

# Лекция

---

## Протокол HTTP – основной транспорт Web

# Приложения, использующие НТТР

- Клиентские (браузеры):

- » MS Internet Explorer
- » Netscape Communicator
- » Opera
- » Apple Safari
- » Mozilla FireFox



- Серверные (Web-серверы):

- » Apache (public domain)
- » MS Internet Information Server (IIS)
- » ...



# Терминология (Web-страницы «полезная нагрузка»)

- Web страница:
  - » адресуется **http-URL**
  - » Состоит из объектов
- В большинстве случаев это:
  - » HTML разметка («скелет»)
  - » Внедренные объекты (рисунки, ActiveX, Java апплеты, Flash ... )

The screenshot shows the CNN.com homepage. At the top, there's a navigation bar with logos for Netscape, MSN, and AOL, along with a search box and a shopping icon. The main header includes the CNN.com logo, editions for Europe and Asia, and a 'Hasse-free Shopping' banner. A date stamp indicates the page was updated on 05:43 p.m. EST (2243 GMT) on 9 February 2002. The main content area is dominated by a large photo of an athlete celebrating a gold medal, with the headline 'First gold' and a sub-headline about the 19th Winter Olympiad. To the right, there's a section for 'BRITAIN'S PRINCESS MARGARET DIES' with a small portrait and a link to the full story. Below this, a 'OTHER TOP NEWS' section lists several international headlines, including 'Iran issues new warning to U.S.', 'Blowing snow smother Plains states', and 'France calls for declaration of Palestinian state'. At the bottom, there are sections for 'QUICK NEWS', 'CNN TV', and 'BUSINESS' with various market updates.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html lang="en">
<head>
  <meta http-equiv="content-type" content="text/html; charset=iso-8859-1">
  <title>CNN.com</title>
