In this lecture, we will discuss…

- Headers
- Status codes

Headers

- ✧ Headers – used for concurrency checks or idempotence
- ✧ Cache Management
  - process if things are out of date
- ✧ Concurrency Management
  - process if things are up to date

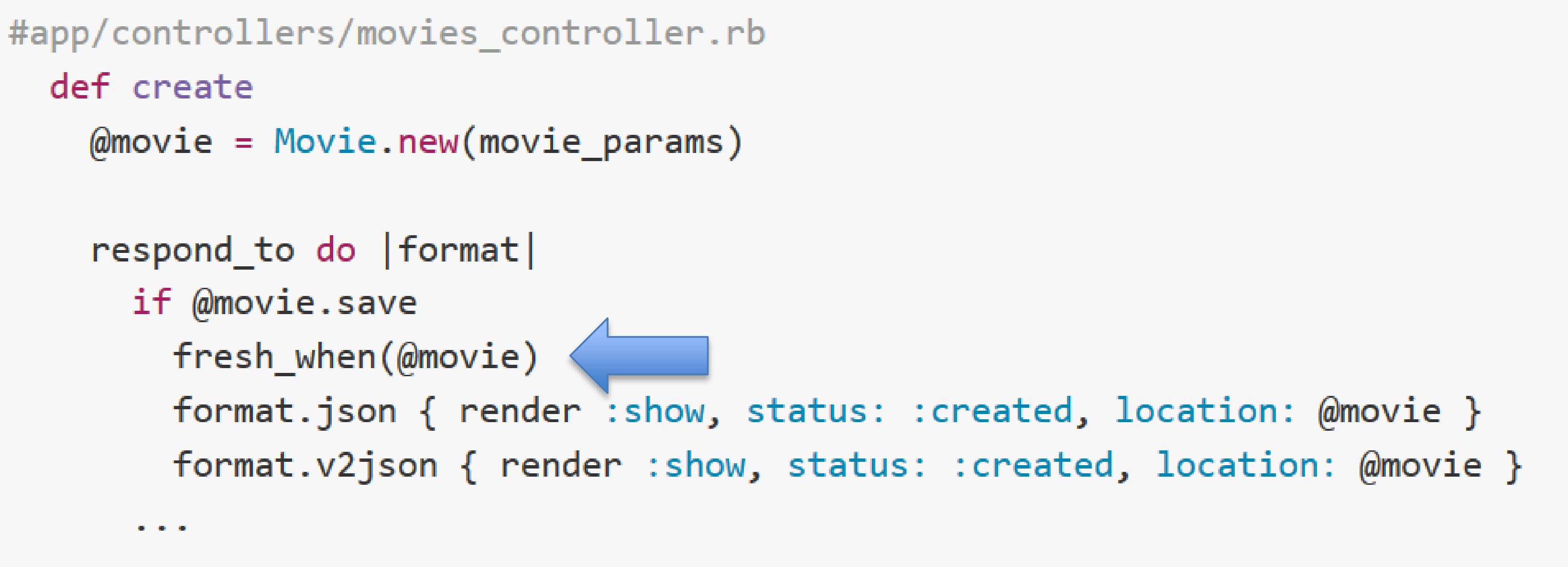
State

- `Etag` (Entity Tags)
  - mechanism used to determine whether the entity or component (images, scripts, stylesheets, page content etc) in the browser's cache matches the one on the origin server
- `Last-Modified` TimeStamp
  - Indicates the most recent modification date/time

Conditions

- Conditions
  - `If-Match`: Etag
  - `If-None-Match`: Etag
  - `If-UnModified-Since`: Last-Modified Timestamp
  - `If-Modified-Since`: Last-Modified Timestamp

Headers – fresh_when



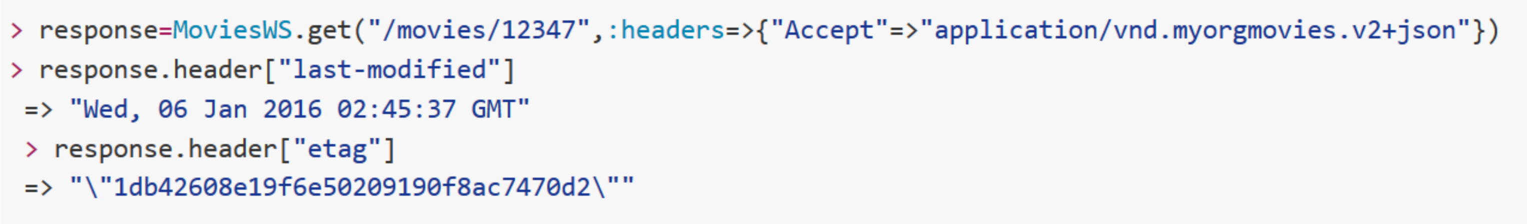
```
#app/controllers/movies_controller.rb
  def create
    @movie = Movie.new(movie_params)

    respond_to do |format|
      if @movie.save
        fresh_when(@movie)
        format.json { render :show, status: :created, location: @movie }
        format.v2json { render :show, status: :created, location: @movie }
      ...
```

