

Финансовый университет при Правительстве РФ, Финансово-
экономический колледж

Тестирование и откатка ПО. Нормативно-правовая база.

2017 г.

Работу выполнили
Студенки 2курса:
Ардуванова Анастасия и
Цивилева Татьяна

Тестирование

- - это процесс обнаружения ошибок в *ПО* путем исполнения выходного кода ПС на тестовых данных, сбора рабочих характеристик в динамике выполнения в конкретной операционной среде, выявления различных ошибок, дефектов, отказов и изъянов, вызванных нерегулярными и аномальными ситуациями или аварийным прекращением работы *ПО*. Важное *место* в проведении верификации и тестирования занимают организационные аспекты - *деятельность* группы специалистов, осуществляющих планирование этих процессов, подготовку тестовых данных и наблюдение за тестированием.

Процессы ЖЦ верификация и валидация программ

- *Верификация и валидация*, как методы, обеспечивают соответственно проверку и *анализ* правильности выполнения заданных функций и соответствия *ПО* требованиям заказчика, а также заданным спецификациям.

Процесс верификации

Цель процесса - убедиться, что каждый *программный продукт* (и/или сервис) проекта отражает согласованные требования к их реализации. Этот процесс основывается:

- 1
 - на стратегии и критериях верификации применительно ко всем рабочим программным продуктам;
- 2
 - на выполнении действий стандарта по верификации;
- 3
 - на *устранении недостатков*, обнаруженных в программных (рабочих и промежуточных) продуктах;
- 4
 - на согласовании результатов верификации с заказчиком.

Процесс валидации

- Цель процесса - убедиться, что специфические требования для программного продукта выполнены, и осуществляется это с помощью:

- 1
 - разработанной стратегии и критериев валидации для всех *рабочих продуктов*;
- 2
 - оговоренных действий по проведению валидации;
- 3
 - демонстрации соответствия разработанных программных продуктов требованиям заказчика и правилам их использования;
- 4
 - согласования с заказчиком полученных результатов валидации.

На других процессах ЖЦ выполняются дополнительные действия:

На других процессах ЖЦ выполняются
дополнительные действия:

- 1
 - проверка и контроль проектных решений с помощью методик и процедур просмотра хода разработки;
- 2
 - обращение к *CASE-системам* , которые содержат процедуры проверки требований к продукту;
- 3
 - просмотры и инспекции промежуточных результатов на соответствие их требованиям для подтверждения того, что ПО имеет корректную реализацию требований и удовлетворяет условиям выполнения системы.

На других процессах ЖЦ выполняются дополнительные действия:

- Таким образом, основные задачи процессов верификации и валидации состоят в том, чтобы *проверить и подтвердить*, что конечный программный продукт отвечает назначению и удовлетворяет требованиям заказчика. Эти процессы взаимосвязаны и определяются, как правило, одним общим термином "*верификация и валидация*" или "*Verification and Validation*" (V&V)

Тестирование программ

- **Тестирование** можно рассматривать, как процесс семантической отладки (проверки) программы, заключающийся в исполнении последовательности различных наборов контрольных тестов, для которых заранее известен результат. Т.е. тестирование предполагает выполнение программы и получение конкретных результатов выполнения тестов.
- Тесты подбираются так, чтобы они охватывали как можно больше типов ситуаций алгоритма программы. Менее жесткое требование - выполнение хотя бы один раз каждой ветви программы.
- Исторически первым видом тестирования была *отладка*.

Тестирование программ

- *Отладка*
 - это проверка описания программного объекта на ЯП с целью обнаружения в нем ошибок и последующее их устранение. Ошибки обнаруживаются компиляторами при их синтаксическом контроле. После этого проводится верификация по проверке правильности кода и валидация по проверке соответствия продукта заданным требованиям.
- Целью тестирования
 - проверка работы реализованных функций в соответствии с их спецификацией. На основе внешних спецификаций функций и проектной информации на процессах ЖЦ создаются функциональные тесты, с помощью которых проводится тестирование с учетом требований, сформулированных на этапе анализа *предметной области*. Методы *функционального тестирования* подразделяются на статические и динамические.