  <meta http-equiv="refresh" content="1800; URL=http://www.cnn.com/?">
  <link rel="StyleSheet" href="http://i.cnn.net/cnn/virtual/2001/style/main.css"
type="text/css">
  <script language="JavaScript1.1"
src="http://i.cnn.net/cnn/virtual/2000/code/main.js"
  type="text/javascript"> </script>
  <script language="JavaScript1.1" type="text/javascript"> </script>
<script language="JavaScript1.1"
src="http://ar.atwola.com/file/adsWrapper.js"></script>
<style type="text/css"></style>
<script language="JavaScript">document.adoffset=0</script>
</head>
```

# Терминология (Http – URL)

## URLs (Universal Resource Locators)

---

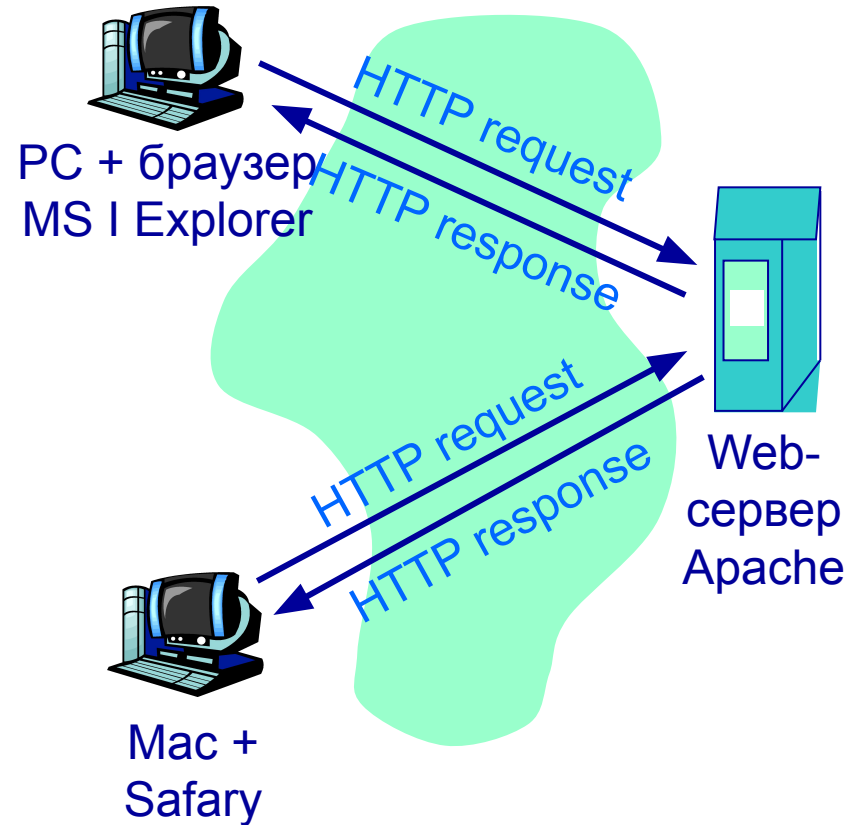


- Компоненты http-URL
  - » Адрес хоста
  - » (порт, не обязательно)
  - » Путь к ресурсу
  - » Доп. параметры (например, параметры серверного сценария)

# Терминология

## HyperText Transfer Protocol (HTTP)

- Протокол уровня прилож.
- Модель клиент/сервер
  - » *клиент*: браузер; запрашивает и отображает результат запроса
  - » *сервер*: формирует и отправляет ответы на запросы
- HTTP/1.0: RFC 1945
- HTTP/1.1: RFC 2616



# HyperText Transfer Protocol

## особенности

---

- HTTP использует TCP sockets
  - » Браузер уст. TCP соединение (на указанный порт 80)
- HTTP сообщения (уровня приложений) форма взаимодействия
- HTTP/1.0: RFC 1945
  - » Каждая пара запрос/ответ interaction per connection
- HTTP/1.1: RFC 2616
  - » Постоянное соединение для многих взаимодействий
- HTTP не имеет «СОСТОЯНИЯ»
  - » Не предусмотрены «сессии»

aside

- Это сделано для упрощения!
  - » Нужно хранить предысторию
  - » Уметь распознавать фатальные сбои; сохранять и восстанавливать сессию

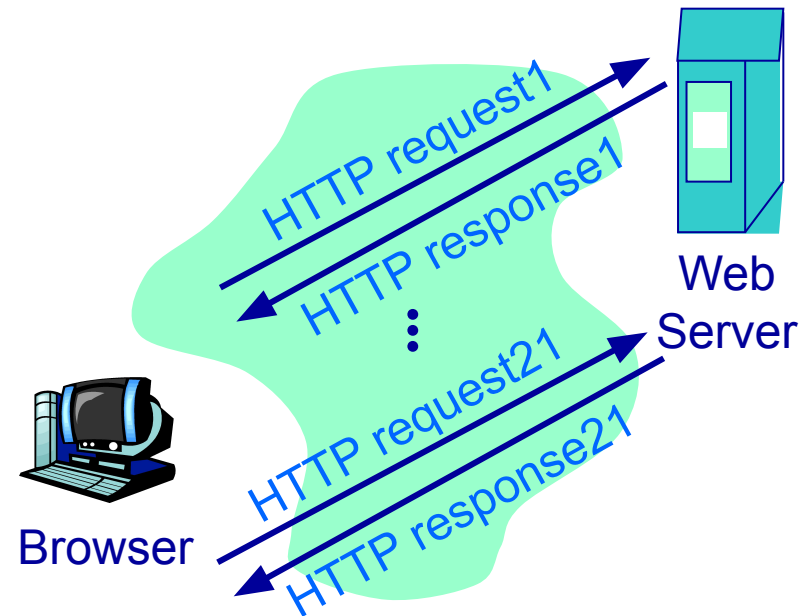
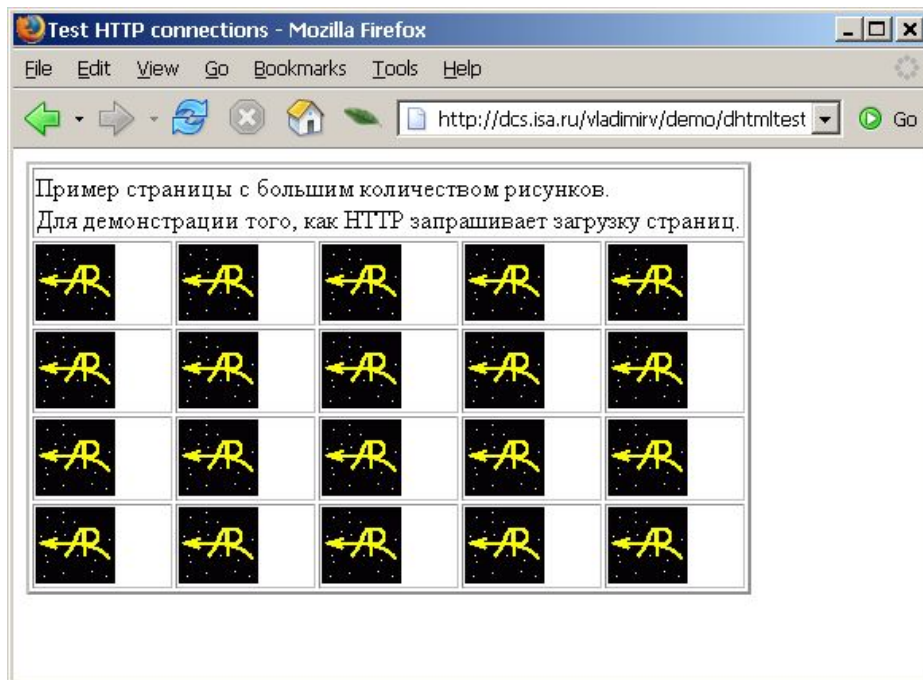
