

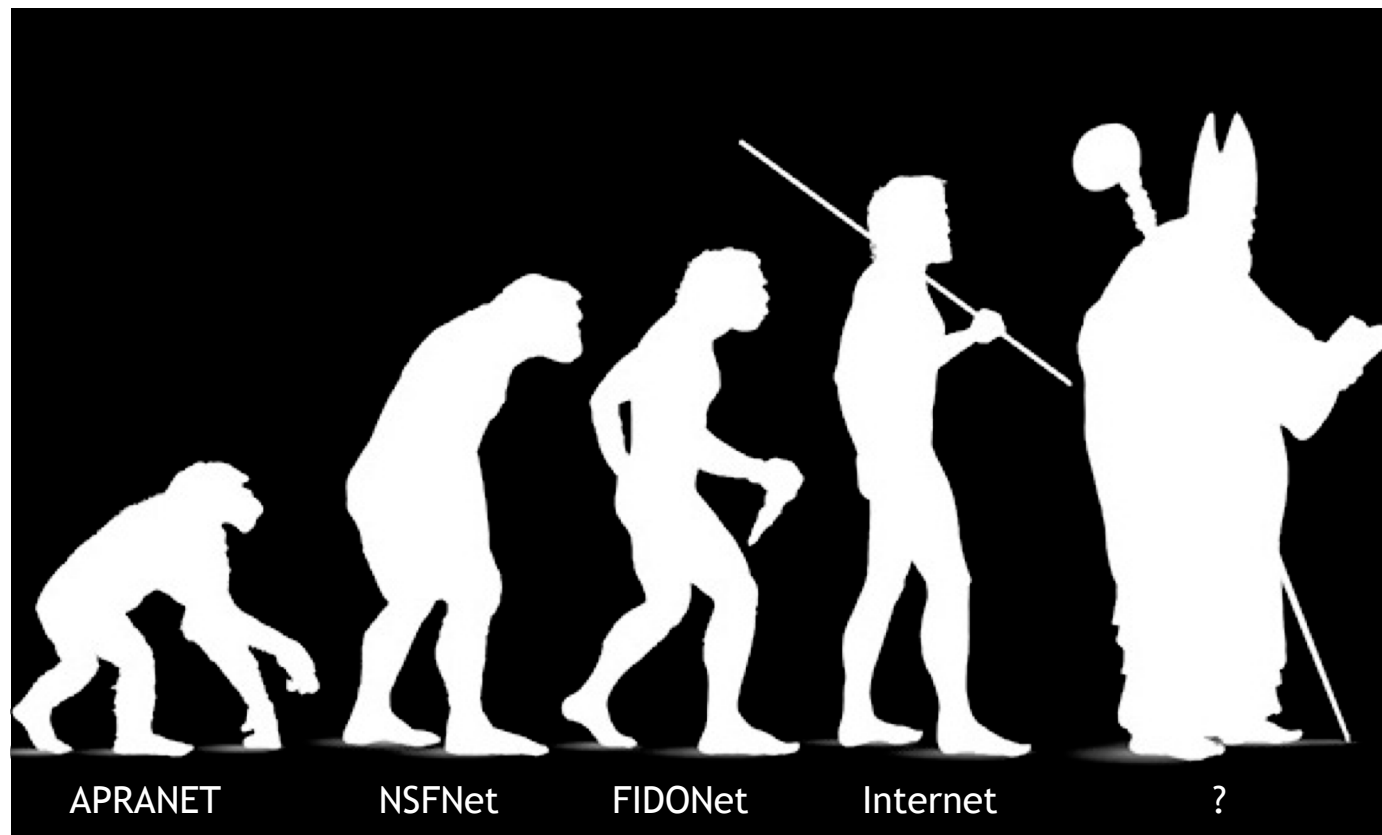
# Введение в Web

Web-программирование

# План лекции

- Введение: Интернет, WWW.
- Компоненты сервиса: URL/URI, HTML, HTTP, CGI.
- Программное обеспечение сервиса WWW: Веб-серверы, Браузеры, Поисковые роботы.
- Веб-приложение.
- Практические примеры.

