STARTING WITH EMBER DATA

HOW TO USE EMBER WITH YOUR REST API?

EMBER.JS BCN MEETUP - MARCH 3, 2016

HELLO... IT'S ME

José San Gil / @jsangilve

I've been using Ember for 14 months

HOW CAN YOU USE EMBER DATA WITH YOUR REST API?

"My API uses JSON format but...it isn't a JSON API".

- Someone starting with Ember

WTF

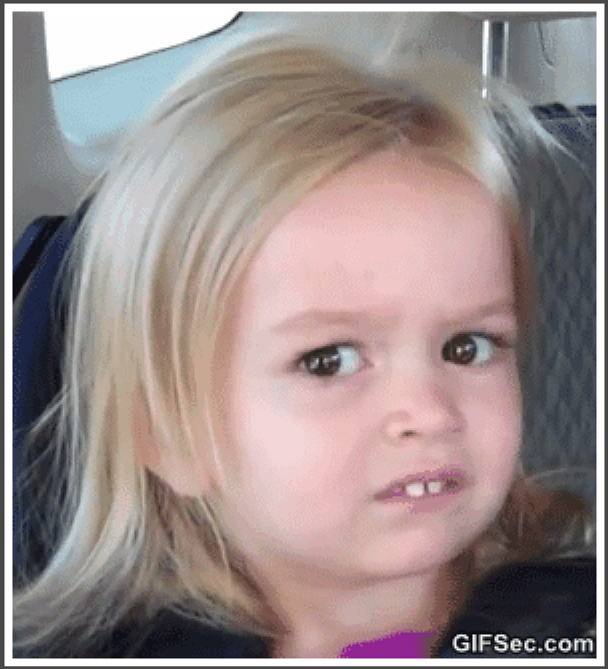
{json:api}

www.jsonapi.org

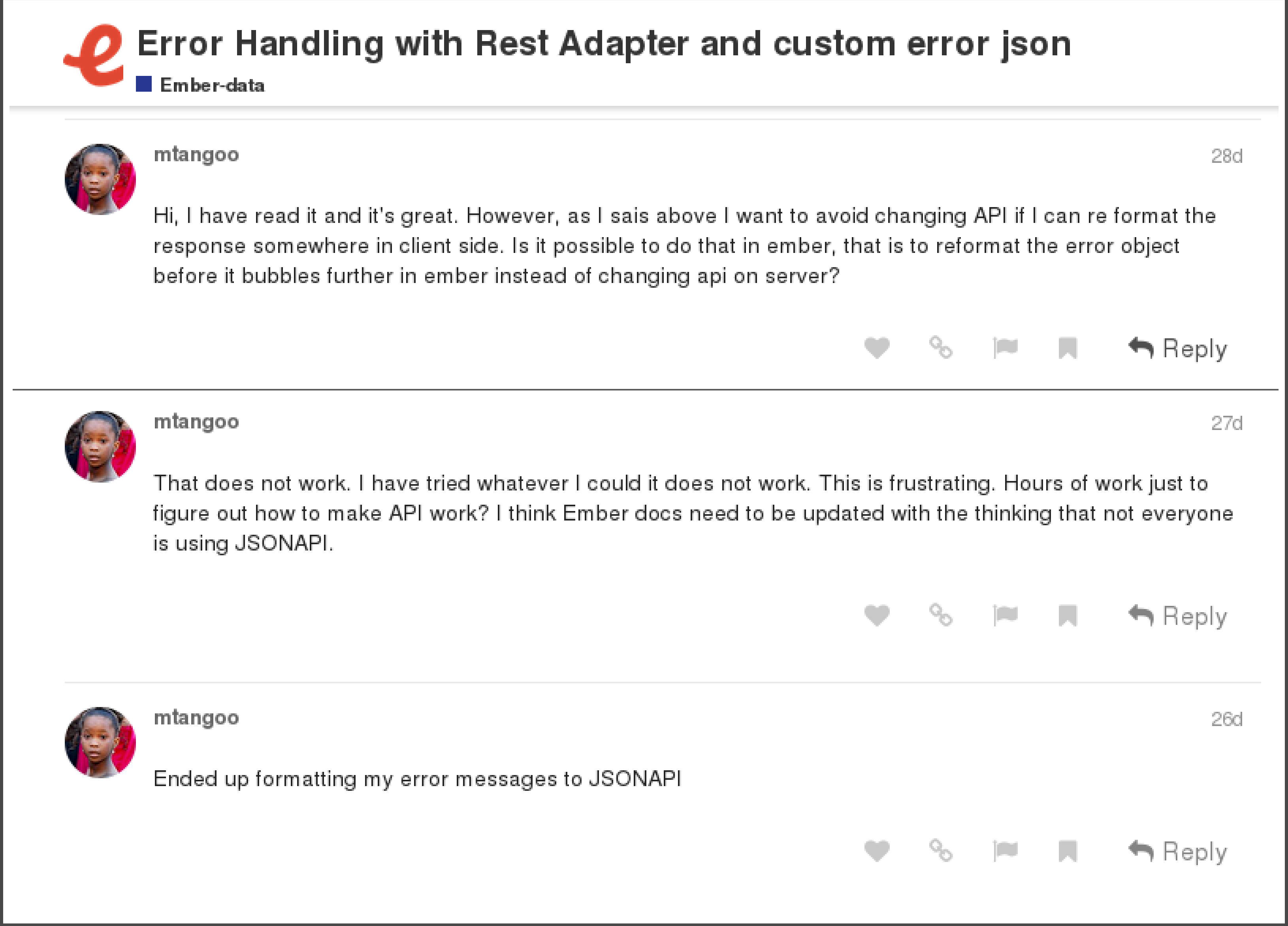
JSON API specification is the default JSON format followed by Ember Data since Ember Data 2.0 (It was introduced with Ember 1.13)

