



# Review Board

Christian Hammond - [chipx86@chipx86.com](mailto:chipx86@chipx86.com)



# Review Board

- An awesome web-based code review tool for companies and open source projects.
- Developed by Christian Hammond and David Trowbridge at VMware in order to improve the code review process.
- Makes code reviews and patch contributions much easier for maintainers and contributors.



# What is code review?

“Code review is systematic examination (often as peer review) of computer source code intended to find and fix mistakes overlooked in the initial development phase, improving both the overall quality of software and the developers' skills.”

— Wikipedia

## Ideally...

- *Humans* look over your code to find your mistakes.
- *Mistakes are caught during* the development phase *before* anybody is affected.
- Helps contributors to provide better patches.
- *Anybody* can review code, not just core developers. ████████

# Code review in Open Source



- Common in larger projects
- Usually only contributors' code is reviewed
- Maintainers assume their code is fine
- Methods:
  - Bugzilla / Trac (Patch tickets)
  - Patches mailing list

# Issues with open source code review



## Raw diffs

- Only a few lines of context
- Non-unified diffs
- Not always easy to describe *where* in a diff to make changes
- Hard to deal with iterations of diffs for large changes

## Bug trackers

- Patches may address several non-duplicate bugs
- Patches are scattered and review requests hard to find

## Mailing Lists

- Not fine-grained enough for some contributors

## Review Board solves this...



- ... with the power of love.
- And some awesome tools.
- Guaranteed to satisfy you and your contributors, or your money back.



# Quick overview of Review Board

- Open source, under the MIT license
- Written in Python using Django
- 2 lead developers
- 34 contributors (and rising)
- Used by over 30 companies/projects, including...
  - Beatport
  - Leapfrog Online
  - Tripwire, Inc.
  - Datacomp
  - NetLogic Microsystems
  - VMware, Inc.
  - Digium
  - PHInaX
  - Wesabe
  - Hedgehog Lab
  - Review Board
  - Yahoo! Web Search
  - IronPort
  - rPath

