Backbone.js + SoundCloud

The Next SoundCloud

Developer tooling

> **How is this different from JSConf?**
>
> There are tons of interesting things happening with JavaScript right now, but we're only focusing on one: building rich, dynamic applications on the web. No compile to js languages, no highly-optimized js runtimes, no Node.

- NodeJS
- Development server
- Build script

Templating: Handlebars

- Enforce good practices

- Pre-compiled

  - Faster to render

  - Smaller to deliver

- Custom helpers

Code style

```javascript
var PlayQueue = require('lib/play-queue'),
    Comments  = require('collections/comments'),
    View      = require('lib/view');

module.exports = View.extend({
  // ...
});
```

Code style

```
var PlayQueue = require('lib/play-queue'),
    Comments  = require('collections/comments'),
    View      = require('lib/view');

module.exports = View.extend({
  template: require('views/my-view.tmpl'),
  css     : require('views/my-view.css')
});
```

CSS AMD modules?

Yeah. CSS AMD modules.

```
define("views/my-view.css", function () {

    var style = document.createElement('style'),
        data = ".myView{margin-top: 20px;} .myV...";

    style.appendChild(
      document.createTextNode(data)
    );
    return style;
});
```

Views as Components

- **Independent** & Reuseable
- Can include subviews with subviews with subviews...
- Some are large, many small
- Must play nice

Getting data

View

Model

API

Have these attrs?
No? Go fetch.

Fetch my data

Data changed

#nextsoundcloud
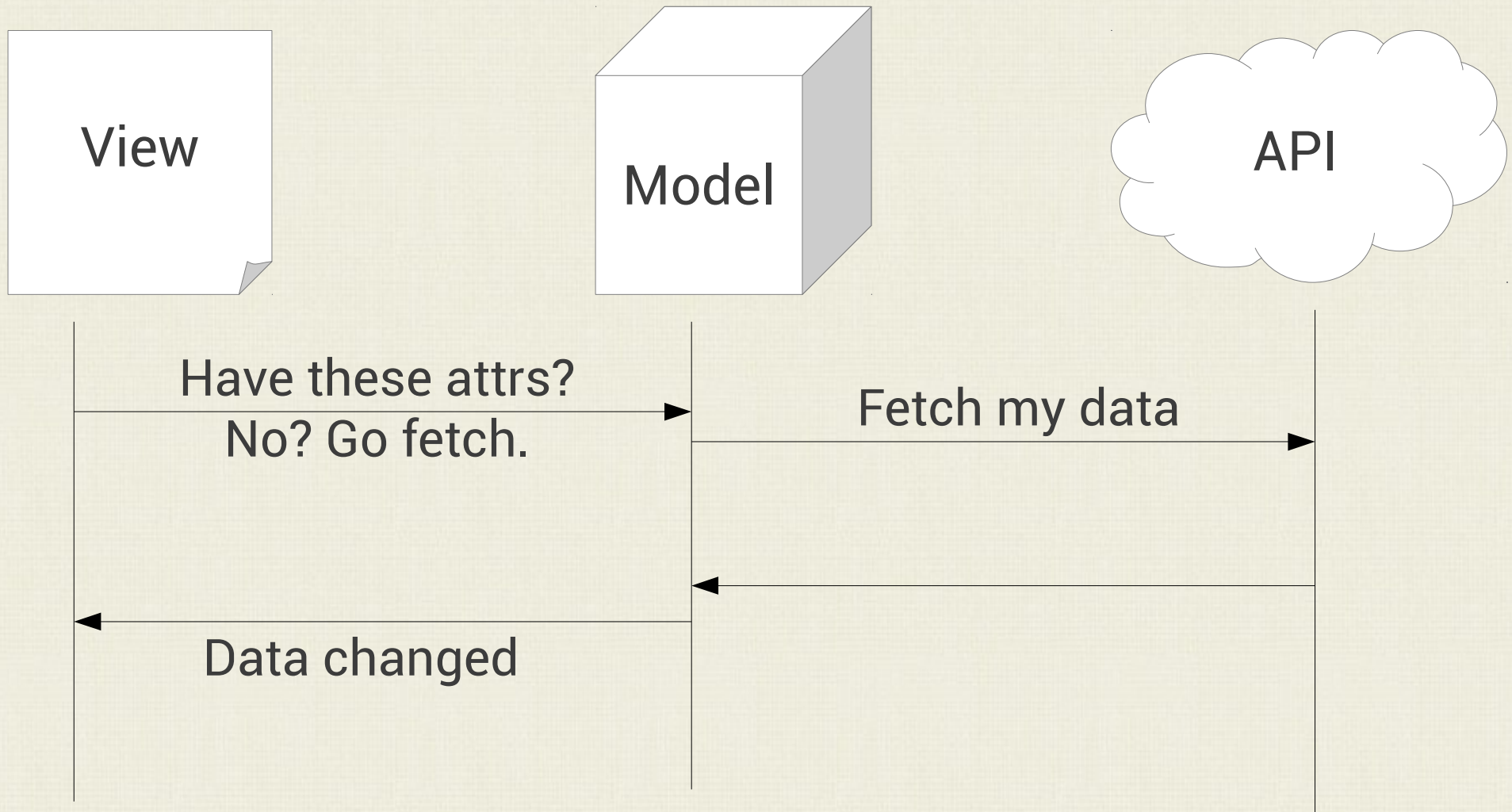
Views added via Templates
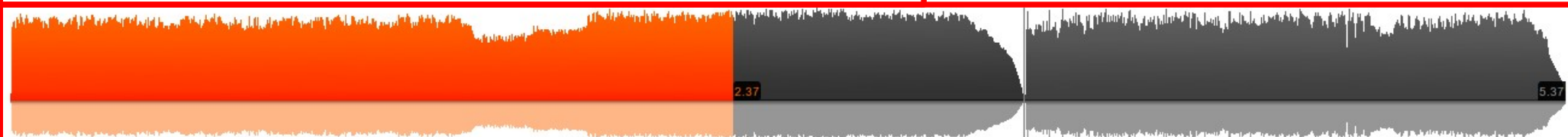
```
<div class="foo">
  {{view "views/sound/play-button"
    resource_id=sound_id
  }}
  {{view "views/user/user-badge"
    resource_id=user_id
  }}
</div>
```