# Hypertext Transfer Protocol

## HTTP example

- URL

`http://dcs.isa.ru/vladimirv/demo/dhtmltest/TestHTTP.html`

- » Страница содержит текст HTML и 20 JPEG и GIF картинок. В итоге – 21 пар запрос/ответ (HTTP).



# HTTP 1.0 Example

**URL** <http://dcs.isa.ru/vladimirv/demo/dhtmltest/TestHTTP.html>

Client

Server

1) Браузер устанавливает TCP соединение **dcs.isa.ru**.

Исп. порт 80 по умолчанию

0) Сервер ждет запросов

3) Клиент посылает сообщение HTTP GET

2) Сервер открывает соединение

Установка  
TCP соедин.

4) Обрабатывает запрос и отправляет HTTP OK с содержимым

5) Сервер закрывает TCP соединение

time



# Нурпертекс Трансфер Протокол

## Формат сообщений НТТР

- Два типа НТТР сообщений: *request* and *response*
  - » ASCII
- НТТР request message:

- » Строка запроса
- » Необяз. Поля заголовков

method <SP> path <SP> version

<CR><LF>

имя заголовка ":" значение <CR><LF>

имя заголовка ":" значение

<CR><LF>

- » «Тело запроса», для POST

<CR><LF>

entity body

# Hypertext Transfer Protocol

## Формат сообщений HTTP

---

- Ответное сообщение
  - » ASCII, данные (binary)
- Структура ответа:

- » Строка  
ответа
- » Доп. поля
- » Результаты

version <SP> code <SP> phrase <CR><LF>

имя заголовка ":" значение <CR><LF>

⋮

имя заголовка ":" значение

<CR><LF>

<CR><LF>

HTML, двоичные данные

# HTTP Message Format

## HTTP коды ответа

---

- Трехзначное число (текст):
  - 200 OK
    - » Все в порядке
  - 301 Moved Permanently
    - » Используйте другой URL, указано в теле ответа
  - 400 Bad Request
    - » Сервер не понял запроса (формат нарушен)
  - 404 Not Found
    - » Запрашиваемый URL не найден
  - 505 HTTP Version Not Supported

# HTTP Формат сообщений

## Typical Request and Response Headers

---

Заголовки  
запроса

