

Курс лекций по дисциплине «Программная инженерия»

Лекция 7. Тестирование

Ст. преподаватель кафедры ИС
Важдаев А.Н.

Управление качеством

- 1865 год – образован комитет, который ныне называется ITU (International Telecommunication Union). ITU является часть ООН. Самыми известными стандартами ITU являются:
 - ISDN (цифровая телефонная связь, объединяющая телефонные сервисы и передачу данных),
 - ADSL (широко известная модемная технология, позволяющая использовать телефонную линию для выхода в Интернет),
 - OSI (модель открытого 7-уровневого сетевого протокола),
 - языки визуального проектирования телекоммуникационных систем, SDL и MSC, влившие позднее в UML.

Управление качеством

- 1946 год – создана организация ISO (International Organization for Standardization). Цель – содействие развитию стандартизации, а также смежных видов деятельности в мире с целью обеспечения международного обмена товарами и услугами, способствование и развитие сотрудничества в интеллектуальной, научно-технической и экономической областях. Вот некоторые стандарты.
 - Серия стандартов ISO 9000.
 - ISO/IEC 90003:2004 – адаптация стандартов ISO 9000 к производству ПО в русле обеспечения качества в жизненном цикле ПО.
 - ISO 9126:2001 – определение качественного ПО и различных атрибутов, описывающих это качество.

Управление качеством

- 1988 год, образование организации ETSI (European Telecommunications Standards Institute), штаб-квартира в г. София Антиполис (Франция). Является независимой, некоммерческой, организацией по стандартизации в телекоммуникационной промышленности (изготовители оборудования и операторы сети) в Европе. Самые известные стандарты – GSM, система профессиональной мобильной радиосвязи TETRA.

Управление качеством

- 1984 год – создание SEI (Software Engineering Institute) на базе университета Карнеги-Меллон в г.Питсбурге (США). Самые известные продукты – стандарт СММ, СММІ.
- 1963 год – создание IEEE (Institute of Electrical and Electronics Engineers). Сейчас IEEE международная некоммерческая ассоциация специалистов в области техники, мировой лидер в области разработки стандартов по радиоэлектронике и электротехнике.
- 1989 год – группа американских IT-компаний (в том числе Hewlett Packard, Sun Microsystems, Canon) организовали OMG (Object Management Group). Сейчас включает около 800 компаний членов. Известные стандарты CORBA, UML, MDA.

Управление качеством

- Наладка качественного процесса, совершенствование процесса. Для комплексного улучшения процессов в компании (подход technology push) компаниями-разработчиками ПО используются стандарты CMM/CMMI, а также по стандартам серии ISO 9000 (с последующей официальной сертификацией).
- Формальные методы – использование математических формализмов для доказательства корректности, спецификации, проверки формального соответствия, автоматической генерации и т.д.:
 - доказательство правильности работы программ,
 - проверка на моделях определенных свойств (model cheking),
 - статический анализ кода по дереву разбора программы,
 - модельно-ориентированное тестирование (model-based testing).

