



Клиентские и серверные веб-технологии



NetCracker®

© 2013 NetCracker Technology Corporation Confidential

План лекции

- Понятие клиентских и серверных веб-технологий.
- HTML.
- CSS.
- JavaScript.
- jQuery.
- AJAX.
- JSON.
- PHP.
- SQL.
- ASP, ASP.NET
- Java.

Понятие клиентских веб-технологий

Клиентские технологии
(browser execution):
выполняются в браузере

- HTML / CSS
- Adobe Flash
- JavaScript
- AJAX
- JSON
- Java

Серверные технологии (server execution):
обрабатывает сервер и
возвращает клиенту только
HTML-код.

- SQL
- PHP
- ASP
- ADO
- VBScript
- Java

Понятие клиентских веб-технологий

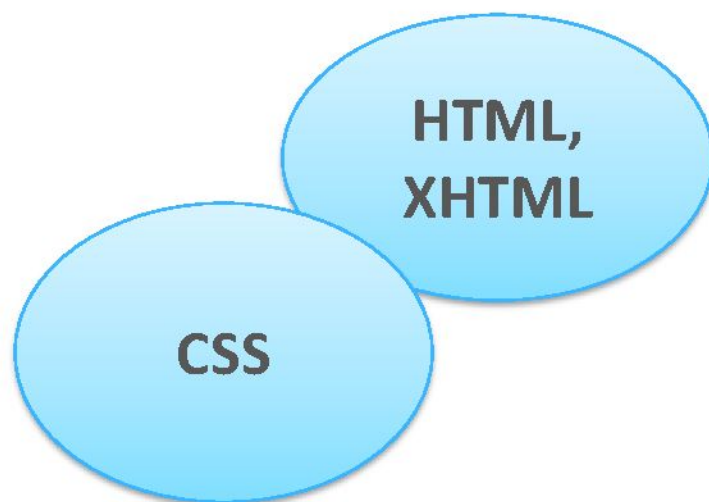
На стороне сервера используются технологии и любые языки программирования, способные осуществлять вывод в стандартную консоль.

Название	Лицензия	Веб-сервер
ASP	проприетарная	специализированный
ASP.NET	проприетарная	специализированный
C/C++	свободная	практически любой
Java	свободная	множество, в том числе свободных
Perl	свободная	практически любой
PHP	свободная	практически любой
Python	свободная	практически любой
Ruby	свободная	практически любой

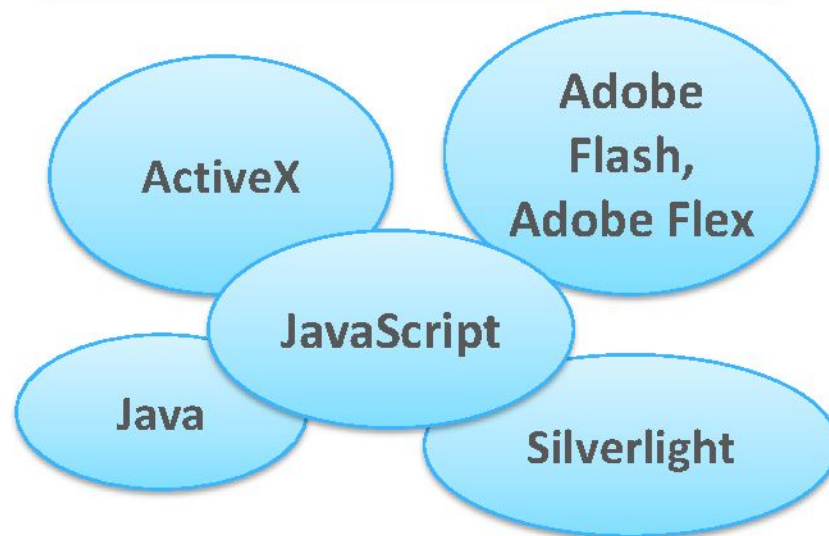
Понятие клиентских веб-технологий

На стороне клиента используются

Для реализации GUI
(Graphical user interface –
графический интерфейс
пользователя):



