



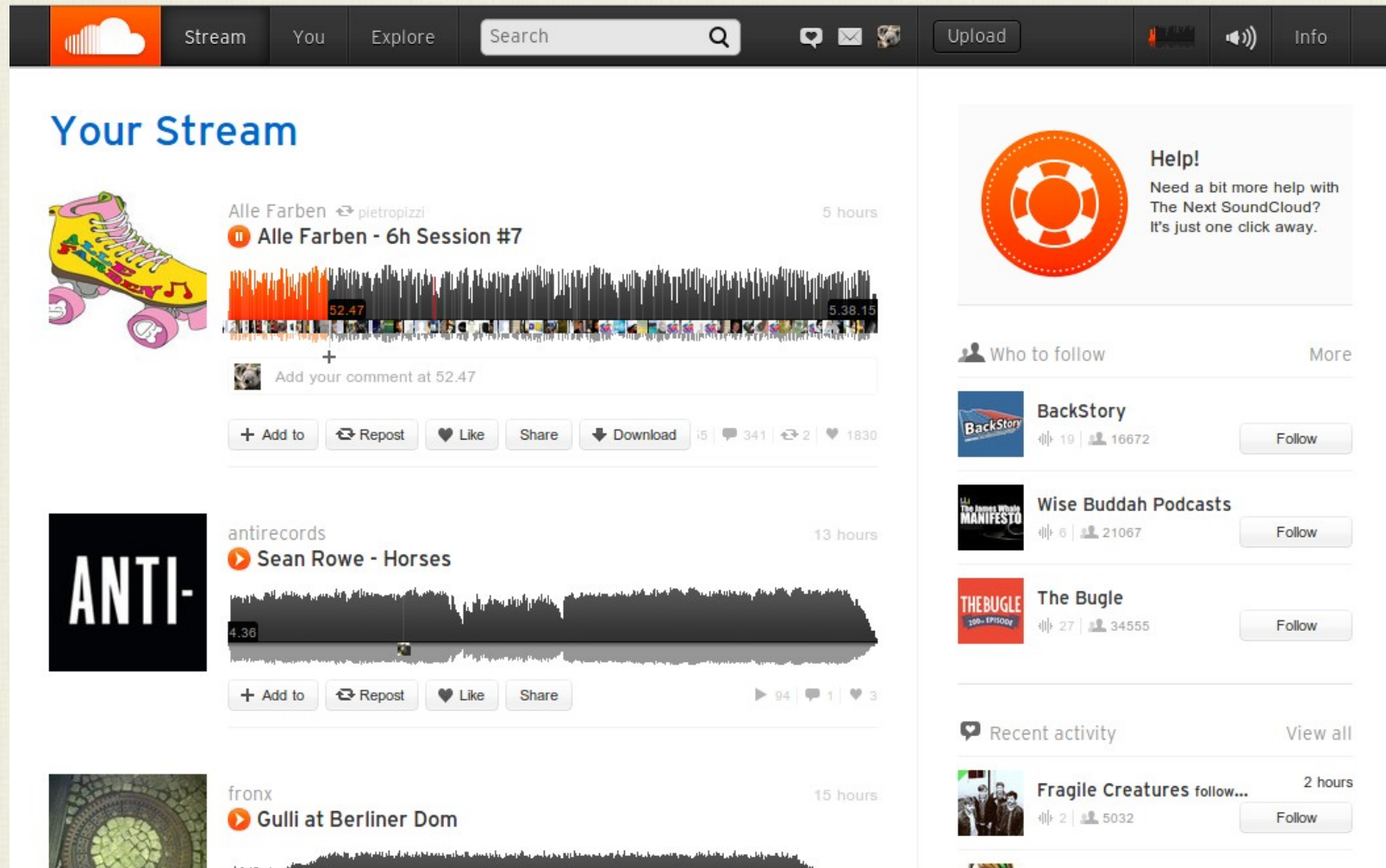
# Backbone.js + SoundCloud

@spadgos

#nextsoundcloud



# The Next SoundCloud



The screenshot displays the user interface of 'The Next SoundCloud'. At the top is a dark navigation bar with the SoundCloud logo, links for 'Stream', 'You', and 'Explore', a search bar, and icons for messages, email, and profile. A 'Upload' button is also present. The main content area is divided into two columns. The left column, titled 'Your Stream', features three audio tracks: 'Alle Farben - 6h Session #7' by Alle Farben (5 hours), 'Sean Rowe - Horses' by antirecords (13 hours), and 'Gulli at Berliner Dom' by fronx (15 hours). Each track includes a waveform, a play button, and interactive buttons for 'Add to', 'Repost', 'Like', 'Share', and 'Download'. The right column contains a 'Help!' section with a lifebuoy icon, a 'Who to follow' section listing 'BackStory', 'Wise Buddah Podcasts', and 'The Bugle', and a 'Recent activity' section showing a follow from 'Fragile Creatures'. The interface uses a clean, modern design with a color palette of orange, grey, and white.

