# HttpCache

## Content Cache

eZ Publish uses Symfony HttpCache to manage content "view" cache with the expiration model.

An additional **X-Location-Id** header is added in the response for identification (for details 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 );
```

### Making your controller user cache aware

If the content you're rendering depends on the user permissions, then an additional header must be set for this as explained in Context aware HTTP cache

- Purge
- Smart HttpCache clearing