

Optimizing for Developer Delight



Rebecca Murphey • JSConf 2013



@rmurphey • rmurphey.com
bazaarvoice.com



★★★★★ 4.2 | 144 Reviews

144

Reviews

88

Questions

50

Answers

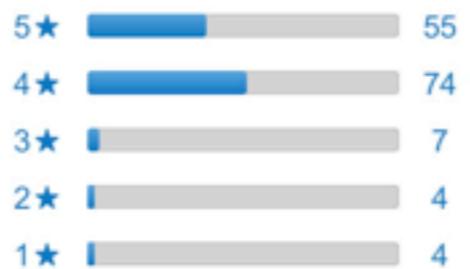
Search topics and reviews

Search

Reviews

[Write a review](#)

Rating Snapshot



Average Customer Ratings

Overall 4.2
Quality 3.9
Value 3.9

Most Helpful Favorable Review

★★★★★

waaaaatahmalon · 5 months ago

LOVED IT, WILL BUY AGAIN

FIVE STARS! this product was splendid! fabulous! incredible! fantastic! wowza! yowza! ...

[Show Full Review](#)

9 of 10 people found this helpful

Most Helpful Critical Review

★★★★★

NOTAJETSKI27 · 5 months ago

I don't know how to steer a jet ski!

I bought a jet ski about a month ago and I have no idea how to steer it. I occasionally just sit on ...

[Show Full Review](#)

5 of 6 people found this helpful

@rmurphrey • rmurphrey.com

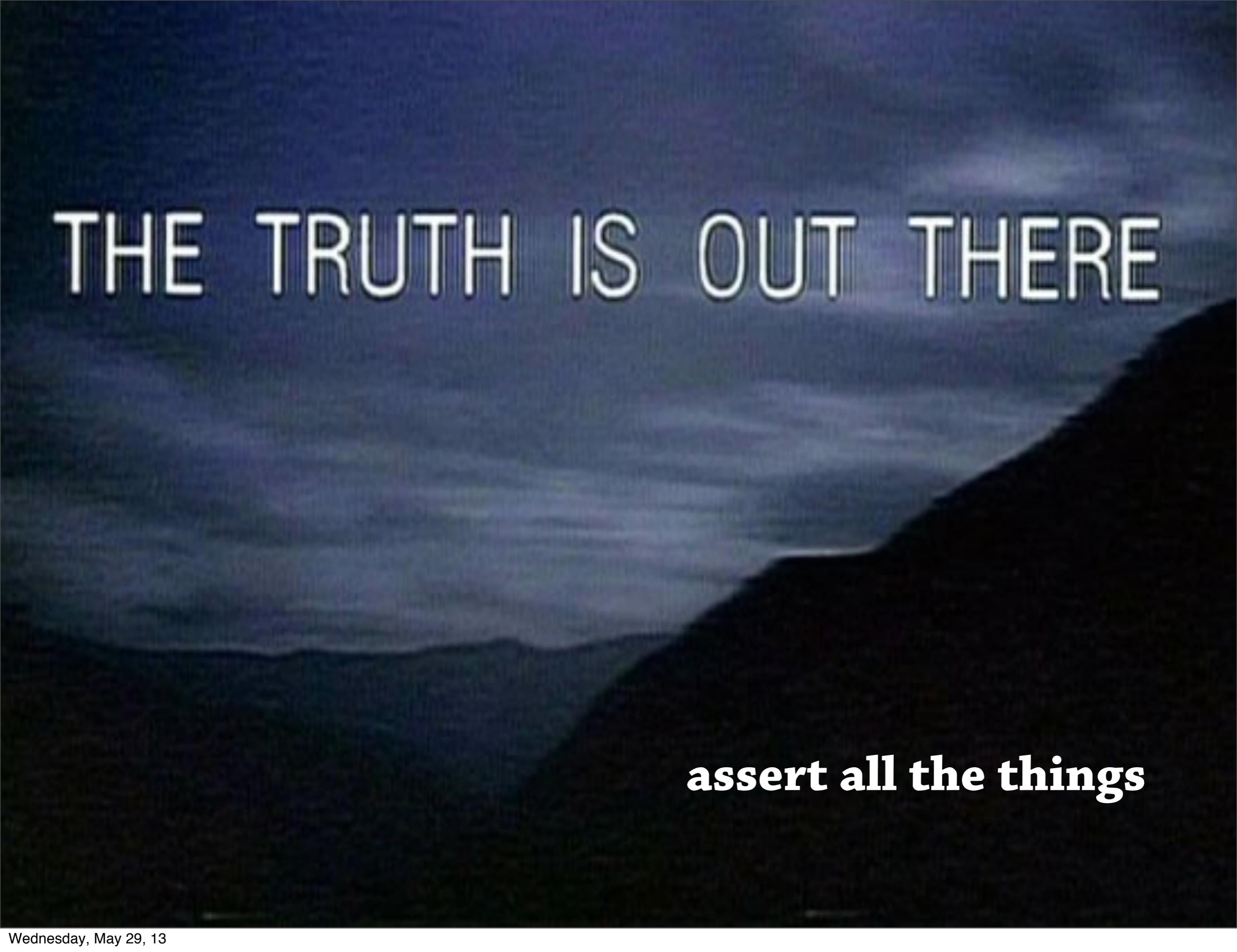
bazaarvoice.com

the problem

- large and growing codebase
- mission-critical application
- constantly expanding team
- mix of junior and senior devs
- mandate for rapid feature development

“developer delight”

- a clear path for getting up to speed
- points of entry for junior & senior devs
- as few surprises as possible: it just works
- isolated complexity
- easy development and debugging
- nothing more difficult than it “should” be