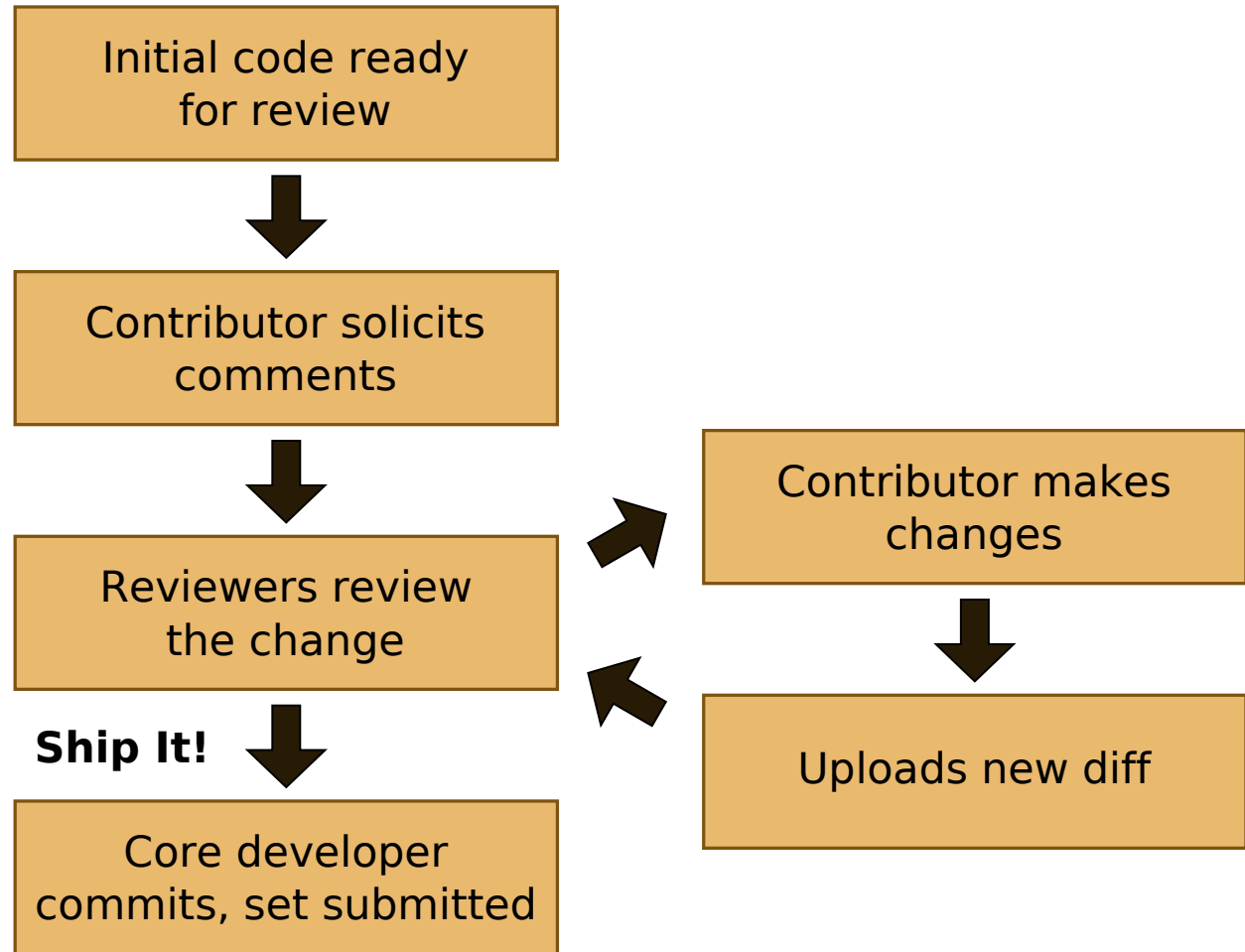


## Supports projects of any size

- Scales from small projects to large umbrella projects
- Works with multiple repositories of many types
  - CVS
  - Bazaar
  - Perforce
  - Git
  - Subversion
  - Mercurial
- Review requests can belong to one or more groups
- Each group can have a dedicated reviews mailing list
- Users can watch multiple groups in the Dashboard
- Quickly search all review requests



# Workflow





# Dashboard

## Easily keep track of...

- Outgoing review requests
- Incoming review requests
  - All indirectly to you (through a group)
  - All directly to you
  - To groups you belong to
  - Groups you're watching
- Starred review requests

★ Starred Reviews	0	All Incoming Review		
Outgoing Reviews	0	✓	...	★ Summary
Incoming Reviews	13	...	★	Make post-r
To Me	1			git-svn is to
reviewboard	13		★	Issue 436: I
			★	Update to p
				with Python
			★	Create a JS
			★	Add support
				post-review
			★	Fix ajax tog
			★	Issue 451: I
		...	★	Allow Usern
			★	Fix some iss
			★	Single-colur
			★	Bazaar SCM
			★	Issue 106 --
				dashboard
			★	Improve jsc





# Review requests

- The central concept of Review Board
- Never deleted
- Can be marked submitted
- Store a variety of information, including:
  - Summary
  - Description
  - Testing done
  - Related bugs (with links to the bug tracker)
  - Target reviewers (groups and people)
  - Related screenshots



# Reviews

- All reviews for a review request shown on one page
- One box per review
- All discussions on a review exist within the review box
- Full context for each comment in a review
  - Lines of the diff being discussed
  - Area of the screenshot



# Reviews

**Josh Stone**

Posted 4 days, 22 hours ago (April 5th, 2008, 4:48 p.m.)