```
Connection: Keep-Alive
User-Agent: Mozilla/4.74 [en] (WinNT; U)
Host: buzzard.cs.unc.edu:8080
Accept: image/gif, image/x-xbitmap, image/jpeg,
        image/pjpeg, image/png, */*
Accept-Encoding: gzip
Accept-Language: en
Accept-Charset: iso-8859-1,*,utf-8
Cookie: SITESERVER=ID=8a064b785a043146e4599174a3d970
```

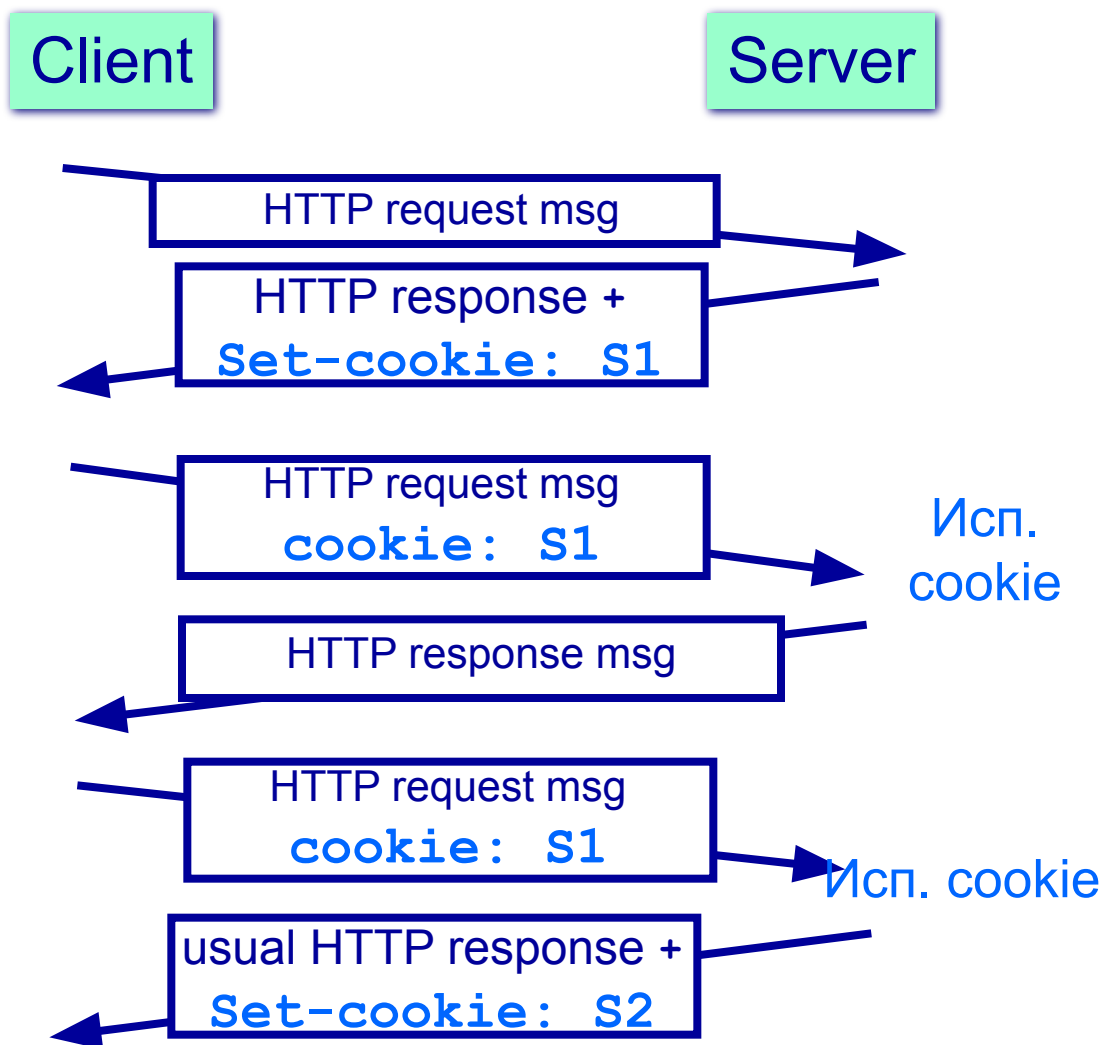
Заголовки  
ответа

