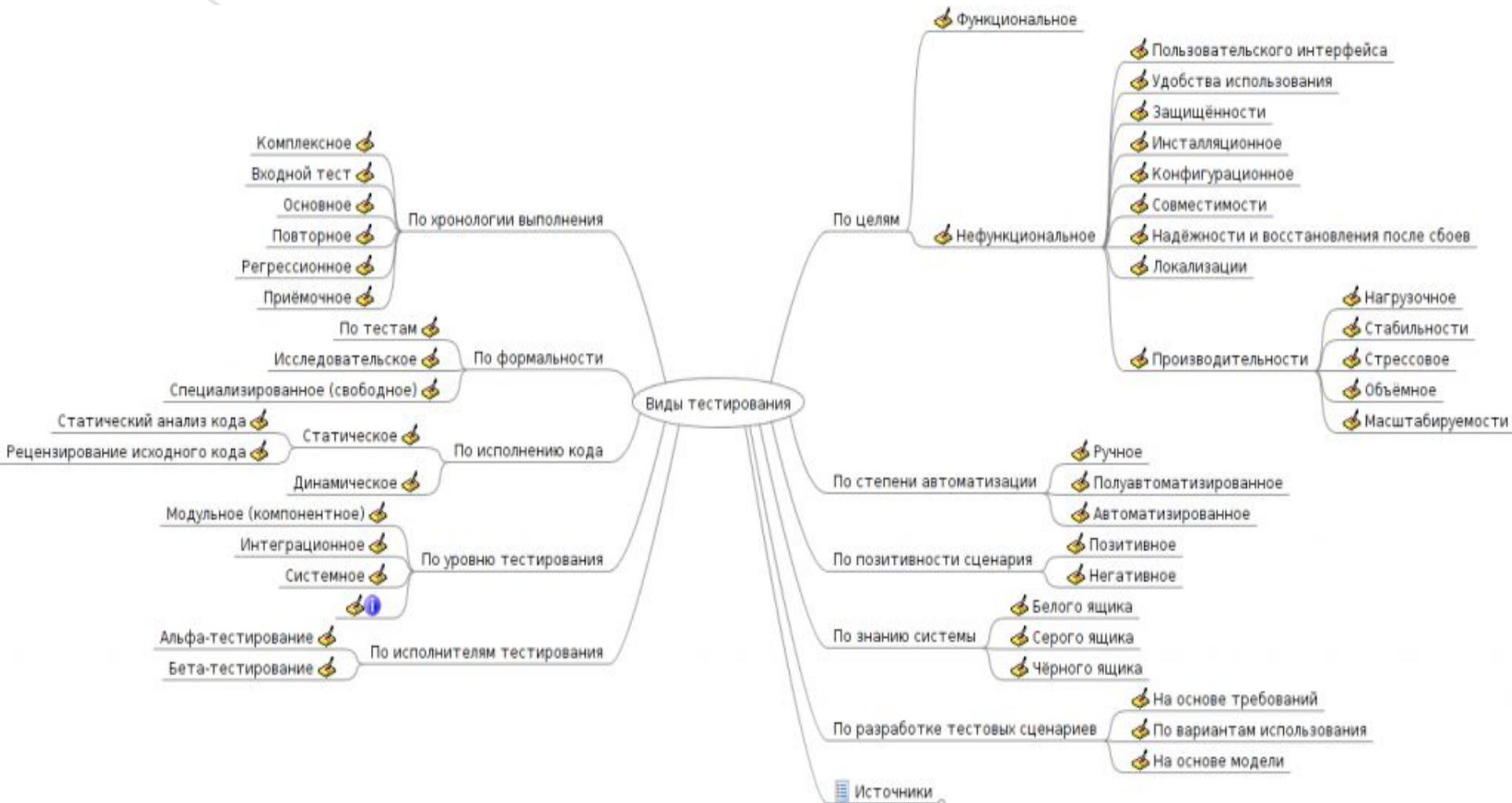


# Программа курса “Введение в тестирование ПО” (20 часов)

Октябрь - Ноябрь, 2017



# - Виды тестирования



## 4. Динамическое тестирование

- Динамическое тестирование



# - Обзор техник тестирования

## **Нефункциональные виды тестирования**

Нефункциональное тестирование описывает тесты, необходимые для определения характеристик программного обеспечения, которые могут быть измерены различными величинами. В целом, это тестирование того, "Как" система работает.

## - Обзор техник тестирования

- нагрузочное тестирование (Performance and Load Testing)
- стрессовое тестирование (Stress Testing)
- тестирование стабильности или надежности (Stability / Reliability Testing)
- объемное тестирование (Volume Testing)
- тестирование масштабируемости (Scalability Testing)

# - Обзор техник тестирования

## **Нагрузочное тестирование (Load Testing)**

или **тестирование производительности (Performance Testing)** - это **автоматизированное тестирование**, имитирующее работу определенного количества бизнес пользователей на каком-либо общем (разделяемом ими) ресурсе.

**Цель:** проверка производительности (времен отклика) на разных нагрузках.

## - Обзор техник тестирования

Термин *нагрузочное тестирование* может быть использован в различных значениях в профессиональной среде тестирования ПО. В общем случае он означает практику моделирования ожидаемого использования приложения с помощью эмуляции работы нескольких пользователей одновременно. Таким образом, подобное тестирование больше всего подходит для многопользовательских систем, чаще — использующих клиент-серверную архитектуру (например, веб-серверов). Однако, и другие типы систем ПО могут быть протестированы подобным способом. Например, текстовый или графический редактор можно заставить прочесть очень большой документ; а финансовый пакет — сгенерировать отчёт на основе данных за несколько лет.

