Rails on Cache

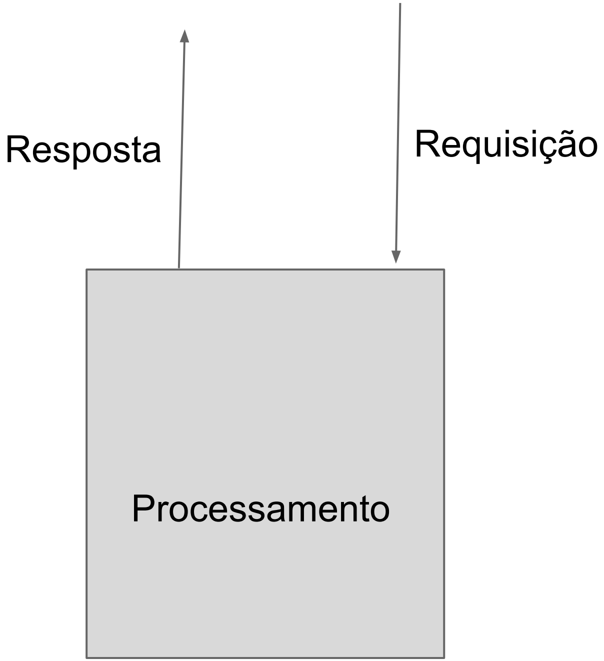
Quem sou eu?

1. Jônatas Rancan
2. Dev desde 2011
3. Ruby on Rails
4. CodeMiner 42

O que veremos?

1. O que é cache?
2. Tipos de armazenamentos.
3. Técnicas de cache em Rails.
    a. Fragment caching
    b. Russian Doll caching
    c. Low level caching
    d. Action caching
    e. Page caching
    f. Condition GET Support (HTTP cache)
4. App exemplo

O que é cache?

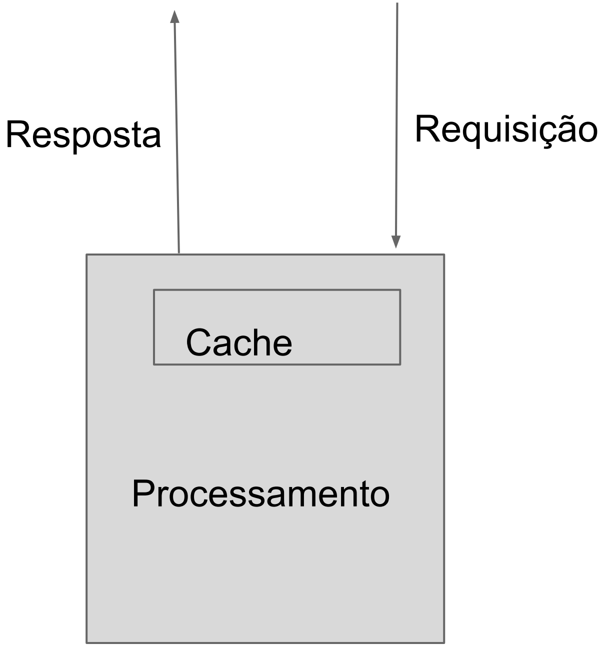


Resposta

Requisição

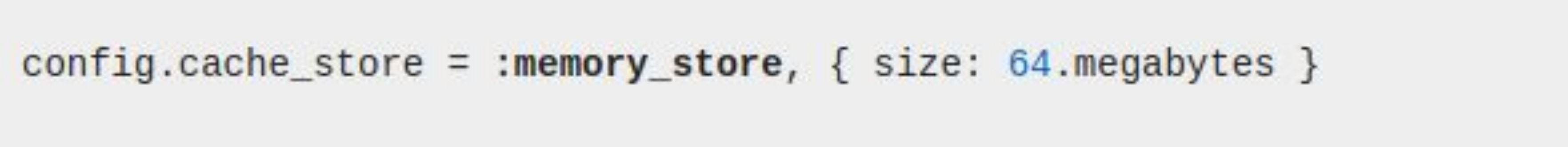
Processamento

O que é cache?

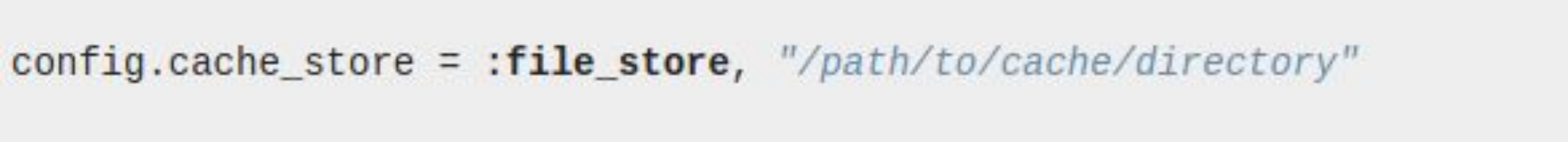


Tipos de Armazenamento

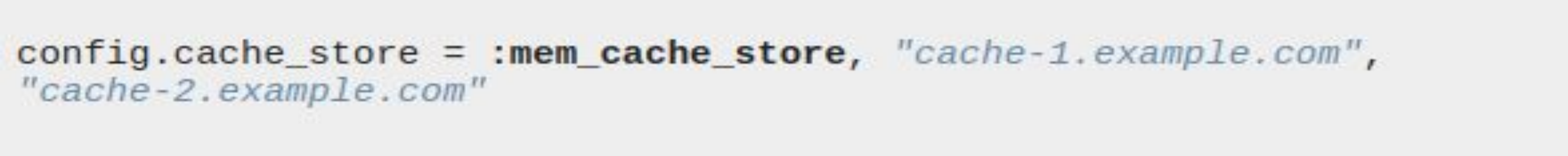
1. Rails 5~5.1
    a. Memória
    b. File system
    c. Memcached



```
config.cache_store = :memory_store, { size: 64.megabytes }
```



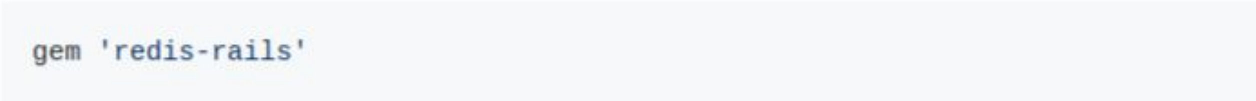
```
config.cache_store = :file_store, "/path/to/cache/directory"
```



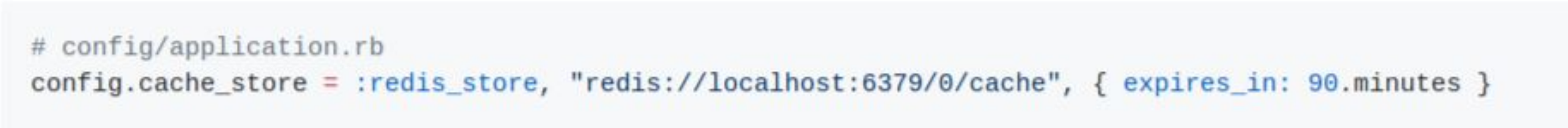
```
config.cache_store = :mem_cache_store, "cache-1.example.com",
"cache-2.example.com"
```

Tipos de Armazenamento

- Redis
  - <= Rails 5.1

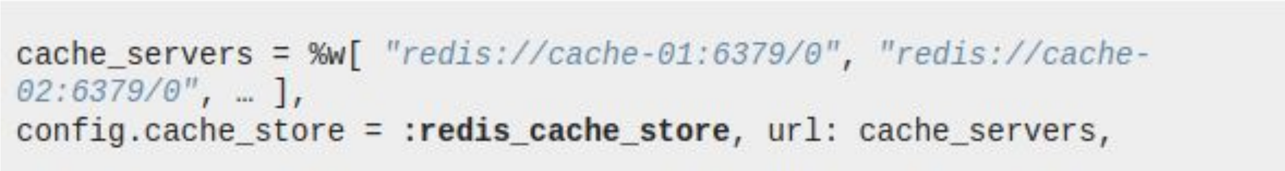


```
gem 'redis-rails'
```



```
# config/application.rb
config.cache_store = :redis_store, "redis://localhost:6379/0/cache", { expires_in: 90.minutes }
```