[Add comment](#)

[/trunk/reviewboard/contrib/tools/post-review](#) (Diff revision 1)

```
853         version_parts =  
re.search('version  
(\d+)\.(\d+)\.(\d+)\.(\d+)',
```

Please note that not all git releases have a four-part version number. For instance, my Cygwin installation has plain-old 1.5.4. You're not using the fourth part anyway, so why not leave that out of the expression?

**Marc Hedlund** 1 day, 17 hours ago (April 8th, 2008, 9:09 p.m.)

Good point. Fixed.

[Add comment](#)



# Diff viewer

## Powerful diff viewer with...

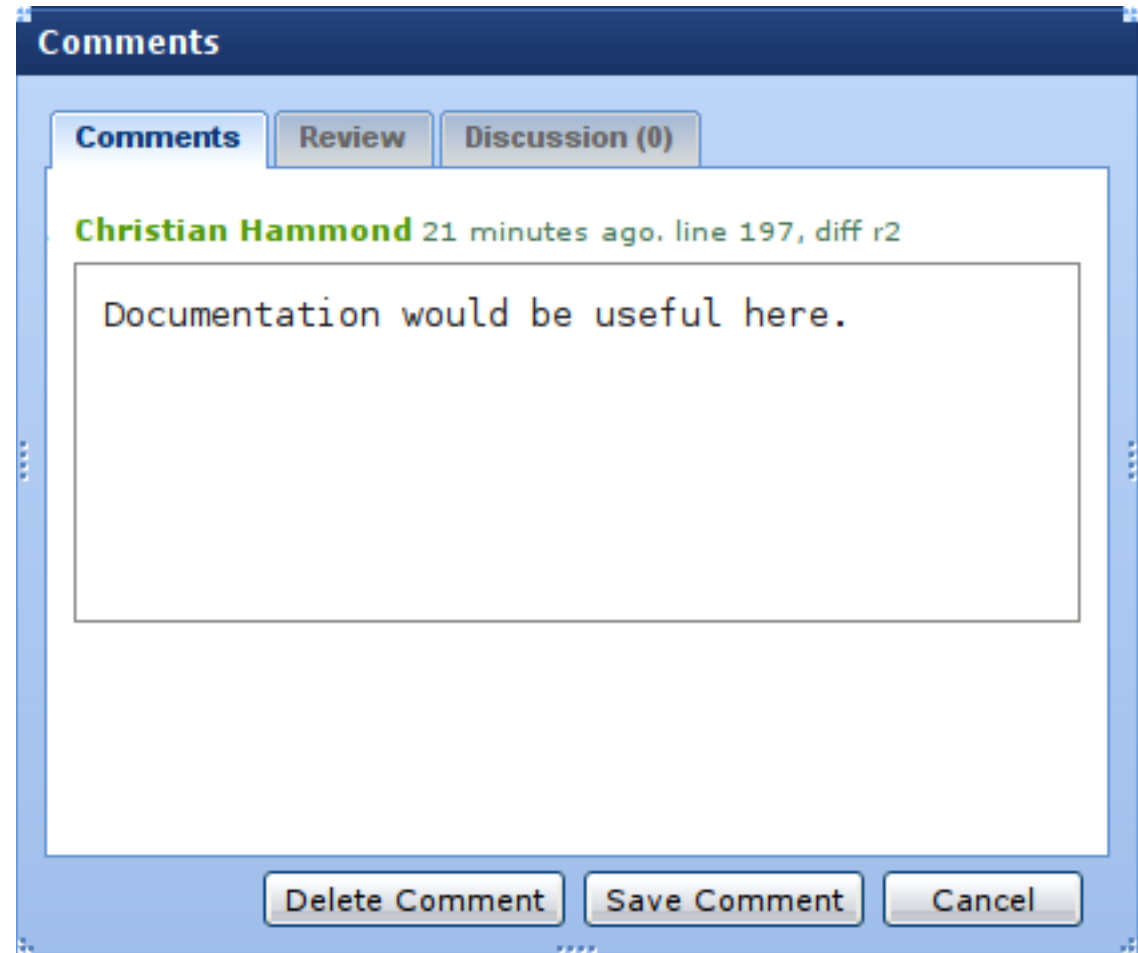
- Syntax highlighting
- Inline commenting
- Inter-line diffs (what part of a line changed)
- Expandable/collapseable context
- Every diff revision stored
- Interdiffs

```
190  
191 return render_to_response(template_name, RequestContext(request, {  
192     'review_requests': review_requests,  
193     'title': title,  
194     'view': view,  
195     'group': group  
196     })))  
199
```