It looks like this...

```json
{
  "data": [{
    "type": "articles",
    "id": "1",
    "attributes": {
      "title": "JSON API paints my bikeshed!",
      "body": "The shortest article. Ever."
    },
    "relationships": {
      "author": {
        "data": {"id": "42", "type": "people"}
      }
    }
  }],
  "included": [
    {
      "type": "people",
      "id": "42",
      "attributes": {
        "name": "John"
      }
    }
  ]
}
```



"...my API uses JSON format but doesn't look like that"



Error Handling with Rest Adapter and custom error json

Ember-data

mtangoo 28d

Hi, I have read it and it's great. However, as I sais above I want to avoid changing API if I can re format the response somewhere in client side. Is it possible to do that in ember, that is to reformat the error object before it bubbles further in ember instead of changing api on server?

mtangoo 27d

That does not work. I have tried whatever I could it does not work. This is frustrating. Hours of work just to figure out how to make API work? I think Ember docs need to be updated with the thinking that not everyone is using JSONAPI.

mtangoo 26d

Ended up formatting my error messages to JSONAPI

DON'T HATE EMBER DATA/JSON API

You just need to learn about these classes:

- JSONAdapter
- RESTAdapter
- JSONSerializer
- RESTSerializer

ADAPTERS

Tell Ember how to connect to your API (URL, Headers)

- DS.Adapter
- DS.RESTAdapter
- DS.JSONAPIAdapter

SERIALIZERS

Tell Ember how to format data sent and received from the server

- DS.Serializer
- DS.JSONSerializer
- DS.RESTSerializer
- DS.JSONAPISerializer

YOU ALREADY KNOW THAT

NO, SERIOUSLY:

You aren't going to use DS.JSONAPIAdapter

You aren't going to use DS.JSONAPISerializer

ALSO..

You have a REST API

- You need DS.RESTAdapter

AND..

