Тестирование программного обеспечения встроенных систем авионики

Институт системного программирования
*Российской академии наук (ИСП РАН) **

1. Цели тестирования программного обеспечения.

Тестирование программного обеспечения (ПО) встроенных систем авионики имеет две взаимодополняющие цели. Первая цель – показать, что ПО удовлетворяет требованиям к нему. Вторая цель – продемонстрировать с высокой степенью доверия, что были устранены ошибки, которые могли бы привести к возникновению отказных ситуаций, определенных процессом оценки безопасности системы. Тестированию, основанному на требованиях, уделяют особое внимание, потому что эту стратегию признают наиболее эффективной в обнаружении ошибок. Поэтому в соответствии со стандартом DO-178B [3] для удовлетворения целей тестирования:

- тестовые варианты должны быть основаны, прежде всего, на требованиях к ПО;
- тестовые варианты должны быть разработаны так, чтобы верифицировать корректность функционирования и сформировать условия, которые выявляют потенциальные ошибки;
- анализ покрытия требований к ПО должен определить, какие требования к ПО не были тестированы;
- анализ структурного покрытия должен определить, какие структуры ПО не были выполнены при тестировании.

2. Тестирование на соответствие стандарту ARINC-653.

Стандарт ARINC-653 [1] — “Avionics Application Software Standard Interface” — разработан компанией ARINC (Aeronautical Radio, Inc.) в 1997 г. Этот стандарт определяет универсальный программный интерфейс APEX (Application/Executive) между операционной системой авиационного компьютера и прикладным ПО. В настоящее время он является основным для реализации в операционных системах реального времени (ОСРВ) для встроенных систем авионики.

Для решения задачи тестирования на соответствие требованиям стандарта и с целью упрощения процесса сертификации коммерческих продуктов в стандарте ARINC-653 присутствует третья часть (Part 3 – Conformity Test Specification), содержащая описание набора тестовых вариантов для проверки соответствия функциональности интерфейса любой ОСРВ требованиям из первой части стандарта ARINC-653. Этот набор включает около 250 различных тестовых вариантов, покрывающих требования ко всем 56 функциям, обязательным для реализации в ОСРВ. Операционная система признается соответствующей стандарту ARINC-653 Part 1, если все тесты из этого набора выполнятся успешно. В настоящее время ведутся работы по созданию спецификации тестового набора для второй части стандарта – Extended services.

В ИСП РАН разработан тестовый набор для проверки ОСРВ на соответствие стандарту ARINC-653 Part 1. В основу этого набора положены тесты, специфицированные в третьей части стандарта. Однако в процессе работы над тестовым набором были обнаружены многочисленные ошибки в спецификации тестов ARINC-653 Part 3. Суть этих ошибок

заключается, в основном, в том, что некоторые проверки в тестах не соответствуют требованиям, зафиксированным в ARINC-653 Part 1, а иногда и нарушают их. По-видимому, спецификация тестового набора никак не адаптировалась к изменениям в спецификации функциональных требований к API. В результате спецификация тестового набора отстала в развитии от спецификации API ОСРВ. Поэтому тестовый набор был исправлен и существенно расширен за счет:

- тестов на некоторые комбинации требований, которые оказались не покрытыми в ARINC-653 Part 3 (в результате в тестируемой ОСРВ были обнаружены дополнительные ошибки, не обнаруживаемые тестами из набора ARINC-653 Part 3);
- тестов на функциональность системы межпроцессного взаимодействия в системных процессах, поддерживающих интерфейс POSIX;
- общих системных тестов для подсистем Interpartition Communication, Intrapartition Communication и Health Monitoring.

В результате тестовый набор включает более 370 тестов, покрывающих все аспекты функциональности отдельных процедур тестируемого интерфейса, а также функциональность подсистем в целом. Набор реализован на языке Си. Структура набора соответствует ARINC-653 Part 3 и включает следующие уровни:

- уровень тестовых макросов;
- уровень тестов;
- уровень тестовой последовательности.