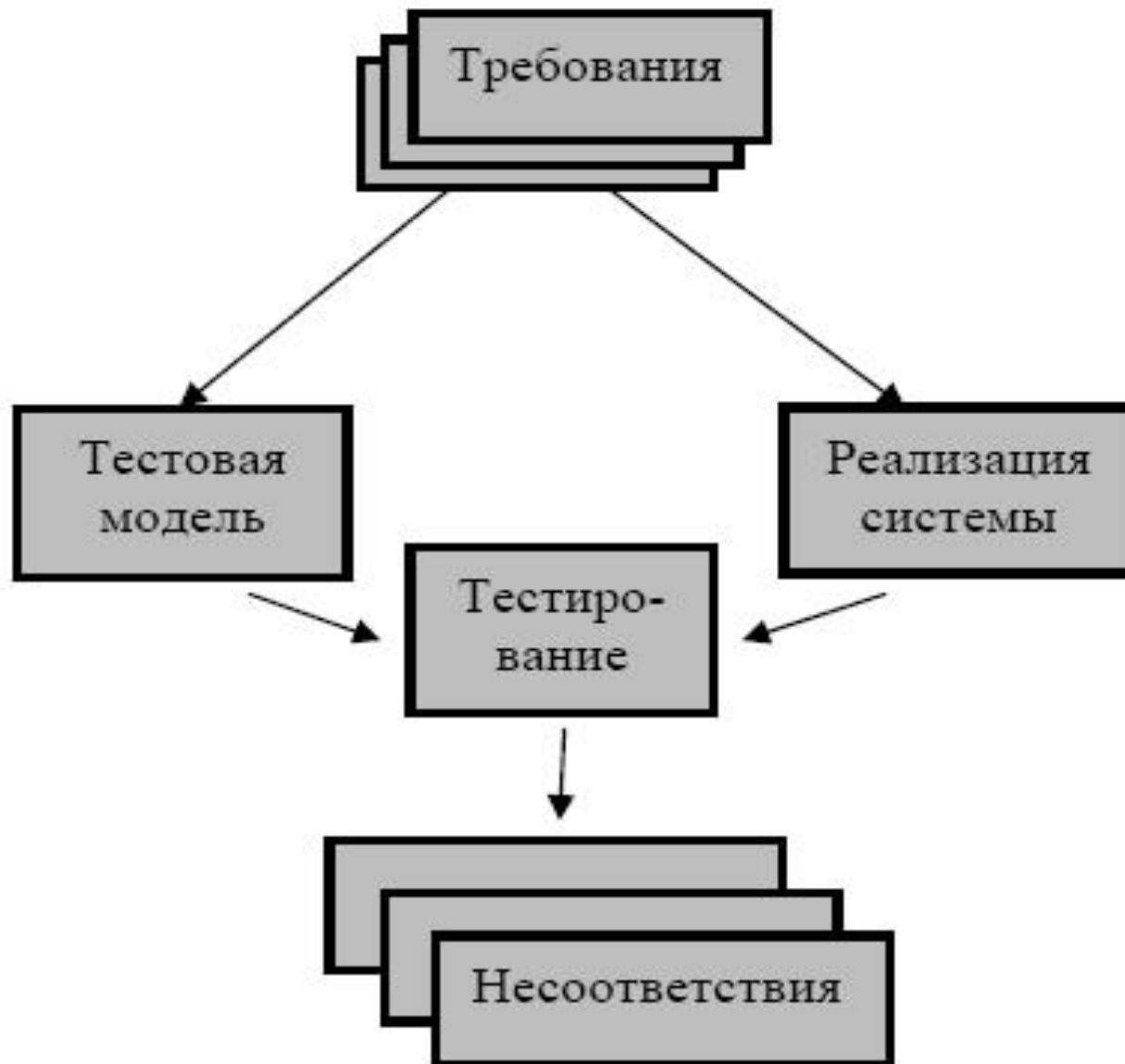
Управление качеством

- Исследование и анализ динамических свойств ПО.
- Обеспечение качества кода.
 - Разработка стандартов оформления кода в проекте и контроль за соблюдением этих стандартов.
 - Регулярный рефакторинг для предотвращения образования из кода "вермишели". *Рефакторинг* – это регулярная деятельность по переписыванию кода, но не с целью добавления новой функциональности, а для улучшения его структуры.
 - Различные варианты инспекции кода, например, техника peer code review.
 - «Вычитка» кода, используемый, например, при разработке критических систем реального времени.
- Тестирование. Самый распространенный способ контроля качества ПО, представленный, фактически, в каждом программном проекте.

