

В. С. Лебедев¹, В. В. Михаэлис¹

¹ *Иркутский государственный университет путей сообщения, г. Иркутск, Российская Федерация*

ОБЗОР ПРОГРАММНЫХ ИНСТРУМЕНТОВ ДЛЯ ПРОВЕДЕНИЯ НАГРУЗОЧНОГО ТЕСТИРОВАНИЯ ИНФОРМАЦИОННЫХ СИСТЕМ

Аннотация. В настоящей работе представлено описание понятия «нагрузочное тестирование», а также определена его цель. В работе выполнен обзор наиболее популярных на данный момент программных средств нагрузочного тестирования. Проведен анализ описанных инструментов нагрузочного тестирования для возможного дальнейшего тестирования программных продуктов. Работа выполнена в рамках научно-исследовательской работы студентов [1].

Ключевые слова: нагрузочное тестирование, анализ инструментов, информационная система, инструмент тестирования.

V.S. Lebedev¹, V.V. Mihaelis¹

¹ *Irkutsk State Transport University, Irkutsk, Russian Federation*

OVERVIEW OF SOFTWARE TOOLS FOR THE ORGANIZATION OF LOAD TESTING OF INFORMATION SYSTEMS

Abstract. This paper provides a description of the concept of "load testing", and also defines its purpose. The paper provides an overview of the currently most popular load testing software. The analysis of the described load testing tools is carried out for possible further testing of software products. The article is written within the framework of research work of students.

Keywords: load testing, tool analysis, information system, testing tool.

Введение. В настоящее время в области разработки ПО наблюдается стремительный рост. Огромное количество компаний специализируются на создании программного обеспечения высокого качества. Рост требований, которые предъявляются к качеству производимых программных продуктов, неизбежен. Одним из главных требований, которые выдвигаются к информационным системам на данный момент, является безотказное функционирование ИС под большими нагрузками. Для обеспечения требуемой производительности необходимо проводить нагрузочное тестирование (НТ).

Родоначальником идеи нагрузочного тестирования считается Кент Бек, создавший методологию разработки через тестирование. Бек также был разработчиком первых библиотек модульного тестирования SUnit для языка SmallTalk и JUnit для языка Java. Библиотека JUnit послужила прототипом для разработки библиотек модульного тестирования на других языках программирования. Совокупность этих библиотек принято называть xUnit библиотеками [2].

НТ выполняет моделирование поведения конкретного числа пользователей, одновременно работающих в системе, создавая нагрузку, идентичную фактической нагрузке. Данный вид тестирования помогает определить основные характеристики, которые требуются для корректной работы определенной информационной системы. Примером такой характеристики будет эффективная работа под максимальной нагрузкой. Также этот вид тестирования дает возможность получить информацию о поведении тестируемой системы в жестких условиях перегруженности и о возможных последствиях в результате сбоев системы из-за превышения максимально допустимой нагрузки.

Определение и цель нагрузочного тестирования. Тестирование информационной системы – это техника контроля качества продукта, которая представляет собой процесс анализа, а также проверки программного продукта с целью сравнения реального и предполагаемого поведения системы для улучшения возможного качества будущего продукта, уменьшения вероятности возникновения дефектов в ходе эксплуатации.

Нагрузочное тестирование – это автоматизированное тестирование, которое имитирует работу заданного заранее количества пользователей, создавая условия, которые идентичны фактической нагрузке на систему.

К настоящему времени тестирование превратилось в сложный процесс, непосредственно встроенный в методологию проектирования. Сейчас существуют различные виды тестирования, направленные на диагностику отдельных частей информационной системы [3].

При проведении нагрузочного тестирования, как правило, используется следующий набор показателей:

- 1) Емкость системы (максимальная нагрузка, при которой система функционирует корректно);
- 2) Время отклика (скорость, откликов на запросы пользователей, получаемых системой);
- 3) Отказоустойчивость (время восстановления после сбоя, время работы до отказа);
- 4) Доступность (время, в течение которого система находится в работоспособном состоянии);
- 5) Масштабируемость (способность системы увеличивать свою производительность).