Для каждой процедуры (сервиса) тестируемого API реализована одна процедура, которая проверяет правильность выполнения тестируемой процедуры API на одном наборе входных параметров путем сравнения полученных при вызове тестируемого сервиса результатов с ожидаемыми. Уровень индивидуальных тестов реализован в виде набора файлов с именами, соответствующими идентификаторам тестов в ARINC-653 Part 3. Уровень тестовой последовательности реализован в виде управляющей тестовой программы и позволяет выполнять тесты по одному в любом порядке с использованием разработанной для каждого теста конфигурации системы. Это исключает возможные взаимовлияния (прежде всего временные) тестов друг на друга во время выполнения.

Разработанный тестовый набор применялся и был апробирован в проекте по тестированию ОСРВ «Багет» [4]. Он позволил обнаружить несколько ошибок в системе, которые невозможно обнаружить исходным тестовым набором, специфицированным в стандарте ARINC-653 Part 3.

3. Тестирование на соответствие стандарту POSIX.

Другой стандарт, широко используемый в ОСРВ для встроенных систем авионики – это стандарт POSIX (Portable Operating System interface for unIX) [2]. Он определяет переносимый интерфейс операционных систем на уровне исходных текстов. Основная спецификация разработана как спецификация IEEE 1003.1 и одобрена в качестве международного стандарта ISO/IEC 9945-1:1990. С точки зрения ОСРВ для встроенных систем авионики наибольший интерес представляют три стандарта: 1003.1a (OS Definition), 1003.1b (Realtime Extensions) и 1003.1c (Threads).

В ИСП РАН разработан тестовый набор для проверки операционной системы на соответствие стандарту POSIX. По сравнению с имеющимися тестовыми наборами для стандарта POSIX (например, сертификационный тестовый набор от Open Group и результаты открытого проекта Open POSIX Test Suite) этот тестовый набор обладает следующими особенностями и преимуществами:

- **формализация требований**: функциональные требования к поведению тестируемой системы, выделенные из текста стандарта, представляются в виде контрактных спецификаций;
- **автоматическая генерация тестов**: тесты или последовательности тестовых воздействий генерируются при выполнении тестового сценария, во время которого

проверяются функциональные требования и обеспечивается тестовое покрытие по заданному критерию;

- использование **техники «тестового агента»** для возможности работы тестового набора на целевой платформе с ограниченными ресурсами (например, на встроенной системе).

Автоматическая генерация тестов из контрактных спецификаций, используемая в разработанном тестовом наборе для стандарта POSIX, делает тестовый набор более управляемым, позволяя описывать требования стандарта в одном месте, а технику их проверки и тестовые данные – в другом. Это значительно облегчает внесение изменений в тесты при развитии стандарта или их адаптации под требования специфической предметной области.

Каталог требований содержит более 10 тысяч требований, выделенных из текста стандарта POSIX. Эти требования формализованы в контрактных спецификациях для 916 функций стандарта общим объемом около 60 тысяч строк. Для тестирования этих функций и подсистем стандарта POSIX тестовый набор содержит 172 тестовых сценария.

Разработанный тестовый набор применялся и был апробирован в проекте OLVER (Open Linux VERification) [5]. В рамках этого проекта тестовый набор использовался для тестирования различных операционных систем семейства Linux. Полученное во время этого тестирования покрытие кода приближается к результатам отладочного тестового набора для библиотеки glibc ОС Linux, созданного разработчиками этой библиотеки, обладающими детальными знаниями об особенностях реализации различных ее функций, и не нацеленного на проверку соответствия каким-либо стандартам. Для некоторых групп функций разработанный тестовый набор обеспечивает большее покрытие кода, чем другие известные наборы для тестирования функциональности.

4. Поддержка работ по тестированию в рамках стандарта DO-178B.

Стандарт DO-178B “Software Consideration in Airborne Systems and Equipment Certification” [3] создан ассоциацией RTCA (Radio Technical Commission for Aeronautics) и определяет требования к процессу разработки и сертификации ПО бортовых авиационных систем. В Европе принят стандарт ED-12B — европейский аналог DO-178B, который определяется EUROCAE (The European organisation for civil aviation equipment). В России с 2003 года действует стандарт ГОСТ Р 51904-2002 «Программное обеспечение встроенных систем: общие требования к разработке и документированию», который также является аналогом стандарта DO-178B. До тех пор, пока все жесткие требования этого стандарта не будут выполнены, вычислительные системы, влияющие на безопасность, никогда не поднимутся в воздух.

