

ТЕСТИРОВАНИЕ НА ОСНОВЕ МОДЕЛЕЙ

Садовский М.В., Медведев О.С.

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

Научный руководитель: Меженная М.М. – канд. техн. наук, доцент, доцент кафедры ИПиЭ

Аннотация. Рассмотрено тестирование на основе моделей, преимущества данного подхода. Затронута возможность использования облачных вычислений при модульном тестировании. А также описаны необходимые требования для автоматизированного тестирования на основе моделей.

Ключевые слова: тестирование, тестирование на основе моделей, MBT

Введение. Качество веб-услуг является важным фактором для предприятий, которые рекламируют или продают свои услуги в интернете. Тестирование сложных веб-сервисов, организованных в сервис-ориентированных архитектурах, является трудной и сложной проблемой. Тестирование на основе моделей (*Model-based testing, MBT*) является одним из решений, позволяющих справиться со сложностью тестирования. При *MBT* тестировщики не определяют тесты напрямую, а определяют структуру и поведение тестируемой системы с помощью моделей. Затем используется стратегия тестирования для получения тестовых случаев автоматически из моделей. Однако *MBT* дает большое количество тестов для сложных систем, которые требуют много ресурсов для их выполнения, что ограничивает его потенциал.

Основная часть. В рамках растущего успеха Интернета веб-сервисы, такие как онлайн-банкинг, стали важным средством предоставления услуг потребителям. Это привело к появлению сервис-ориентированных архитектур (*service-oriented architectures, SOA*) парадигмы, в которой программное обеспечение декомпозируется на отдельные веб-сервисы, слабо связанные друг с другом [1]. Важным аспектом *SOA* является то, что *Web*-сервисы описываются только с помощью их интерфейсов, независимо от их реализации и расположения. *SOA*-приложения объединяют *Web*-сервисы и определяют шаблоны для их взаимодействия. Этот процесс называется оркестровкой сервисов. Высокое качество имеет решающее значение для успеха веб-сервисов. Однако тестирования отдельных веб-сервисов в изоляции недостаточно, поскольку необходимо убедиться, что веб-сервис может функционировать в рамках полного *SOA*-приложения, а не только сам по себе. Таким образом, необходимо протестировать и известные оркестрации. Это приводит к ряду проблем при тестировании сервис-ориентированных решений, например:

- использование сервисов может быстро измениться, если они используются в новых или других оркестровках;
- исходный код или другая структурная информация сервисов часто недоступна, доступны только их интерфейсы. Это не позволяет использовать методы тестирования "белого ящика". Более того, это усложняет определение корректных имитационных объектов, необходимых для модульного тестирования сервисов;
- непредвиденная эволюция услуг других поставщиков в рамках оркестровки услуг может привести к всевозможным осложнениям в работе и обеспечении качества услуг, например, из-за изменения интерфейсов или поведения;
- создание тестовой среды со всеми необходимыми компонентами, такими как серверы приложений, конфигурации брандмауэров и мониторинг, само по себе уже является сложной задачей.

Тестирование на основе моделей (*Model-based testing, MBT*) обеспечивает решение многих проблем тестирования программного обеспечения. Модели обеспечивают высокий уровень абстракции, который позволяет компактно определить сложное поведение. Благодаря этому такие модели, как машины состояний, могут отражать поведение целых протоколов и

оркестров и, следовательно, подходят для тестирования *SOA*. Количество тестов, получаемых из модели, зависит от стратегии тестирования и сложности моделируемого программного обеспечения. Обычно стратегии тестирования пытаются получить определенное покрытие системы, например, путем выполнения всех событий системы или путем перевода системы во все ее логические состояния хотя бы один раз. Однако более сложные покрытия, например, когда должны быть охвачены все пары или даже тройки возможных событий, обычно не могут быть достигнуты из-за экспоненциально растущего числа тестов. Это особенно проблематично для большого и сложного программного обеспечения, такого как оркестрованные сервисы с множеством возможных путей взаимодействия. Таким образом, усилия по тестированию ограничены ресурсами, доступными для выполнения тестов[2].

