WIKIPEDIA

Query string

An address bar on Google Chrome showing a URL with the query string `title=Query_string&action=edit`

On the World Wide Web, a **query string** is the part of a uniform resource locator (URL) which assigns values to specified parameters. The query string commonly includes fields added to a base URL by a Web browser or other client application, for example as part of an HTML form.[1]

A web server can handle a Hypertext Transfer Protocol request either by reading a file from its file system based on the URL path or by handling the request using logic that is specific to the type of resource. In cases where special logic is invoked, the query string will be available to that logic for use in its processing, along with the path component of the URL.

Contents

Structure

Typical URL containing a query string is as follows:

```
http://example.com/over/there?name=ferret
```

When a server receives a request for such a page, it may run a program, passing the query string, which in this case is, `name=ferret` unchanged, to the program. The question mark is used as a separator, and is not part of the query string.[2][3]

Web frameworks may provide methods for parsing multiple parameters in the query string, separated by some delimiter.[4] In the example URL below, multiple query parameters are separated by the ampersand, "&":

```
http://example.com/path/to/page?name=ferret&color=purple
```

The order of the queries doesn't matter (`name=ferret&color=purple` and `color=purple&name=ferret` both produce the same results) and the exact structure of the query string is not standardized. Methods used to parse the query string may differ between websites.

A link in a web page may have a URL that contains a query string. HTML defines three ways a user agent can generate the query string:

- an HTML form via the `<form>...</form>` element
- a server-side image map via the `ismap` attribute on the `<img>` element with a `<img ismap>` construction
- an indexed search via the now deprecated `<isindex>` element

Web forms

One of the original uses was to contain the content of an HTML form, also known as web form. In particular, when a form containing the fields `field1`, `field2`, `field3` is submitted, the content of the fields is encoded as a query string as follows:

```
field1=value1&field2=value2&field3=value3...
```

- The query string is composed of a series of field-value pairs.
- Within each pair, the field name and value are separated by an equals sign, "=".
- The series of pairs is separated by the ampersand, "&" (or semicolon, ";" for URLs embedded in HTML and not generated by a `<form>...</form>`. See below).

While there is no definitive standard, most web frameworks allow multiple values to be associated with a single field (e.g. `field1=value1&field1=value2&field2=value3`).[5][6]

For each field of the form, the query string contains a pair *`field`*`=`*`value`*. Web forms may include fields that are not visible to the user; these fields are included in the query string when the form is submitted

This convention is a W3C recommendation.[4] W3C recommends that all web servers support semicolon separators in addition to ampersand separators[7] to allow application/x-www-form-urlencoded query strings in URLs within HTML documents without having to entity escape ampersands.

The form content is only encoded in the URL's query string when the form submission method is GET. The same encoding is used by default when the submission method is POST, but the result is submitted as the HTTP request body rather than being included in a modified URL.[1]

Indexed search

Before forms were added to HTML, browsers rendered the `<isindex>` element as a single-line text-input control. The text entered into this control was sent to the server as a query string addition to a GET request for the base URL or another URL specified by the `action` attribute.[8] This was intended to allow web servers to use the provided text as query criteria so they could return a list of matching pages.[9]

When the text input into the indexed search control is submitted, it is encoded as a query string as follows:

```
argument1+argument2+argument3...
```

- The query string is composed of a series of arguments by parsing the text into words at the spaces.
- The series is separated by the plus sign, '+'.

Though the `<isindex>` element is deprecated and most browsers no longer support or render it, there are still some vestiges of indexed search in existence. For example, this is the source of the special handling of plus sign, '`+`' within browser URL percent encoding (which today, with the deprecation of indexed search, is all but redundant with `%20`). Also some web servers supporting CGI (e.g., Apache) will process the query string into command line arguments if it does not contain an equals sign, '`=`' (as per section 4.4 of CGI 1.1). Some CGI scripts still depend on and use this historic behavior for URLs embedded in HTML.

URL encoding

Some characters cannot be part of a URL (for example, the space) and some other characters have a special meaning in a URL: for example, the character# can be used to further specify a subsection (or fragment) of a document. In HTML forms, the character `=` is used to separate a name from a value. The URI generic syntax uses URL encoding to deal with this problem, while HTML forms make some additional substitutions rather than applying percent encoding for all such characters. SPACE is encoded as '`+`' or "`%20`".[10]

HTML 5 specifies the following transformation for submitting HTML forms with the "get" method to a web server.[1] The following is a brief summary of the algorithm:

- Characters that cannot be converted to the correct charset are replaced with HTML numeric character references[11]
- SPACE is encoded as '`+`' or '`%20`'
- Letters (`A`–`Z` and `a`–`z`), numbers (`0`–`9`) and the characters '`*`','`-`','`.`' and '`_`' are left as-is
- + is encoded by %2B
- All other characters are encoded as `%HH` hex representation with any non-ASCII characters first encoded as UTF-8 (or other specified encoding)

The octet corresponding to the tilde ("~") is permitted in query strings by RFC3986 but required to be percent-encoded in HTML forms to "`%7E`".

The encoding of SPACE as '`+`' and the selection of "as-is" characters distinguishes this encoding from RFC 3986.

Example

If a form is embedded in an HTML page as follows:

```
<form action="cgi-bin/test.cgi"  method="get">
  <input type="text" name="first" />
  <input type="text" name="second" />
  <input type="submit" />
</form>
```

and the user inserts the strings “this is a field” and “was it clear (already)?” in the two text fields and presses the submit button, the program `test.cgi` (the program specified by the `action` attribute of the `form` element in the above example) will receive the following query string:`first=this+is+a+field&second=was+it+clear+%28already%29%3F`

If the form is processed on the server by a CGI script, the script may typically receive the query string as an environment variable named `QUERY_STRING`.