  - >= Rails 5.2



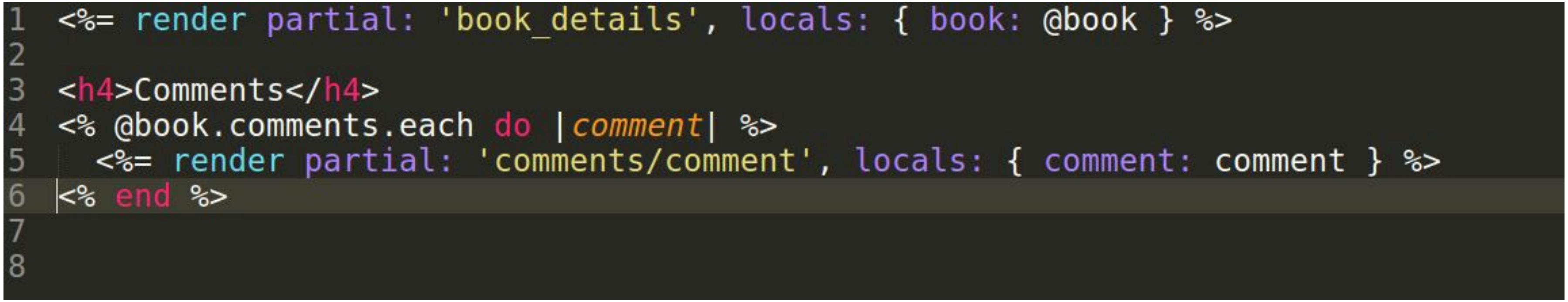
```
cache_servers = %w[ "redis://cache-01:6379/0", "redis://cache-
02:6379/0", … ],
config.cache_store = :redis_cache_store, url: cache_servers,
```

Tipos de Armazenamento

- **ActiveSupport::Cache::Store**
  - **ActiveSupport::Cache::MemoryStore**
  - **ActiveSupport::Cache::FileStore**
  - **ActiveSupport::Cache::MemCacheStore**
  - **ActiveSupport::Cache::RedisCacheStore**
  - **ActiveSupport::Cache::NullStore**
  - config.cache_store = SuaClasseDeCache.new

Fragment Caching

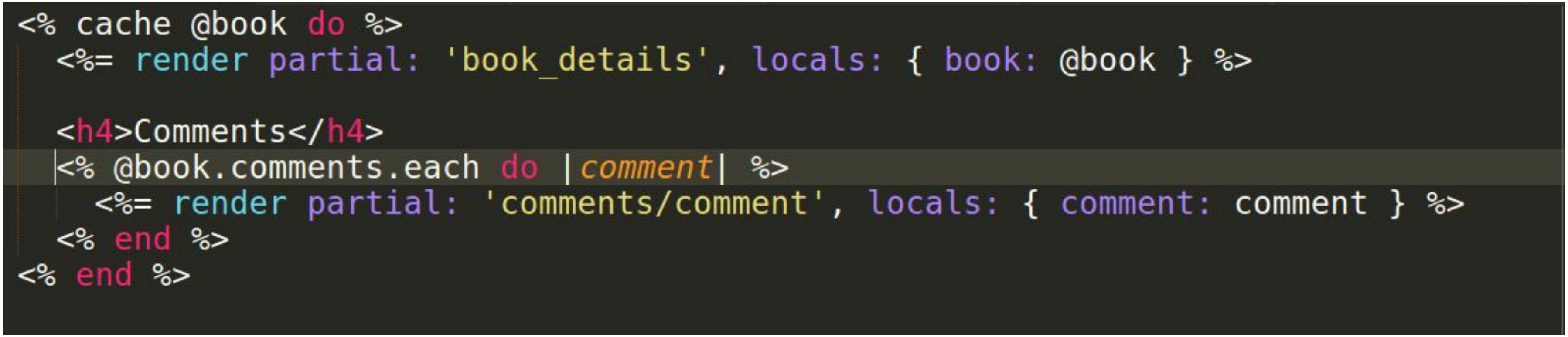
- Cachear “fragmentos” da view



```
<%= render partial: 'book_details', locals: { book: @book } %>

<h4>Comments</h4>
<% @book.comments.each do |comment| %>
  <%= render partial: 'comments/comment', locals: { comment: comment } %>
<% end %>
```

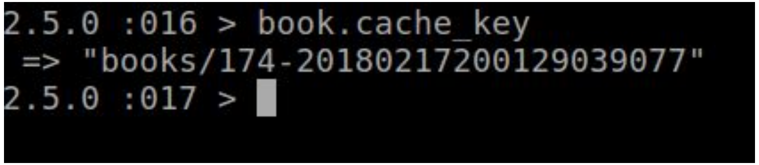
Fragment Caching



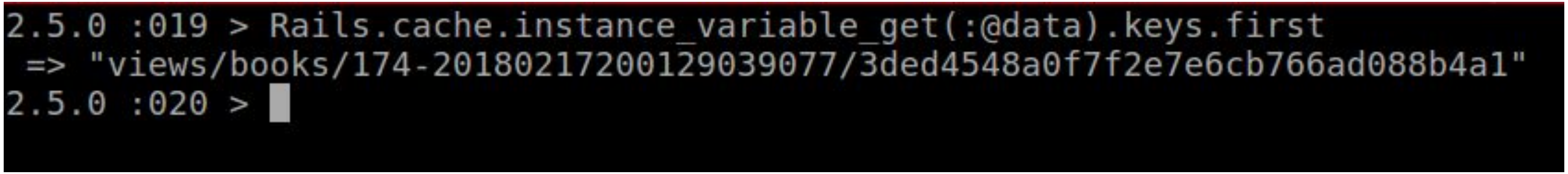
```
<% cache @book do %>
  <%= render partial: 'book_details', locals: { book: @book } %>

  <h4>Comments</h4>
  <% @book.comments.each do |comment| %>
    <%= render partial: 'comments/comment', locals: { comment: comment } %>
  <% end %>
<% end %>
```

Fragment Caching

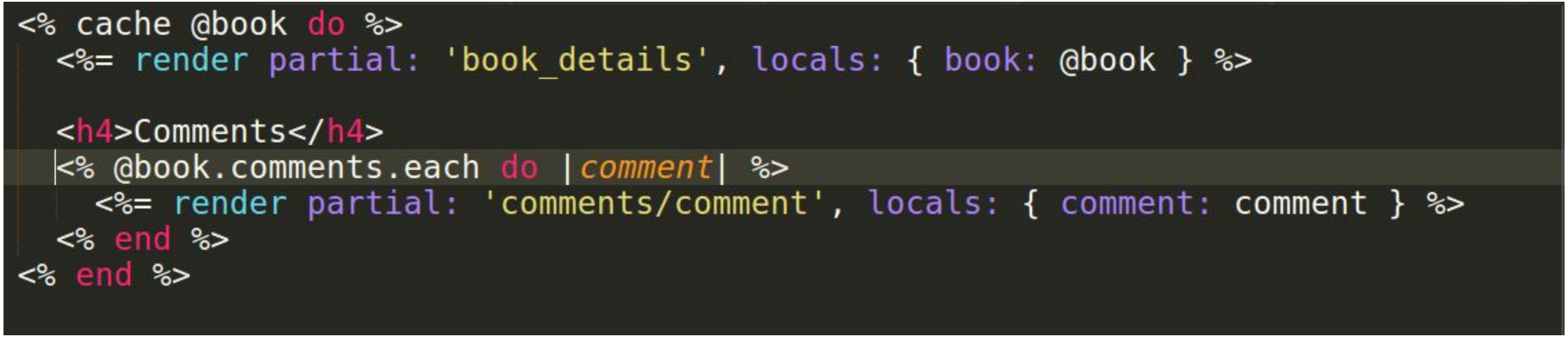


```
2.5.0 :016 > book.cache_key
 => "books/174-20180217200129039077"
2.5.0 :017 >
```



```
2.5.0 :019 > Rails.cache.instance_variable_get(:@data).keys.first
 => "views/books/174-20180217200129039077/3ded4548a0f7f2e7e6cb766ad088b4a1"
2.5.0 :020 >
```