Тестирование

- **Тестирование** – это проверка соответствия между реальным поведением программы и ее ожидаемым поведением в специально заданных, искусственных условиях.
- *Ожидаемое поведение программы.* Исходной информацией для тестирования является знание о том, как система должна себя вести, то есть требования к ней или к ее отдельной части. Самым распространенным способом тестирования является тестирование методом **черного ящика**, то есть когда реализация системы недоступна тестировщикам, а тестируется только ее интерфейс. Существует тестирование методом **белого ящика**, когда код программ доступен тестировщикам и используется в качестве источника информации о системе.

Тестирование



Тестирование

- *Специально заданные, искусственные условия*, – те условия, где осуществляется тестирование. При этом ключевым аспектом здесь является наличие **тестов** – воспроизводимых шагов манипуляции с системой, приводящих к ее некорректной работе.
- Тесты могут быть "ручными" и автоматизированными. "Ручной" тест – это последовательность действий тестировщика, которую он (или разработчик) может воспроизвести и ошибка произойдет. Как правило, в средствах контроля ошибками такие последовательности действий содержатся в описании ошибки. Автоматический тест – это некоторая программа, которая воздействует на систему и проверяет то или иное ее свойство. Автоматический тест, по сравнению с "ручным", можно легко воспроизводить без участия человека.

Тестирование

- **Модульное тестирование** - тестируется отдельный модуль, в отрыве от остальной системы. Самый распространенный случай применения – тестирования модуля самим разработчиком, проверка того, что отдельные модули, классы, методы делают действительно то, что от них ожидается.
- **Интеграционное тестирование** – две и более компонент тестируются на совместимость.
- **Системное тестирование** – это тестирование всей системы в целом, как правило, через ее пользовательский интерфейс. При этом тестировщики, менеджеры и разработчики акцентируются на том, как ПО выглядит и работает в целом, удобно ли оно, удовлетворяет ли она ожиданиям заказчика.