# Введение



# Введение

Интернет - это глобальная компьютерная сеть, объединяющая сотни миллионов компьютеров в общее информационное пространство. Интернет представляет свою инфраструктуру для прикладных сервисов различного назначения, самым популярным из которых является Всемирная Паутина – World Wide Web (www).

World Wide Web (www, web, рус.: веб, Всемирная Паутина) — распределенная информационная система, предоставляющая доступ к гипертекстовым документам по протоколу HTTP.

WWW — сетевая технология прикладного уровня стека TCP/IP, построенная на клиент-серверной архитектуре и использующая инфраструктуру Интернет для взаимодействия между сервером и клиентом.

# Введение

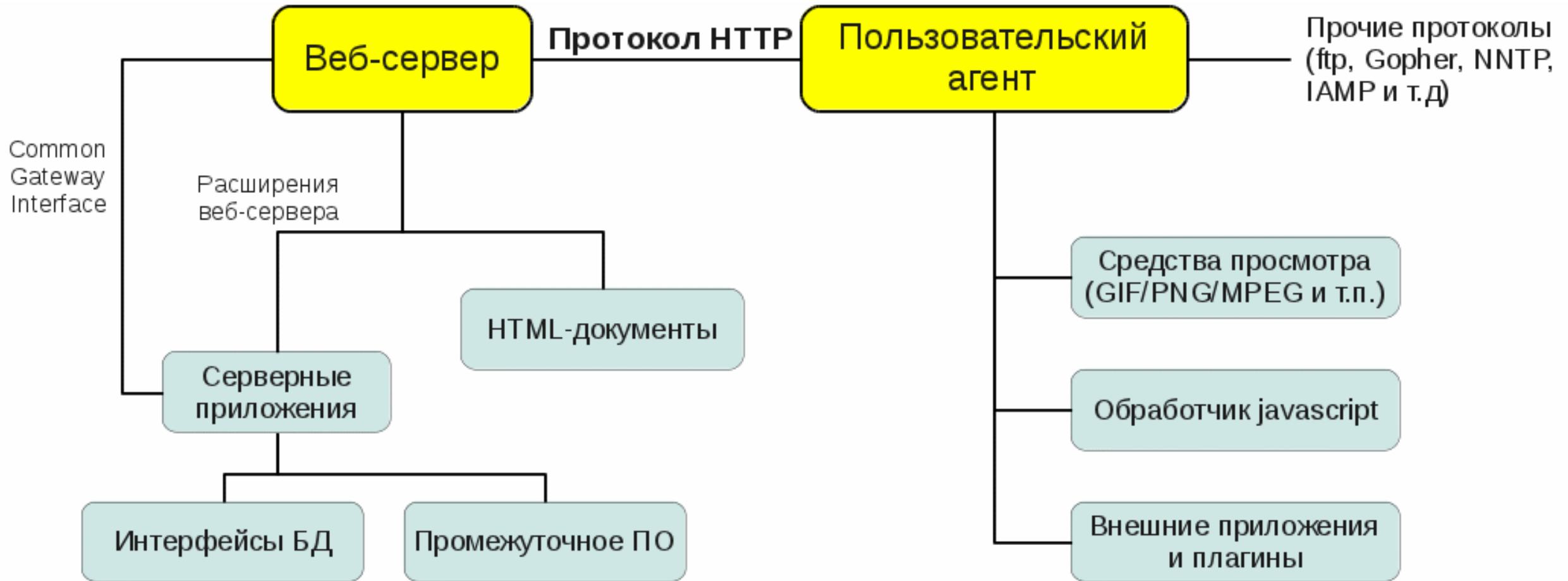


# Введение

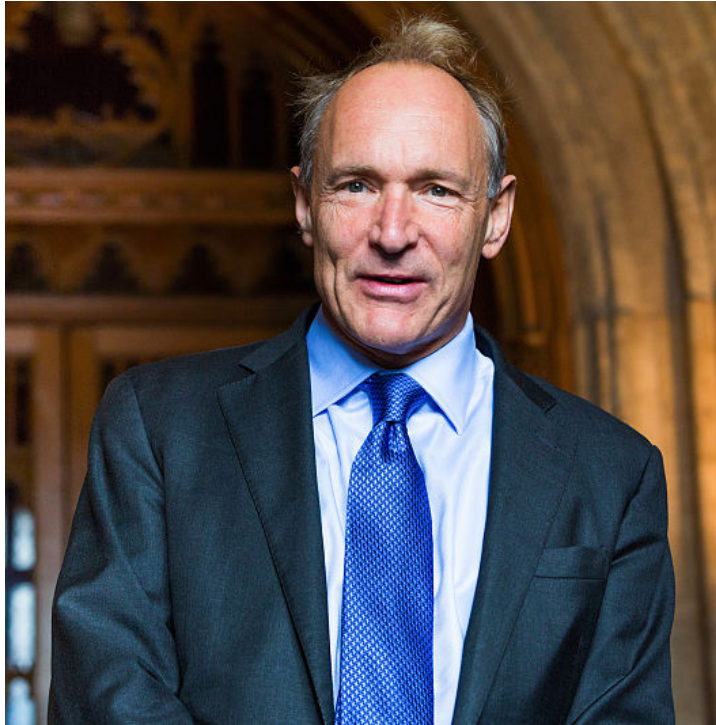
Серверы WWW (веб-серверы) — это хранилища гипертекстовой (в общем случае) информации, управляемые специальным программным обеспечением.

В основе WWW — взаимодействие между веб-сервером и браузерами по протоколу HTTP (HyperText Transfer Protocol). Веб-сервер — это программа, запущенная на сетевом компьютере и ожидающая клиентские запросы по протоколу HTTP.

# Архитектура сервиса WWW



# Введение



Сэр Тимоти Джон Бернерс-Ли — британский учёный-физик, изобретатель Всемирной паутины (совместно с Робертом Кайо), автор URI, HTTP и HTML. Действующий глава Консорциума Всемирной паутины (W3C). Автор концепции семантической паутины и множества других разработок в области информационных технологий. 16 июля 2004 года Королева Великобритании Елизавета II произвела Тима Бернерса-Ли в Рыцари-Командоры за «службу во благо глобального развития Интернета».