**Для формирования и
обработки запросов, создания
интерактивного и
независимого от браузера
интерфейса:**



HTML



HTML (HyperText Markup Language) –

язык разметки гипертекста

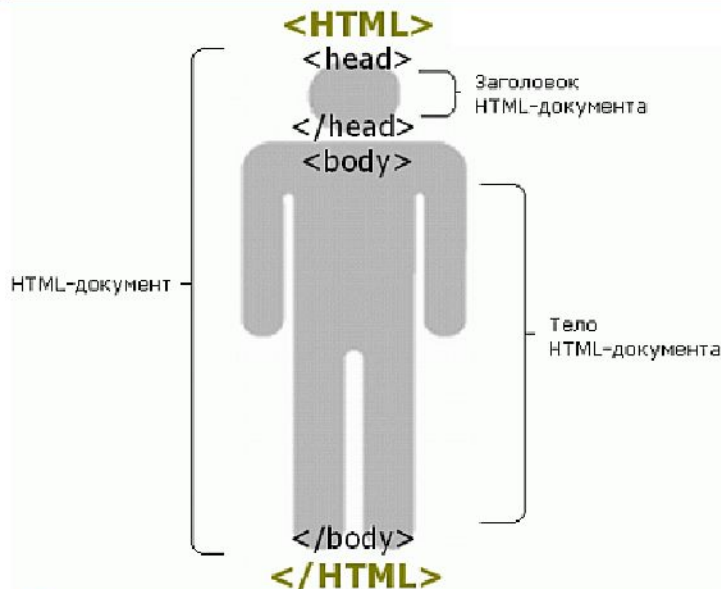
Стандартный язык разметки документов.

Теговый язык.

Система верстки веб-страниц, которая определяет, какие элементы и как должны располагаться в документе.

```
<!DOCTYPE html>
<html>
<!-- created 2010-01-01 -->
<head>
  <title>sample</title>
</head>
<body>
  <p>Voluptatem accusantium
    totam rem aperiam.</p>
</body>
</html>
```

HTML



HTML интерпретируется браузерами и отображается в виде документа в понятной для человека форме.

HTML-страница — это обычный текстовый файл, который содержит набор инструкций для интерпретации внутри браузеров.



CSS (Cascading Style Sheets) –

каскадные таблицы стилей

```
body {
  margin: 2;
  padding: 5;
  border: 1;
  width: 100%;
  background: #ffffff;
  min-width: 900px;
  font-size: 80%;
}

#header {
  border-bottom: 3px solid #aaaaaa;
}

#dimensions {
  clear: both;
  background: #eeeeee;
  border-top: 4px solid #ffffff;
  border-bottom: 4px solid #ffffff;
  margin: 4;
  padding: 6px 10px !important;
  text-align: left;
}
```

Формальный язык описания внешнего вида документа, написанного с использованием языка разметки.

Каскадные таблицы стилей дают возможность **хранить содержимое отдельно от его представления.**

Стандарт на основе текстового формата, **определяющий представление данных в браузере.**

Набор параметров форматирования, который применяется к элементам документа, чтобы **изменить их внешний вид.**



Связанные стили

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
  "http://www.w3.org/TR/html4/strict.dtd
```

Содержимое файла mysite.css

```
h1 {
  color: #000080;
  font-size: 200%;
  font-family: Arial, Verdana, sans-serif;
  text-align: center;
}
p {
  padding-left: 20px;
}
```

Глобальные стили

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
  "http://www.w3.org/TR/html4/strict.dtd">
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
    <title>Глобальные стили</title>
    <style type="text/css">
      h1 {
        font-size: 120%;
        font-family: Verdana, Arial, Helvetica, sans-serif;
        color: #333366;
      }
    </style>
  </head>
  <body>
    <h1>Hello, world!</h1>
  </body>
</html>
```

Внутренние стили

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
  "http://www.w3.org/TR/html4/strict.dtd">
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
    <title>Внутренние стили</title>
  </head>
  <body>
    <p style="font-size: 120%; font-family: monospace;
    color: #cd66cc">Пример текста</p>
  </body>
</html>
```



JavaScript

- прототипно-ориентированный сценарный язык программирования.
- скриптовый язык, предназначенный для создания интерактивных веб-страниц.



Расположение JavaScript кода внутри страницы

```
<!DOCTYPE HTML>
<html>
<body>
  <p>Начало документа...</p>
```

```
    <script>
      alert('Привет, Мир!');
    </script>
```

```
  <p>...Конец документа</p>
</body>
</html>
```

Когда браузер читает HTML-страничку, и видит `<script>` - он первым делом читает и выполняет код, а только потом продолжает читать страницу дальше.

Расположение JavaScript кода в заголовке HEAD

```
<html>
  <head>
    <script type="text/javascript">
      function count_rabbits() {
        for(var i=1; i<=3; i++) {
          alert("Из шляпы достали "+i+" кролика!")
          // оператор + соединяет строки
        }
      }
    </script>
  </head>
  <body>
    <input type="button" onclick="count_rabbits()"
      value="Считать кролей!"/>
  </body>
</html>
```

Расположение JavaScript кода во внешнем файле