Headers

DEMO

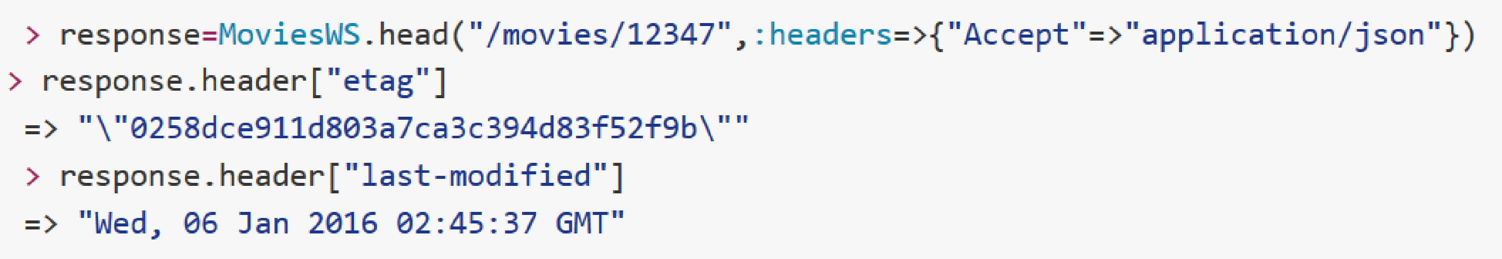
Status

- ✧ Etag and Last-Modified Headers are returned for `GET`



```
> response=MoviesWS.get("/movies/12347",:headers=>{"Accept"=>"application/vnd.myorgmovies.v2+json"})
> response.header["last-modified"]
 => "Wed, 06 Jan 2016 02:45:37 GMT"
  > response.header["etag"]
 => "\"1db42608e19f6e50209190f8ac7470d2\""
```

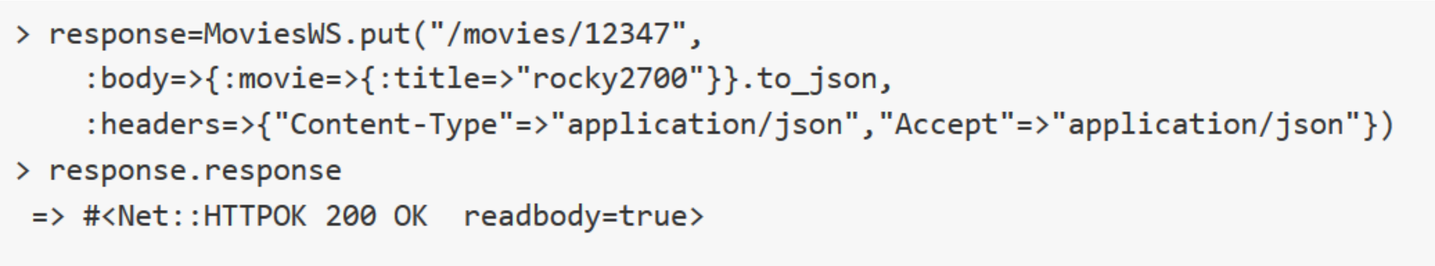
- ✧ Etag and Last-Modified Headers are returned for `HEAD`



```
 > response=MoviesWS.head("/movies/12347",:headers=>{"Accept"=>"application/json"})
> response.header["etag"]
 => "\"0258dce911d803a7ca3c394d83f52f9b\""
 > response.header["last-modified"]
 => "Wed, 06 Jan 2016 02:45:37 GMT"
```

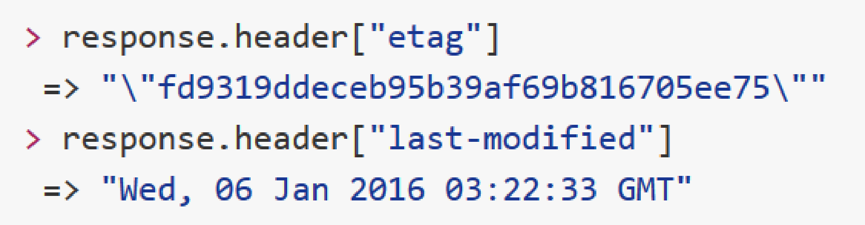
Updates

- Changes via `PUT`



```
> response=MoviesWS.put("/movies/12347",
    :body=>{:movie=>{:title=>"rocky2700"}}.to_json,
    :headers=>{"Content-Type"=>"application/json","Accept"=>"application/json"})
> response.response
 => #<Net::HTTPOK 200 OK  readbody=true>
```

- `Etag` and `Last-Modified` have changed



```
> response.header["etag"]
 => "\"fd9319ddeceb95b39af69b816705ee75\""
> response.header["last-modified"]
 => "Wed, 06 Jan 2016 03:22:33 GMT"
```

Concurrent Update Issue

✧ Nested Resource Update

- Update Movie
- Add Role to Movie
- Add same Role to Movie (second time)
- `Etag` is updated but `Last-Modified` will not
- Can be fixed by using `touch`

IF_UNMODIFIED_SINCE

- If HTTP_IF_UNMODIFIED_SINCE is supplied
  - Use fresh_when to populate response with current Etag and Last-Modified
  - Continue if request date later than or equal to current state
  - Else report conflict and make no changes

Summary

- ✧ Cache Management and Concurrency Management can be successfully managed with status and headers

What’s Next?

- ✧ Caching