



Life of a Change

Contributing to the WebM Project

John Koleszar
Software Engineer

So you want to contribute to WebM...

- Developing a Change
 - Getting Started
 - Testing your change
 - Uploading your change
 - Code review
 - Criteria For Submission
- Open Discussion



Photo: National Park Service

Getting Started

Contributor License Agreement

Must complete the CLA before your change can be accepted upstream.

Individual

- May sign electronically
<http://code.google.com/legal/individual-cla-v1.0.html>

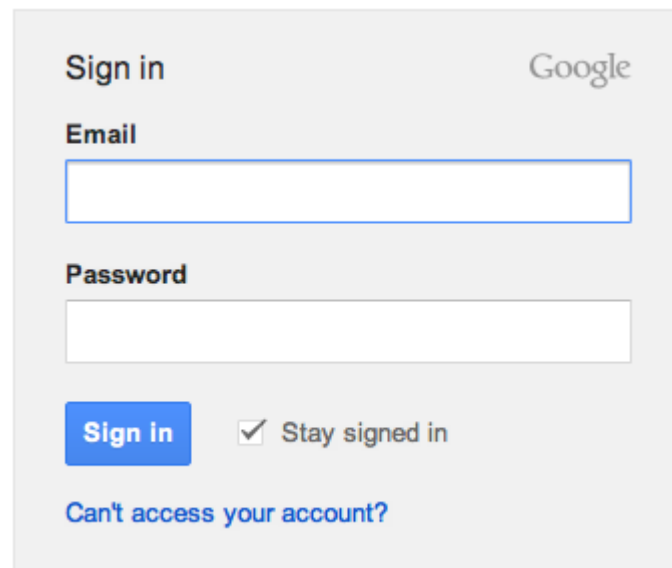
Corporate

- List of people allowed to contribute
- Signed by an authorized representative
- Print, Sign, Scan, (E)mail
<http://code.google.com/legal/corporate-cla-v1.0.html>

Getting Started

Create A Google Account

- Register a username/password with Google
 - Not a "GMail Account," can use your existing email address.
 - Used for authenticating to WebM tools (code review, build, dashboard)



Sign in Google

Email

Password

☒ Stay signed in

[Can't access your account?](#)

Getting Started

Register With Gerrit Code Review

- Visit <http://gerrit.chromium.org>
 - Register link in the top right
- Real Name
- Preferred Email
- Username
 - Hint: Try to use the username you use on your development machine
- Public Key
 - Used for SSH authentication when pushing changes
 - You may already have one you can reuse

Getting Started Walkthrough

Visit our website:

- Step-by-step guides
- Screencasts

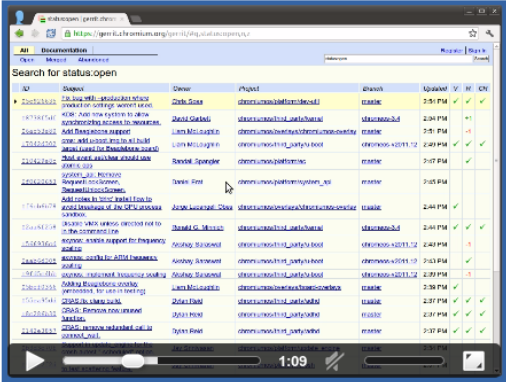
The WebM Project | Submitting Patches - Google Chrome

The WebM Project | Submi x

sudo2.mtv.corp.google.com:4331/code/contribute/submitting-patches/

Create An Account

Browse to <http://gerrit.chromium.org/> and complete the registration process. You must register using [OpenID](#). OpenID provides secure single-sign-on, so you don't need to create a new account or remember another password. If you have a Google account or a Yahoo ID, you're all set. There are many other OpenID providers available. You may already have an OpenID.



The screenshot shows a table of patches in the Gerrit interface. The table has columns for ID, Subject, Owner, Project, Branch, Updated, and a status column with checkmarks. The patches are listed in descending order of update time.

Contributor Agreement

You will be required to execute a [contributor agreement](#) to ensure that the WebM Project has the rights to distribute your changes. Individuals can execute this agreement online. Corporations must submit this agreement via mail or fax. OpenID provides secure single-sign-on, without revealing your passwords to this website.

Create or Upload your SSH Public Key

You must have an SSH public key to upload changes to Gerrit. GitLab has some good guides to

<http://www.webmproject.org/code/contribute/submitting-patches/>

Develop Your Change

Get The Code

Clone from git

```
$ git clone review:webm/libvpx
```



Checkout experimental branch

```
$ git checkout experimental
```

- Forked from master in June 2011
- Stripped down to make easier to hack on
- Changes generally not ported from master (except tests)

Build

```
$ configure --enable-experimental \  
            --enable-internal-stats
```

```
$ make
```

See EXPERIMENT_LIST **in the** configure **script**

Develop Your Change

Test Your Change

- Check out the Contributors Guide

<http://www.webmproject.org/code/contribute/>

```
$ run_tests 200 500 50 baseline
$ run_tests 200 500 50 mytweak
$ visual_metrics.py "*stt" baseline mytweak >
mytweak.html
```



