

## Лекция 2

**Веб-сервер** — сервер, принимающий HTTP-запросы от клиентов, обычно веб-браузеров, и выдающий им HTTP-ответы, как правило, вместе с HTML-страницей, изображением, файлом, медиа-поток или другими данными<sup>[1]</sup>.

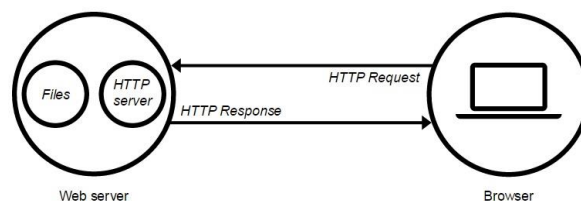
### Краткая характеристика

Понятие "Веб-сервер" может относиться как к самому серверу, как физическому хранилищу, так и к программному обеспечению.

С точки зрения железа, веб-сервер — это компьютер который хранит ресурсы сайта (HTML документы, CSS стили, JavaScript файлы и другое) и доставляет их на устройство конечного пользователя (веб-браузер и т.д.). Обычно подключен к сети Интернет и может быть доступен через доменное имя, например mozilla.org.

С точки зрения ПО, веб-сервер включает в себя некоторые вещи, которые контролируют доступ пользователей к размещенным на сервере файлам, это минимум HTTP сервера. HTTP сервер это часть ПО которая расшифровывает URL (веб-адрес) и HTTP (протокол который использует браузер для просмотра веб-страниц). Простыми словами, когда браузеру нужен файл размещенный на веб-сервере, браузер запрашивает его через HTTP. Когда запрос достигает нужного веб-сервера (железо), сервер HTTP (ПО) передает запрашиваемый документ обратно, также через HTTP.

Чтобы опубликовать веб-сайт, нужен либо статический, либо динамический веб-сервер.



Взаимодействие браузера и веб-сервера

Статический веб-сервер, или стек, состоит из компьютера (железо) с сервером HTTP (ПО). На слэнге это называется “статикой”, потому что сервер посылает размещенные на нем файлы в браузер “как есть”.

Динамический веб-сервер состоит из статического веб-сервера плюс дополнительного программного обеспечения, наиболее часто сервером приложений и базы данных. Мы называем его “динамический”, потому что сервер приложений изменяет исходные файлы перед отправкой в браузер по HTTP.

Например, для получения итоговой страницы, которую вы видите в браузере, сервер приложений может заполнить HTML шаблон данными из базы данных. Такие сайты, как MDN (Mozilla Developer Network) или Википедия состоят из тысяч веб-страниц, но они не являются реальными HTML документами, лишь несколько HTML шаблонов и

гигантские базы данных. Эта структура упрощает и ускоряет сопровождение веб-приложений и доставку контента.

## Цель и функции веб-сервера

Главная задача веб сервера принимать HTTP-запросы от пользователей, обрабатывать их. Затем выдавать HTTP-ответы, файлы, HTML-страницы.

Цель веб-сервера - обслуживать одновременно большое количество клиентов, максимально эффективно используя ресурсы.

Любой веб сервер, для удобства его использования пользователями, должен иметь удобный веб-браузер. Он передает веб серверу запросы, преобразованные в URL-адреса интернет - ресурсов.

Наряду со стандартными функциями, некоторые веб серверы имеют дополнительные. Так, к примеру, соответствующее программное обеспечение может фиксировать число обращений пользователей к тому или иному ресурсу, записывать их в отдельный журнал. А еще они могут поддерживать HTTPS, что не менее важно для защищенного соединения между сайтами и пользователями. Зачастую веб-сервер устанавливается вместе с почтовым сервером. Это позволяет пользователям быстро переходить на страничку почты прямо с сайта, нажав всего лишь на одну гиперссылку.

## Хостинг файлов

Во-первых, веб-сервер хранит файлы веб-сайта, а именно все [HTML](#) документы и связанные с ними ресурсы, включая изображения, CSS стили, JavaScript файлы, шрифты и видео.