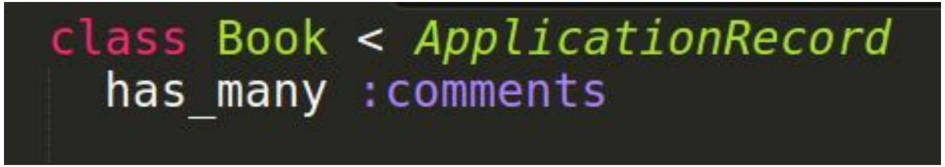
Fragment Caching



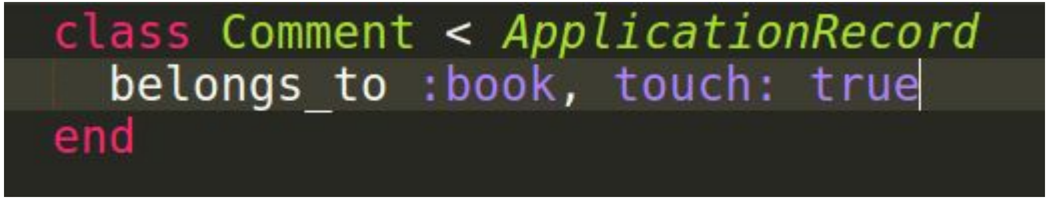
```
<% cache @book do %>
  <%= render partial: 'book_details', locals: { book: @book } %>

  <h4>Comments</h4>
  <% @book.comments.each do |comment| %>
    <%= render partial: 'comments/comment', locals: { comment: comment } %>
  <% end %>
<% end %>
```

Fragment Caching



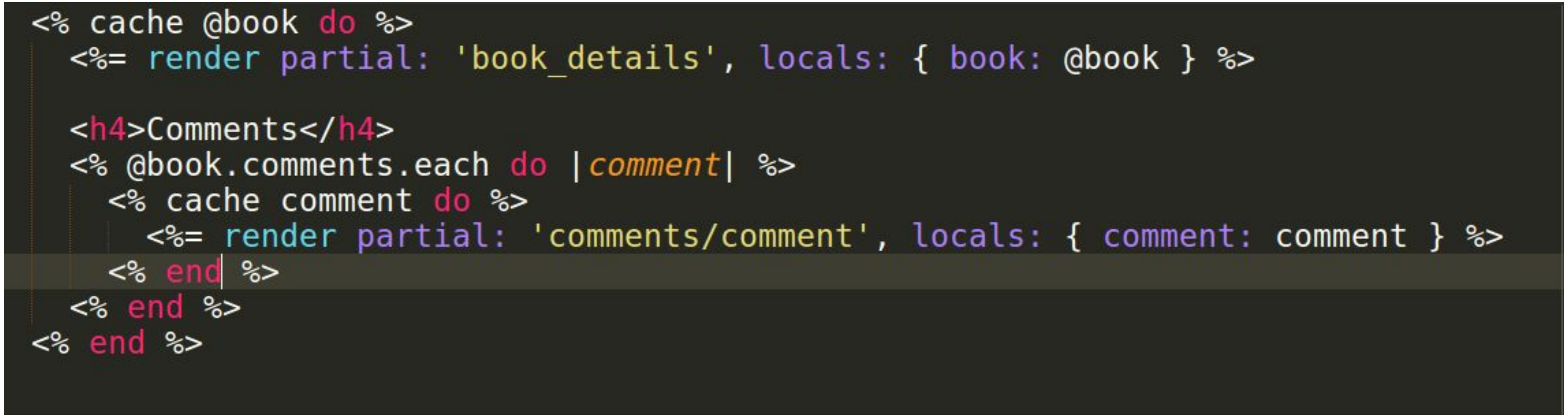
```ruby
class Book < ApplicationRecord
  has_many :comments
```



```ruby
class Comment < ApplicationRecord
  belongs_to :book, touch: true
end
```

Russian Doll Caching

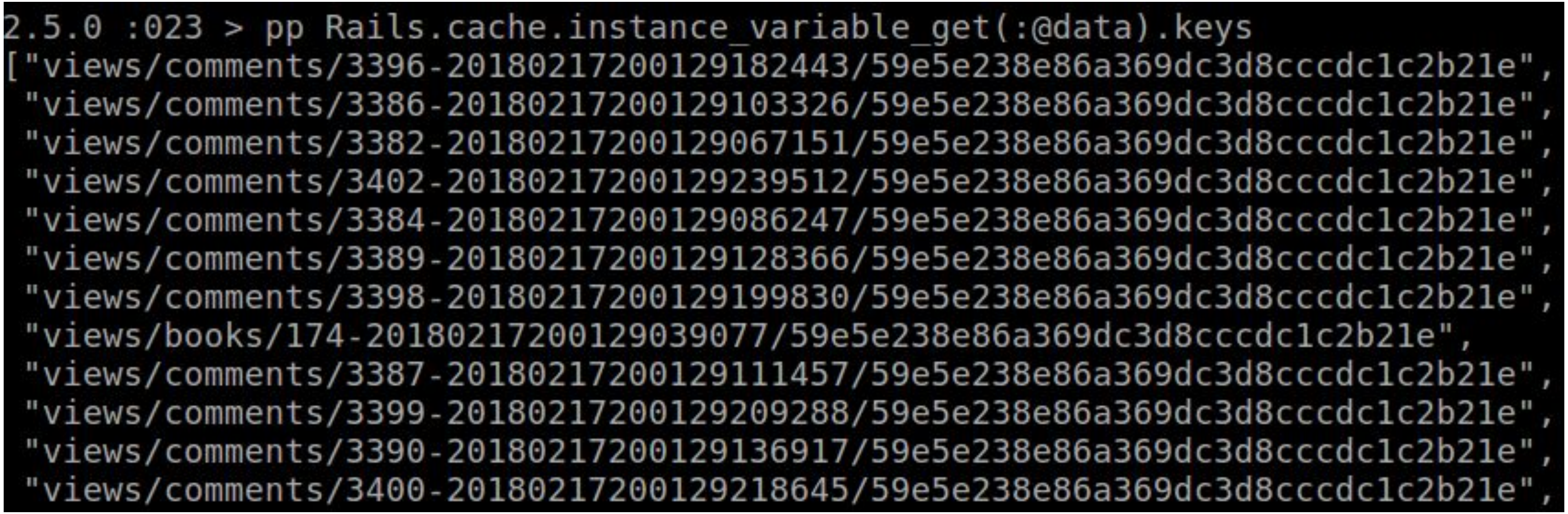
Russian Doll Caching



```
<% cache @book do %>
  <%= render partial: 'book_details', locals: { book: @book } %>

  <h4>Comments</h4>
  <% @book.comments.each do |comment| %>
    <% cache comment do %>
      <%= render partial: 'comments/comment', locals: { comment: comment } %>
    <% end %>
  <% end %>
<% end %>
```

