ИСПОЛЬЗОВАНИЕ СТАТИЧЕСКОГО АНАЛИЗА КОДА ДЛЯ ПОВЫШЕНИЯ ЭФФЕКТИВНОСТИ РАЗРАБОТКИ ПРОГРАММНОГО ПРОДУКТА

Кумаков В.В.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Карпович С.Е. – доктор технич. наук, профессор

В статье производится обзор статического анализа кода и его влияние на разработку программного обеспечения.

Статический анализ кода — это процесс выявления ошибок и недочетов в исходном коде программ. Статический анализ позволяет находить уязвимости и его можно использовать в процессе разработки, интегрируя в настроенные процессы. Статический анализ можно рассматривать как автоматизированный процесс обзора кода.

С одной стороны, хочется регулярно осуществлять обзор кода. С другой — это слишком дорого. Компромиссным решением являются инструменты статического анализа кода. Они обрабатывают исходные тексты программ и выдают программисту рекомендации обратить повышенное внимание на определенные участки кода. При этом анализаторы не устают и проверяют весь код, который затрагивается правками в файлах. Конечно, программа не заменит полноценного обзора кода, выполняемого коллективом программистов. Однако, соотношение польза/цена делает использование статического анализа весьма полезной практикой, применяемой многими компаниями.

Задачи, решаемые с помощью статического анализа кода, можно разделить на 3 категории:

1. Выявление ошибок в программах.
2. Рекомендации по оформлению кода.
3. Подсчет метрик.

Как и у любой другой методологии выявления ошибок, у статического анализа есть свои сильные и слабые стороны. Нет одного идеального метода тестирования программ. Наилучший результат можно получить, используя сочетание различных подходов, таких как: хороший стиль кодирования, статический анализ кода, динамический анализ кода, юнит-тестирование, регрессионное тестирование и так далее.

Важное преимущество статического анализа состоит в возможности найти многие ошибки сразу после их появления в коде, а значит, их исправление обойдётся очень дешево. Дело в том, что чем раньше ошибка выявлена, тем меньше стоимость ее исправления. Средняя стоимость исправления дефектов на разных этапах разработки представлена в таблице 1.

Таблица 1 – Средняя стоимость исправления дефектов в зависимости от времени их обнаружения

Этап разработки	Стоимость исправлений
Программирование	1
Тестирование	10
Поддержка	10-25

Инструменты статического анализа позволяют выявить большое количество ошибок, характерных для этапа написания кода, что существенно снижает стоимость разработки всего проекта. Чем больше проект, тем больше ошибок на 1000 строк кода он содержит.

Одним из основных алгоритмов статического анализа является анализ потока данных. Задача такого анализа — определить в каждой точке программы некоторую информацию о данных, которыми оперирует код. Информация может быть разная, например, тип данных или значение. В зависимости от того, какую информацию нужно определить, можно сформулировать задачу анализа потока данных.

Безусловным преимуществом статического анализа является полное покрытие анализируемого кода. Также к плюсам статического анализа можно отнести то, что для его запуска нет необходимости выполнять приложение в рабочей среде. Статический анализ можно внедрять на самых ранних стадиях разработки, минимизируя стоимость найденных уязвимостей и тем самым повышая эффективность разработки программного обеспечения.

Список использованных источников:

1. Статический анализ кода [Электронный ресурс]. – Режим доступа : https://www.viva64.com/ru/a/0087/. – Дата доступа: 20.04.2020.

2. A Few Billion Lines of Code Later: Using Static Analysis to Find Bugs in the Real World [Электронный ресурс]. – Режим доступа : https://cacm.acm.org/magazines/2010/2/69354-a-few-billion-lines-of-code-later/fulltext. – Дата доступа: 15.04.2020.