Технически, вы можете разместить все эти файлы на своем компьютере, но гораздо удобнее хранить их на выделенном веб-сервере, который<sup>[2]</sup>:

- всегда запущен и работает
- постоянно в сети Интернет
- имеет тот же IP адрес все время (не все провайдеры предоставляют статический IP адрес для домашнего подключения)
- обслуживается на стороне

По всем этим причинам, поиск хорошего хостинг-провайдера является ключевой частью создания вашего сайта. Рассмотрите различные предложения компаний и выберите то, что соответствует вашим потребностям и бюджету (предложения варьируются от бесплатных до тысяч долларов в месяц).

## HTTP

Во-вторых, веб-сервер обеспечивает поддержку HTTP (hypertext transfer protocol). Как следует из названия, HTTP указывает, как передавать гипертекст (т.е. связанные веб-документы) между двумя компьютерами.

Протокол представляет собой набор правил для связи между двумя компьютерами. HTTP является текстовым протоколом без сохранения состояния и обладает следующими свойствами:

- Текстовый (Все команды представлены в текстовом виде и пригодны для восприятия человеком) .
- Не сохраняет состояние(ни клиент, ни сервер, не помнят о предыдущих соединениях. Например, опираясь только на HTTP, сервер не сможет вспомнить введенный вами пароль или на каком шаге транзакции вы находитесь. Для таких задач, вам потребуется сервер приложений.)

HTTP задает строгие правила, как клиент и сервер должны общаться. Мы рассмотрим непосредственно HTTP далее в техническом разделе. Вот некоторые из них:

1. Только клиенты могут отправлять HTTP запросы, и только на сервера. Сервера отвечают только на HTTP запросы клиента.
2. Когда запрашивается физический файл, клиент должен сформировать URL для сервера.
3. Веб-сервер должен ответить на каждый HTTP запрос, по крайней мере с сообщением об ошибке.

На веб-сервере, HTTP сервер отвечает за обработку входящих запросов и ответ на них:

- При получении запроса, HTTP сервер сначала проверяет существует ли ресурс по данному URL.
- Если это так, веб-сервер отправляет содержимое файла обратно в браузер. Если нет, сервер приложений создает необходимый ресурс.
- Если это не возможно, веб-сервер возвращает сообщение об ошибке в браузер, чаще всего “404 Not Found”. (Это ошибка настолько распространена, что многие веб-дизайнеры тратят большое количество времени на разработку 404 страниц об ошибках.)

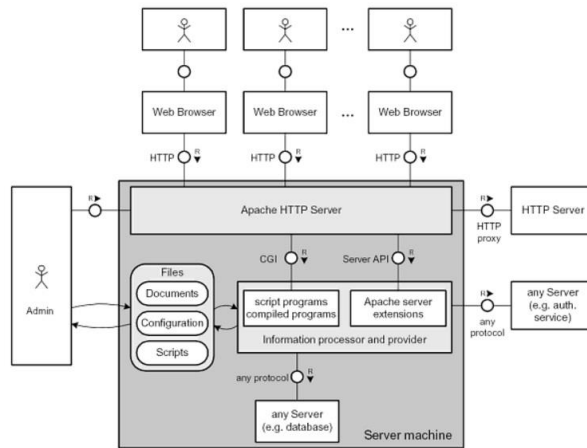
## Популярные веб-сервера

### Apache HTTP Server

Основная статья: [Apache HTTP Server](#)

[Apache HTTP Server](#) является проектом, развиваемый The Apache Software Foundation, в рамках которого разрабатывается кроссплатформенный HTTP сервер с открытым исходным кодом. Входит в состав LAMP (комплект из Linux, Apache, MySQL, PHP) и WAMP (комплект из Windows, Apache, MySQL, PHP). Другими словами, это полнофункциональный, расширяемый веб-сервер, полностью поддерживающий протокол HTTP/1.1 и распространяющийся с открытым исходным кодом<sup>[3]</sup>.