THE TRUTH IS OUT THERE

assert all the things



► **TypeError: Cannot read property '0' of undefined**

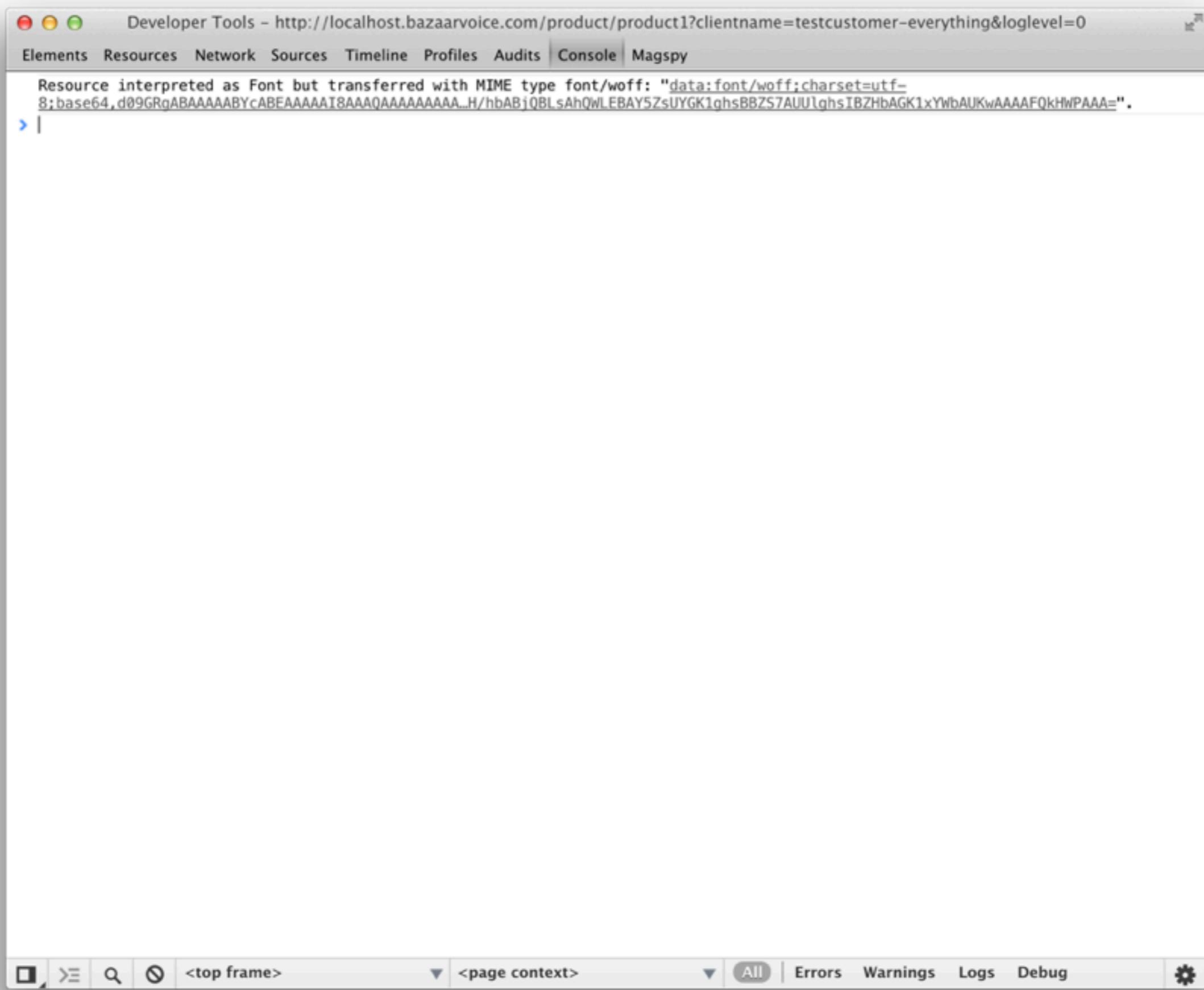
```
BVReporter.assert(  
  _(config).isObject(),  
  'config is object'  
);
```

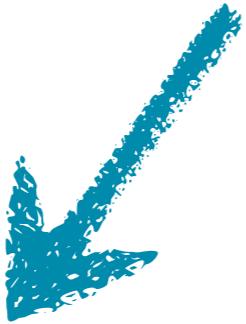
```
BVReporter.assert(  
  !_config.subjectId.isDefined(),  
  'config.subjectId is defined'  
);
```

```
assert : function (assertion, message) {  
  if (!assertion) {  
    throw new Error('Assertion failed: ' + message);  
  }  
}
```



log the lifecycle





```
BVReporter.group(BVReporter.INFO, 'Component Initialization');
```

```
_ (componentData).each(function (cd, componentName) {  
  var c = new BComponent(cd);  
  this.components[componentName] = c;  
}, this);
```

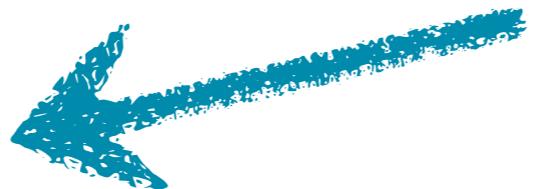
```
BVReporter.groupEnd(BVReporter.INFO, 'Component Initialization');
```



```
▼ Rendering top view BVRRContainer[reviewContentList1<Reviews>]
  ▼ View contentList attaching
    View contentList is acting as its own event manager.
  ▼ View contentFilterButton attaching
    View contentFilterButton found an event manager: BVRRContainer[reviewContent
     View: 4.017ms
  ▼ View dropdownable attaching
    View dropdownable found an event manager: BVRRContainer[reviewContentList1<F
     View: 1.979ms
  ▼ View headToHead attaching
    View headToHead found an event manager: BVRRContainer[reviewContentList1<Rev
  ▼ View contentItem attaching
    View contentItem found an event manager: BVRRContainer[reviewContentList1<Rev
    ▼ View avatar attaching
      View avatar found an event manager: BVRRContainer[reviewContentList1<Rev
       View: 5.587ms
    ▼ View feedback attaching
      View feedback found an event manager: BVRRContainer[reviewContentList1<Rev
       View: 1.224ms
    ▼ View secondaryContentList attaching
      View secondaryContentList found an event manager: BVRRContainer[reviewContentList1<Rev
      ▼ View secondaryContentItemCollection attaching
        View secondaryContentItemCollection found an event manager: BVRRContainer[reviewContentList1<Rev
         View: 2.738ms
      ▼ View secondaryContentItemCollection attaching
        View secondaryContentItemCollection found an event manager: BVRRContainer[reviewContentList1<Rev
```