```
Date: Fri, 02 Feb 2001 19:10:11 GMT
Server: Apache/1.3.9 (Unix) (Red Hat/Linux)
Last-Modified: Tue, 30 Jan 2001 21:48:14 GMT
ETag: "1807135e-67-3a77369e"
Accept-Ranges: bytes
Content-Length: 103
Connection: close
Content-Type: text/plain
```

# HTTP поддержка сессий

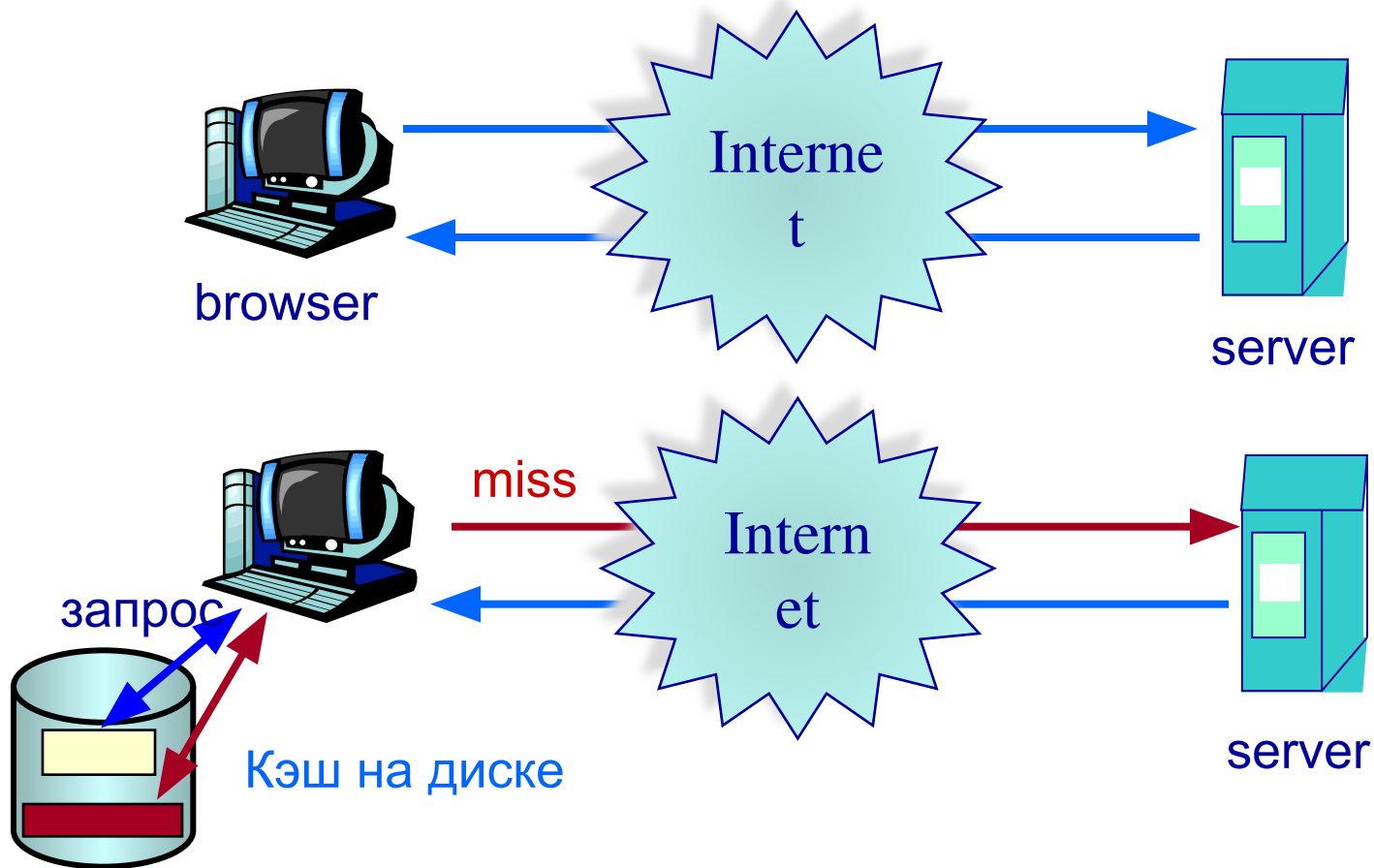
## Cookies

- Сервер шлет `Set-cookie:<value>`
- Браузер сохраняет и отправляет при след. запросах `cookie: <value>`
- Сервер исп. cookie
  - » Для аутентификации
  - » Запоминает предпочтения пользователя, имитирует сессию.



# HTTP

## Кэширование в браузере



- Браузер экономит на излишних обращениях к серверу (Last-modified параметр)

# HTTP

## Условный GET

- Указывает время кэширования

`If-modified-since:<date>`

- Сервер отправляет свежую версию, только если это нужно

- Если не нужно:

`HTTP/1.0 304 Not Modified`