```
<script type="text/javascript" src="/Путь/к/script.js"></script>
```

```
<html>
  <head>
    <script src="rabbits.js"></script>
  </head>
  <body>
    <script>
      count_rabbits();
    </script>
  </body>
</html>
```

Содержимое файла rabbits.js:

```
function count_rabbits() {
  for(var i=1; i<=3; i++) {
    alert("Кролик номер "+i)
  }
}
```

Библиотека JavaScript, фокусирующаяся на взаимодействии JavaScript и HTML



Возможности

- Движок кроссбраузерных CSS-селекторов Sizzle, выделившийся в отдельный проект;
- Переход по дереву DOM, включая поддержку XPath как плагина;
- События;
- Визуальные эффекты;
- AJAX-дополнения;
- JavaScript-плагины.

Отделяет поведение от структуры HTML

Упрощает и ускоряет написание JavaScript кода

Содержит команды позволяющие создавать анимацию и обработчики событий

```
$(document).ready(function() {  
    $(".button").click(function() {  
        $(".button").hide();  
        $("#wrap1").addClass("add");  
        $("#wrap1").animate({height:280},2000);  
        $("#wrap1").animate({width:400},2000);  
        $("#wrap1").animate({padding:20},2000,function() {  
            $("#text1").hide(2000,function() {$("#text2").show(2000)});  
        });  
    });  
});
```

Добавление jQuery на страницу

1

Скачать ее с официального сайта: <http://jquery.com/>

2

Добавить ее на страницу. Для этого следующий код должен быть добавлен на страницу в секцию head:

```
<script type="text/javascript" src="путь_к_скачанному_файлу/jquery.js">  
</script>
```

3

Для того, чтобы использовать jQuery удаленно просто добавьте на страницу в секцию head следующий код:

```
<script type="text/javascript"  
src="http://ajax.googleapis.com/ajax/libs/jquery/1.5/jquery.min.js"></script>
```

AJAX

Подход к построению интерактивных пользовательских интерфейсов веб-приложений, заключающийся в «фоновом» обмене данными браузера с веб-сервером.

В результате, при обновлении данных веб-страница не перезагружается полностью, и веб-приложения становятся быстрее и удобнее.



Не новый язык программирования или разметки, AJAX — не самостоятельная технология, а **концепция использования нескольких смежных технологий** - эффективный способ совместного использования **HTML, CSS, JavaScript и DOM**.

AJAX использует

HTML
в качестве
"каркаса"

CSS
для
оформления

DOM
для извлечения или
изменения
информации на
странице

**Объект
XMLHttpRequest**
для асинхронного
обмена данными с
сервером

JavaScript
для связи
перечисленных
технологий между
собой

JSON

Текстовый формат обмена данными, основанный на JavaScript и обычно используемый именно с этим языком. Легко читается людьми.

JSON

JavaScript Object Notation

Может использоваться с AJAX вместо XML. Поддержка формата JSON встроена в JavaScript.

```
{
  "firstName": "Иван",
  "lastName": "Иванов",
  "address": {
    "streetAddress": "Московское ш., 101",
    "city": "Ленинград",
    "postalCode": 101101
  },
  "phoneNumbers": [
    "812 123-1234",
    "916 123-4567"
  ]
}
```

За счёт своей лаконичности по сравнению с XML, формат JSON может быть более подходящим для сериализации сложных структур.

Уместен в задачах обмена данными как между браузером и сервером (AJAX), так и между самими серверами (программные HTTP-интерфейсы).

JSON

JSON

JSON или XML?

XML

- Простота использования;
- Упрощенный синтаксис обеспечивающий легкость прочтения кода.

- Поддерживается почти всеми языками программирования;
- Расширяем.

```
{
bookbase: [
{
title:"Гарри Поттер и философский камень",
author:"Джоан Ролинг",
cost:500,
quantity:15
},
.....
]
}
```

```
<?xml version="1.0" encoding="UTF-8" ?>
<bookstore>
<book id='1'>
<title>Гарри Поттер и
философский камень </title>
<author>Джоан Ролинг</author>
<cost>500</cost>
<quantity>15</quantity>
</book>
.....
</bookstore>
```



PHP (Hypertext Preprocessor) — скриптовый язык программирования общего назначения, интенсивно применяемый для разработки веб-приложений.

Мощный инструмент для **быстрого создания динамических и интерактивных веб-страниц.**

Один из популярных сценарных языков (наряду с JSP, Perl и языками, используемыми в ASP.NET) благодаря своей **простоте, скорости выполнения, богатой функциональности, кроссплатформенности и распространению исходных кодов на основе лицензии PHP.**