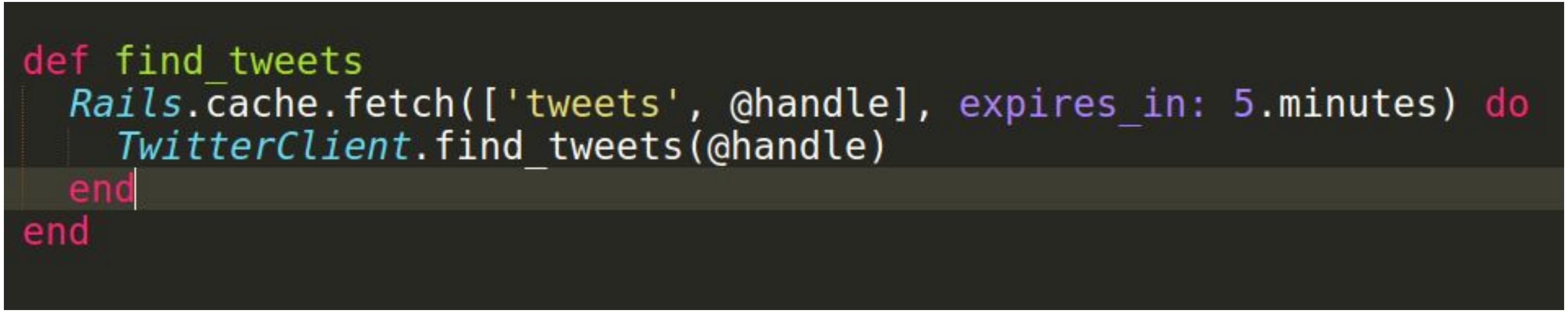
Russian Doll Caching



```
2.5.0 :023 > pp Rails.cache.instance_variable_get(:@data).keys
["views/comments/3396-20180217200129182443/59e5e238e86a369dc3d8cccdc1c2b21e",
 "views/comments/3386-20180217200129103326/59e5e238e86a369dc3d8cccdc1c2b21e",
 "views/comments/3382-20180217200129067151/59e5e238e86a369dc3d8cccdc1c2b21e",
 "views/comments/3402-20180217200129239512/59e5e238e86a369dc3d8cccdc1c2b21e",
 "views/comments/3384-20180217200129086247/59e5e238e86a369dc3d8cccdc1c2b21e",
 "views/comments/3389-20180217200129128366/59e5e238e86a369dc3d8cccdc1c2b21e",
 "views/comments/3398-20180217200129199830/59e5e238e86a369dc3d8cccdc1c2b21e",
 "views/books/174-20180217200129039077/59e5e238e86a369dc3d8cccdc1c2b21e",
 "views/comments/3387-20180217200129111457/59e5e238e86a369dc3d8cccdc1c2b21e",
 "views/comments/3399-20180217200129209288/59e5e238e86a369dc3d8cccdc1c2b21e",
 "views/comments/3390-20180217200129136917/59e5e238e86a369dc3d8cccdc1c2b21e",
 "views/comments/3400-20180217200129218645/59e5e238e86a369dc3d8cccdc1c2b21e",
```

Low Level Caching

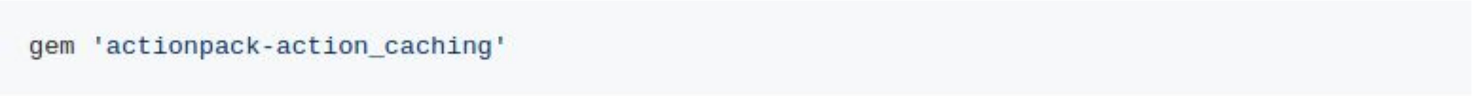
- Utilizado para cachear resultado de query, chamada de API
- Rails.cache.fetch



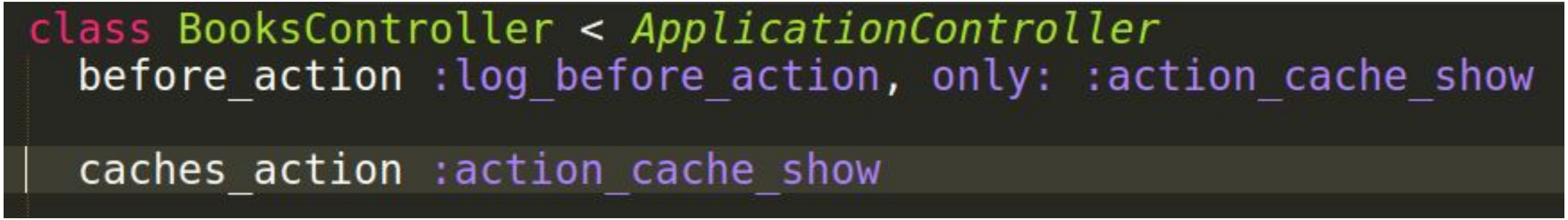
```ruby
def find_tweets
  Rails.cache.fetch(['tweets', @handle], expires_in: 5.minutes) do
    TwitterClient.find_tweets(@handle)
  end
end
```

Action Cache

- Cache a nível de action



```
gem 'actionpack-action_caching'
```

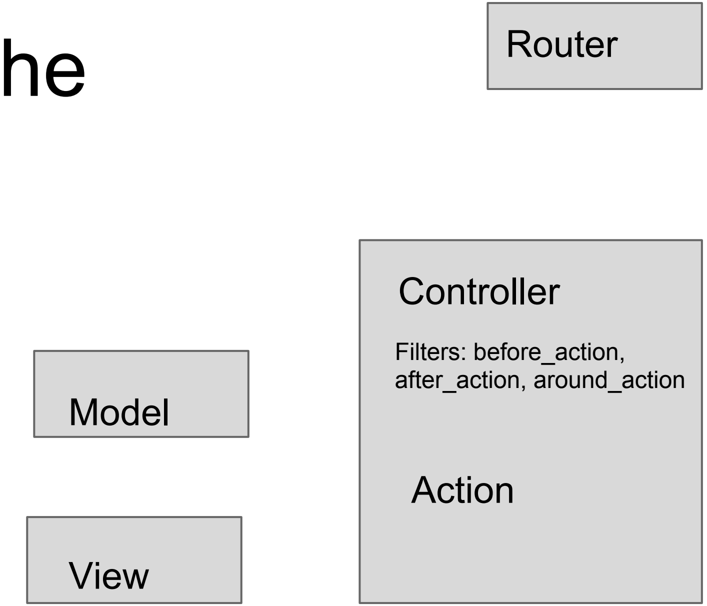


```
class BooksController < ApplicationController
  before_action :log_before_action, only: :action_cache_show

  caches_action :action_cache_show
```