eliminate temptation

```
define([
  'ENV', 
  'underscore',
  'bv/api/fetch'
], function (ENV, _, api) {
```

```
ENV.trigger('showLoading');
```

```
ENV.trigger(  
  'track:error',  
  new Error('Missing parent in Collection : ' + this.name)  
);
```

```
ENV.trigger('track:conversion', data, conversion);
```

```
if (
  !ENV.get('userTokenDfd') ||
  ENV.get('userTokenDfd').state() === 'resolved'
) {
  ENV.set({ userTokenDfd : $.Deferred() });
}
```

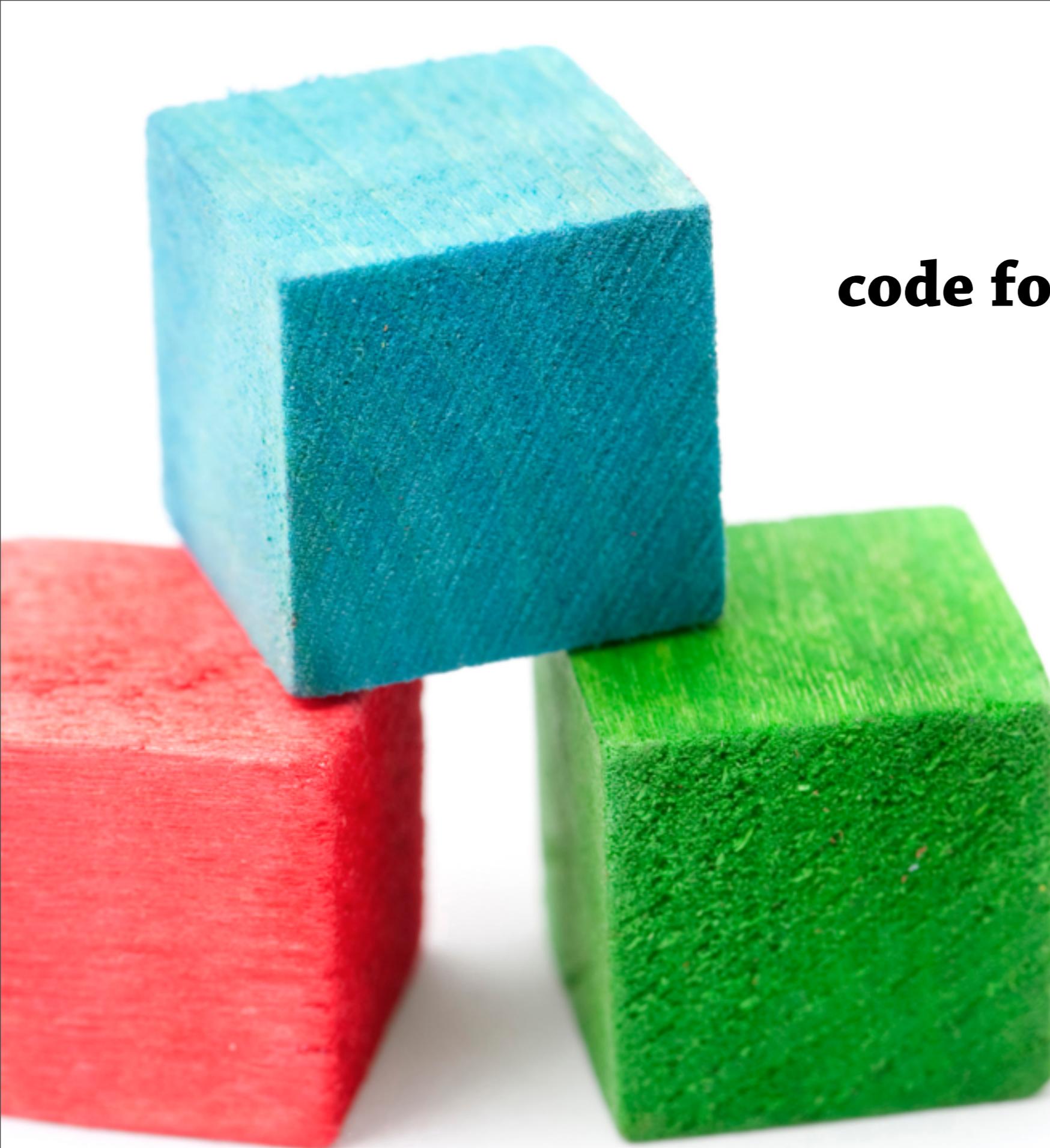
```
this.on('pageto', this.fetchPage, ENV);
```

```
ENV.trigger('showLoading');
```

```
LoadingOverlay.show();
```

```
ENV.trigger(  
  'track:error',  
  new Error('Missing parent in Collection : ' + this.name)  
);  
  
BVTracker.error( new Error(  
  'Missing parent in Collection : ' + this.name  
) );  
  
ENV.trigger('track:conversion', data, conversion);  
  
BVTracker.conversion(data, conversion);
```

```
this.on('pageto', this.fetchPage, ENV);  
this.subscribe('pageto', this.fetchPage);
```