All Tomorrows Parties ⅈ⌇ 2   5 hours

⏸ Nightmare Before Christmas curated by Shellac

2.37   5.37

↻ Repost   ♥ Like   Share

This set contains **2 sounds**, total time: **5.37**

All Tomorrows Parties
Mission Of Burma - Second Television
▶ 128   ♥ 5   ⏸ + ↻ ♥

All Tomorrows Parties
Mission Of Burma - Dust Devil
▶ 88   💬 1   ♥ 3

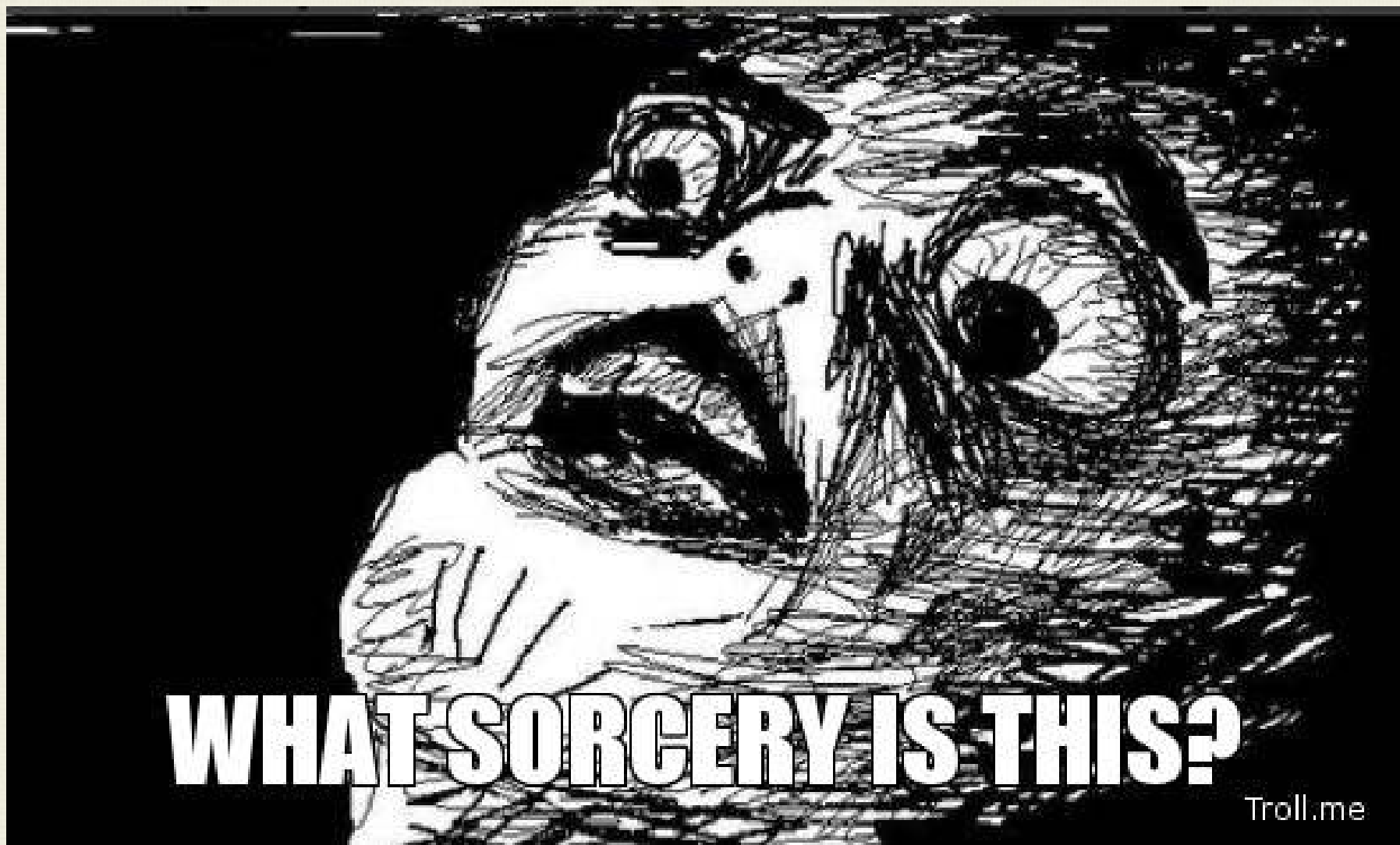**All Tomorrows Parties**
ⅈ⌇ 82 | 👥 41404   Following

Models

- Identity map behaviour

```
var soundA = new Sound({ id: 123, title: 'Foo' }),
    soundB = new Sound({ id: 123, genre: 'techno' });

soundA === soundB;    // true
soundB.get('title'); // 'Foo'
soundA.get('genre'); // 'techno'
```

#nextsoundcloud

How?

- Instance store is just an object (in essence)

- Override Model constructor

```
01 store = {};
02 Sound = Backbone.Model.extend({
03   constructor: function (attrs) {
04     var id = attrs.id;
05     if (store[id]) {
06       return store[id]; // ← return the other one
07     }
08     store[id] = this;
09     // regular instantiation...
10   }
11 });
```

#nextsoundcloud

Problem solved!

- Models fetched once, rendered many times

- Models + events can be used to synchronise views

- Nice side effect: full use of the response

Sub-resources in a response

```
https://api.soundcloud.com/tracks/52167545.json
{
  "id": 52167545,
  "user_id": 2,
  "duration": 71523,
  "user": {
    "id": 2,
    "permalink": "eric",
    "username": "Eric",
    "avatar_url": "https://i1.sndcdn..."
  },
  "created_with": {
    "id": 124,
    "name": "SoundCloud iOS"
  }
  // …
}
```