# Адресация веб-ресурсов. URL, URN, URI

URL (RFC 1738) — унифицированный локатор (указатель) ресурсов, стандартизированный способ записи адреса ресурса в www и сети Интернет.

**<схема>://<логин>:<пароль>@<хост>:<порт>/<полный-путь-к-ресурсу>**

- Схема - схема обращения к ресурсу: http, ftp, gopher, mailto, news, telnet, file, man, info, whatis, ldap, wais и т.п.
- логин:пароль - имя пользователя и его пароль, используемые для доступа к ресурсу.
- Хост - доменное имя хоста или его IP-адрес.
- Порт - порт хоста для подключения.
- Полный-путь-к-ресурсу - уточняющая информация о месте нахождения ресурса (зависит от протокола).

# Примеры URL

<http://example.com> #запрос стартовой страницы по умолчанию

<http://www.example.com/site/map.html> #запрос страницы в указанном каталоге

<http://example.com:81/script.php> #подключение на нестандартный порт

<http://example.org/script.php?key=value> #передача параметров скрипту

<ftp://user:pass@ftp.example.org> #авторизация на ftp-сервере

<http://192.168.0.1/example/www> #подключение по ip-адресу

<file:///srv/www/htdocs/index.html> #открытие локального файла

<gopher://example.com/1> #подключение к серверу gopher

<mailto://user@example.org> #ссылка на адрес эл.почты

# Язык гипертекстовой разметки HTML

**HTML** (HyperText Markup Language) — стандартный язык разметки документов во Всемирной паутине. Большинство веб-страниц созданы при помощи языка HTML. Язык HTML интерпретируется браузером и отображается в виде документа, в удобной для человека форме.