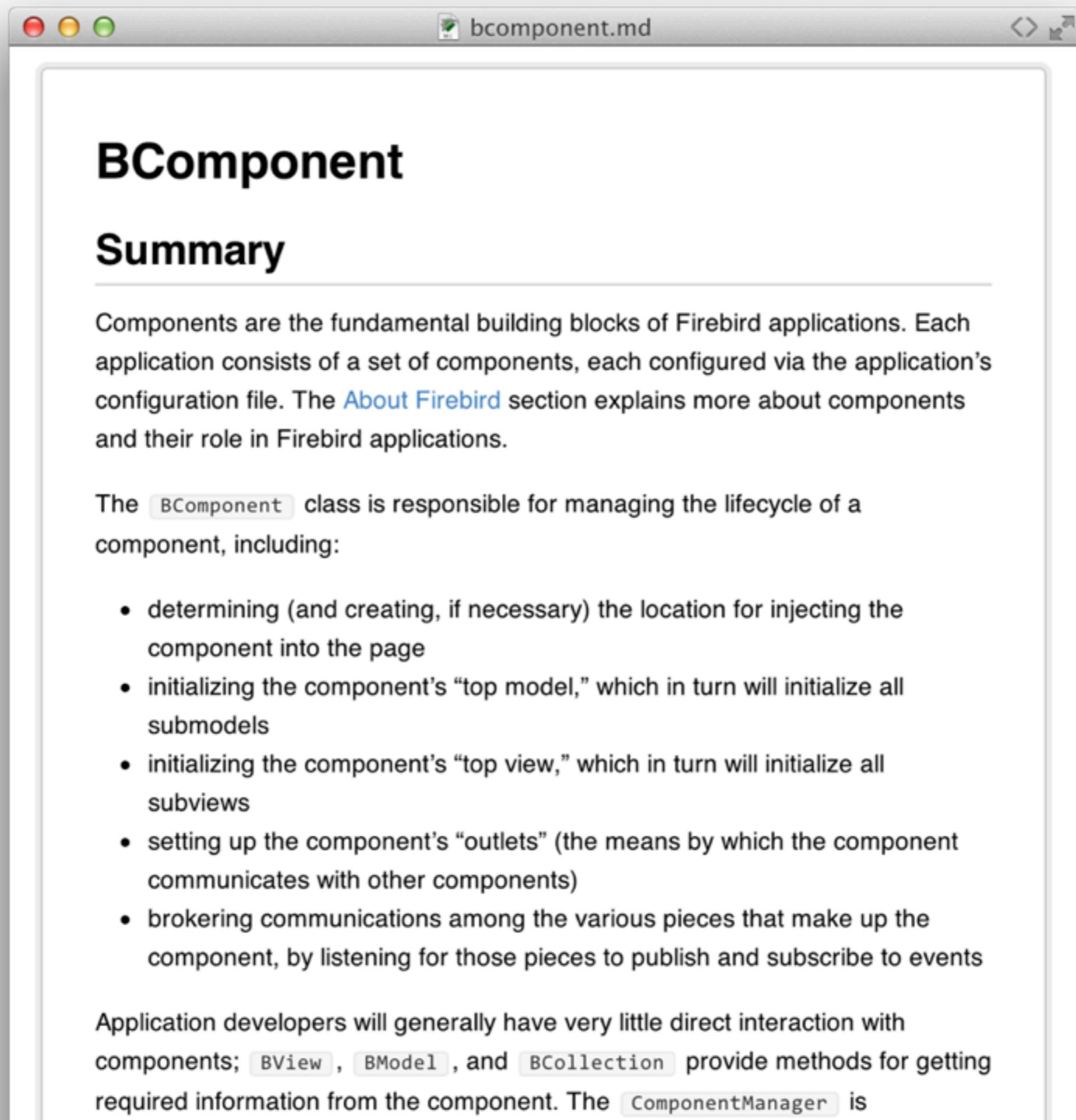
code for every concept

```
ENV.trigger(  
  'track:error',  
  new Error('Missing parent in Collection : ' + this.name)  
);
```

```
BVTracker.error( new Error(  
  'Missing parent in Collection : ' + this.name  
) );
```

```
"reviewContentList" : {  
  "features" : {  
    "self" : ["headToHead", "contentFilter", "contentItemC  
    "contentItem" : ["has:stars", "has:secondaryRatings",  
    "pagination" : ["ugcCount"],  
    "secondaryContentList" : ["secondaryContentItemCollect  
    "secondaryContentItem" : ["avatar", "feedback"],  
    "contentFilter" : ["has:filterButton"]  
  },  
  // ...  
},
```

```
var components = ENV.get('components');
components.each(function (component, index) {
  var view = init.initView(component, index);
  // ...
});
```



The screenshot shows a Mac OS X window titled "bcomponent.md". The window contains a summary of the BComponent class. It states that components are the fundamental building blocks of Firebird applications and that the BComponent class manages the lifecycle of a component, including determining the location for injecting the component into the page, initializing the component's "top model" and "top view", setting up "outlets", and brokering communications among various pieces. Application developers interact with components through BView, BModel, and BCollection, and the ComponentManager is used for required information.

BComponent

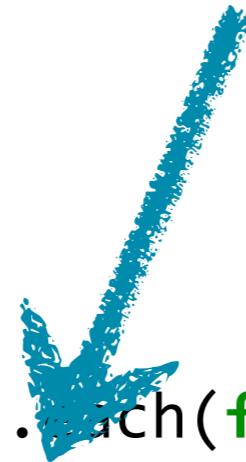
Summary

Components are the fundamental building blocks of Firebird applications. Each application consists of a set of components, each configured via the application's configuration file. The [About Firebird](#) section explains more about components and their role in Firebird applications.