## Reviewing diffs

- Comment directly on lines of code in the diff viewer
- Comments can span multiple lines
- Review requests can have comments on multiple diffs (even interdiffs)
- Commented lines will appear directly in the review





# Reviewing screenshots

- Comment directly on screenshots
- Click and drag to select a region
- Bordered region indicates a comment
- Commented regions appear inline in the review



# Additional features



## **E-mail support**

- Reviewers receive e-mails for review requests and discussions

## **Reports view**

- See what you put up for review and reviewed for others during a span of time
- Useful for status reports

## **Search**

- Full text search for review requests

## **Easter eggs**



## Works with many setups

### **Web Servers**

- Apache
- lighttpd

### **Databases**

- MySQL
- PostgreSQL
- SQLite

### **Additional requirements**

- memcached

### **Source Code Repositories**

- Bazaar
- CVS
- Git
- Mercurial
- Perforce
- Subversion



# Extendable

## **JSON API**

- post-review tool
- Third-party IDE plugins in development

## **Custom web UIs**

- Experimental iPhone UI

## **Extension support coming in 2.0**

- Customize pages
- Add to the dashboard
- Add to the review request page
- Perform actions on review request or review updates



## post-review tool

- Command line script to create and update review requests.
- Generates the diff files for you and uploads them in one go.

- Easy to use:

```
$ post-review -c          - Creates a new review request  
$ post-review -r review_request_id - Updates an existing review request
```

- Easy to set up with projects:

### **Subversion:**

```
$ svn propset reviewboard:url http://reviews.example.com .
```

### **Other repository types:**

```
$ echo "REVIEWBOARD_URL = 'http://reviews.example.com' " > .reviewboardrc
```

# post-review installation and requirements



- Requires Python 2.4+ and simplejson
- Easily installs on Linux, Windows and MacOS X
- Just copy the script to a bin directory and you're ready to go!



## post-review tips and tricks

- Allows for posting a review request based on a range of committed revisions (useful for reviewing code on branches)

```
$ post-review --revision-range=1042:1045
```

- Pre-fill review request fields when posting

```
$ post-review --summary= "<summary text >"  
  --description= "<description text >"  
  --testing-done= "<testing done text >"
```

- Auto-generate review requests for a user (requires that the script logs in as a user with special permissions).

```
$ post-review --submit-as= <username >
```

- Automatically publish the review request

```
$ post-review --publish
```



## post-review and changesets

- Even easier!
- Sets the summary, description, and other fields for you
- Ties together change numbers and review requests
- So far only Perforce is supported

```
$ post-review 12345
```



# Setup

- Installation could be easier...
  - Dealing with server configuration files mostly
  - Not all dependencies are packaged on most distributions yet
- But we're working on it!
  - Building a Review Board VM for 1.0
  - Working to get Ubuntu packages (and hopefully other distros later)
- Follow our GettingStarted wiki page for step-by-step instructions



# Administration

- Administration happens in the Django-provided Admin UI
- Adding users, review groups, etc. is very easy
- Improving the Admin UI for 2.0
  - Upgrade notices
  - Project news
  - Extension management
  - Site customization
  - Server statistics