Цель проведения нагрузочного тестирования - гарантировать технологическое качество информационной системы после развертывания информационной системы и обеспечение соответствия заявленным технологическим требованиям в процессе эксплуатации информационной системы [4].

Инструменты. На сегодняшний день на рынке программного обеспечения представлено большое количество различных инструментов, позволяющих проводить нагрузочное тестирование. Существуют не только бесплатные приложения с открытым кодом, но и дорогостоящие мощные инструменты, осуществляющие автоматизированное тестирование. Однако, даже с таким большим разнообразием представленных вариантов, не всегда легко выбрать необходимый инструмент НТ для проведения качественного тестирования конкретного программного продукта.

Согласно проведенному исследованию в статье [5] наиболее популярными на рынке являются такие инструменты нагрузочного тестирования как Apache JMeter, HP LoadRunner и Blazemeter.

1. Apache JMeter – на текущий момент является одним из наиболее популярных инструментов для нагрузочного тестирования с открытым исходным кодом. Инструмент разработан в 2003 году и за время своего развития расширил свою функциональность, а также получил широкое признание, как хороший аналог платным программным продуктам.

Инструмент разработан на Java, поэтому является кроссплатформенным. Возможна работа как в графическом режиме, так и с помощью консоли. JMeter работает с такими протоколами как HTTP, HTTPS, FTP, LDAP, SOAP, TCP.

Таблица 1.

Преимущества и недостатки инструмента Jmeter

Преимущества	Недостатки
Удобный интерфейс	Не поддерживает JavaScript
Анализ результатов и кэширование	Есть предел использования памяти, из-за

	которого возникают ошибки
Базовая версия легко расширяется установкой плагинов	Иногда возникают трудности с тестированием сложных приложений
Реализация на Java	
Графическое разделение на блоки для упрощения разработки скриптов	
Удобство работы с многопоточностью	
Визуализация данных	
Динамический ввод	

2. HP LoadRunner представляет собой комплексный инструмент нагрузочного тестирования, обнаруживающий проблемы с производительностью в больших и сложных приложениях. LoadRunner применяется для тестирования ERP-систем, технологий Web 2.0, а также устаревших системных приложений.

Данный инструмент позволяет осуществлять тестировщикам комплексный анализ производительности тестируемой системы. Основная специализация инструмента направлена на выявление слабых мест до момента, когда система будет внедрена или развернута. В конечном итоге это позволяет пользователям оценить компоненты системы по отдельности до того, как они начнут эксплуатироваться.

Кроме этого, LoadRunner может использоваться для выявления проблем с производительностью при будущих обновлениях системы. Он содержит продвинутые функции, предоставляемые пользователям, по прогнозированию возможного роста требований к производительности системы. Благодаря такому прогнозированию возможно предсказать затраты программного и аппаратного обеспечения, в результате чего достаточно просто достигается повышение масштабируемости и производительности системы [6].

Таблица 2.

Преимущества и недостатки инструмента HP LoadRunner

Преимущества	Недостатки
Обнаружение узких мест на уровне системы, конечного пользователя и кода	Высокая стоимость
Позволяет обнаружить первоначальную причину проблемы производительности	Использование большого количества памяти, в случае нехватки ресурсов аварийно завершает работу
Позволяет сократить затраты за счет исключения времени простоя, которое возникло из-за проблем с производительностью	Стоимость зависит от количества имитируемых пользователей
Позволяет тестировать на производительность устаревшие приложения	
Возможность проводить тестирование мобильных приложений	
Позволяет снизить затраты благодаря прогнозированию масштабируемости и производительности	
Уменьшает время тестирования для более быстрого ввода систем в эксплуатацию	
Позволяет отслеживать использование данного инструмента	
Возможность получать доступа к тестовым ресурсам через браузерный интерфейс	
Оптимальное распределение нагрузки	

3. BlazeMeter – это программный продукт, который позволяет выполнять полное интегрированное нагрузочное тестирование, включающее множество проверок производительности на базе облачных систем. Данный продукт совместим с JMeter и достаточно хорошо подходит для тестирования программного обеспечения разноплановой направленности (от настольных программ до продуктов мобильных операционных систем).