# Протокол HTTP

**HTTP** (HyperText Transfer Protocol) — протокол передачи гипертекста, текущая версия HTTP/1.1 (RFC 2616). Этот протокол изначально был предназначен для обмена гипертекстовыми документами, сейчас его возможности существенно расширены в сторону передачи двоичной информации.

**HTTP** — протокол прикладного уровня, но используется также в качестве «транспорта» для других прикладных протоколов, в первую очередь, основанных на языке XML (SOAP, XML-RPC, SiteMap, RSS и проч.).

# Общий шлюзовый интерфейс CGI

**CGI** (Common Gateway Interface) — механизм доступа к программам на стороне веб-сервера. Спецификация CGI была разработана для расширения возможностей сервиса `www` за счет подключения различного внешнего программного обеспечения. При использовании CGI веб-сервер представляет браузеру доступ к исполнимым программам, запускаемым на его (серверной) стороне через стандартные потоки ввода и вывода.

# Веб-серверы

**Веб-сервер** — это сетевое приложение, обслуживающее HTTP-запросы от клиентов, обычно веб-браузеров. Веб-сервер принимает запросы и возвращает ответы, обычно вместе с HTML-страницей, изображением, файлом, медиа-потокком или другими данными.

**Веб-серверы** — основа Всемирной паутины.

- Apache;
- IIS (Internet Information Services);
- Google Web Server (GWS);
- nginx [engine x].

# Браузеры

**Браузер, веб-обозреватель** (web-browser) — клиентское приложение для доступа к веб-серверам по протоколу HTTP и просмотра веб-страниц. Как правило браузеры дополнительно поддерживают и ряд других протоколов (например, ftp, file, mms, rtp).

# Поисковые роботы

**Поисковый робот** («веб-паук», краулер) — программа, являющаяся составной частью поисковой системы и предназначенная для перебора страниц Интернета с целью занесения информации о них в базу данных поисковика. По принципу действия паук напоминает обычный браузер.



# Веб-приложения

**Веб-приложение** — клиент-серверное приложение, в котором клиент взаимодействует с сервером при помощи браузера, а за сервер отвечает — веб-сервер. Логика веб-приложения распределена между сервером и клиентом, хранение данных осуществляется, преимущественно, на сервере, обмен информацией происходит по сети.



# Веб-приложения

Веб-приложение состоит из клиентской и серверной частей, тем самым реализуя технологию «клиент-сервер».

Клиентская часть реализует пользовательский интерфейс, формирует запросы к серверу и обрабатывает ответы от него.

**Технологии для разработки сервера:** ASP, ASP.NET, C/C++, Java, Perl, PHP, Python, Ruby, Nodejs.

**Технологии для разработки клиента:** HTML, XHTML, CSS, ActiveX, Adobe Flash, Adobe Flex, Java, JavaScript, Silverlight.

Практика

# HTML

```
<html>
```

```
  <head>
```

стили, заголовок, метайнформация

```
  </head>
```

```
  <body>
```

содержимое

```
  </body>
```

```
</html>
```

# CSS

```
селектор1, селектор2 {  
    свойство1: значение1;  
    свойство2: значение2;  
}
```

```
.rect {  
    width: 50px;  
    height: 50px;  
    margin-bottom: 20px;  
    background: black;  
}
```

# Box Model



# Пример: HTML + CSS

```
<!DOCTYPE html>
```

```
<html>
```

```
  <head>
```

```
    <style type="text/css">
```

```
      .rect {
```

```
        width: 50px;
```

```
        height: 50px;
```

```
        margin-bottom: 20px;
```

```
        background: black;
```

```
      }
```

```
    .wrapper {
```

```
      background: pink;
```

```
    }
```

```
  </style>
```

```
</head>
```

```
<body>
```

```
  <div class="wrapper">
```

```
    <div class="rect"></div>
```

```
    <div class="rect"></div>
```