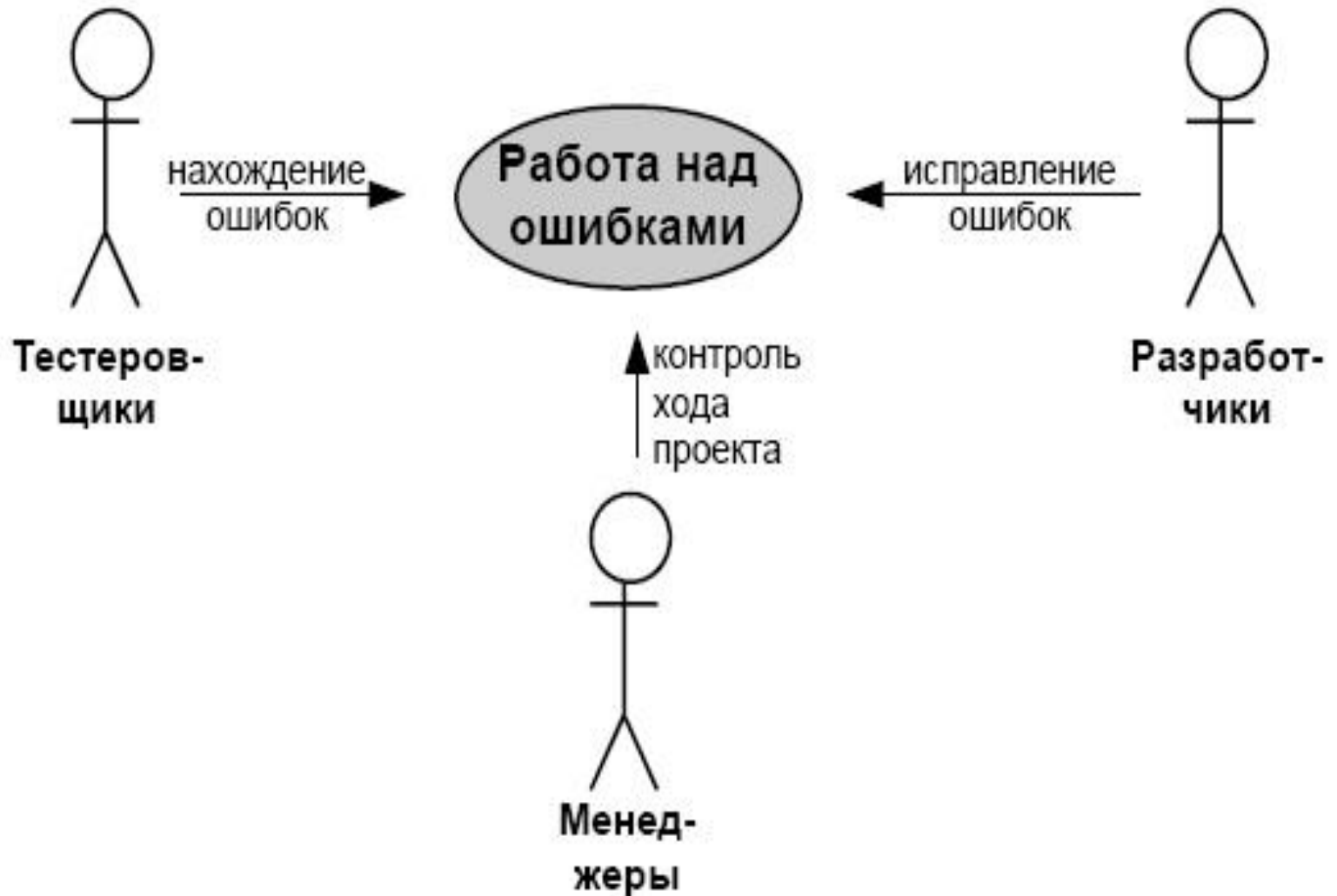
Тестирование

- **Регрессионное тестирование** – тестирование системы в процессе ее разработки и сопровождение на регресс.
- **Нагрузочное тестирование** – тестирование системы на корректную работу с большими объемами данных.
- **Стрессовое тестирование** – тестирование системы на устойчивость к непредвиденным ситуациям. Этот вид тестирования нужен далеко не для каждой системы, так как подразумевает высокую планку качества.
- **Приемочное тестирование** – тестирование, выполняемое при приемке системы заказчиков. Более того, различные стандарты часто включают в себя наборы приемочных тестов.

Работа с ошибками

- Между программистами и тестировщиками необходим специальный интерфейс общения. Ведь ошибок находится много, их исправление требует времени, и их исправления разработчиками тестировщики должны удостовериться, что они действительно исправлены.
- Менеджерам нужна статистика по найденным и исправленным ошибкам – это хороший инструмент контроля проекта. Чтобы справиться с этим потоком информации и обеспечить необходимые в работе, удобные сервисы, существует специальный класс программных средств – **средства контроля ошибок** (bug tracking systems).

Работа с ошибками



Работа с ошибками

- Как правило, описание ошибки в системе контроля ошибок имеет следующие основные атрибуты:
- ответственного за ее проверку – тестировщика, который ее нашел и который проверяет, что исправления, сделанные разработчиком, действительно устраняют ошибку;
- ответственного за ее исправление – разработчика, которому ошибка отправляется на исправление;
- состояние, например, ошибка найдена, ошибка исправлена, ошибка закрыта, ошибка вновь проявилась и т.д.
- Этот список существенно дополняется в различных программных средствах контроля ошибок, но это основные атрибуты.

Работа с ошибками

- Использование подобных систем стало общей практикой в разработке ПО. Они включают в себя:
- базу данных для хранения ошибок;
- интерфейс к этой базе данных для внесения новых ошибок и задания их многочисленных атрибутов, для просмотра ошибок на основе различных фильтров;
- сетевой доступ, так как проекты все чаще оказываются распределенными;
- программный интерфейс для возможностей программной интеграции таких систем с другим ПО, поддерживающим разработку ПО (например, со средствами непрерывной интеграции – они могут автоматически вносить в базу данных найденные при автоматическом прогоне тестов ошибки).