Action Cache

Sem Cache



Router

Controller

Filters: before_action, after_action, around_action

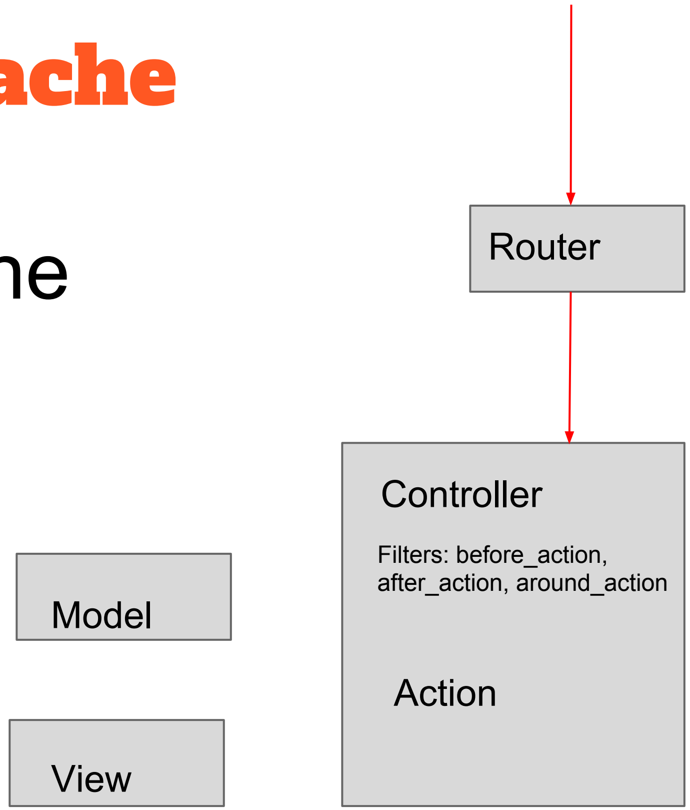
Action

Model

View

Action Cache

Sem Cache



Router

Controller

Filters: before_action, after_action, around_action

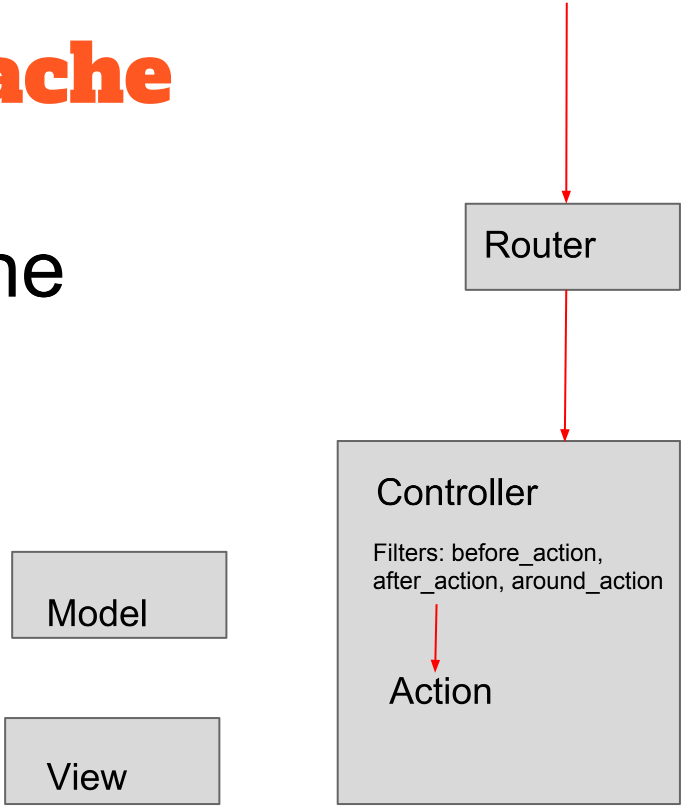
Action

Model

View

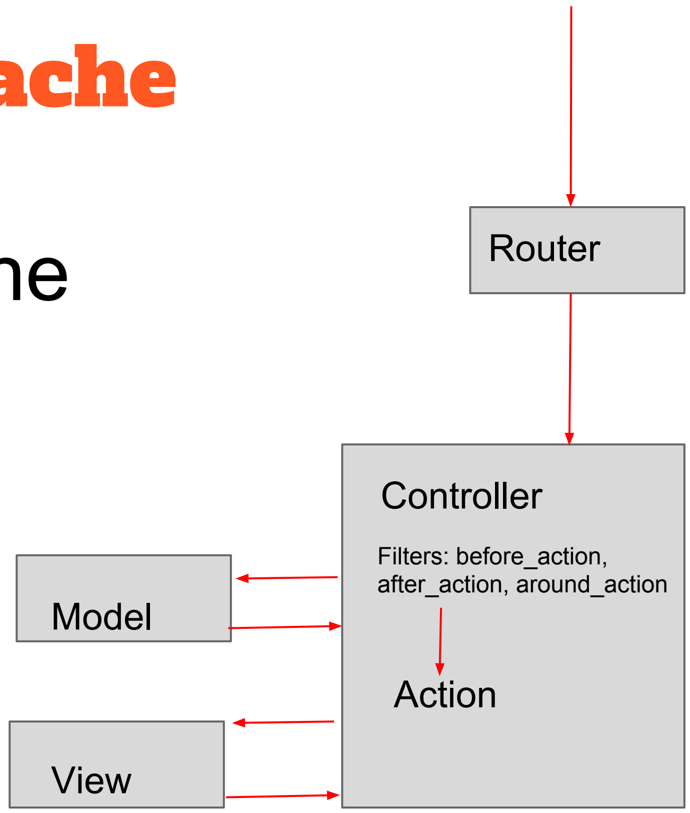
Action Cache

Sem Cache



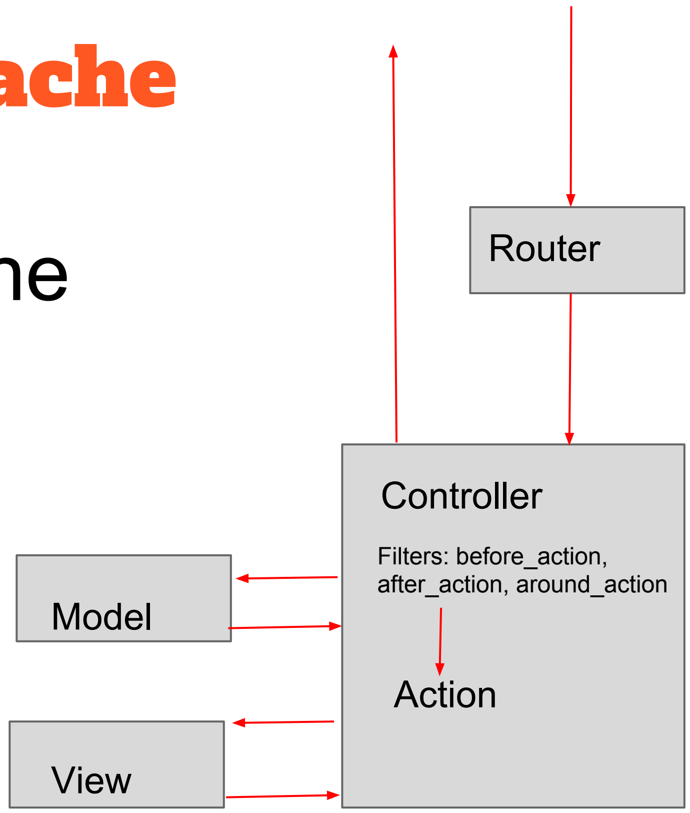
Action Cache

Sem Cache



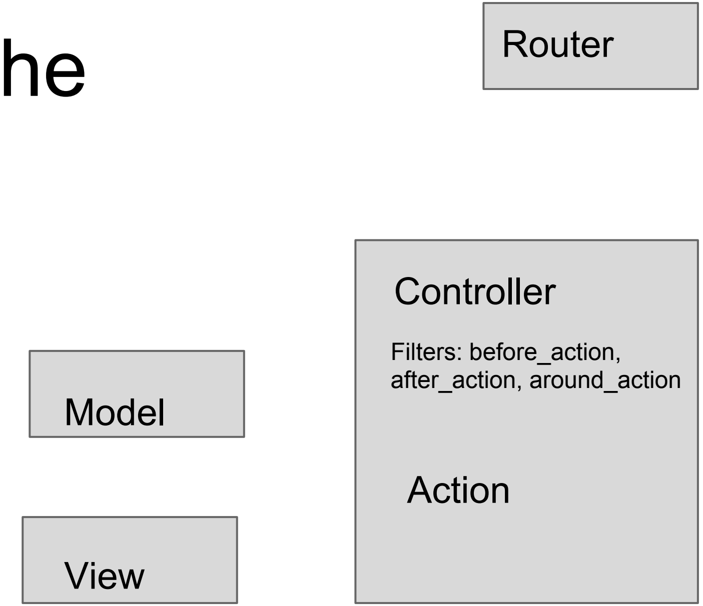
Action Cache

Sem Cache



Action Cache

Com Cache



Router

Controller

Filters: before_action, after_action, around_action

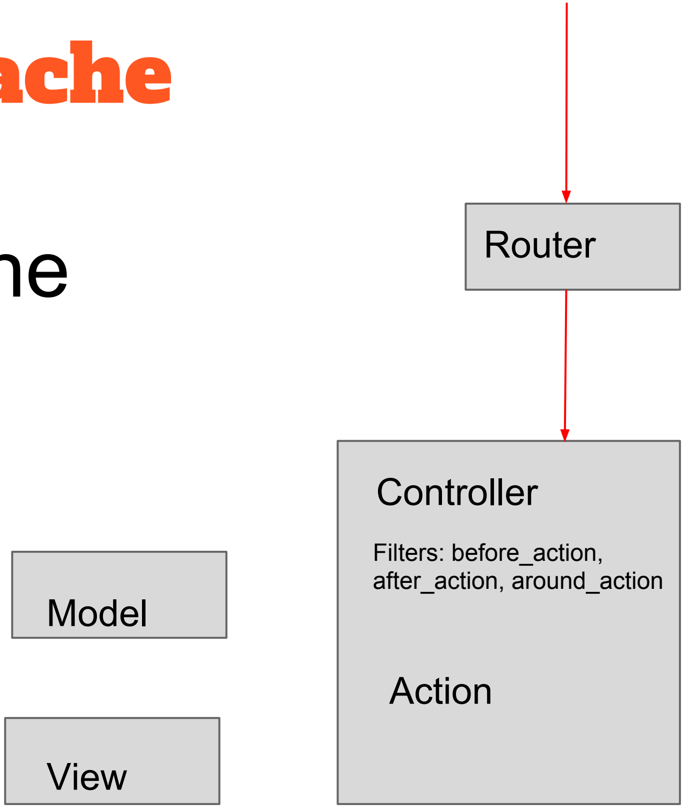
Action

Model

