

# HttpCache

## Content Cache

### Limitation

Shared HTTP cache is only available for anonymous users. Logged in users will be served the same cache as for anonymous users, except for restricted access.

**For personal information display, you must use sub-requests with ESI or Hinclude.** Sub-controller would not use cache or make the cached response vary with Cookie (individual cache).

eZ Publish uses [Symfony HttpCache](#) to manage content cache, with both [expiration and validation model](#). Hence an **ETag** is computed for every content/version and sent in the Http response. It is also possible to use expiration model to get lightning fast responses.

An additional **X-Location-Id** header is added in the response for identification (see [cache purge document](#)).

## Configuration

```
ezpublish:
  system:
    my_siteaccess:
      content:
        view_cache: true      # Activates HttpCache for content
        ttl_cache: true      # Activates expiration based HttpCache for
content (very fast)
        default_ttl: 60      # Number of seconds an Http response is valid in
cache (if ttl_cache is true)
```

## Making your controller content cache aware

Sometimes you need that your controller's cache expires in the same time than a specific content (i.e. [ESI sub-requests with render twig helper](#), for a menu for instance). To be able to do that, you just need to add **X-Location-Id** header to the response object:

```
use Symfony\Component\HttpFoundation\Response;

// In a controller
// "Connects" the response to location #123 and sets a max age (TTL) of 1 hour.
$response = new Response();
$response->headers->set( 'X-Location-Id', 123 );
$response->setSharedMaxAge( 3600 );
```

- [Purge](#)