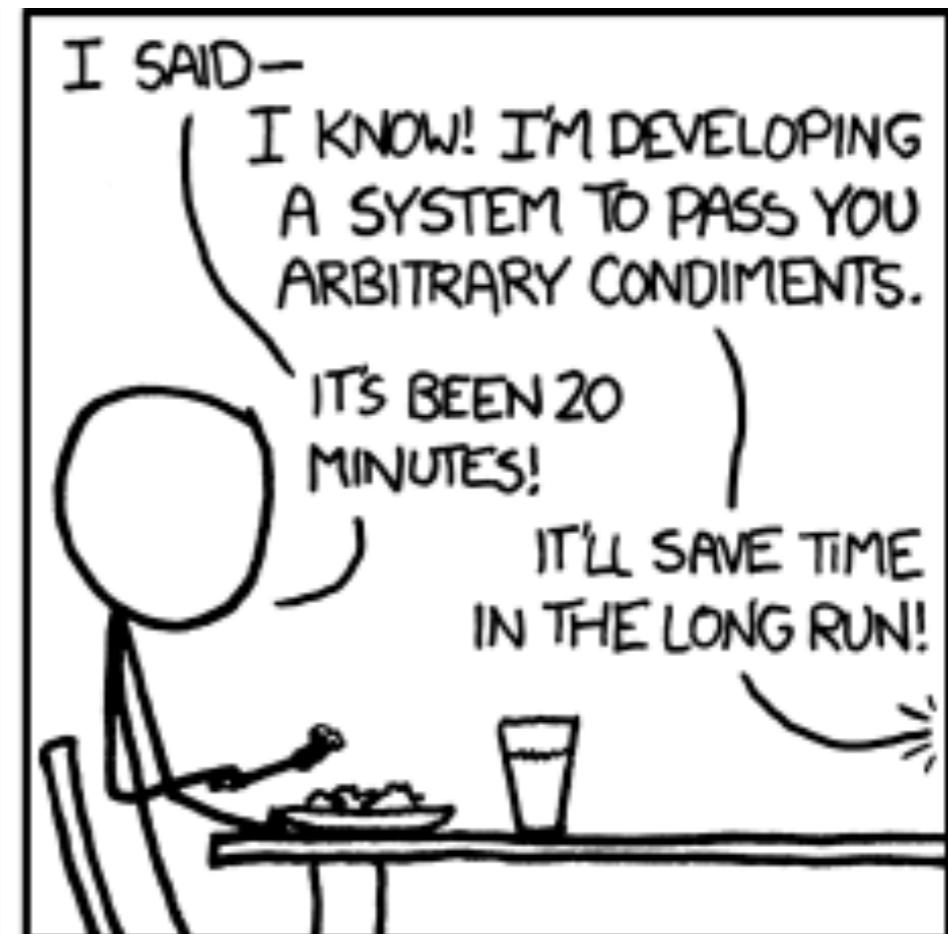
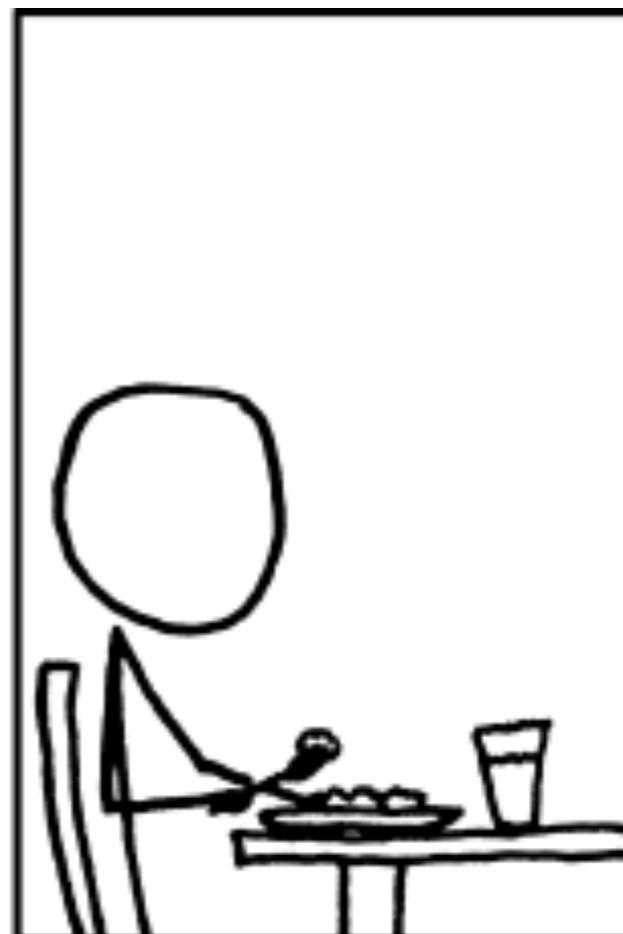
The `BComponent` class is responsible for managing the lifecycle of a component, including:

- determining (and creating, if necessary) the location for injecting the component into the page
- initializing the component's "top model," which in turn will initialize all submodels
- initializing the component's "top view," which in turn will initialize all subviews
- setting up the component's "outlets" (the means by which the component communicates with other components)
- brokering communications among the various pieces that make up the component, by listening for those pieces to publish and subscribe to events

Application developers will generally have very little direct interaction with components; `BView`, `BModel`, and `BCollection` provide methods for getting required information from the component. The `ComponentManager` is



```
_ (componentData).each(function (config, componentName) {  
  var c = new Component(config);  
  this.components[componentName] = c;  
  var view = c initView(index);  
  // ...  
}, this);
```



automate everything

 **F.B.I.**  fbi.firebird.bazaarvoice.com       

FIREBIRDINSPECTOR **ENVIRONMENT:** **QA**

1 [Home](#) / sup?

Browse a customer's packages

For each deployment zone, see which packages are currently installed for PREVIEW, STAGING, and PRODUCTION.

Go!

Load Firebird via a scout file

Preview a client's site using one of their currently deployed scout files.

Go!

Get Pacman package info

Get info about a package from the Pacman API.

Go!

Packages deployed since

See the full list of clients who have last deployed a package in a given date range.

  **Go!**

Dexter

Split apart batch requests.

Go!



2. rebecca.murphey@rmurphey-mbpro: ~/code/firebird (zsh)

→ **firebird** git:(master) grunt local-config

Running "local-config" task

path.existsSync is now called `fs.existsSync`.

>> Removing old configs

>> Using path /Users/rebecca.murphey/code/firebird

>> Updated config/local/nginx-dev.conf

>> Updated config/local/nginx-build.conf

>> Updated config/BV/Loyalty/API/Client/Leaderboard calls

>> Hey! If you aren't using `withClientName` method in config/local/nginx-build.conf
✓ should set client name using `withClientName` method

✓ should respect limit

✓ should set page number

Done, without errors.

