

РОЛЬ АВТОМАТИЧЕСКОГО ТЕСТИРОВАНИЯ В ПРОЦЕССЕ РАЗРАБОТКИ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Очень часто современные программные продукты разрабатываются в сжатые сроки и при ограниченных бюджетах проектов. Программирование сегодня перешло из разряда искусства, став при этом ремеслом для многих миллионов специалистов. Но, к сожалению, в такой спешке разработчики зачастую игнорируют необходимость обеспечения информационной безопасности и защищенности своих продуктов, подвергая тем самым пользователей своих продуктов неоправданному риску. Контроль качества (тестирование) считается важным этапом в процессе разработки программного обеспечения, потому что это обеспечивает безопасность, удобство и надежность создаваемого продукта. Автоматизация этого процесса несомненно принесет большую пользу и сэкономит время и ресурсы компании. Для автоматизации тестирования есть множество программных продуктов, средств и решений, среди которых наиболее популярные WinRunner, SilkTest, Rational Robot являются лидерами рынка автоматизации процессов тестирования, но это тема отдельной статьи, мы же рассмотрим полезность автоматизации в целом. Существует несколько фаз в традиционном тестировании программного продукта:

- Тестирование спецификации
- Дымовое или приемочное тестирование
- Регрессионное тестирование
- Полное функциональное тестирование
- Нагрузочное или стресс-тестирование
- Тестирование документации

Попробуем разобраться, какие фазы тестирования необходимо автоматизировать и почему?

ДЫМОВОЕ ТЕСТИРОВАНИЕ



Дымовое или *приемочное тестирование* (*smoke test*) – это такой набор тестов, после которого можно сказать, что программный продукт запускается. К примеру, после сборки программы-калькулятора дымовым тестом для нее будет запуск самого калькулятора. Если ошибок при запуске не происходит, то дымовой тест считается пройденным. В более сложных продуктах в дымовое тестирование могут включаться дополнительные, но очень простые критерии, такие как «внешний вид» или проверка запуска и остановки сервисов. Если программа не прошла дымовой тест, то ее отправляют на доработку. Дело в том, что разработчики пишут отдельные компоненты одного приложения, но когда эти компоненты объединяют, нередко получается так, что совместно они работать не могут, следовательно, нет смысла тестировать продукт в целом.

В процессе разработки часто бывает так, что новая версия с исправленными ошиб-

ками выпускается каждый день, а иногда и несколько раз в день. Дымовое тестирование прежде всего должно быть автоматизировано, потому что сразу после сборки новой версии программы нам необходимо в кратчайшие сроки убедиться в том, что продукт запускается. Автоматический тест справится с подобной задачей за считанные секунды, и сборку программы можно будет считать успешной. Если же этим будет заниматься человек, то времени на проверку будет уходить гораздо больше, а отдел контроля качества будет простаивать.

На практике это реализуется примерно так: после того как в исходный код продукта внесены все необходимые изменения, продукт отправляют на сборку – этот процесс достаточно долгий и может занимать от получаса до нескольких часов. Сразу после этого в автоматическом режиме собирается установочная программа. Затем установочная программа автоматически копируется на тестовый компьютер, и там уже проходит автоматический тест. Если результат теста удовлетворительный, установочная программа копируется на общедоступный ресурс, и менеджерам продукта (обычно по почте) приходит оповещение о том, что дымовое тестирование пройдено и можно приступать к остальной части тестирования. В противном случае приходит оповещение о необходимости доработать продукт. Во время разработки, сборки и дымового тестирования отдел контроля качества тестирует предыдущую версию программы. Вся автоматизация обычно выполняется скриптами или же специальной тестовой оболочкой.

Таким образом, автоматизация дымового тестирования – это наилучшая практика для экономии времени отдела тестирования. Ведь если продукт нерабочий, а его установили уже все сотрудники отдела, то работа приостановится, пока отдел качества не разберется, что допущены ошибки не в самой программе, а в сборке составных частей. Также будет затрачено время на повторное удаление и установку предыдущей версии программы.

РЕГРЕССИОННОЕ ТЕСТИРОВАНИЕ



Регрессионное тестирование – это набор тестов, включающий в себя проверку большинства ранее найденных ошибок, а также проверку основной функциональности. Проводится, как правило, на каждой новой версии программного продукта.

Регрессионное тестирование является наиболее важной фазой тестирования непосредственно перед окончанием работ над продуктом, так как непосредственно перед релизом продукта крайне необходимо проверить не только основную функциональность, но и то, что ни одна из ранее найденных ошибок не повторяется в финальной версии.