Состав архитектуры Apache HTTP Server:



Архитектура Apache HTTP Server

Ядро, написанное на языке программирования C, в чьи функциональные возможности входит:

- обработка конфигурационных файлов
- протокол HTTP
- система загрузки модулей

Система текстовой конфигурации, состоящая из трех уровней:

- Конфигурация сервера (httpd.conf).
- Конфигурация виртуального хоста (httpd.conf с версии 2.2, extra/httpd-vhosts.conf).
- Конфигурация уровня директории (.htaccess).

Мультипроцессорные модели (MPM), которые используются для работы с различными серверными операционными системами. (worker, pre-fork и др.) Система модулей для обеспечения:

- Поддержки языков программирования.
- Добавления функций.
- Исправление ошибок или модификация основных функций.
- Усиления безопасности.

Механизм виртуальных хостов, он позволяет полноценно обслуживать на одном IP-адресе множество сайтов (доменных имён), отображая для каждого из них собственное содержимое.

Веб-сервер Apache разрабатывается и поддерживается открытым сообществом разработчиков под эгидой Apache Software Foundation и включён во многие программные продукты, среди которых СУБД Oracle и IBM WebSphere. С апреля 1996 и до настоящего времени является самым популярным HTTP-сервером в Интернете.

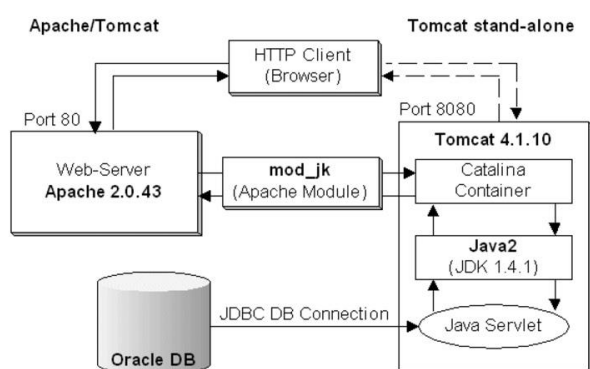
## Apache Tomcat

Основная статья: [Apache Tomcat](#)

[Apache Tomcat](#) - это контейнер, который позволяет вам использовать интернет приложения такие, как Java сервлеты и JSP (серверные страницы Java). Пакеты [Apache Tomcat 6.0](#) в Ubuntu поддерживают два варианта запуска Tomcat<sup>[4]</sup>:

1. Использовать Tomcat как классический одиночный экземпляр на всю систему, который будет запускаться при загрузке системы от имени непривилегированного пользователя tomcat6.
2. Развернуть частные экземпляры, которые будут запускаться с правами вашего собственного пользователя, и вам придется запускать и останавливать их самостоятельно.

Tomcat используется в качестве самостоятельного веб-сервера, в качестве сервера контента в сочетании с веб-сервером Apache HTTP Server, а также в качестве контейнера сервлетов в серверах приложений JBoss и GlassFish. Компоненты Tomcat:



### Архитектура Apache Tomcat

1. Catalina — контейнер сервлетов Tomcat'а, который реализует спецификацию сервлетов Servlet API. Servlet API является основой для всех остальных технологий Java, касающихся Web и дает возможность динамически генерировать любой web-контент, используя любые библиотеки, доступные для java. Архитектором Catalina являлся Craig McClanahan.
2. Coyote — компонент стека HTTP Tomcat'а, который поддерживает протокол HTTP 1.1 для веб-серверов или контейнера приложений. Coyote прослушивает входящие соединения на определенном TCP порту сервера, пересылает запросы в механизм Tomcat для обработки запросов и отправляет ответ назад запрашивающему клиенту.
3. Jasper — механизм JSP Tomcat'а. Tomcat 5.x использует Jasper 2, который является реализацией спецификации JavaServer Pages 2.0 Sun Microsystems. Jasper анализирует JSP-файлы, чтобы компилировать их в Java код, как сервлеты (которые могут быть обработаны с помощью Catalina). Во время выполнения, Jasper может автоматически обнаруживать изменения JSP-файла и перекомпилировать его. В Jasper 2, были добавлены важные особенности:
  - JSP библиотеки тегов объединения - Каждый тег разметки в файле JSP обрабатывается классом обработчика тегов. Объекты класса обработчика тега может быть объединены и использованы повторно в целом JSP сервлете.
  - Фоновая JSP компиляция - в то время как происходит перекомпиляция измененного JSP Java-кода, старая версия все еще доступна для серверных

