# Протокол HTTP

**HTTP** – прот.перед.гипертекста – прот.прикл.ур-ня перед.д-х (изнач.– в виде гипертекст.док-ов).

Основа – технол.«К-С», т.е. предполаг.сущ-ние К, кот. инициир. соед. и посыл.запрос, и С, кот. ожид. соед. для получ.запроса, вып.необх.д-вия и возвращ. обратно сообщ.с рез-том.

**Спецификация HTTP**

* Название: HyperText Transfer Protocol
* Уровень (по модели OSI): прикладной
* Семейство: TCP/IP
* Создан: 1990г
* Порт/ID: 80/TCP
* Назначение протокола: доступ к гипертексту, ныне стал универсальным
* Спецификация: RFC 1945, RFC 2616
* Основные реализации (клиенты): веб-браузеры (IE, Opera, Mozilla, GCh)
* Основные реализации (серверы): Apache, IIS

**HTTP** исп.в кач-ве «транспорта» для др.прот. прикл.ур( SOAP, XML-RPC..)

Обмен сообщениями идет по обыкновенной схеме «запрос-ответ». Запросы и Ответы мб конвейеризованы в соединении. Конвейерная обработка позвол.клиенту делать мн-во запросов не ожидая ответа на каждый => одиночное TCP соед, исп-ние кот намного более эффективно, теряет меньше времени.

Загрузка сети уменьш.с умен-нием числа пакетов, вызванных открытием TCP соед. => дает протоколу дост.время для опред-ния сост.загрузки сети.

Пост.соед.обеспеч.мех, согласно кот.К и С м.сообщ.о разрыве TCP соед. Это

**Достоинства HTTP**

* **Простота -** легко создавать клиентские приложения
* **Расширяемость –** благод.внедр.своих собств.заг., сохраняя совмест. с др.К и С
* **Распространенность**- различн.документац.по протоколу на многих языках мира, включение удобных в использовании средств разработки в популярные IDE, поддержка протокола в качестве клиента многими программами.

**Недостатки HTTP**

* **Большой р-р сообщ -**  возраст.нагрузка на оборуд.при формир., обработке и передаче сообщ (решение - кешировние)
* **Отсутствие «навигации»**
* **Нет поддержки распред-сти -** при выс.нагрузках на С протокол беспомощ.

**Программное обеспечение**

Все ПО для работы с протоколом делится на **3 большие категории**:

\***серверы** как осн.поставщики услуг хранения/обработки инфы(обработка запросов

\***клиенты** – конечные потребители услуг сервера (отправка запроса)

\***прокси** для выполнения транспортных служб.

**Структура протокола**

1. Стартовая строка – определяет тип сообщения
2. Заголовки – характериз.тело сообщ, пар-ры передачи и прочие сведения
3. Тело сообщ –д-е сообщ. Обяз.д.отдел.от заголовков пустой строкой.

Заголовки и тело м.отсутств., но старт.строка – обяз, т.к.указ тип запроса/отв

Старт.строка запроса: метод, uri, версия

Старт.строка ответа: версия, код состояния, пояснение

**Методы:**

\*OPTIONS – исп.для опред-ния возм-стей веб-С или пар-ров соед для рес-са.

\*GET – узнать содержимое ресурса; м. начать какой-то процесс и передать инфу

\*HEAD – анал.get. Отличие – отсутствует тело в ответе С.

\*POST – для передачи пользовательских д-х задан.ресурсу.

\*PUT – загрузки содержимого на указ.URI (не было – созд; есть – измен.)

**Доп.методы:**

\*PATCH – как put, тольео к фрагменту ресурса.

\*DELETE – удаляет указ.рес-с

\*TRACE – вовзращ.запрос так, что К м.увидеть какую инфу промеж.С доб/измен в запросе

\*LINK – устан.связь указ.рес.с др

\*UNLINK – убирает связь указ.рес.с др

**Код состояния** – число из 3 арабский цифр; первая строка ответа С.

Первая цифра – класс сост.

1хх – информационный

2хх – успешное принятие/обработка запроса К

3хх – перенаправление (для успеха – другой запрос)

4хх – ошибка клиента

5хх – ошибка сервера

**Заголовки –** строки в сообщ, содеж.пару пар-р – знач (имя – знач).

Content-type: text/plain; charset= utf-8

**Основные группы заголовков:**

1. Основные – д.включ.в люб.сообщ К и С
2. Запроса – в запросах К
3. Ответа – для ответа от С
4. Сущности – сопровожд.каждую сущность сообщ