# Administration

## Review Board administration

Welcome, Admin. Document

### Site administration

Auth		
Groups	<a href="#">+ Add</a>	<a href="#">Change</a>
Users	<a href="#">+ Add</a>	<a href="#">Change</a>
Sites		
Sites	<a href="#">+ Add</a>	<a href="#">Change</a>
Accounts		
Profiles	<a href="#">+ Add</a>	<a href="#">Change</a>
Diffviewer		
Diff set historys	<a href="#">+ Add</a>	<a href="#">Change</a>
Diff sets	<a href="#">+ Add</a>	<a href="#">Change</a>
File diffs	<a href="#">+ Add</a>	<a href="#">Change</a>
Reviews		
Comments	<a href="#">+ Add</a>	<a href="#">Change</a>
Default reviewers	<a href="#">+ Add</a>	<a href="#">Change</a>
Review groups	<a href="#">+ Add</a>	<a href="#">Change</a>

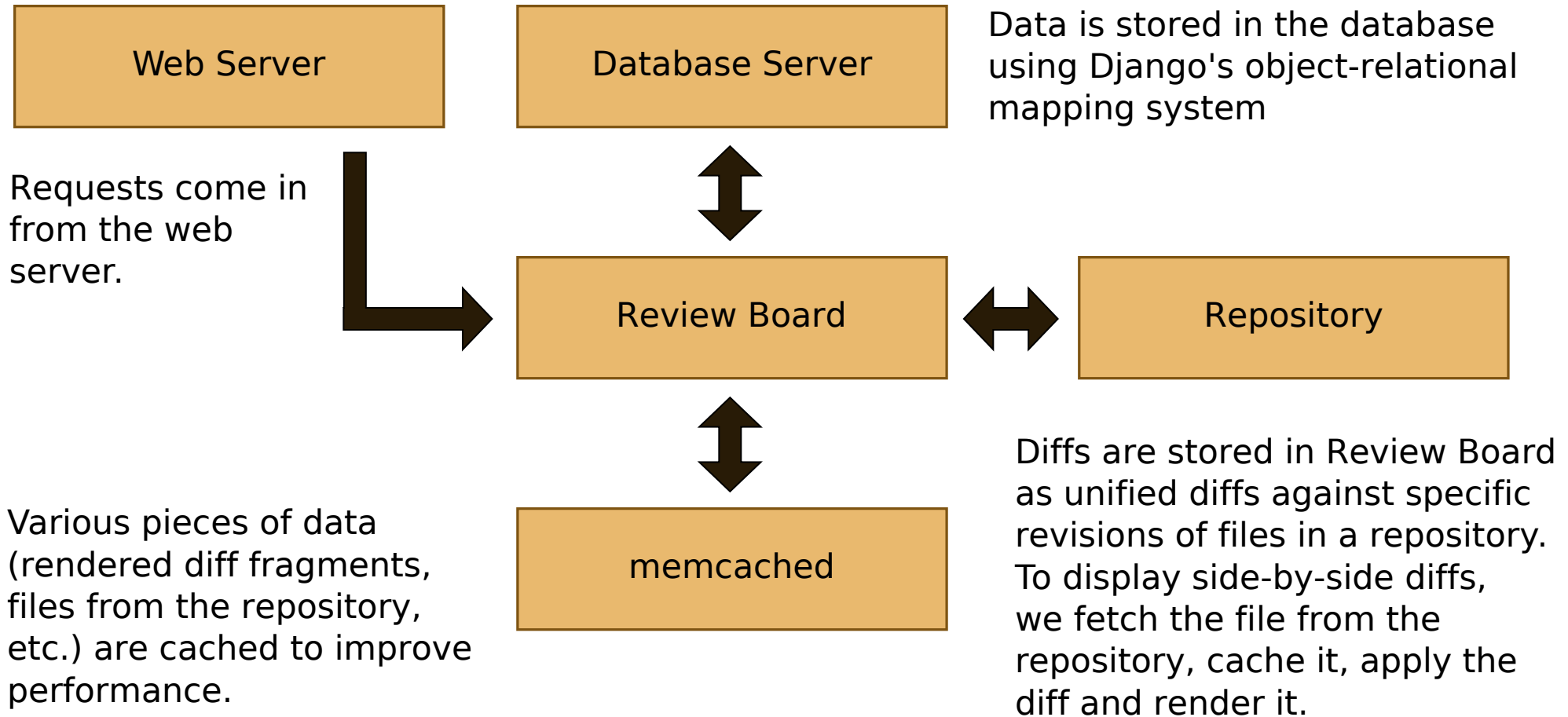
### Recent A

#### My Acti

- [Diff Set](#)  
Diff set l
- [Diff Set](#)  
Diff set l
- [Diff Set](#)  
Diff set l
- [Diff Set](#)  
Diff set l
- [Diff Set](#)  
Diff set l