Статические методы тестирования

- Статические методы используются при проведении инспекций и рассмотрении спецификаций компонентов без их выполнения. Техника статического анализа заключается в методическом просмотре (или обзоре) и анализе структуры программ, а также в доказательстве их правильности. Статический анализ направлен на анализ документов, разработанных на всех этапах ЖЦ и заключается в инспекции исходного кода и сквозного контроля программы.

Статические методы тестирования

- *Инспекция ПО* - это статическая проверка соответствия программы заданным спецификациями, проводится путем анализа различных представлений результатов проектирования (документации, требований, спецификаций, схем или исходного кода программ) на процессах ЖЦ. Просмотры и инспекции результатов проектирования и соответствия их требованиям заказчика обеспечивают более высокое качество создаваемых ПС.

Эффективность такой проверки заключается в том, что привлекаемые эксперты пытаются взглянуть на проблему "со стороны" и подвергают ее всестороннему критическому анализу

Динамические методы тестирования

- *Динамические методы тестирования* используются в процессе выполнения программ. Они базируются на графе, связывающем причины ошибок с ожидаемыми реакциями на эти ошибки. В процессе тестирования накапливается информация об ошибках, которая используется при оценке надежности и качества ПС.

Краткая характеристика.

- Систематические методы тестирования делятся на методы, в которых программы рассматриваются как "черный ящик" (используется информация о решаемой задаче), и методы, в которых программа рассматривается как "белый ящик" (используется структура программы). Этот вид называют тестированием с управлением по данным или управлением по входу-выходу. Цель - выяснение обстоятельств, при которых поведение программы не соответствует ее спецификации. При этом количество обнаруженных ошибок в программе является критерием качества входного тестирования.
- Цель динамического тестирования программ по принципу "черного ящика" - выявление одним тестом максимального числа ошибок с использованием небольшого подмножества возможных входных данных.

Методы "черного ящика" обеспечивают:

- 1
 - эквивалентное разбиение;
- 2
 - анализ граничных значений;
- 3
 - применение функциональных диаграмм, которые в соединении с реверсивным анализом дают достаточно полную информацию о функционировании тестируемой программы.

Средства разработки и отладки программного обеспечения

- Для разработки программного обеспечения в настоящее время используются **интегрированные среды разработки**, содержащие в своем составе текстовые редакторы, компиляторы, *редакторы связей, загрузчики* и симуляторы.

Текстовые редакторы служат для создания текста программы. Как правило, они имеют более ограниченные возможности *по* сравнению с универсальными программами этого типа и ориентированы на особенности написания программ на используемых языках программирования.

Если программы предыдущих типов используются при любом процессе программирования, то *симуляторы* являются специфическим программным средством, используемым в процессе проектирования МПС.

Средства разработки и отладки программного обеспечения

- **Симуляторы** предоставляют пользователю возможность выполнить **тестирование** и **отладку** разработанного программного обеспечения на программно-логической модели микропроцессора.

Симулятор охватывает сразу несколько процессоров одного семейства. Выбор конкретного типа МП среди моделей семейства обеспечивается соответствующими опциями *меню*. При этом моделируется работа ЦП, всех портов ввода/вывода, прерываний и другой *периферии*.

Изначально *отладка* программ с использованием симуляторов велась на уровне машинных команд в символьных обозначениях регистров. В состав современных симуляторов входят также отладчики на языках высокого уровня, поскольку в комплект разработчика, как правило, входит и соответствующий *компилятор*.

Комплексная отладка микропроцессорных систем

- Как правило, *микропроцессорная система* на основе МК или ЦСП - это система реального времени, то есть *корректность* ее функционирования зависит от времени выполнения отдельных программ и скорости работы аппаратуры. Поэтому система считается отлаженной после того, как рабочие программы правильно функционируют на *действительной* аппаратуре системы в реальных условиях.
- Дополнительным свойством, которым должны обладать средства комплексной отладки *по* сравнению со средствами автономной отладки, является возможность управления поведением МПС и сбора информации о ее поведении в реальном времени.

Комплексная отладка микропроцессорных систем

- На этапе комплексной отладки *микропроцессорной системы* используются следующие **основные приемы**:

- 1
 - пошаговое отслеживание поведения системы;
- 2
 - останов функционирования системы при возникновении определенного события;
- 3
 - чтение и изменение содержимого памяти или регистров системы в момент останова;
- 4
 - отслеживание поведения системы в реальном времени.