View

Action Cache

Com Cache



Router

Controller

Filters: before_action, after_action, around_action

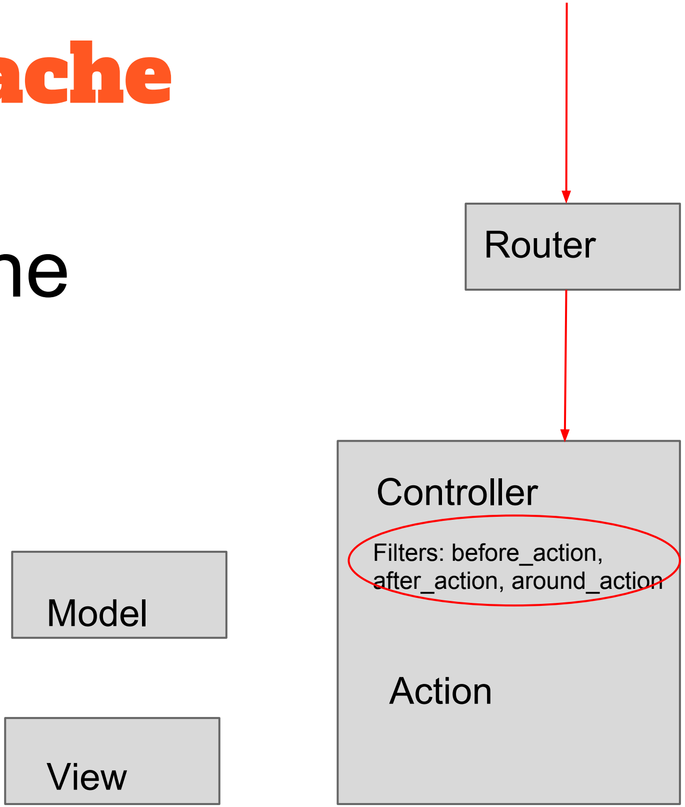
Action

Model

View

Action Cache

Com Cache



Router

Controller

Filters: before_action, after_action, around_action

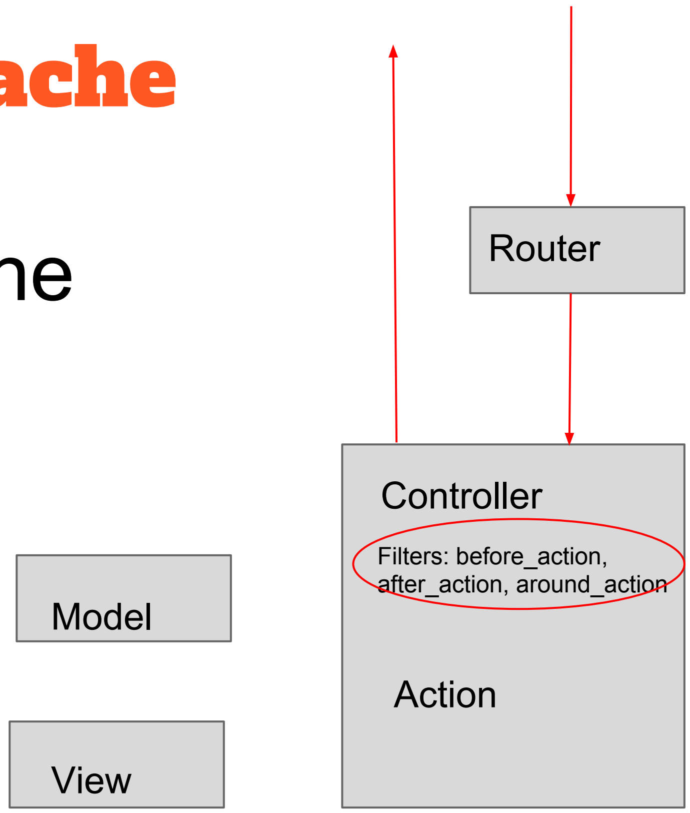
Action

Model

View

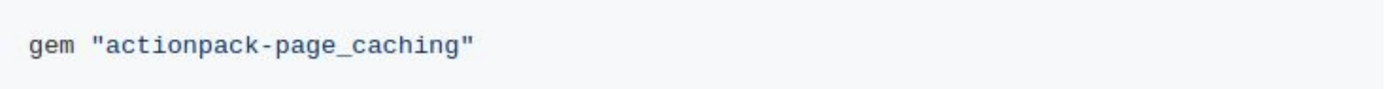
Action Cache

Com Cache

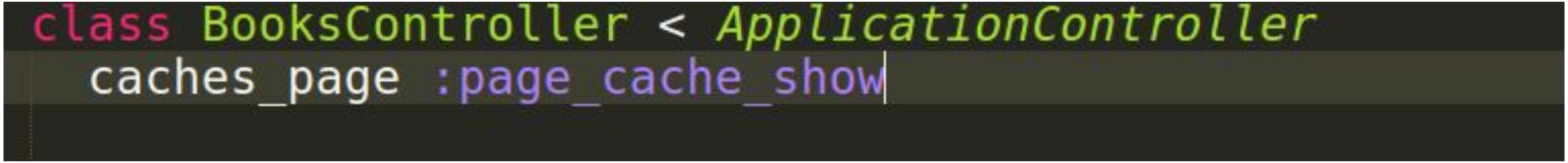


Page Cache

- Cache a nível de página
- Salva no sistema de arquivos



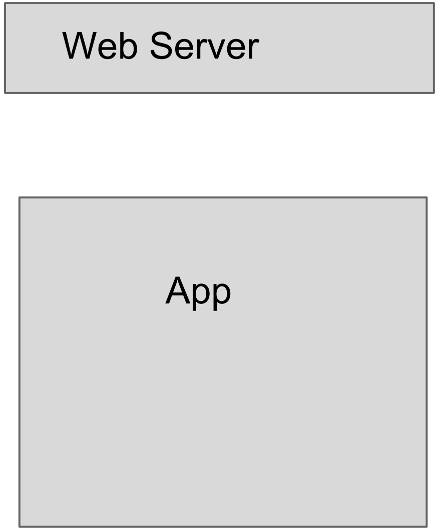
```
gem "actionpack-page_caching"
```



```
class BooksController < ApplicationController
  caches_page :page_cache_show
```