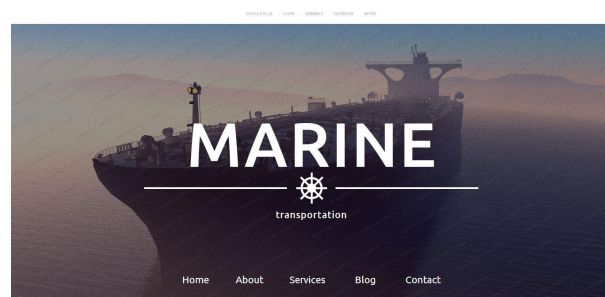
```
    <div class="rect"></div>
```

```
  </div>
```

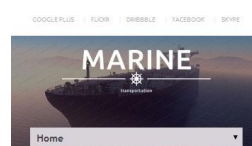
```
</body>
```

```
</html>
```

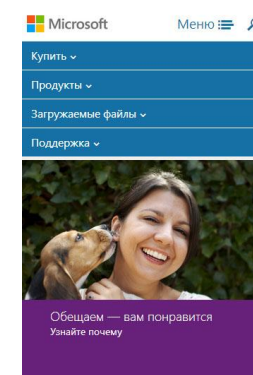
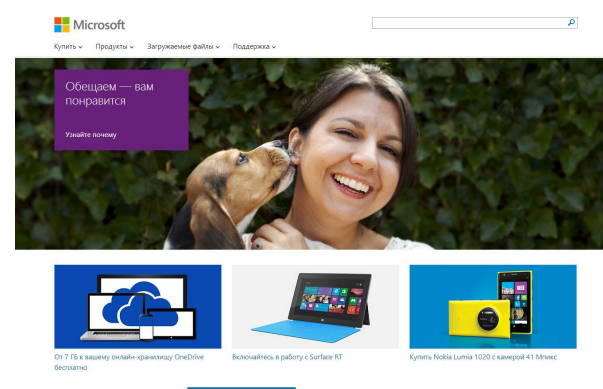
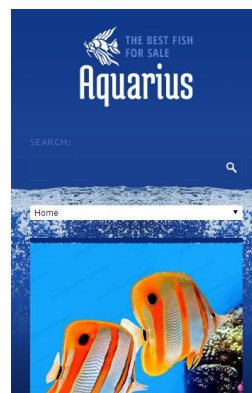
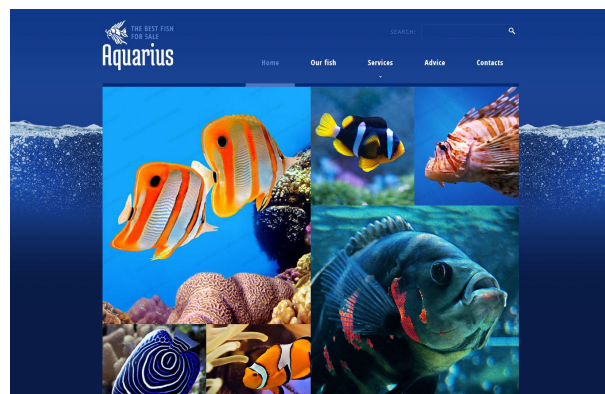
# Адаптивный дизайн



The largest transportation fleet



The largest transportation fleet





# Полезности

- Browser dev tools
- <http://htmlbook.ru>
- <https://webref.ru>
- <https://developer.mozilla.org/ru/docs/Web>
- <https://html5book.ru>
- Webstorm / VS Code
- Sublime Text / Atom / Notepad++

# Лабораторная работа 1

## Адаптивная вёрстка компоновки (wireframe) web-приложения

В решении должно быть минимум 4 блока (например: профиль, общая информация, друзья, стена), которые определённым образом распределены по странице. Пользуясь медиа-запросом, нужно реализовать адаптивную верстку. При маленьких размерах экрана блоки растягиваются или центруются.

Вся инфа будет здесь:  
[vk.com/itmoweb2019](https://vk.com/itmoweb2019)