Tracking

A program receiving a query string can ignore part or all of it. If the requested URL corresponds to a file and not to a program, the whole query string is ignored. However, regardless of whether the query string is used or not, the whole URL including it is stored in the server log files.

These facts allow query strings to be used to track users in a manner similar to that provided by HTTP cookies. For this to work, every time the user downloads a page, a unique identifier must be chosen and added as a query string to the URLs of all links the page contains. As soon as the user follows one of these links, the corresponding URL is requested to the server. This way, the download of this page is linked with the previous one.

For example, when a web page containing the following is requested:

```
<a href="foo.html">see my page!</a>
<a href="bar.html">mine is better</a>
```

a unique string, such as `e0a72cb2a2c7` is chosen, and the page is modified as follows:

```
<a href="foo.html?e0a72cb2a2c7" >see my page!</a>
<a href="bar.html?e0a72cb2a2c7" >mine is better</a>
```

The addition of the query string does not change the way the page is shown to the user. When the user follows, for example, the first link, the browser requests the page `foo.html?e0a72cb2a2c7` to the server, which ignores what follows `?` and sends the page `foo.html` as expected, adding the query string to its links as well.

This way, any subsequent page request from this user will carry the same query string `e0a72cb2a2c7`, making it possible to establish that all these pages have been viewed by the same user. Query strings are often used in association with web beacons.

The main differences between query strings used for tracking and HTTP cookies are that:

1. Query strings form part of the URL, and are therefore included if the user saves or sends the URL to another user; cookies can be maintained across browsing sessions, but are not saved or sent with the URL.
2. If the user arrives at the same web server by two (or more) independent paths, it will be assigned two different query strings, while the stored cookies are the same.
3. The user can disable cookies, in which case using cookies for tracking does not work. However, using query strings for tracking should work in all situations.
4. Different query strings passed by different visits to the page will mean that the pages are never served from the browser (or proxy, if present) cache thereby increasing the load on the web server and slowing down the user experience.

Compatibility issues

According to the HTTP specification:

> Various ad hoc limitations on request-line length are found in practice. It is RECOMMENDED that all HTTP senders and recipients support, at a minimum, request-line lengths of 8000 octets.[12]

If the URL is too long, the web server fails with the 414 Request-URI Too Long HTTP status code.

The common workaround for these problems is to use POST instead of GET and store the parameters in the request body. The length limits on request bodies are typically much higher than those on URL length. For example, the limit on POST size, by default, is 2 MB on IIS 4.0 and 128 KB on IIS 5.0. The limit is configurable on Apache2 using the `LimitRequestBody` directive, which specifies the number of bytes from 0 (meaning unlimited) to 2147483647 (2 GB) that are allowed in a request body.[13]

See also

- Common Gateway Interface (CGI)
- HTTP cookie
- HyperText Transfer Protocol (HTTP)
- Semantic URLs

- URI scheme
- UTM parameters
- Web beacon

References

1. [1] (https://www.w3.org/TR/html52/sec-forms.html#form-submission-algorithm) HTML5.2, W3C recommendation, 14 December 2017
2. T. Berners-Lee, W3C/MIT; R. Fielding, Day Software; L. Masinter, Adobe Systems (January 2005). "RFC 3986" (http://tools.ietf.org/html/rfc3986#section-3) "Syntax Components" (section 3).
3. T. Berners-Lee, W3C/MIT; R. Fielding, Day Software; L. Masinter, Adobe Systems (January 2005). "RFC 3986" (http://tools.ietf.org/html/rfc3986#section-3.4) "Query" (section 3.4).
4. Forms in HTML documents (http://www.w3.org/TR/REC-html40/interact/forms.html#form-content-type) W3.org. Retrieved on 2013-09-08.
5. "ServletRequest (Java EE 6)" (http://docs.oracle.com/javaee/6/api/javax/servlet/ServletRequest.html#getParameterValues(java.lang.String)) *docs.oracle.com*. 2011-02-10. Retrieved 2013-09-08.
6. "uri – Authoritative position of duplicate HTTP GET query keys" (https://stackoverflow.com/questions/1746507/authoritative-position-of-duplicate-http-get-query-keys) *Stack Overflow*. 2013-06-09. Retrieved 2013-09-08.
7. Performance, Implementation, and Design Notes (http://www.w3.org/TR/1999/REC-html401-19991224/appendix/notes.html#h-B.2.2). W3.org. Retrieved on 2013-09-08.
8. "<isindex>" (https://developer.mozilla.org/en-US/docs/Web/HTML/Element/isindex) *HTML (HyperText Markup Language)*.
9. "HTML/Elements/isindex" (http://www.w3.org/wiki/HTML/Elements/isindex). *W3C Wiki*.
10. "HTML URL Encoding Reference" (https://www.w3schools.com/tags/ref_urlencode.asp). W3Schools. Retrieved May 1, 2013.
11. The *application/x-www-form-urlencoded* encoding algorithm (https://www.w3.org/TR/html52/sec-forms.html#application-x-www-form-urlencoded-encoding-algorithm) HTML5.2, W3C recommendation, 14 December 2017
12. HTTP/1.1 Message Syntax and Routing (http://tools.ietf.org/html/rfc7230#section-3.1.1) ietf.org. Retrieved on 2014-07-31.
13. core – Apache HTTP Server (http://httpd.apache.org/docs/2.2/mod/core.html#limitrequestbody) Httpd.apache.org. Retrieved on 2013-09-08.

External links

- RFC 3986