Your API uses JSON Format

You have to choose between:

1. DS.JSONSerializer
2. DS.RESTSerializer

DIFFERENCE:

DS.JSONSerializer

Expects simple data without extra information or attributes

```
HTTP GET /api/team/1
{
  "id": 1,
  "name": "Leicester City FC",
  "nickname": "The Foxes",
  "founded": 1884,
  "ground": "King Power Stadium",
  "manager": "Claudio Ranieri",
  "league": 1
}
```

DS.JSONSerializer

Another endpoint. Serializer will expect...

```
HTTP GET /api/leagues
[
 { "id": 1, "name": "Premier League", "confederation": "UEFA", "fo
 { "id": 2, "name": "Primera División España", "confederation": "U
 { "id": 3, "name": "Serie A", "confederation": "UEFA", "founded":
 { "id": 4, "name": "Bundesliga", "confederation": "UEFA", "founde
]
```

IN THE OTHER HAND

DS.RESTSerializer

We have this model

```
// app/models/team.js
import DS from 'ember-data';

const { attr, belongsTo } = DS;

export default DS.Model.extend({
  name: attr('string'),
  nickname: attr('string'),
  founded: attr('number'),
  ground: attr('string'),
  manager: attr('string'),
  league: belongsTo('league')
});
```

DS.RESTSerializer

Expects an extra attribute matching model name

```
HTTP GET /api/team/1
{
  "team": {
    "id": 1,
    "name": "Leicester City FC",
    "nickname": "The Foxes",
    "founded": 1884,
    "ground": "King Power Stadium",
    "manager": "Claudio Ranieri",
    "league": 1
  }
}
```

SIDELOADING

DS.RESTSerializer

It's capable of processing sideloaded records included in the same response

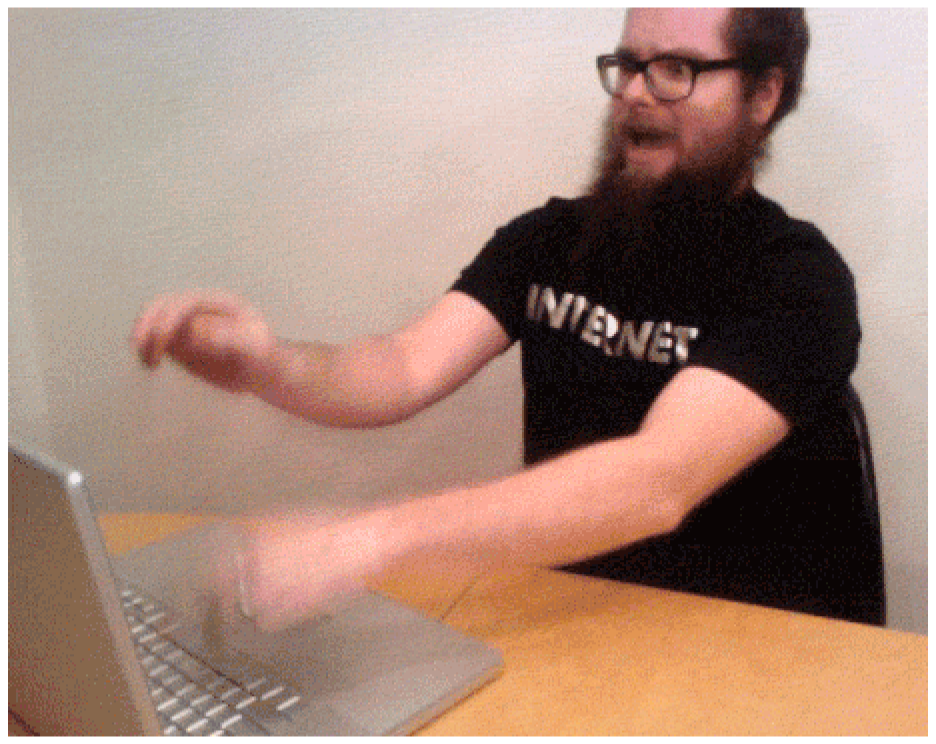
```
HTTP GET /api/team/1
{
  "team": {
    "id": 1,
    "name": "Leicester City FC",
    "nickname": "The Foxes",
    "founded": 1884,
    "ground": "King Power Stadium",
    "manager": "Claudio Ranieri",
    "league": 1
  },
  "league": {
    "id": 1,
    "name": "Premier League",
    "confederation": "UEFA",
    "founded": 1992,
    "numberOfTeams": 20,
    "country": "England"
  }
}
```

DS.RESTSerializer

Multiple records...

```
HTTP GET /api/leagues
{
  "leagues": [
    { "id": 1, "name": "Premier League", "confederation": "UEFA",
    { "id": 2, "name": "Primera División España", "confederation":
    { "id": 3, "name": "Serie A", "confederation": "UEFA", "founde
    { "id": 4, "name": "Bundesliga", "confederation": "UEFA", "fou
  ]
}
```

SHOW ME THE CODE!