Для преодоления ограниченности ресурсов целесообразно использовать облачные вычисления. С помощью облачных вычислений можно арендовать вычислительную инфраструктуру по требованию. Более того, одной из особенностей облачных вычислений является эластичность, которая позволяет динамически масштабировать вычислительную инфраструктуру в зависимости от текущих вычислительных потребностей. Это естественным образом подходит для масштабирования *MBT* и позволяет проводить крупномасштабные тестовые кампании с автоматически генерируемыми тестами для сложных *SOA*-оркестров.

Международный совет по квалификации тестирования программного обеспечения (*The International Software Testing Qualifications Board, ISTQB*) определяет *MBT* как "тестирование на основе или с использованием моделей" [3]. Это означает, что концептуальная модель *SUT* используется для создания тестов для системы. Это определение довольно общее и допускает различные тестовые артефакты, которые могут быть получены из моделей, например:

- абстрактные тестовые случаи, которые обеспечивают высокоуровневое описание задач для ручного тестирования. Тестовые примеры могут содержать не все необходимые значения или могут отсутствовать очевидные промежуточные шаги, что может привести к отклонениям при определении конкретных тестовых примеров на основе абстрактных тестов;
- конкретные тестовые случаи, предоставляющие всю необходимую информацию для повторяемого ручного тестирования. Может оказаться возможным автоматическое генерирование тестов с помощью ручных усилий;
- конкретные тестовые случаи, доступные в виде автоматически исполняемого тестового сценария, который может быть скомпилирован и исполнен.

Для автоматизированного *MBT* важно, чтобы тестовая модель была богата подробной информацией о *SUT*; в частности, необходимы три вещи: поведенческая модель, описание интерфейса и информация развертывания.

Первым требованием для автоматизированного *MBT* является структурное описание *SUT*, включающее интерфейсы, открываемые *SUT*. Описание интерфейса должно описывать операции, которые могут быть вызваны, а также структуры данных, используемые для ввода и вывода этими операциями.

Вторым требованием для автоматизированного *MBT* является поведенческая модель *SUT*. Задача поведенческой модели – определить, как должен вести себя *SUT* при взаимодействии с окружающей средой. На основе этого определения используется стратегия тестирования для получения тестовых случаев из поведенческой модели. Комбинация поведенческой модели и стратегии тестирования определяет, какие и сколько тестовых примеров будут получены из модели.

Третьим требованием для автоматизированной *MBT* является информация о развертывании *SUT*. Существует два места, где может содержаться эта информация.

1. Эта информация становится частью тестового набора, в котором выполняются тестовые случаи. В данном случае эта информация не содержится в модели, а добавляется непосредственно во время выполнения теста.

2. Информация содержится в модели и входит в состав сгенерированных тестовых примеров.

Заключение. Таким образом при рассмотрении процесса тестирования на основе моделей, наилучшим вариантом является привлечение облачных сервисов, для проведения вычислений. При этом необходимо соблюдать требования для автоматизированного тестирования.

Список литературы

1. Erl, T.: *Service-Oriented Architecture: Concepts, Technology, and Design*. Pearson Education India, Karnataka (2005).
2. Barcelona Liédana, M.A., López-Nicolás, G., García-Borgon òn, L.: *Practical experiences in the usage of midas in the logistics domain*. *J. Softw. Tools Technol. Transf. (STTT)* (2016).
3. *International Software Testing Qualifications Board (ISTQB): Standard glossary of terms used in Software Testing, Version 2.1* (2010).

UDC 004.054

MODEL-BASED TESTING

Sadovskiy M.V., Medvedev O.S.

Belarusian State University of Informatics and Radioelectronics, Minsk, Republic of Belarus

Mezhennaya M.M. – PhD, assistant professor, associate professor of the department of EPE

Annotation. The paper considers model-based testing, the advantages of this approach. The possibility of using cloud computing in unit testing is touched upon. It also describes the necessary requirements for automated model-based testing.

Keywords: testing, model-based testing, MBT