# - Обзор техник тестирования

При этом происходит:

- измерение времени выполнения выбранных операций при определенных интенсивностях выполнения этих операций
- определение количества пользователей, одновременно работающих с приложением
- определение границ приемлемой производительности при увеличении нагрузки (при увеличении интенсивности выполнения этих операций)
- исследование производительности на высоких, предельных, стрессовых нагрузках
- оптимизация производительности приложения, включая настройки серверов и оптимизацию кода
- подбор соответствующей для данного приложения аппаратной (программной платформы) и конфигурации сервера

# - Обзор техник тестирования

The **Apache JMeter**<sup>™</sup> application is open source software, a 100% pure Java application designed to load test functional behavior and measure performance. It was originally designed for testing Web Applications but has since expanded to other test functions.

Ability to load and performance test many different server/protocol types:

- Web - HTTP, HTTPS
- SOAP / REST
- FTP
- Database via JDBC
- LDAP
- Message-oriented middleware (MOM) via JMS
- Mail - SMTP(S), POP3(S) and IMAP(S)
- MongoDB (NoSQL)
- Native commands or shell scripts
- TCP

# - Обзор техник тестирования

**loadimpact.com**



**loadstorm.com**



## - Обзор техник тестирования

### **Стрессовое тестирование (Stress Testing)**

Стрессовое тестирование позволяет проверить насколько приложение и система в целом работоспособны в условиях стресса и также оценить способность системы к регенерации, т.е. к возвращению к нормальному состоянию после прекращения воздействия стресса. Стрессом в данном контексте может быть повышение интенсивности выполнения операций до очень высоких значений или аварийное изменение конфигурации сервера. Также одной из задач при стрессовом тестировании может быть оценка деградации производительности, таким образом цели стрессового тестирования могут пересекаться с целями тестирования производительности.

# - Обзор техник тестирования

Необходимость **стресс-тестирования** диктуется следующими факторами:

- Большая часть всех систем разрабатываются с допущением о функционировании в нормальном режиме и даже в случае, когда допускается возможность увеличения нагрузки, реальные объёмы её увеличения не принимаются во внимание.
- В случае SLA-контракта (соглашения об уровне услуг) стоимость отказа системы в экстремальных условиях может быть очень велика.
- Обнаружение некоторых ошибок или дефектов в функционировании системы не всегда возможно с использованием других типов тестирования.
- Тестирования, проведенного разработчиком, может быть недостаточно для эмуляции условий при которых происходит отказ системы.
- Предпочтительнее быть готовым к обработке экстремальных условий системы, чем ожидать её отказа.

# - Обзор техник тестирования

## **Объемное тестирование (Volume Testing)**

Задачей объемного тестирования является получение оценки производительности при увеличении объемов данных в базе данных приложения, при этом происходит:

- измерение времени выполнения выбранных операций при определенных интенсивностях выполнения этих операций
- может производиться определение количества пользователей, одновременно работающих с приложением

# - Обзор техник тестирования

## **Тестирование стабильности или надежности (Stability / Reliability Testing)**

Задачей тестирования стабильности (надежности) является проверка работоспособности приложения при длительном (многочасовом) тестировании со средним уровнем нагрузки. Времена выполнения операций могут играть в данном виде тестирования второстепенную роль. При этом на первое место выходит отсутствие утечек памяти, перезапусков серверов под нагрузкой и другие аспекты влияющие именно на стабильность работы.

## - Обзор техник тестирования

### **Тестирование масштабируемости (англ. **Scalability Testing**)**

Этот вид тестирования ПО предназначен для проверки его способности по увеличению и уменьшению масштаба любых его нефункциональных возможностей. При этом приложение должно быть способно выполнять пользовательскую нагрузку, поддерживать необходимое количество транзакций и объем данных.

## - Обзор техник тестирования

### **Тестирование Установки или Installation Testing**

Тестирование установки направленно на проверку успешной инсталляции и настройки, а также обновления или удаления программного обеспечения.

В настоящий момент наиболее распространена установка ПО при помощи **инсталляторов** (специальных программ, **которые сами по себе так же требуют надлежащего тестирования**).

В реальных условиях инсталляторов может не быть. В этом случае придется самостоятельно выполнять установку программного обеспечения, используя документацию в виде инструкций или *readme* файлов, шаг за шагом описывающих все необходимые действия и проверки.

## - Обзор техник тестирования

В распределенных системах, где приложение разворачивается на уже работающем окружении, простого набора инструкций может быть мало. Для этого, зачастую, пишется план установки (**Deployment Plan**), включающий не только шаги по инсталляции приложения, но и шаги отката (**roll-back**) к предыдущей версии, в случае неудачи. Сам по себе **план установки также должен пройти процедуру тестирования** для избежания проблем при выдаче в реальную эксплуатацию. Особенно это актуально, если установка выполняется на системы, где каждая минута простоя - это потеря репутации и большого количества средств, например: банки, финансовые компании или даже баннерные сети.

## - Обзор техник тестирования

Поэтому тестирование установки можно назвать одной из важнейших задач по обеспечению качества программного обеспечения.

Именно такой комплексный подход с написанием планов, пошаговой проверкой установки и отката инсталляции, полноправно можно назвать **тестированием установки** или **Installation Testing**.

***Вопросы?***



**QA Analyst /  
Performance  
Testing**