@spadgos

#nextsoundcloud



# 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

```
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?



@spadgos

#nextsoundcloud



# 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



# 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 2

5 hours

## Nightmare Before Christmas curated by Shellac



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  
82 | 41404

Following

@spadgos

#nextsoundcloud





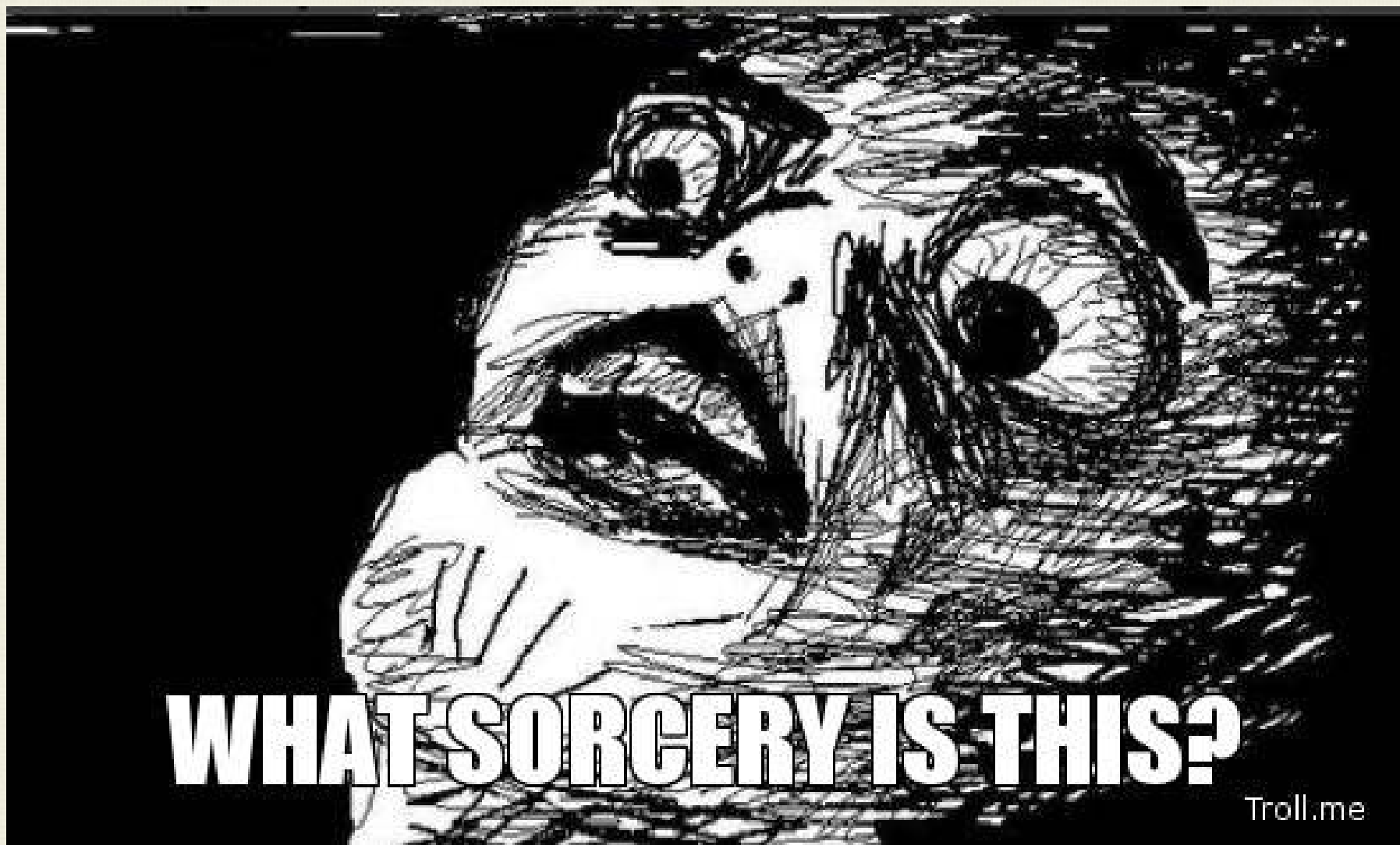
# Models

- 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'
```