→ **firebird** git:(master) grunt feature:myNewFeature

✓ should set client name using `withClientName` method

✓ should respect limit

✓ should set page number

Running "feature:myNewFeature" (feature) task

Test the parameter path.existsSync is now called `fs.existsSync`.

Do not enable links

>> Generated: static/dev/scripts/bv/ugc/feature/model/myNewFeature.hbs

✓ should map links to static/dev/scripts/bv/ugc/feature/view/myNewFeature.hbs

✓ no links

>> Generated: static/dev/template/myNewFeature.hbs

Enable links

✓ should map links to static/dev/scripts/bv/ugc/feature/view/myNewFeature.hbs

✓ should map links to static/dev/template/myNewFeature.hbs

✓ should be able to enable multiple exactly same links.

904 tests complete (2 seconds)



document

about-firebird.md

About Firebird

Firebird the Framework

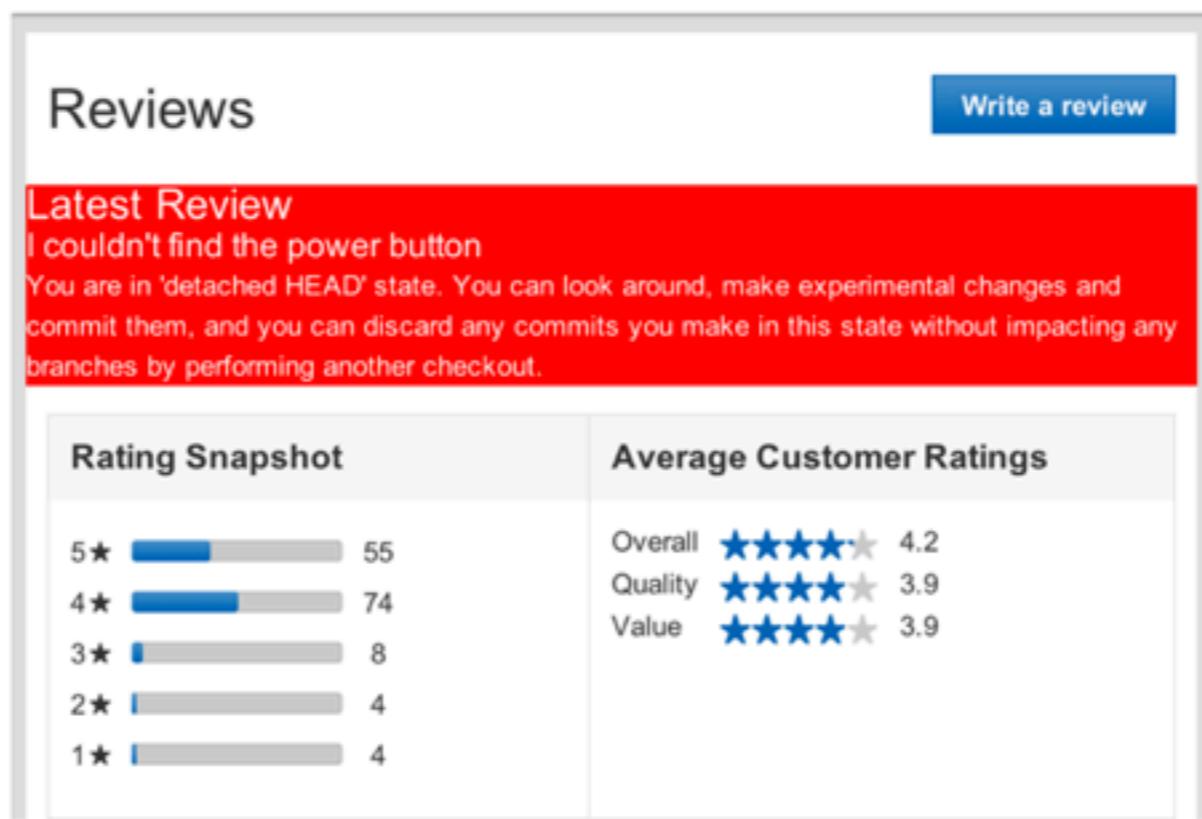
Firebird is the framework that powers the consumer-facing interface for Conversations 2013 (C2013) applications. It is a configuration-driven framework that uses a client-specific configuration file to generate the user interface for an application *at runtime*. It was designed with a few key concepts in mind:

- **Self-serve** A growing client base means we need to reduce the effort required to launch each new client. Whereas Firebird's predecessor provided a base application on top of which implementation engineers could apply customizations via JavaScript and CSS, a scalable solution requires that clients themselves be able to control how their application looks, feels, and behaves. Customers make these decisions via the Config Hub tool; Firebird takes the output of the Config Hub tool and turn it into an application.
- **Flexibility** In order to support the various decisions that customers can make, Firebird supports arbitrary combinations of arbitrary components, and arbitrary interactions between them. These combinations and interactions are described entirely in the application's configuration; there is never any custom code for a given application. Firebird turns an application's configuration into a working user interface *at runtime*.
- **Performance** While customers can choose arbitrary combinations of arbitrary components, Firebird must ensure that the production files for a given application contain *only* what the application requires in order to minimize the size of the file that a customer's users have to download. Generally, Firebird must also ensure that its applications have no adverse

example-feature.md

Creating a Feature

This document will step through the process of creating a simple feature and adding it to Firebird. The feature we will add is a “Latest Review” box, which will be displayed at the top of the review list. When we’re done, we’ll have something that looks like this:



The screenshot shows a product review page with a red “Latest Review” box. The box contains a single review: “I couldn’t find the power button”. Below the box is a “Rating Snapshot” table with the following data:

| Rating | Percentage |
|--------|------------|
| 5★ | 55 |
| 4★ | 74 |
| 3★ | 8 |
| 2★ | 4 |
| 1★ | 4 |

On the right, there is a “Average Customer Ratings” section with three rows: Overall (4.2), Quality (3.9), and Value (3.9), each accompanied by a 5-star rating icon.

