EXHIBIT 6

EXPERT REPORT OF
PATRICK MCCLORY

IN THE MATTER OF:

PERSONALWEB TECHNOLOGIES, LLC, and
LEVEL 3 COMMUNICATIONS, LLC
adv.
TWITCH INTERACTIVE, INC.

UNITED STATES DISTRICT COURT
NORTHERN DISTRICT OF CALIFORNIA
CASE NO. 5:18-CV-05619-BLF

AUGUST 23, 2019

71. If I am being precise, the 'standard' for web servers clearly is the file node + last-modified time + size (for directly served static content, not coming from an application server).

72. It is fairly rare to have a standalone web server with no app server, especially in the 2012-2014 timeframe.

73. For specific HTTP servers, if an ETag is not present for a file, the web server will generate an ETag for the asset:

- Litespeed: file inode, last-modified time + size—see https://www.litespeedtech.com/docs/webserver/config/tuning#fileETag (Exhibit 16).
- Apache: file inode, last-modified time + size—see https://httpd.apache.org/docs/2.4/mod/core.html#fileetag (Exhibit 17).
- Nginx: file inode, last-modified time + size—see http://lxr.nginx.org/source/src/http/ngx_http_core_module.c#1582 (Exhibit 18).

74. That being said, this default HTTP server behavior happens when an ETag is not present, and one would have to look at what the application is passing to the web server for a given request.

75. There are many, many examples of file-content based ETags in PHP apps dating back many years:

- https://gist.github.com/oliworx/4951478 (Exhibit 19)
- https://css-tricks.com/snippets/php/intelligent-php-cache-control/ (Exhibit 20)

- https://stackoverflow.com/questions/18218643/php-etag-generation-using-php (Exhibit 21)

76. Additionally, a few popular PHP frameworks from 2012 also used file-content based ETags:

- Kohana uses an SHA1 of the rendered content: see https://github.com/kohana/core/blob/bdbe81afb5a09cee4269d2e2210a0d293265231a/classes/Kohana/Response.php#L650 (Exhibit 22).
- Symfony uses a weak ETag: see https://github.com/symfony/symfony/blob/2.0/src/Symfony/Component/HttpFoundation/Response.php#L674 (Exhibit 23).
- Laravel uses an MD5 of the rendered content: see https://github.com/laravel/framework/blob/5.8/src/Illuminate/Http/Middleware/SetCacheHeaders.php#L32 (Exhibit 24).
- Python has a number of toolsets, but Django and Flask are two very popular frameworks:
    - Python Django uses the response content's md5—see https://github.com/django/django/blob/master/django/utils/cache.py#L100-L103 (Exhibit 25).
    - The same Python Django framework, but from 2012, uses the response content's md5: see https://github.com/django/django/blob/4b27813198ae31892f1159d437e492f7745761a0/django/utils/cache.py#L97-L100 (Exhibit 26).

- Flask uses the Werkzeug web server and Werkzeug, by default, generates ETags via an MD5 of the response data (body)—see https://github.com/pallets/werkzeug/blob/71cf9902012338f8ee98338fa7bba50572606637/src/werkzeug/http.py#L775 (Exhibit 27).

77. Rack, the web server commonly paired with Ruby on Rails, directly digests the content body to generate ETags:

- Version 1.2: see https://github.com/rack/rack/blob/rack-1.2/lib/rack/etag.rb#L14 (Exhibit 28).
- Version 1.3: see https://github.com/rack/rack/blob/rack-1.3/lib/rack/etag.rb#L26 (Exhibit 29).
- … Version 1.6: see https://github.com/rack/rack/blob/1-6-stable/lib/rack/etag.rb#L28 (Exhibit 30).
- The latest version is 2.0: see https://github.com/rack/rack/blob/2-0-stable/lib/rack/etag.rb#L29 (Exhibit 31).

78. Apache Tomcat (Java) has historically defaulted to a weak ETag with the ability to override. Websphere (IBM) utilizes the underlying setup in their web server based on apache tomcat: see https://developer.ibm.com/answers/questions/334787/using-etag-with-websphere-portal/ (Exhibit 32).

79. The ETag workflow seems less mature in the ASP.net/IIS world with lots of ‘here is how I implemented it’ docs along with a lot of weak ETag defaults.