Develop Your Change

Test Your Change

Standard Test Sequences

- <http://media.xiph.org/video/derf/>
- Soon: webmproject.org

YouTube

- Google Internal
- 360p, HD

| CIF | | HD |
|---------------|-----------------|---------------|
| akiyo | ice | City |
| bowing | intros | Crew |
| bridge-close | mobile | Cyclists |
| bridge-far | mobile_calendar | GipsMotion |
| bus | mother_daughte | GipsStationar |
| cheer | r | y |
| city | news | Jets |
| coastguard | pamphlet | Night |
| container | paris | Ped |
| crew | sign_irene | Sherrif |
| deadline | silent | Blue Sky |
| flower | soccer | Park Joy |
| flower_garden | stefan | Shields |
| football | students | Crowd Run |
| foreman | tempete | Riverbed |
| galleon | tennis | Sunflower |
| hall_monitor | vtc1nw | Old Town |
| harbour | washdc | |
| highway | waterfall | |
| husky | | |

Develop Your Change

Test Your Change

Legacy: libvpx-tester

- Run server-side as a presubmit check
- Generally too hard to run locally
- Good coverage
- Deprecated. Porting to gtest unit test framework underway.

Latest and Greatest: gtest

- Easy to run during development
 - `$ make test`
- Write one!

Develop Your Change Walkthrough

Screencast to be made available on webmproject.org

Upload Your Change

Uploading your change triggers:

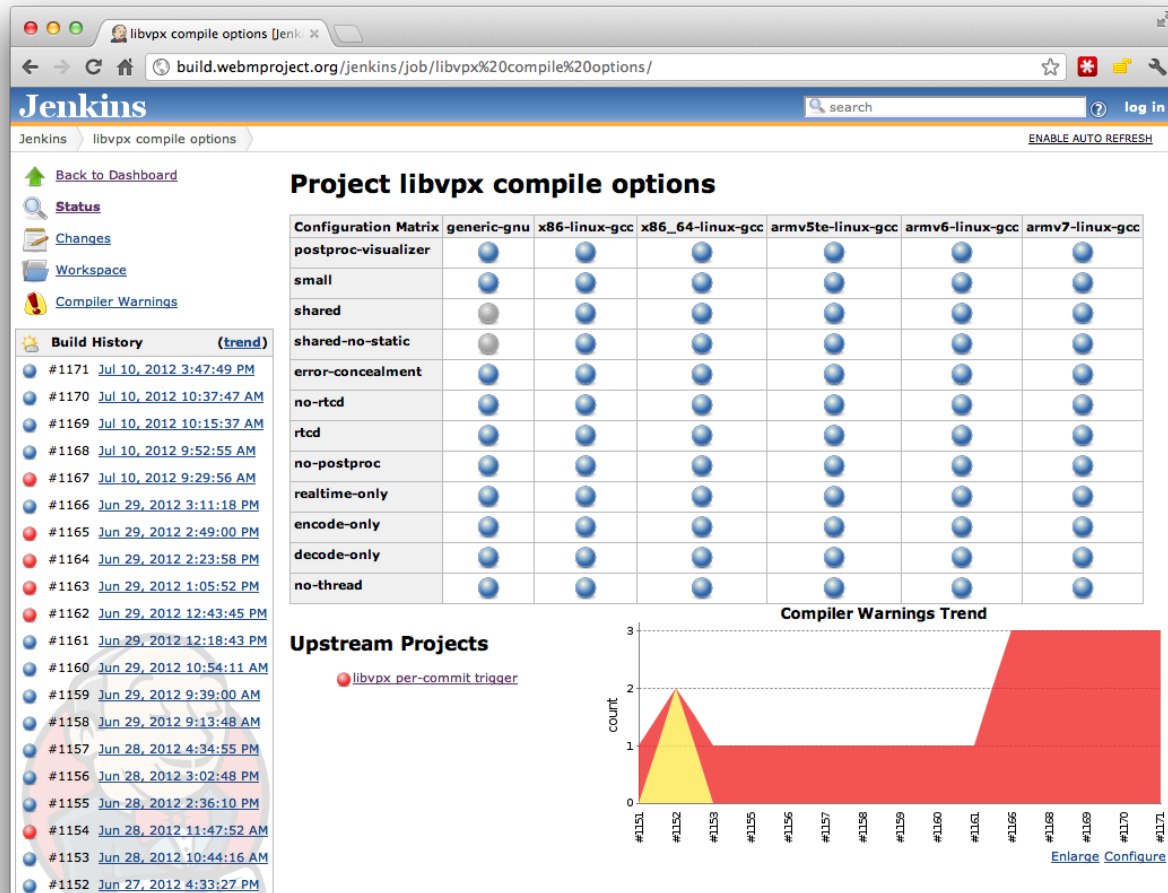
- Notification on IRC
- Email notification to users "watching" the project
- Automated Build - <http://build.webmproject.org>
 - Compilation Tests
 - Unit Tests
 - Linters
 - Fuzzing
 - Quality Tests (Dashboard launching 2012Q3)
- Build result -> Gerrit -> Email
 - First time committers builds will fail