Latest Review Feature

If you’re following along with this example, you should create a “feature branch” from the master branch of the repo:

```
git checkout -b example_feature
```

This way, you can commit as you go without affecting the master branch. If you

troubleshooting.md

Troubleshooting

When things go wrong, there are a few things to try:

- Open your browser console and see what's up. BVReporter does lots of logging – you can [change the log level](#) and perhaps get more information about what's going wrong.
- Take a look at the end of logs/nginx-error.log. It may point you in the right direction.
- Make sure you've got the latest from bazaarvoice/master. If the problem exists on a feature branch, see if the problem exists on master.

Common Issues

Local Testcustomer Config Problems

Symptoms

- Sample pages for a particular local testcustomer are not loading properly; can produce a variety of javascript errors

Fixes:

- Try `grunt copy-clients` – something about the config may have changed and you need to get the update.
- Check your browser dev tools Network log – you may have the dreaded SSL error. If one of the requests is red, right-click to view it in another tab, and then accept the SSL cert.

directory-structure.md

Firebird Directory Structure

This document describes the key directories for Firebird development. It is not an exhaustive description of all directories, just the ones you are likely to work with often.

/

The root of the repo contains several useful shell scripts. You will use **serveLocal.sh** to run a local instance of Firebird. You will use **vagrant.sh** if you want to use [vagrant](#) to run Firebird.

The root also contains **grunt.js**, which loads all of our [Grunt](#) tasks; and **.jshintrc**, which some [JSHint](#) editor plugins will use to verify that code complies with our formatting requirements.

build

This directory contains most of the files concerned with generating Firebird builds. It also contains the **build/tasks** directory, which contains all of our Grunt tasks, and the **build/templates** directory, which contains all of the templates associated with the Grunt tasks. Our Grunt tasks are not well documented at the moment, but you may be interested in the [scaffolding](#) task if you're working on feature development.

config

This directory contains configs for various environments. When you set up

contributing-to-firebird.md

Contributing to Firebird

This document assumes you set up Firebird according to [these instructions](#).

You should never do work in the master branch, and you should keep the master branch updated from the main repo by pulling often – things change fast and it's easy to get behind.

```
git checkout master
git pull bv master      # assumes your bazaarvoice remote is named "bv"
git push origin master  # assumes your remote is named "origin"
```

When you are ready to start working on a new feature, you should create a branch.

```
git checkout master          # make sure you're on master first
git pull bv master          # assumes your bazaarvoice remote is named "bv"
git checkout -b my_new_branch  # create a new branch based on master
```

You can update your branch with the latest master, which is a good thing to do so they don't get out of sync.

```
git checkout master
git pull bv master
git checkout my_new_branch
git merge master
```

When you are ready for others to see your work, you should push your branch

changelog.md

Changelog

This page is to document changes to Firebird that are *developer facing*.

The changelogs are written by hand. If the git command from the last entry is `git log abcdef..ghijkl --merges` then you should run `git log ghijkl..HEAD --merges` to see what's new, and then look to see what the last commit is in order to come up with the git command you'll put into the new entry.

2013-05-15 `git log 41b7d8..ebcd3f --merges`

- Added a test to verify serverrender isn't broken; this test only runs locally, not in TeamCity.
- Added `enableLinks` option to the paragraphize Handlebars helper, which will linkify any URLs in the text. This should only be used for client content.
- Added utility to provide a standard way to get IE version information.
- BComponent now waits for all models (not just top model) to call `.dataReady()` before rendering the component's view. This fixes an issue where headToHead was not rendering because its data was not ready at component render time.
- Fixed an issue where publish/subscribe didn't include content type information.
- Tests can now specify `.implementation()` to use something other than `Main Implementation`

2013-05-04 `git log 312fd8..41b7d8 --merges`

firebird/docs at master · b...  GitHub, Inc. [US] <https://github.com/bazaarvoice/firebird/tree/master/docs>        

This repository   Explore Gist Blog Help  rmurphrey   

bazaarvoice / firebird    Pull Request  Unwatch  Star 18  Fork 41

Code Network Pull Requests 1 Issues 48 Wiki Graphs Settings

rise from the ashes 

 [Clone in Mac](#)  [ZIP](#)   <git@github.com:bazaarvoice/firebird.git>  [Read+Write access](#)

branch: **master**  Files [Commits](#) [Branches 51](#) [Tags 1](#)

firebird / docs /

 [History](#)

