HTTP Extensions - **Content Negotiation**

Content Negotiation is the mechanism that is used for serving different representations, like language of a document, image format, or content encoding, of a resource at the same URI, so that the user agent or software can specify which is best suited for the user. Since not all users have the same preferences of how elements should be presented, content negotiation is basically the process of selecting the best representation for a given response when there are multiple representations available. Any response that has an entity-body may be subjected to content negotiation, including error responses.

Server-driven Content Negotiation

Also known as proactive content negotiation, the browser or any user-agent sends HTTP headers that describe the preferred choice of the user along with the URL. The server uses the headers as hints then an internal server-specific algorithm chooses the best content to serve to the client. The selection is based on the available representations of the response and the contents of particular header fields in the request message, or on other information pertaining to the request.

Request Header Fields:

1. Accept - lists the MIME types of media resources that the agent is willing to process. The lists of MIME types are comma-separated, each combined with a quality factor and a parameter indicating the relative degree of preference between the different MIME types.
2. Accept-CH - lists configuration data that can be used by the server to select an appropriate response. Valid values are:
3. Device-Memory - indicates the approximate amount of device RAM.
4. DPR - indicates the client’s device pixel ratio.
5. Viewport-Width - indicates the layout viewport width in CSS pixels.
6. Width - indicates the resource width in physical pixels.
7. Accept-Charset - indicates what kinds of character encodings are understood by the user-agent.
8. Accept-CH-Lifetime – used with the Device-Memory value of the Accept-CH header and indicates the amount of time the device should choose to share the amount of device memory with the server.
9. Accept-Encoding – defines the acceptable content-encoding supported compressions. The value is a q-factor list that indicates the priority of the encoding values. The default value identity is at the lowest priority.
10. Accept-Language – used to indicate the language preference of the user. A default values is often set according to the language of the graphical interface of the user agent, but most browsers allow to set different language preferences.
11. User-Agent – identifies the browser sending the request. This may contain a space-separated list of product tokens (name followed by a ’\’ and a version number) and comments (free string delimited by parantheses).
12. Vary – sent by the web browser in its response. It indicates the list of headers that were used by the server during the server-driven content negotiation phase. The Vary header is needed in order to inform the cache of the decision criteria so that it can reproduce it, allowing the cache to be functional while preventing serving incorrect content to the user.

Agent-driven Negotiation

This negotiation allows the server to send back a page containing links to the available alternative resources when faced by an ambiguous request. This resolves the issue of the server-driven negotiation not being able to scale well and sending of headers are done on every request. In agent-driven negotiation, the user is presented the resources and choose the one to use.

Transparent Negotiation

Transparent Negotiation is a combination of both server-driven and agent-driven negotiation. When a cache is supplied with a form of the list of available representations of the response (as in agent-driven negotiation), and the dimensions of variance are completely understood by the cache, the cache becomes capable of performing server-driven negotiation on behalf of the origin server for subsequent requests on that resource. Because of this, distributing the negotiation work and removing the second request delay of agent-driven negotiation when the cache is able to correctly guess the right response is possible.