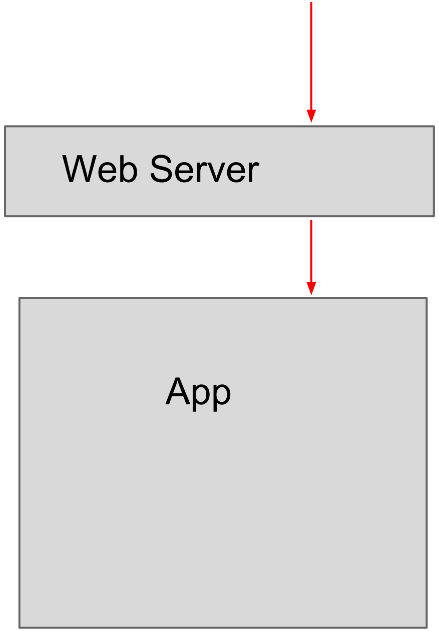
Page Cache

Sem Cache



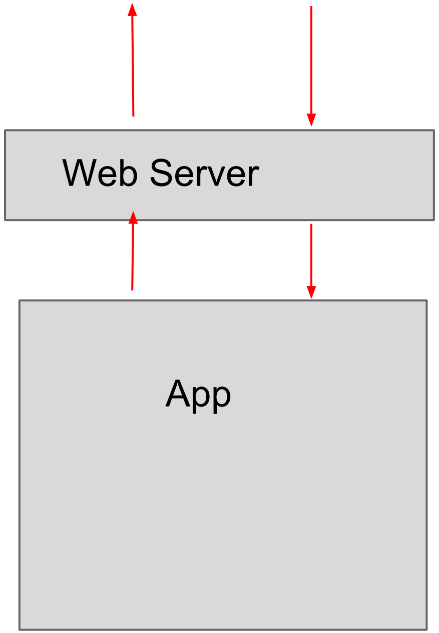
Page Cache

Sem Cache



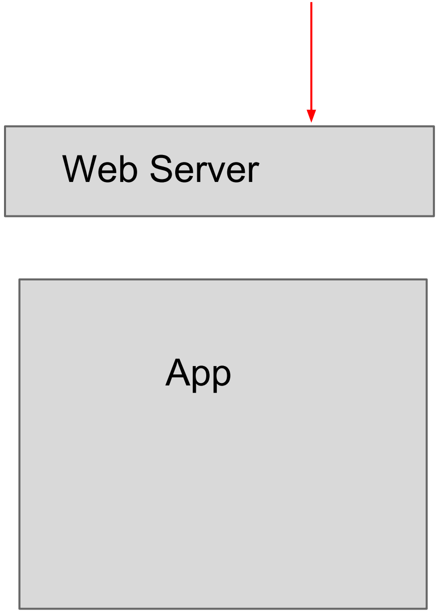
Page Cache

Sem Cache



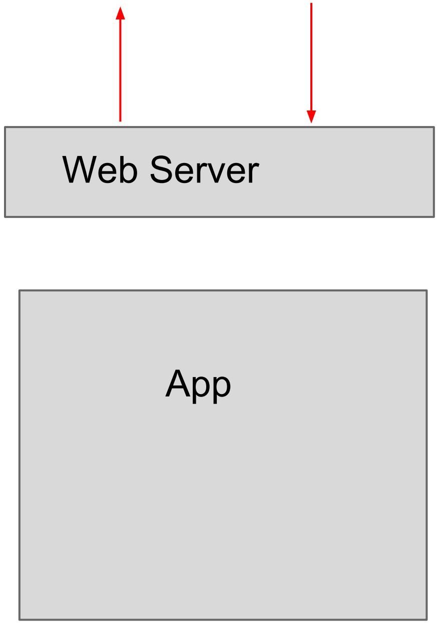
Page Cache

Com Cache



Page Cache

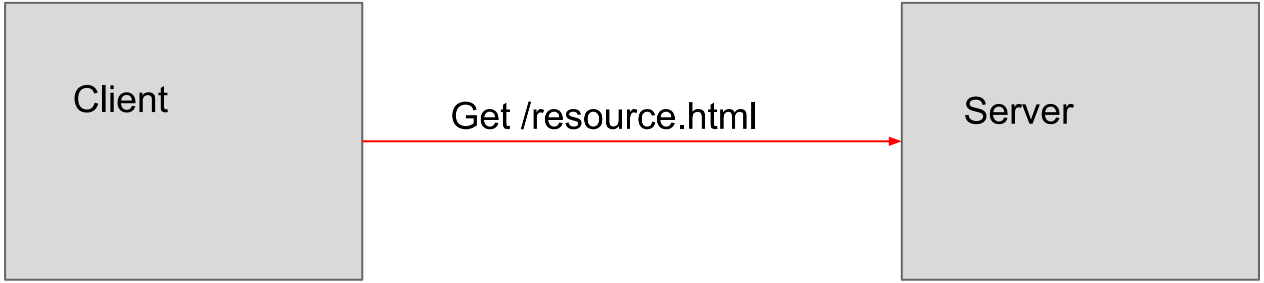
Com Cache



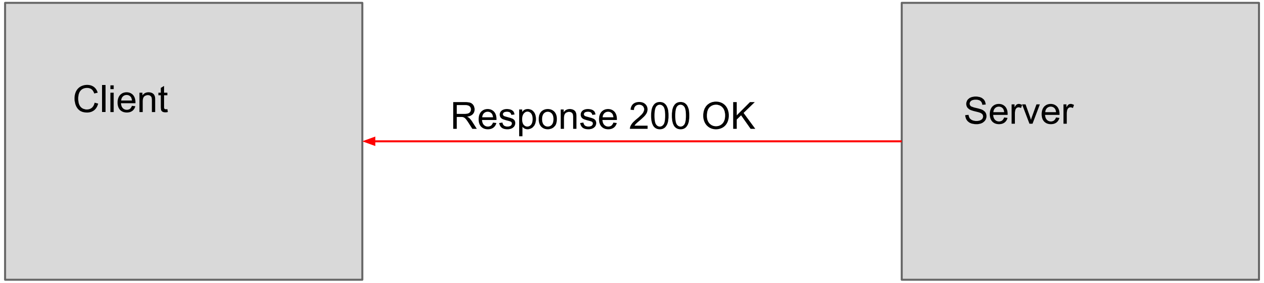
Conditional GET support

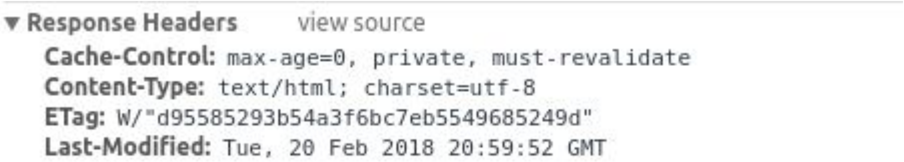
- Cache HTTP
- Usa cache to browser
- Etag
- If-None-Match

Conditional GET support



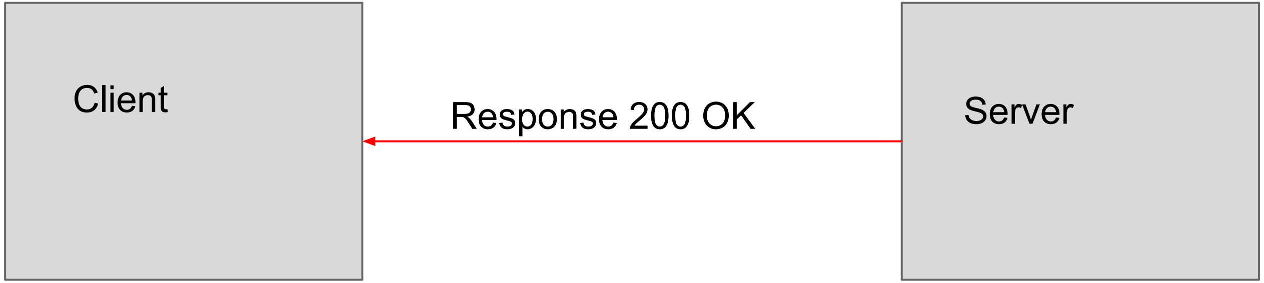
Conditional GET support

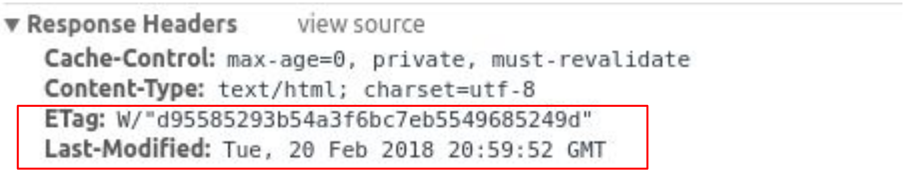




```
▼ Response Headers      view source
    Cache-Control: max-age=0, private, must-revalidate
    Content-Type: text/html; charset=utf-8
    ETag: W/"d95585293b54a3f6bc7eb5549685249d"
    Last-Modified: Tue, 20 Feb 2018 20:59:52 GMT
```

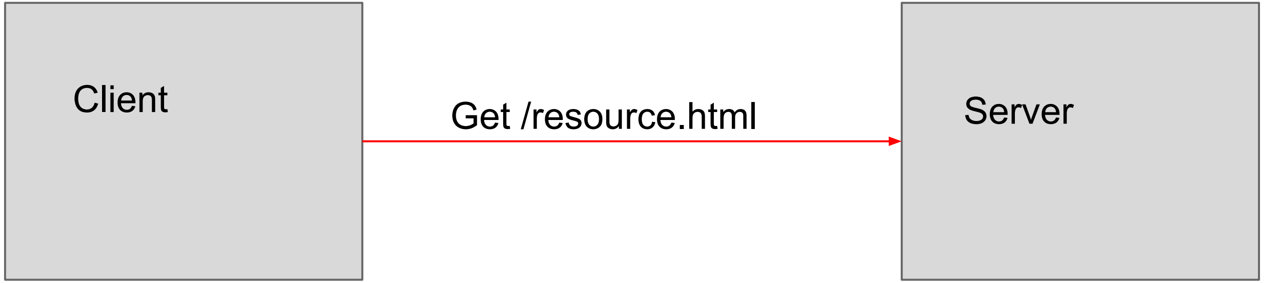
Conditional GET support





```
Cache-Control: max-age=0, private, must-revalidate
Content-Type: text/html; charset=utf-8
ETag: W/"d95585293b54a3f6bc7eb5549685249d"
Last-Modified: Tue, 20 Feb 2018 20:59:52 GMT
```