```
<!DOCTYPE html>
<html>
<body>

<?php
echo "My first PHP script!";
?>

</body>
</html>
```

Output Result:

My first PHP script!

PHP файл:

PHP файл может **содержать** текст, HTML, CSS, JavaScript, и PHP код.

PHP код **исполняется на сервере**, а **результат** возвращается в браузер в виде обычного **HTML документа**.

Расширение файлов «.php»

PHP скрипт начинается с обозначения **<?php** и заканчивается **?>**:

```
<?php  
// PHP code goes here  
?>
```



SQL (Structured Query Language — структурированный язык запросов)

универсальный компьютерный язык, применяемый для создания, модификации и управления данными в реляционных базах данных.

- Информационно-логический язык.
- SQL нельзя назвать языком программирования.
- SQL остаётся единственным механизмом связи между прикладным программным обеспечением и базой данных. В то же время, современные СУБД, а, также, информационные системы, использующие СУБД, предоставляют пользователю развитые средства визуального построения запросов.

Преимущества

- Независимость от конкретной СУБД
- Наличие стандартов
- Декларативность

Недостатки

- Несоответствие реляционной модели данных
- Сложность
- Отступления от стандартов
- Сложность работы с иерархическими структурами

ASP, ASP.NET



ASP (Active Server Pages) —
активные серверные страницы



Первая технология компании Microsoft, позволяющая **динамически создавать веб-страницы на стороне сервера.**

Работает на операционных системах линейки **Windows NT** и на веб-сервере **Microsoft IIS.**

Не является языком программирования — это **технология предварительной обработки**, позволяющая **подключать программные модули во время процесса формирования веб-страницы.**

Относительная популярность ASP основана на **простоте используемых языков сценариев (VBScript или JScript) и возможности использования внешних COM-компонентов.**

Технология ASP получила своё развитие в виде **ASP.NET — новой технологии создания веб-приложений, основанной на платформе Microsoft .NET.**
В начале 2011 технология **ASP** заменена **ASP.NET.**



Java (Джава) — объектно-ориентированный язык программирования.

Приложения Java обычно компилируются в специальный байт-код, поэтому они **могут работать на любой виртуальной Java-машине (JVM)** вне зависимости от компьютерной архитектуры.

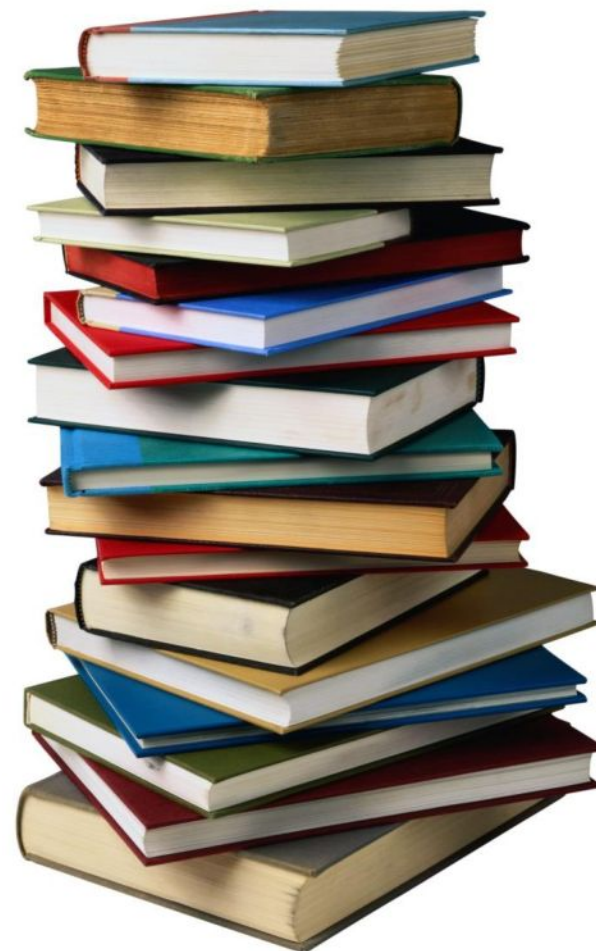
Полная независимость байт-кода от операционной системы и оборудования.

Гибкая система безопасности - исполнение программы полностью контролируется виртуальной машиной.

Любые операции, которые превышают установленные полномочия программы (например, попытка несанкционированного доступа к данным или соединения с другим компьютером) вызывают немедленное прерывание.

Дополнительные материалы

- www.google.com
- www.w3schools.com/
- <http://htmlbook.ru/>
- <http://www.intuit.ru>
- <http://www.wisdomweb.ru/>



Q&A





Thank you!