Инструмент позволяет загружать и настраивать свои скрипты с помощью интуитивно понятного пользовательского интерфейса, используя при этом инструменты с открытым исходным кодом, такие как Apache JMeter, Selenium, Gatling, Grinder, Locust и многие другие. Есть возможность контролировать потоки, скорость загрузки и т.д, причем все это доступно в режиме реального времени.

Таблица 3.

Преимущества и недостатки инструмента BlazeMeter

Преимущества	Недостатки
Полная совместимость JMeter	Простые не детализированные отчеты
Доступно создание тестов с числом пользователем до 1 миллиона	Высокая стоимость если необходимо проводить тестирование, имитируя более 1000 пользователей
Быстрая настройка тестов	
Запуск возможен из облака или локально	
Позволяет запускать тесты из нескольких географических местоположений	
Мобильного тестирования имитируется с реальных устройств	
Поддержка нескольких сторонних инструментов	
Наличие интеграции с наиболее популярными инструментами	
Вывод результатов тестирования в режиме реального времени	

Проведя анализ указанных выше инструментов, можно сказать, что задача выбора наилучшего инструмента для осуществления нагрузочного тестирования информационных систем на данный момент не имеет универсального решения, которое подходило бы для абсолютно всех возможных ситуаций. При поиске подходящего инструмента нагрузочного тестирования для конкретной ИС необходимо внимательно изучить все доступные варианты и в конечном итоге выбрать тот, который наиболее всего соответствует текущим потребностям и поставленным целям.

Заключение. Проверка нагрузоустойчивости необходима для того, чтобы избежать неожиданностей в функционировании ИС, а также обеспечить бесперебойную работу и снизить стоимость владения ПО на всех этапах жизненного цикла. Для выборочной или полной проверки можно использовать сотни сервисов и платформ — в том числе частично или полностью бесплатных.

Рассмотренные выше инструменты являются лишь небольшой частью арсенала тестировщика. Однако уже на приведенных примерах можно увидеть, что несмотря на длительное развитие и широкую функциональность, ни один из инструментов не предоставляет простой

возможности для расширения, программирования поведения, представления данных о проведенных тестах в различных форматах, полноценной возможности тестирования сетевых инфраструктур.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Белинская С. И., Козыревская А. В., Климова Н. А., Лучников В. А., Михаэлис В.В., Михаэлис С. И., Петрова Л. В., Черепанова А. Л. Методическое и организационное обеспечение научно-исследовательской работы студентов кафедры «Информатика» ИРГУПС // Информационные технологии и проблемы математического моделирования сложных систем. 2009. № 7. С. 154-163.
2. Лило Я.Г., Тюменцев Е.А. Повторное использование модульных тестов для организации нагрузочного, стресс-тестирования, тестирования стабильности // Математические структуры и моделирование 2011, вып. 24, с. 69–74
3. Бородин А.А., Финогеев А.Г. Основные этапы развития нагрузочного тестирования информационных систем // Теоретические и прикладные аспекты современной науки 2014, вып. 4-1, с. 89-97
4. Проведение нагрузочного тестирования [Электронный ресурс]. – URL: <https://its.1c.ru/db/metod8dev/content/5811/hdoc> (дата обращения: 15.12.2020)
5. Коткова Ю.А. Сравнение и выбор инструмента для нагрузочного тестирования веб-приложений // ACTUALSCIENCE 2016, вып. 12, с. 199-200
6. Лучшие инструменты для нагрузочного тестирования [Электронный ресурс]. – URL: <https://www.performance-lab.ru/blog/luchshie-instrumenty-dlya-nagruzochnogo-testirovaniya> (дата обращения: 15.12.2020)