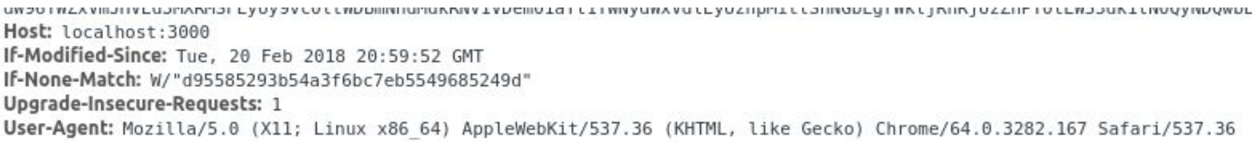
Conditional GET support



Client

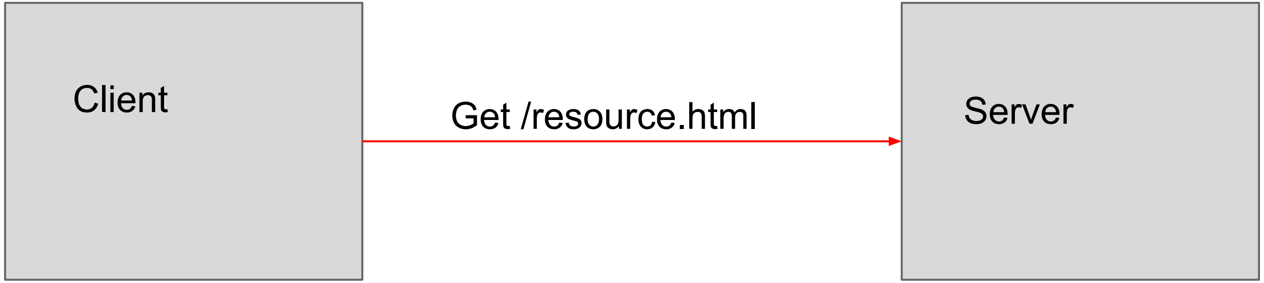
Get /resource.html

Server



```
Host: localhost:3000
If-Modified-Since: Tue, 20 Feb 2018 20:59:52 GMT
If-None-Match: W/"d95585293b54a3f6bc7eb5549685249d"
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/64.0.3282.167 Safari/537.36
```

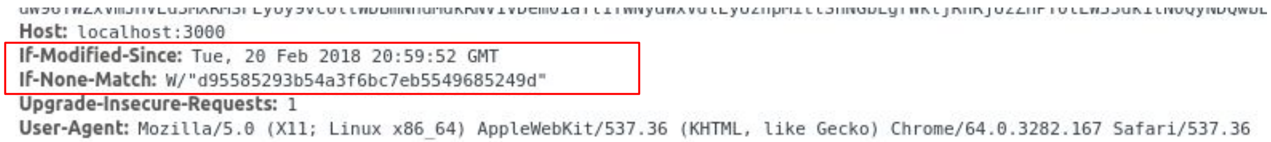
Conditional GET support



Client

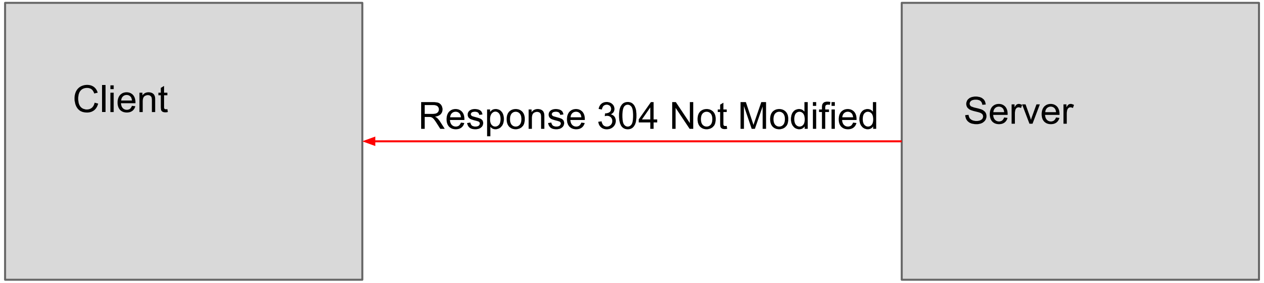
Get /resource.html

Server

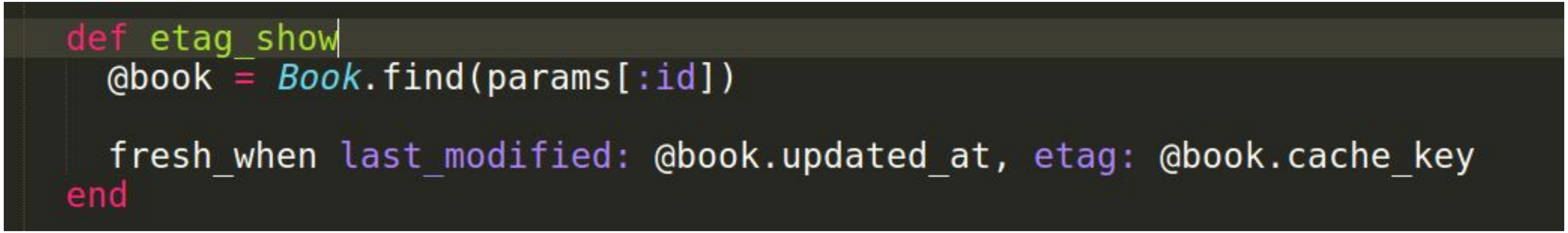


```
Host: localhost:3000
If-Modified-Since: Tue, 20 Feb 2018 20:59:52 GMT
If-None-Match: W/"d95585293b54a3f6bc7eb5549685249d"
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/64.0.3282.167 Safari/537.36
```

Conditional GET support

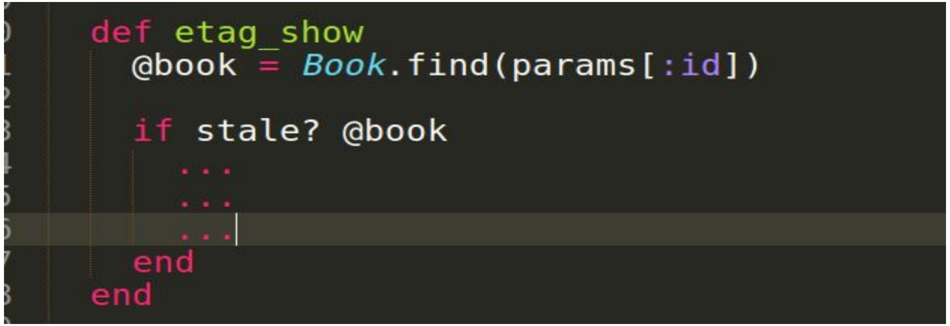


Conditional GET support



```
def etag_show
  @book = Book.find(params[:id])

  fresh_when last_modified: @book.updated_at, etag: @book.cache_key
end
```



```
def etag_show
  @book = Book.find(params[:id])

  if stale? @book
    ...
    ...
    ...
  end
end
```

That's all Folks!

Links Uteis

- http://guides.rubyonrails.org/caching_with_rails.html
- http://hawkins.io/2012/07/advanced_caching_part_1-caching_strategies/
- https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/ETag
- https://en.wikipedia.org/wiki/HTTP_ETag
- https://www.justinweiss.com/articles/a-web-server-vs-an-app-server/
- https://github.com/jonatasrancan/rails-on-cache