Releasing

- Instance store must let go at some point

- When a model is 'constructed': `usage++`

- When a model is 'released': `usage--`

- Periodically, remove unused models
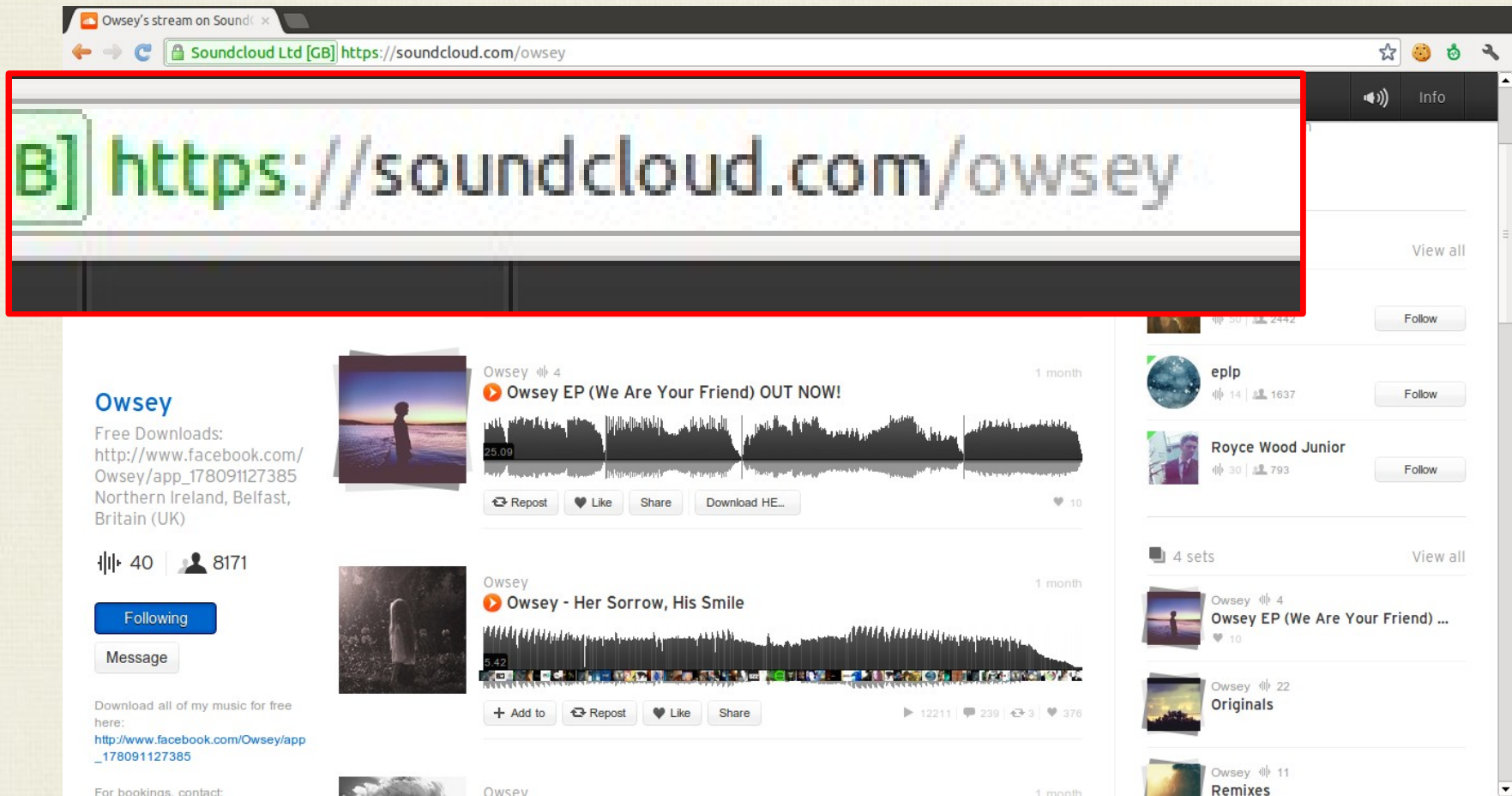
Backbone + SoundCloud

#nextsoundcloud

(Not so) Private beta