**Protocol**

A **protocol** is a system of rules that define how data is exchanged within or between computers. Communications between devices require that the devices agree on the format of the data that is being exchanged. The set of rules that defines a format is called a protocol.

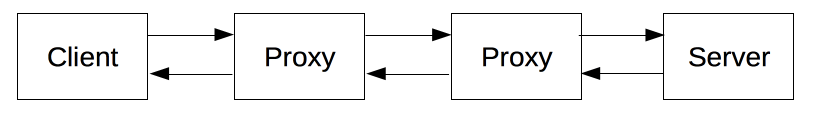
HTTP is a protocol for fetching resources(ДЛЯ ПОЛУЧЕНИЯ РЕСУРСОВ) such as HTML documents. It is the foundation of any data exchange on the Web and it is a client-server protocol, which means requests are initiated by the recipient, usually the Web browser. A complete document is reconstructed from the different sub-documents fetched, for instance, text, layout description, images, videos, scripts, and more. 

Clients and servers communicate by exchanging individual messages (as opposed to a stream of data). The messages sent by the client, usually a Web browser, are called requests and the messages sent by the server as an answer are called responses.

Designed in the early 1990s, HTTP is an extensible protocol which has evolved over time. It is an application layer protocol that is sent over TCP, or over a TLS-encrypted TCP connection, though any reliable transport protocol could theoretically be used. Due to its extensibility, it is used to not only fetch hypertext documents, but also images and videos or to post content to servers, like with HTML form results. HTTP can also be used to fetch parts of documents to update Web pages on demand.

HTTP is a client-server protocol: requests are sent by one entity, the user-agent (or a proxy on behalf of it). Most of the time the user-agent is a Web browser, but it can be anything, for example, a robot that crawls the Web to populate and maintain a search engine index.

Each individual request is sent to a server, which handles it and provides an answer called the response. Between the client and the server there are numerous entities, collectively called proxies, which perform different operations and act as gateways (ШЛЮЗЫ) or caches, for example



Прокси-сервер — это промежуточная программа или компьютер, используемый при навигации по различным сетям Интернета. Они облегчают доступ к контенту во всемирной паутине. Прокси перехватывает запросы и возвращает ответы; он может пересылать запросы или нет (например, в случае кеша), и он может изменять их (например, изменять заголовки на границе между двумя сетями).

A proxy can be on the user's local computer, or anywhere between the user's computer and a destination server on the Internet. In general there are two main types of proxy servers: 1)- A **forward proxy** that handles requests from and to anywhere on the Internet 2)- A **reverse proxy (Обратный прокси-сервер,)** taking requests from the Internet and forwarding them to servers in an internal network.

In reality, there are more computers between a browser and the server handling the request: there are routers, modems, and more. Thanks to the layered design of the Web, these are hidden in the network and transport layers. HTTP is on top, at the application layer.

### [Client: the user-agent](https://developer.mozilla.org/en-US/docs/Web/HTTP/Overview#client_the_user-agent)

The user-agent is any tool that acts on behalf of the user. This role is primarily performed by the Web browser, but it may also be performed by programs used by engineers and Web developers to debug their applications. The browser is **always** the entity initiating the request. It is never the server (though some mechanisms have been added over the years to simulate server-initiated messages).

To display a Web page, the browser sends an original request to fetch the HTML document that represents the page. It (SERVER) then parses this file, making additional requests corresponding to execution scripts, layout information (CSS) to display, and sub-resources contained within the page (usually images and videos). The Web browser then combines these resources to present the complete document, the Web page. Scripts executed by the browser can fetch more resources in later phases and the browser updates the Web page accordingly

A Web page is a hypertext document. This means some parts of the displayed content are links, which can be activated (usually by a click of the mouse) to fetch a new Web page, allowing the user to direct their user-agent and navigate through the Web. The browser translates these directions into HTTP requests, and further interprets the HTTP responses to present the user with a clear response (web page).

[**The Web server**](https://developer.mozilla.org/en-US/docs/Web/HTTP/Overview#the_web_server)

On the opposite side of the communication channel is the server, which *serves* the document as requested by the client. A server appears as only a single machine virtually; but it may actually be a collection of servers sharing the load (load balancing), or a complex piece of software interrogating other computers (like cache, a DB server, or e-commerce servers), totally or partially generating the document on demand. A server is not necessarily a single machine, but several server software instances can be hosted on the same machine. With HTTP/1.1 and the [Host](https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Host) header, they may even share the same IP address.

 Каждый запрос и каждый ответ состоит из элементов: