

A Practical Guide to Hypermedia APIs

Eric Oestrich
@ericoestrich
@smartlogicsolutions





woofound.com



Steve Klabnik

@steveklabnik



 Follow



I just saw a diff where someone converted their app + client (same repo) to a hypermedia api: +900, -5000 LOC

 Reply  Retweet  Favorited

13
RETWEETS

4
FAVORITES



5:53 PM - 13 Nov 12 · Embed this Tweet

That was this project

Prepare yourself

**Please note that this has been
heavily stripped down**

Don't want to give everything away

But first

Hypermedia Basics

Resources \neq Models

HTTP is your friend

methods and status codes

**Client knows only one
URL**

"/"

Media types

HAL

Sample HAL Response

```
{
  "status": "Processing",
  "_embedded": {
    "items": [
      {
        "name": "Book",
        "_links": { "self": { "href" => "http://example.com/items/1" } }
      }
    ]
  },
  "_links": {
    "self" => { "href" => "http://example.com/orders/1" }
  }
}
```

Server

Controllers

- Use `respond_to` and `respond_with`

Models

- Doesn't have to be a subclass of ActiveRecord::Base

Serializers

- Disable root
- Links
- Embedded resources
- Changing serializer representations

Client

Basics

- Uses Faraday
- Basic auth over SSL

Resources

- Data only
- Convenience methods
- Links

Loaders

- Pull data down from the API
- Only GETs

Services

- Very similar to Loaders
- Push data back up

Integrating with Rails

Problems

- There are no ids
- Speed

Resources

- Creating a Hypermedia Client
 - <http://blog.oestrich.org/2012/12/creating-a-hypermedia-client/>
- ActiveSupport::Serializers
 - https://github.com/rails-api/active_model_serializers
- Abusing ActiveSupport::Serializers for HAL
 - <http://blog.oestrich.org/2012/10/abusing-active-model-serializers-for-hal/>
- <httpstatus.es>

Useful gems for apis

- `rspec_api_documentation`
- `Raddocs`
- `letter`
- `json_spec`
- `vcr`
- `webmock`
- `cane`

Hobostove

Campfire for the command line

<https://github.com/oestrich/hobostove>