Так как этот набор тестов является достаточно большим, особенно в завершающем этапе тестирования, он является к тому же достаточно сложным. Сложность также заключается в том, что один человек, проводя один и тот же набор тестов снова и снова, устает, со временем меньше уделяет внимания тщательной проверке, поэтому вероятность заметить небольшую ошибку сильно снижается. Кроме того, надо держать в голове список ранее найденных ошибок, не упустить ни одной детали основной функциональности.

Мы видим, что подобный набор тестов достаточно трудно выполнять вручную, а если учесть важность этого набора, то без автоматизации просто не обойтись.

На практике данный подход реализуется следующим образом: составляется набор тестов для проверки всей основной функ-

циональности. Далее тест отлаживается и дополняется. Можно провести аналогию с дымовым тестированием, но есть и принципиальная разница – дымовой тест может сказать, запускается ли продукт или нет, а регрессионный автоматический тест находит ошибки и предоставляет отчетность, по которой можно судить о состоянии продукта и сроках окончания работ. Еще одним важным плюсом регрессионного автоматического теста является то, что становится легко проверять ранее найденные ошибки: после того как ошибка найдена, она добавляется в тест, и при каждом следующем запуске можно четко сказать, исправлена ли ошибка.

Автоматическое регрессионное тестирование выполняет очень ресурсоемкую часть тестирования, тем самым освобождая ресурсы отдела тестирования для поиска более сложных ошибок, которые практически невозможно и нерентабельно находить автоматическим тестом.

НАГРУЗОЧНОЕ ТЕСТИРОВАНИЕ



Нагрузочное или *стресс-тестирование* подразумевает проверку программы в стрессовых ситуациях, таких как наличие большого объема входных параметров, нехватка дискового пространства, маломощный процессор и так далее. Очевидно, что не все стрессовые ситуации можно автоматизировать, например, в условиях ограниченного дискового пространства или маломощ-

ного процессора ресурсы нужны не только самому программному продукту, но и программе, проводящей автоматический тест. Но если рассмотреть ситуацию большого объема входных значений, то в этой ситуации автоматизация будет большим плюсом. Формирование большого входного массива, набор огромных списков, переключение всевозможных элементов управления – со всеми этими задачами автоматический тест справится без каких-либо трудностей и будет повторять формирование столько раз, сколько потребуется. После того как программный продукт обработает все входящие данные, надо проанализировать результат, который не очень удобно просматривать глазами из-за большого объема результирующей информации. Автоматическая проверка тем и хороша, что независимо от объема проверяемой информации, всегда можно быть уверенным в достоверности результата.

В нагрузочном тестировании автоматизация позволяет повысить точность результата тестирования и, конечно же, экономит время, выделенное на тестирование продукта.

Автоматизация в целом не только экономит время на разработку, что является актуальной проблемой, но увеличивает надежность и безопасность создаваемых продуктов. В данный момент на рынке существует 8 крупных компаний, которые занимаются разработкой средств автоматизации тестирования. Развитость рынка очевидна, но до сих пор есть множество областей, которые не покрыты существующими средствами автоматизации. Спрос на такие средства растет с каждым годом, что говорит о возрастающей отдаче и о все большей распространенности автоматизации процессов тестирования.

Очевидны плюсы как самих таких компаний, так и инженеров по качеству: надежность проверки продукта возрастает, время на тестирование сокращается, работа тестирующего становится менее стрессовой. Конечно, автоматические тесты никогда не смогут заменить человека, но могут облегчить и усовершенствовать работу инженера по качеству программного обеспечения.

Литература

1. *Винниченко И.В.* Автоматизация процессов тестирования. СПб.: Питер, 2005.
2. *Дастин Э., Рэшка Д., Пол Д.* Автоматизированное тестирование программного обеспечения. М.: Лори, 2003.
3. *Тамре Л.* Введение в тестирование программного обеспечения. М.: Вильямс, 2003.
4. *Кент Б.* Экстремальное программирование: разработка через тестирование. СПб.: Питер, 2003.
5. Тестирование и качество ПО [Электронный ресурс]; Актуальность вопросов тестирования безопасности и защищенности программных продуктов; *Полаженко С.*; <http://software-testing.ru/lib/polazhenko/security-testing-actuality.htm>



Наши авторы, 2006.
Our authors, 2006.

*Колмогоров Константин Алексеевич,
аспирант факультета КТ и У
кафедры ВТ СПбГУ ИТМО.*