запросов. Старый JSP сервлет удаляется только когда новый JSP сервлет закончил перекомпиляцию.

- Компилятор Java JDT - Jasper 2 может использовать Eclipse, JDT (Средства разработки Java) компилятор Java вместо Apache Ant и JAVAC.

Некоторые из свободных ресурсов и объединений Apache Tomcat включают Tomcatexpert.com (a SpringSource спонсорское сообщество разработчиков и операторов, которые работают с Apache Tomcat в крупномасштабных производственных средах) и Apache Tomcat MuleSoft (который имеет учебные руководства по установке, обновлению, Настройка, мониторинг, устранение неполадок и крепления различные версии Tomcat).

## nginx

Основная статья: [nginx](#)

nginx (engine x) является HTTP-сервером изначально написанным Игорем Сысоевым. nginx также может использоваться как почтовый прокси-сервер, а также TCP/UDP прокси-сервер общего назначения<sup>[5]</sup>. Уже длительное время он обслуживает серверы многих высоконагруженных российских сайтов, таких как Яндекс, Mail.Ru, ВКонтакте и Рамблер. Согласно статистике Netcraft nginx обслуживал или проксировал 27.80% самых нагруженных сайтов в октябре 2016 года. Рабочие процессы в [Nginx](#) одновременно обслуживают множество соединений, обеспечивая их вызовами ОС (операционной системы) epoll (Linux), select и kqueue (FreeBSD).



Логотип Nginx

Данные, полученные от клиента, разбираются посредством конечного автомата. Обработку разобранного запроса осуществляет цепочка модулей, задаваемая конфигурацией. Формирование ответа клиенту происходит в буферах, которые могут указывать на отрезок файла или хранить данные в памяти. Последовательность передачи данных клиенту определяется цепочками, в которые группируются буферы. В структурном отношении HTTP-сервер Nginx разделён на виртуальные серверы, которые в свою очередь делятся на location. Виртуальному серверу или директиве можно задать порты и адреса для приёма соединений. Для location можно задать точный URI, часть URI, или регулярное выражение. Для оперативного управления памятью служат пулы, являющиеся последовательностью заранее выбранных блоков памяти. Один блок, выделяемый изначально под пул, имеет длину от 1 до 16 килобайт. Он разделён на области – занятую и незанятую. По мере заполнения последней выделение нового объекта обеспечивается образованием нового блока.

nginx считается очень быстрым HTTP сервером. Вместо Apache или совместно с ним Nginx используют, чтобы ускорить обработку запросов и уменьшить нагрузку на сервер. Дело в том, что огромные возможности, заложенные модульной архитектурой

Apache, большинству пользователей не требуются. Платить же за эту невостребованную функциональность приходится значительным расходом системных ресурсов. Для обычных сайтов, как правило, характерно явное «засилье» статических файлов (изображений, файлов стилей, JavaScript), а не скриптов. Никакого специального функционала для передачи этих файлов посетителю ресурса не требуется, так как задача весьма проста. А, значит, и веб-сервер для обработки таких запросов должен быть простым и легковесным, как Nginx. Географическая классификация клиентов по их IP-адресу производится в nginx посредством специального модуля. Система Radix tree позволяет оперативно работать с IP-адресами, занимая минимум памяти.