

If you are interested in understanding the internals of Google Test, building from source, or contributing ideas or modifications to the project, then this document is for you.

Introduction

First, let's give you some background of the project.

Licensing

All Google Test source and pre-built packages are provided under the [New BSD License](#).

The Google Test Community

The Google Test community exists primarily through the [discussion group](#) and the GitHub repository. You are definitely encouraged to contribute to the discussion and you can also help us to keep the effectiveness of the group high by following and promoting the guidelines listed here.

Please Be Friendly

Showing courtesy and respect to others is a vital part of the Google culture, and we strongly encourage everyone participating in Google Test development to join us in accepting nothing less. Of course, being courteous is not the same as failing to constructively disagree with each other, but it does mean that we should be respectful of each other when enumerating the 42 technical reasons that a particular proposal may not be the best choice. There's never a reason to be antagonistic or dismissive toward anyone who is sincerely trying to contribute to a discussion.

Sure, C++ testing is serious business and all that, but it's also a lot of fun. Let's keep it that way. Let's strive to be one of the friendliest communities in all of open source.

As always, discuss Google Test in the official GoogleTest discussion group. You don't have to actually submit code in order to sign up. Your participation itself is a valuable contribution.

Working with the Code

If you want to get your hands dirty with the code inside Google Test, this is the section for you.

Compiling from Source

Once you check out the code, you can find instructions on how to compile it in the [README](#) file.

Testing

A testing framework is of no good if itself is not thoroughly tested. Tests should be written for any new code, and changes should be verified to not break existing tests before they are submitted for review. To perform the tests, follow the instructions in [README](#) and verify that there are no failures.

Contributing Code

We are excited that Google Test is now open source, and hope to get great patches from the community. Before you fire up your favorite IDE and begin hammering away at that new feature, though, please take the time to read this section and understand the process. While it seems rigorous, we want to keep a high standard of quality in the code base.

Contributor License Agreements

You must sign a Contributor License Agreement (CLA) before we can accept any code. The CLA protects you and us.

- If you are an individual writing original source code and you're sure you own the intellectual property, then you'll need to sign an [individual CLA](#).
- If you work for a company that wants to allow you to contribute your work to Google Test, then you'll need to sign a [corporate CLA](#).

Follow either of the two links above to access the appropriate CLA and instructions for how to sign and return it.

Coding Style

To keep the source consistent, readable, diffable and easy to merge, we use a fairly rigid coding style, as defined by the [google-styleguide](#) project. All patches will be expected to conform to the style outlined [here](#).

Updating Generated Code

Some of Google Test's source files are generated by the Pump tool (a Python script). If you need to update such files, please modify the source (`foo.h.pump`) and re-generate the C++ file using Pump. You can read the PumpManual for details.

Submitting Patches

Please do submit code. Here's what you need to do:

1. A submission should be a set of changes that addresses one issue in the [issue tracker](#). Please don't mix more than one logical change per submittal, because it makes the history hard to follow. If you want to make a change that doesn't have a corresponding issue in the issue tracker, please create one.
2. Also, coordinate with team members that are listed on the issue in question. This ensures that work isn't being duplicated and communicating your plan early also generally leads to better patches.
3. Ensure that your code adheres to the [Google Test source code style](#).
4. Ensure that there are unit tests for your code.
5. Sign a Contributor License Agreement.
6. Create a Pull Request in the usual way.

Google Test Committers

The current members of the Google Test engineering team are the only committers at present. In the great tradition of eating one's own dogfood, we will be requiring each new Google Test engineering team member to earn the right to become a committer by following the procedures in this document, writing consistently great code, and demonstrating repeatedly that he or she truly gets the zen of Google Test.

Release Process

We follow a typical release process:

1. A release branch named `release-X.Y` is created.
2. Bugs are fixed and features are added in trunk; those individual patches are merged into the release branch until it's stable.
3. An individual point release (the `z` in `X.Y.z`) is made by creating a tag from the branch.

4. Repeat steps 2 and 3 throughout one release cycle (as determined by features or time).
5. Go back to step 1 to create another release branch and so on.

This page is based on the [Making GWT Better](#) guide from the [Google Web Toolkit](#) project. Except as otherwise [noted](#), the content of this page is licensed under the [Creative Commons Attribution 2.5 License](#).