

Нагрузочное тестирование

План лекции

1. Терминология или Введение.
2. Виды нагрузочного тестирования.
3. Принципы нагрузочного тестирования.
4. Цели нагрузочного тестирования.
5. Этапы проведения нагрузочного тестирования.

Нагрузочное тестирование

Нагрузочное тестирование (Load Testing) или **тестирование производительности** (Performance Testing) - это автоматизированное тестирование, имитирующее работу определенного количества пользователей на каком-либо общем для них ресурсе.



Терминология

- **Виртуальный пользователь (Virtual User)** - программный процесс, циклически выполняющий моделируемые операции.
- **Итерация (Iteration)** – это один повтор выполняемой в цикле операции.
- **Интенсивность выполнения операции (Operation Intensity)** - частота выполнения операции в единицу времени, в тестовом скрипте задается интервалом времени между итерациями.
- **Нагрузка (Loading)** - совокупное выполнение операций на общем ресурсе.
- **Производительность (Performance)** - количество выполняемых операций за период времени (N операций за M часов)

Терминология

- **Масштабируемость приложения (Application Scalability)** - пропорциональный рост производительности при увеличении нагрузки.
- **Профиль нагрузки (Performance Profile)** - это набор операций с заданными интенсивностями, полученный на основе сбора статистических данных либо определенный путем анализа требований к тестируемой системе.
- **Нагрузочной точкой** называется рассчитанное (либо заданное Заказчиком) количество виртуальных пользователей в группах, выполняющих операции с определенными интенсивностями.

Виды нагрузочного тестирования

- **Тестирование производительности (Performance testing)**- определение масштабируемости приложения под нагрузкой.
- **Стрессовое тестирование (Stress Testing)** позволяет проверить насколько приложение и система в целом работоспособны в условиях стресса и также оценить способность системы к регенерации.
- **Объемное тестирование (Volume Testing)** - получение оценки производительности при увеличении объемов данных в базе данных приложения.

Виды нагрузочного тестирования

- **Тестирование стабильности или надежности (Stability / Reliability Testing)** - проверка работоспособности приложения при длительном тестировании со средним уровнем нагрузки.
- **Тест на масштабируемость (Scalability Test)** оценивает способность системы увеличивать производительность пропорционально увеличению масштаба нефункциональных возможностей.

Принципы нагрузочного тестирования

1. Уникальность запросов
2. Время отклика системы
3. Зависимость времени отклика системы от степени распределённости этой системы
4. Разброс времени отклика системы
5. Точность воспроизведения профилей нагрузки

Цели нагрузочного тестирования

1. Оценка производительности и работоспособности приложения на этапе разработки и передачи в эксплуатацию.
2. Оценка производительности и работоспособности приложения на этапе выпуска новых релизов.
3. Оптимизация производительности приложения, включая настройки серверов и оптимизацию кода.
4. Подбор соответствующей для данного приложения аппаратной (программной платформы) и конфигурации сервера.

Этапы проведения нагрузочного тестирования

1. Анализ требований и сбор информации о тестируемой системе
2. Конфигурация тестового стенда для нагрузочного тестирования
3. Разработка модели нагрузки
4. Выбор инструмента для нагрузочного тестирования
5. Создание и отладка тестовых скриптов
6. Проведение тестирования
7. Анализ результатов
8. Подготовка, отправка и публикация отчета по проведенному нагрузочному тестированию

Анализ требований

- **Время отклика** (время необходимое для открытия страницы или получения ожидаемого результата)
- **Интенсивность** (число запросов в секунду)
- **Используемые ресурсы** (загрузка процессора, количество используемой памяти и т.д.)
- **Максимальное количество пользователей** (определяет число пользователей, способных работать с системой в условиях заданной конфигурации)

Конфигурация стенда

Основные причины, по которым не всегда получается продублировать конфигурацию системы на тестовом стенде:

- Сложность дублирования дорогого серверного железа для тестовых нужд
- Ограничения на использование лицензий требуемого программного обеспечения
- Закрытость архитектуры приложения со стороны заказчика по соображениям безопасности
- Трудность воссоздания или транспортировки базы данных приложения
- Сложность воссоздания требуемой архитектуры сети
- и многое другое

Разработка модели нагрузки

- список тестируемых операций
- интенсивность выполнения операций
- зависимость изменения интенсивности выполнения операций от времени