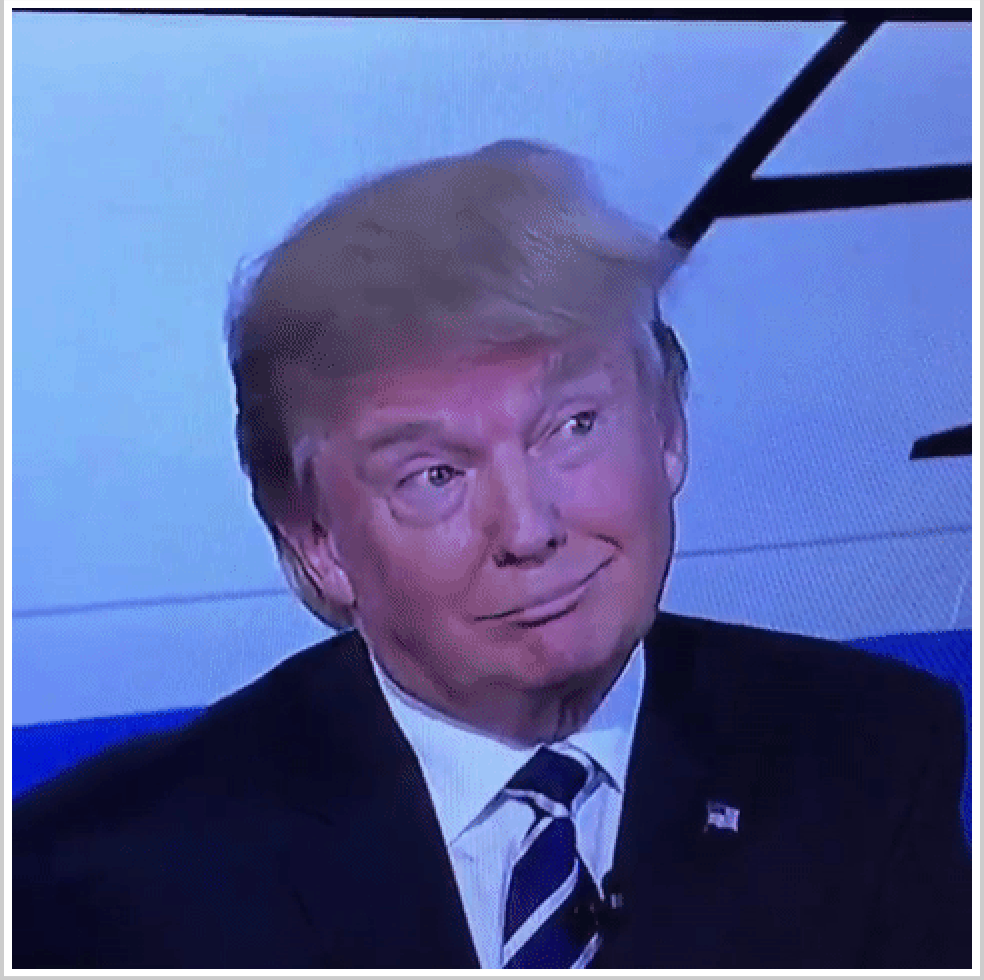


SUMMARY

1. Don't hate JSON API
2. Don't hate your API because it's not JSON API compliant
3. Embrace JSONAdapter / RESTAdapter / JSONSerializer / RESTSerializer

Thal

QUESTIONS?



THANKS

José San Gil - @jsangilve