- [Screen](#)  
Review i
- [Review](#)  
Repositc
- [Galago](#)  
Repositc
- [Galago](#)  
Repositc
- [Galago](#)  
Repositc
- [Galago](#)  
Repositc
- [Review](#)  
Repositc

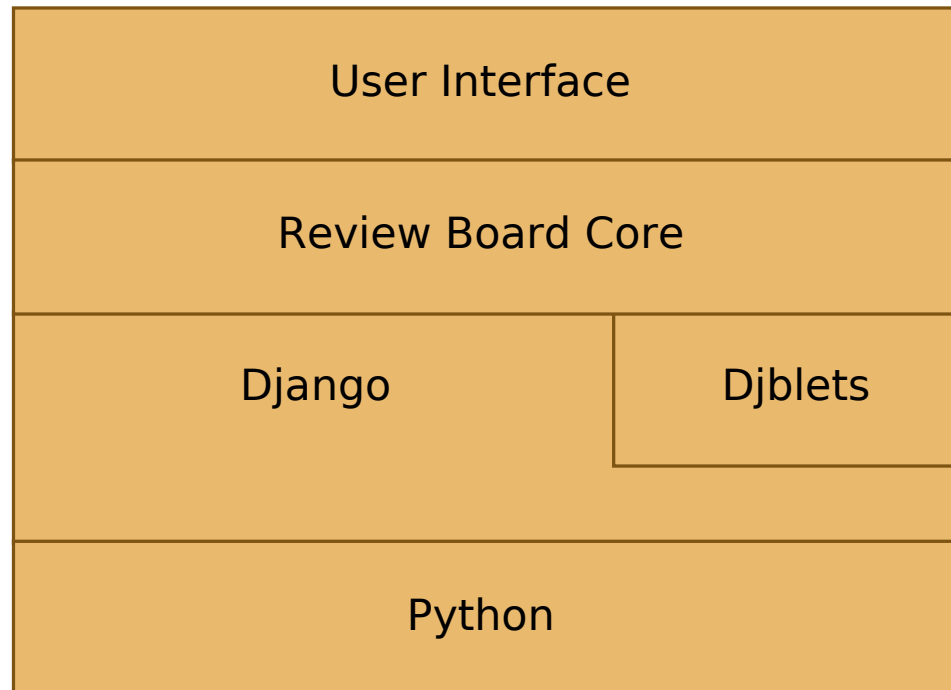


# High-level architecture





# Codebase architecture





# Detailed codebase architecture

post-review	iPhone UI		Web UI		
JSON API			Reports		Yahoo! UI Extensions Library
Diff Viewer		Reviews			
SCM Tools					Yahoo! UI Library
Perforce	CVS	Subversion	Git	Mercurial	



## Future plans

### **Coming soon...**

- Review Board 1.0!
- Appliance VM

### **And in 2.0...**

- Extensions support
- New administration UI
- Better support for distributed SCMs (Git, Mercurial, Bazaar)
- Inline help and documentation



# Getting Started

Project: <http://www.review-board.org/>

Mailing List: [reviewboard@googlegroups.com](mailto:reviewboard@googlegroups.com)

Development: <http://reviewboard.googlecode.com/>