Инструменты нагрузочного тестирования

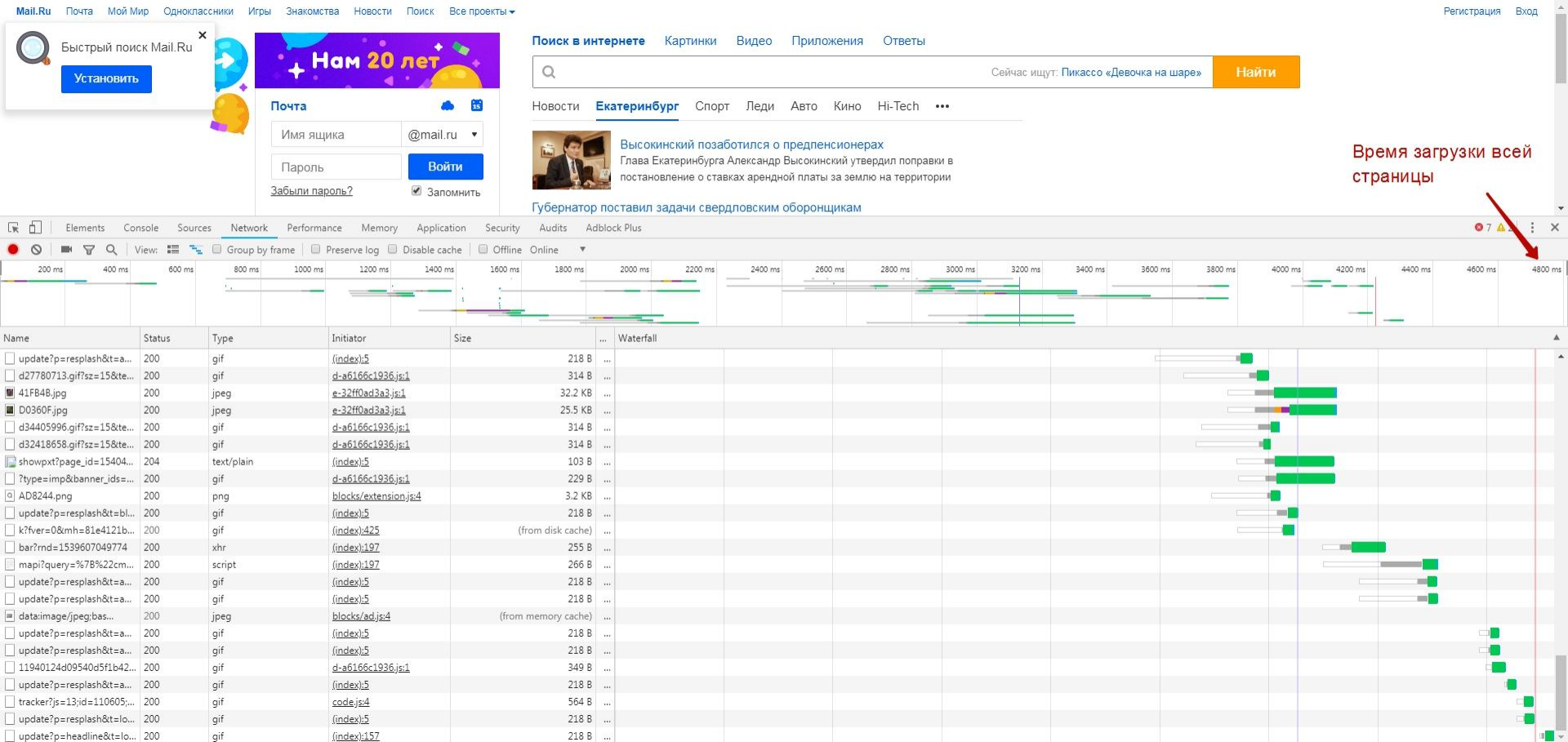
Коммерческие инструменты: Hewlett-Packard (Mercury Interactive), IBM Rational, Borland (Segue), SmartBear, Neotys, HP Performance Center (включает HP LoadRunner), Rational Performance Tester, Silk Performer, LoadComplete Web Load Testing, NeoLoad

Бесплатные инструменты: Jmeter, Grinder

Создание и отладка тестов

- внедрить приемы обеспечения качества: использование стандартов, шаблонов и инструкций
- договориться об общем использовании одинаковой структуры каталогов, архитектуры скриптов, именовании функций, переменных и транзакций

Проведение тестирования



Время загрузки всей страницы

Анализ результатов

1. Время выполнения запроса(минимальное, максимальное, среднее)
2. Количество выполненных запросов и процент ошибок
3. Пропускная способность
4. Потребление ресурсов центрального процессора
5. Потребление оперативной памяти
6. Потребление сетевых ресурсов
7. Работа с дисковой подсистемой

Подготовка, отправка и публикация отчета по проведенному нагрузочному тестированию

популярные ошибки при тестировании и интерпретации результатов теста:

- Нет четкой цели, для чего нужно тестирование данного проекта.
- Нередко проводят неправильное нагрузочное тестирование и получают ошибочный вердикт по результатам тестирования.
- Получают большое количество результатов и не могут их растолковать, не знают, что с ними делать.
- Неправильное представление результатов нагрузочного тестирования клиенту.
- DDOS тестируемой системы.