Методология сертификации, изложенная в стандарте DO-178B, требует от поставщиков программных решений «доказанного» качества разработанного ими ПО. В качестве метода такого доказательства стандарт предлагает поставщикам разработать тестовый набор, основанный на требованиях к ПО и удовлетворяющий следующим условиям:

- в наборе существуют тестовые варианты для каждого требования к ПО;
- тестовые варианты удовлетворяют критериям тестирования области определения (normal range test cases) и тестирования на устойчивость к ошибкам (robustness test cases).

Для поддержки этих работ по тестированию в ИСП РАН разработаны технология UniTESK[6] и процесс FOREST [7] для разработки тестового набора для ПО. Основные задачи этого подхода состоят в том, чтобы:

- построить формальную спецификацию требований к ПО;
- разработать на ее основе тестовый набор для тестирования соответствия интерфейса ПО этим требованиям.

Процесс разработки тестового набора состоит из четырех этапов. Результаты каждого этапа используются при выполнении последующих, но могут быть использованы и отдельно.

Этап 1. Требования извлекаются из первичных документов (ARINC-653 Part 1 и 2, POSIX и т.п.) и систематизируются. В результате получается каталог требований, в котором требования сформулированы максимально однозначно, каждому требованию сопоставлен уникальный идентификатор, требования классифицированы, и, возможно, установлены связи между отдельными требованиями. Каталог требований используется на последующих этапах и позволяет в дальнейшем оценивать адекватность тестирования или полноту реализации ПО в терминах исходного текста стандарта.

Этап 2. Выполняется анализ требований. В качестве основного средства анализа используются концептуальные модели требований. В ходе разработки и анализа концептуальных моделей происходит реструктуризация знания о предметной области, и производится проверка свойств требований, таких как адекватность, полнота, и т.п. Разработка модели опирается на каталог требований, построенный на первом этапе.

Этап 3. Представление требований в формальном виде. Требования из каталога записываются с использованием математического формализма контрактных спецификаций. Перевод требований из текстового представления в формальное облегчается тем, что знания о предметной области были реструктурированы и проанализированы на первых двух этапах.

Этап 4. Разработка тестов с использованием контрактных спецификаций. На этом этапе разрабатываются тестовые сценарии, обеспечивающие проверку всех требований из каталога, зафиксированных в контрактных спецификациях. Тестовые сценарии пишутся на языке спецификаций и выполняются под управлением инструмента UniTESK. По результатам выполнения тестовых сценариев создается отчет о проведенном тестировании, показывающий полноту покрытия и возможные нарушения требований, включенных в каталог требований на первом этапе.

Применение этого подхода позволяет разработать тестовый набор, обеспечивающий трассируемость тестов к исходным требованиям, зафиксированным в тексте стандарта. Это позволит упростить идентификацию и модификацию тестов, которые требуют доработки, если исходные требования подверглись изменениям. В результате использование технологии UniTESK и процесса FOREST обеспечивает тестирование ПО в соответствии с требованиями стандарта DO-178B и тем самым облегчает сертификацию программных систем по этому стандарту.

5. Литература.

1. ARINC. ARINC Specification 653-2: Avionics Application Software Standard Interface Part 1 - Required Services. Aeronautical Radio INC, Maryland, USA, 2005.
2. IEEE Std 1003.1, 2004 Edition. The Open Group Technical Standard. Base Specification, Issue 6.
3. RTCA/DO-178B, "Software Considerations in Airborne Systems and Equipment Certification". http://www.rtca.org
4. Операционная система ос2000. http://www.niisi.ru/intro1.htm
5. http://www.linuxtesting.ru
6. http://www.unitesk.ru
7. В.В.Кулямин, Н.В.Пакулин, О.Л.Петренко, А.А.Сортов, А.В.Хорошилов «Формализация требований на практике». Препринт N.13. М.: ИСП РАН, 2006.