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
Getting Started
Contributor License Agreement

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

Individual
- May sign electronically

Corporate
- List of people allowed to contribute
- Signed by an authorized representative
- Print, Sign, Scan, (E)mail
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)
Getting Started
Register With Gerrit Code Review

- Visit [http://gerrit.chromium.org](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

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

```bash
$ 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

<table>
<thead>
<tr>
<th>CIF</th>
<th>HD</th>
</tr>
</thead>
<tbody>
<tr>
<td>akiyo</td>
<td>ice</td>
</tr>
<tr>
<td>bowing</td>
<td>intros</td>
</tr>
<tr>
<td>bridge-close</td>
<td>mobile</td>
</tr>
<tr>
<td>bridge-far</td>
<td>mobile_calendar</td>
</tr>
<tr>
<td>bus</td>
<td>mother_daughter</td>
</tr>
<tr>
<td>cheer</td>
<td>news</td>
</tr>
<tr>
<td>city</td>
<td>pamphlet</td>
</tr>
<tr>
<td>coastguard</td>
<td>paris</td>
</tr>
<tr>
<td>container</td>
<td>sign_irene</td>
</tr>
<tr>
<td>crew</td>
<td>silent</td>
</tr>
<tr>
<td>deadline</td>
<td>soccer</td>
</tr>
<tr>
<td>flower</td>
<td>stefan</td>
</tr>
<tr>
<td>flower_garden</td>
<td>students</td>
</tr>
<tr>
<td>football</td>
<td>tempete</td>
</tr>
<tr>
<td>foreman</td>
<td>tennis</td>
</tr>
<tr>
<td>galleon</td>
<td>vtc1nw</td>
</tr>
<tr>
<td>harbour</td>
<td>washdc</td>
</tr>
<tr>
<td>highway</td>
<td>waterfall</td>
</tr>
<tr>
<td>husky</td>
<td>City</td>
</tr>
<tr>
<td></td>
<td>Crew</td>
</tr>
<tr>
<td></td>
<td>Cyclists</td>
</tr>
<tr>
<td></td>
<td>GipsMotion</td>
</tr>
<tr>
<td></td>
<td>GipsStationary</td>
</tr>
<tr>
<td></td>
<td>Jets</td>
</tr>
<tr>
<td></td>
<td>Night</td>
</tr>
<tr>
<td></td>
<td>Ped</td>
</tr>
<tr>
<td></td>
<td>Sherrif</td>
</tr>
<tr>
<td></td>
<td>Blue Sky</td>
</tr>
<tr>
<td></td>
<td>Park Joy</td>
</tr>
<tr>
<td></td>
<td>Shields</td>
</tr>
<tr>
<td></td>
<td>Crowd Run</td>
</tr>
<tr>
<td></td>
<td>Riverbed</td>
</tr>
<tr>
<td></td>
<td>Sunflower</td>
</tr>
<tr>
<td></td>
<td>Old Town</td>
</tr>
</tbody>
</table>
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](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

![LCOV code coverage report]

<table>
<thead>
<tr>
<th>Directory</th>
<th>Line Coverage</th>
<th>Functions</th>
<th>Branches</th>
</tr>
</thead>
<tbody>
<tr>
<td>/usr/lib/gcc/x86_64-linux-gnu/4.4.3/include</td>
<td>100.0%</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>/var/jenkins/workspace/libvpx/coverage_report/libvpx</td>
<td>100.0%</td>
<td>100.0%</td>
<td>35/70</td>
</tr>
<tr>
<td>vpx</td>
<td>65.8%</td>
<td>54.8%</td>
<td>85.3%</td>
</tr>
<tr>
<td>vpx/common</td>
<td>54.8%</td>
<td>45.8%</td>
<td>85.3%</td>
</tr>
<tr>
<td>vpx/common/generic</td>
<td>100.0%</td>
<td>-</td>
<td>0/0</td>
</tr>
<tr>
<td>vpx/common/x86</td>
<td>53.8%</td>
<td>45.8%</td>
<td>85.3%</td>
</tr>
<tr>
<td>vpx/decoder</td>
<td>94.1%</td>
<td>84.1%</td>
<td>85.3%</td>
</tr>
<tr>
<td>vpx/encoder</td>
<td>80.1%</td>
<td>7207/9002</td>
<td>85.3%</td>
</tr>
<tr>
<td>vpx/encoder/x86</td>
<td>100.0%</td>
<td>90/90</td>
<td>85.3%</td>
</tr>
<tr>
<td>vpx</td>
<td>85.7%</td>
<td>45.8%</td>
<td>85.3%</td>
</tr>
<tr>
<td>vpx/external</td>
<td>0.0%</td>
<td>0.0%</td>
<td>0/0</td>
</tr>
<tr>
<td>vpx/avc</td>
<td>45.5%</td>
<td>45.3%</td>
<td>45.8%</td>
</tr>
<tr>
<td>vpx/mm</td>
<td>90.9%</td>
<td>80.0%</td>
<td>85.3%</td>
</tr>
<tr>
<td>vpx/mivt</td>
<td>75.2%</td>
<td>71.1%</td>
<td>85.3%</td>
</tr>
</tbody>
</table>

Generated by: LCOV version 1.9
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?