REFERENCES

1. Belinskaya S. I., Kozyrevskaya A. V., Klimova N. A., Luchnikov V. A., Mikhaelis V.V., Mikhaelis S. I., Petrova L. V., Cherepanova A. L. Metodicheskoe i organizatsionnoe obespechenie nauchno-issledovatel'skoy raboty studentov kafedry «Informatika» IRGUPS // Informatsionnye tekhnologii i problemy matematicheskogo modelirovaniya slozhnykh sistem. 2009. № 7. P. 154-163.
2. Lilo YA.G., Tyumentsev Ye.A. Povtornoye ispol'zovaniye modul'nykh testov dlya organizatsii nagruzochnogo, stress-testirovaniya, testirovaniya stabil'nosti // Matematicheskiye struktury i modelirovaniye 2011, No. 24, 69–74 p.
3. Borodin A.A., Finogeyev A.G. Osnovnyye etapy razvitiya nagruzochnogo testirovaniya informatsionnykh sistem // Teoreticheskiye i prikladnyye aspekty sovremennoy nauki 2014, No. 4-1, 89-97 p.
4. Provedeniye nagruzochnogo testirovaniya [Conducting stress testing] [Electronic resource]. – URL: <https://its.1c.ru/db/metod8dev/content/5811/hdoc> (date of the application: 15.12.2020)
5. Kotkova YU.A. Sravneniye i vybor instrumenta dlya nagruzochnogo testirovaniya veb-prilozheniy // ACTUALSCIENCE 2016, No. 12, 199-200 p.
6. Luchshiye instrumenty dlya nagruzochnogo testirovaniya [The best tools for stress testing] [Electronic resource]. – URL: <https://www.performance-lab.ru/blog/luchshie-instrumenty-dlya-nagruzochnogo-testirovaniya> (date of the application: 15.12.2020)

Информация об авторах

Вадим Сергеевич Лебедев – магистрант кафедры «Информационные системы и защита информации», Иркутский государственный университет путей сообщения, г. Иркутск, e-mail: lebedevs97@yandex.ru

Владимир Вячеславович Михаэлис – к. п. н., доцент кафедры «Информационные системы и защита информации», Иркутский государственный университет путей сообщения, г. Иркутск, e-mail: mihaelis_vv@irgups.ru

Authors

Vadim Sergeevich Lebedev – master student of the Department «Information Systems and Information Security», Irkutsk State Transport University, Irkutsk, e-mail: lebedevvs97@yandex.ru

Vladimir Vyacheslavovich Mikhaelis – Ph. D., associate Professor of the Department «Information systems and information protection», Irkutsk State Transport University, Irkutsk, e-mail: mihaelis_vv@irgups.ru.

Для цитирования

Лебедев В.С., Михаэлис В.В. Обзор программных инструментов для проведения нагрузочного тестирования информационных систем // «Информационные технологии и математическое моделирование в управлении сложными системами»: электрон. науч. журн. – 2021. – №2(10). – С. 40-45 – DOI: 10.26731/2658-3704.2021.2(10).40-45 – Режим доступа: <https://ismm.irgups.ru/toma/210-2021>, свободный.. – Загл. с экрана. – Яз. рус., англ. (дата обращения: 29.04.2021)

For citations

Lebedev V.S., Mikhaelis V.V. Overview of software tools for the organization of load testing of information systems // *Informacionnye tehnologii i matematicheskoe modelirovanie v upravlenii slozhnymi sistemami: ehlektronnyj nauchnyj zhurnal* [Information technology and mathematical modeling in the management of complex systems: electronic scientific journal], 2021. No. 2. P.40-45. DOI: 10.26731/2658-3704.2021.2(10). 40-45 [Accessed 29/04/21]