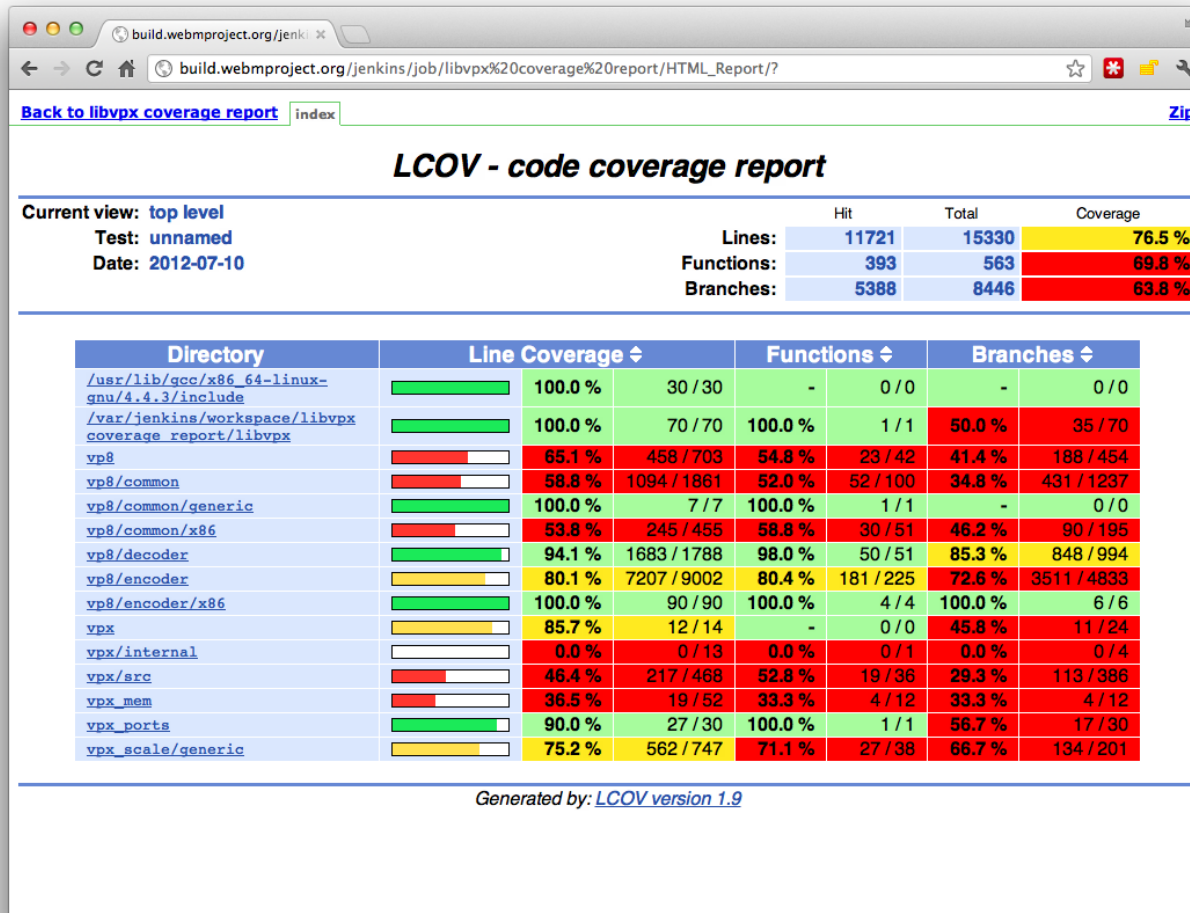
Or upload as draft:

- Publish to only a few people

Upload Your Change Compilation Tests



Upload Your Change Unit Tests



Participate in Code Review

Gerrit Code Review

- A reviewer will review your change, and say:
 - +1 - Looks Good To Me, but someone else must approve
 - -1 - I'd prefer you didn't submit this
- Verified/Fails bit if the reviewer actually tests the code
 - Set by Jenkins
- Any user can perform +1/-1 code reviews
- An approver will review your change, and say:
 - +2 - Looks Good To Me, approved.
 - -2 - Do not submit.
- Make comments in the web UI - per line, or top level
 - Drafts until Published
 - Emailed to author and reviewers
 - But don't respond via email!
- Not a good medium for larger discussion

Participate in Code Review Discussion Forums



Mailing List

`codec-devel@webmproject.org`



IRC

`#vp8` on FreeNode

Participate in Code Review

Criteria For Submission

- Quality up, bitrate down, low complexity!
 - Otherwise, trade 10% speed (optimized) for 1% quality
 - If it's not a clear win, can still submit as an experiment
- Verification
 - Server-side presubmit checks must pass
 - Need to see RD curves if change affects quality
 - Use `visual_metrics.py` script (until dashboard launches)
 - Otherwise must wait for independent verification
- Approval
 - Approvers don't approve their own changes
 - You can be an approver too!
 - Nominate new approvers who have submitted several deeply technical patches
- Success!

Open Discussion

- Questions?
- Are there additional barriers we should work on removing?
- Are there any codec development practices out there in industry that we should adopt?
- Thoughts on development practices as format starts to solidify?