@spadgos

#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 });
```





# 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"  
  }  
  // ...  
}
```

@spadgos

#nextsoundcloud



# 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

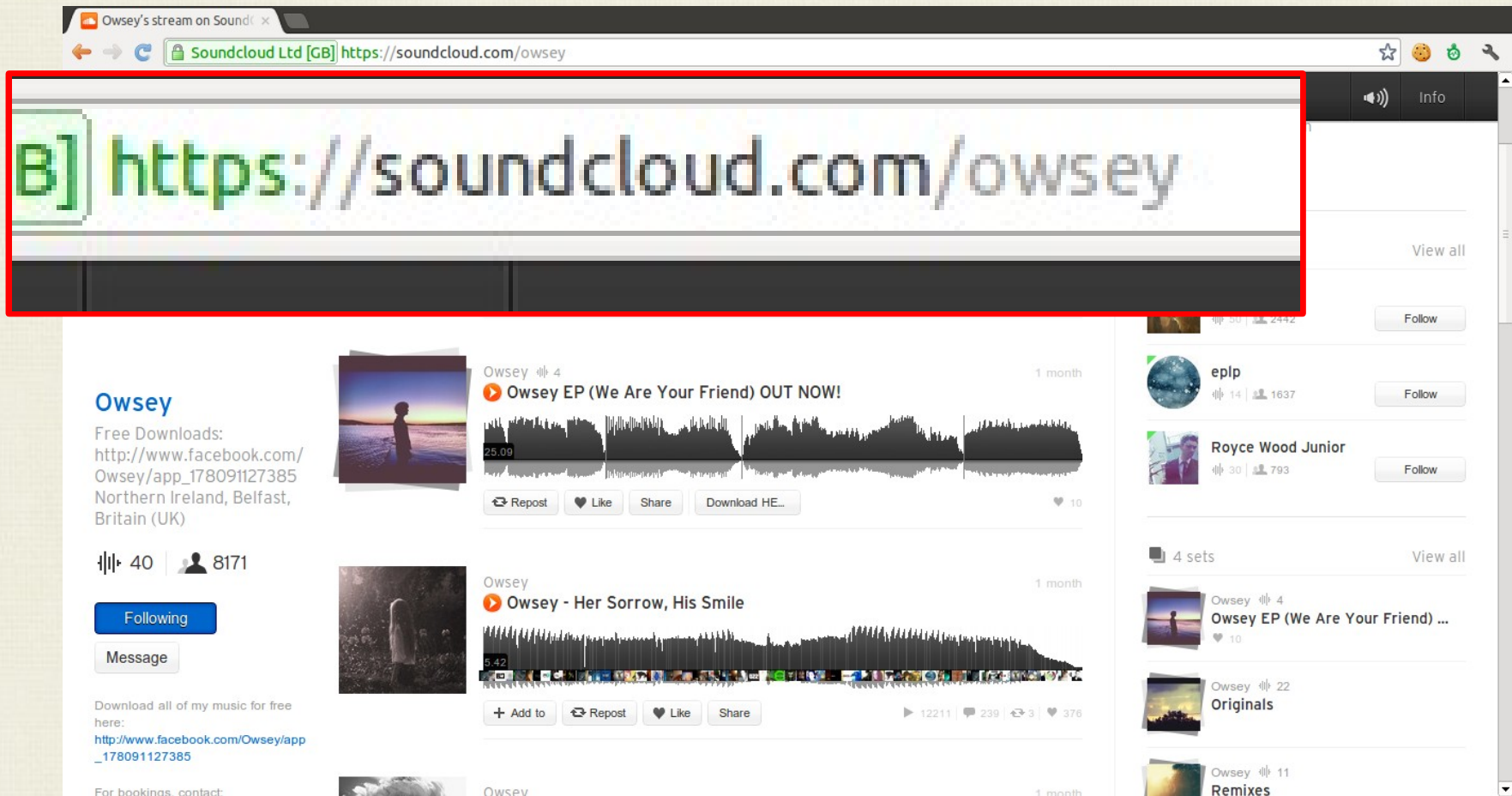


@spadgos

#nextsoundcloud



# (Not so) Private beta



@spadgos

#nextsoundcloud