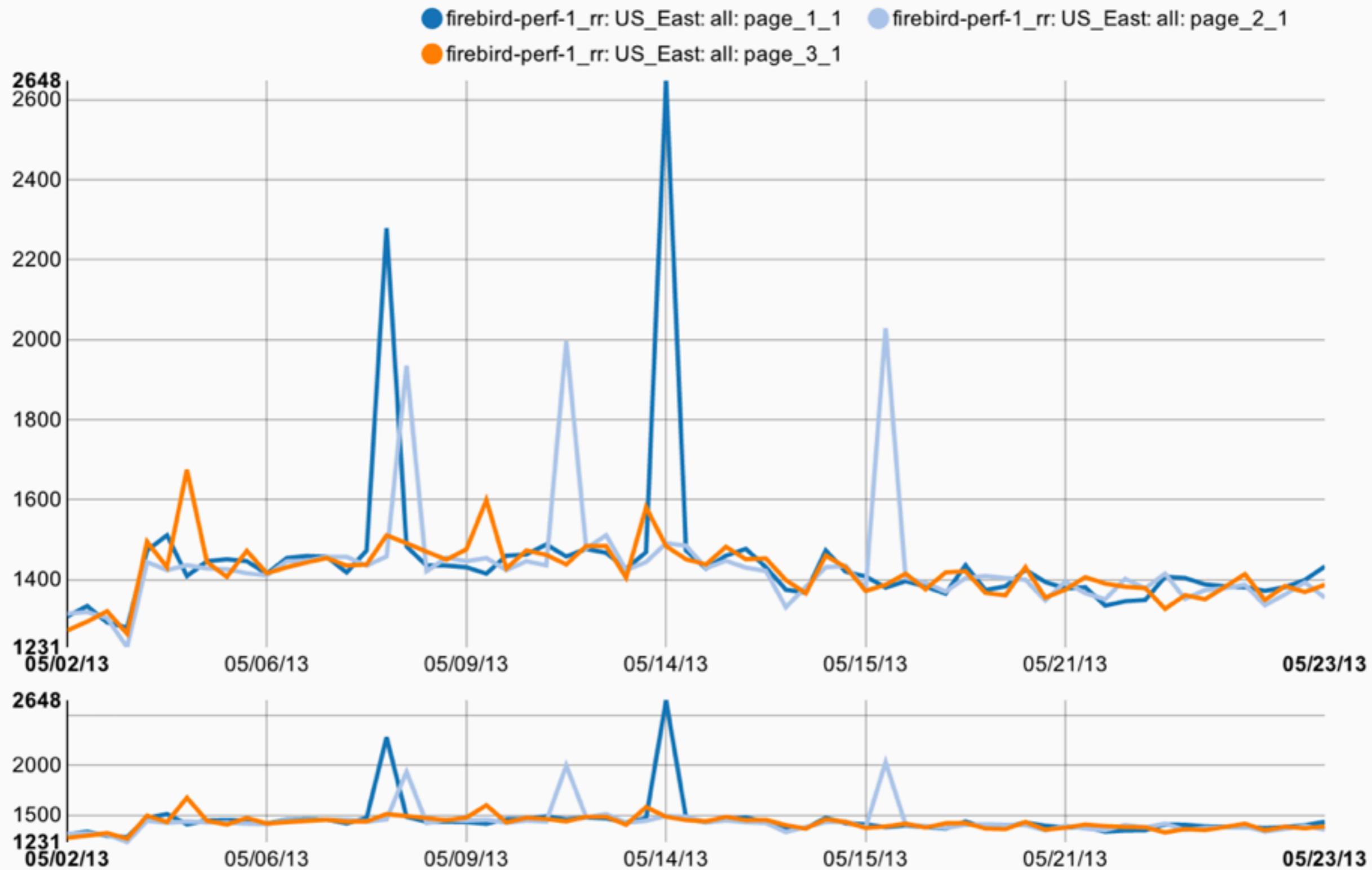
| doc update | | |
|---|---------------------|--|
|  owiber | authored 2 days ago | latest commit dc6aa7ab3a  |
| .. | | |
|  about-firebird.md | 25 days ago | first stab at "about firebird" [rmurphey] |
|  bcollection.md | 22 days ago | collection docs [rmurphey] |
|  bcomponent.md | 22 days ago | component, component manager docs [rmurphey] |
|  bdata.md | 8 days ago | more docs about how the data stuff works, plus tests [rmurphey] |
|  bevented.md | 22 days ago | fix typo [rmurphey] |
|  bmodel.md | 9 days ago | attachDataReady -> setupDataProcessing [rmurphey] |
|  bmodelbase.md | 21 days ago | docs updates [rmurphey] |
|  bv-ui-call-lifecycle.md | a month ago | better name for file [rmurphey] |
|  bview.md | 13 days ago | more ENV removal [rmurphey] |
|  bvreporter-logging.md | a month ago | Add docs. [savetheclocktower] |



**measure progress
(and side effects)**

| File | Halstead | Cyclomatic | Effort | Volume | Vocabulary | Complexity | Score |
|--|--------------------|------------|-----------|--------|------------|------------|-------|
| static/dev/scripts/bv/util/formMemory.js | 27.5 | | 13612 | 495.00 | 22 | 117.78 | |
| static/dev/scripts/bv/util/geoLocation.js | | | | | | 121.10 | |
| static/dev/scripts/bv/util/grammr.js | | | | | | 134.68 | |
| static/dev/scripts/bv/util/janrain.js | | | | | | 115.98 | |
| static/dev/scripts/bv/util/janrain.js:19 - <anonymous> is too complicated | | | | | | | |
| static/dev/scripts/bv/util/loadingOverlay.js | 24.32857142857143 | 5 | 33833 | 1390.7 | 48 | 128.49 | |
| static/dev/scripts/bv/util/localeMapping.js | | | | | | 65.540 | |
| static/dev/scripts/bv/util/normalizeParams.js | | | | | | 137.85 | |
| static/dev/scripts/bv/util/pageInfo.js | | | | | | 99.865 | |
| static/dev/scripts/bv/util/pageInfo.js:21 - getPageInfo is too complicated | | | | | | | |
| static/dev/scripts/bv/util/queryrouter.js | 35.52173913043478 | 7 | 28157 | 792.67 | 42 | 129.17 | |
| static/dev/scripts/bv/util/referenceCache.js | | | | | | 132.18 | |
| static/dev/scripts/bv/util/session.js | | | | | | 125.13 | |
| static/dev/scripts/bv/util/socialAvatar.js | | | | | | 131.29 | |
| static/dev/scripts/bv/util/spiffybox.js | | | | | | 109.55 | |
| static/dev/scripts/bv/util/spiffybox.js:131 - open is too complicated | | | | | | | |
| static/dev/scripts/bv/util/spiffybox.js:131 - open is too complicated | 30.079646017699115 | 16 | 1.2453e+5 | 4139.9 | | | |

Time to PageView Event (Repeat View): rr

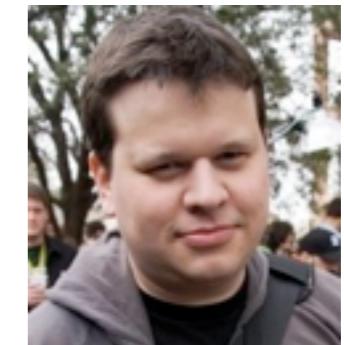


“I wish I’d had this documentation when I was building that feature last week.”

“So I can just add my own logging and they’ll show up in these groups automatically?”

“The assertion stuff just caught an error in my code that I would have missed.”

are we there yet?



@rmurphrey • rmurphrey.com

bazaarvoice.com