

# TensorFlow Builds

This directory contains all the files and setup instructions to run all the important builds and tests. You can run it yourself!

## Run It Yourself

You have two options when running TensorFlow tests locally on your machine. First, using docker, you can run our Continuous Integration (CI) scripts on tensorflow devel images. The other option is to install all TensorFlow dependencies on your machine and run the scripts natively on your system.

### Run TensorFlow CI Scripts using Docker

1. Install Docker following the [instructions on the docker website](#).
2. Start a container with one of the devel images here: <https://hub.docker.com/r/tensorflow/tensorflow/tags/>.
3. Based on your choice of the image, pick one of the scripts under [https://github.com/tensorflow/tensorflow/tree/master/tensorflow/tools/ci\\_build/linux](https://github.com/tensorflow/tensorflow/tree/master/tensorflow/tools/ci_build/linux) and run them from the TensorFlow repository root.

### Run TensorFlow CI Scripts Natively on your Machine

1. Follow the instructions at <https://www.tensorflow.org/install/source>, but stop when you get to the section "Configure the installation". You do not need to configure the installation to run the CI scripts.
2. Pick the appropriate OS and python version you have installed, and run the script under tensorflow/tools/ci\_build/.

## TensorFlow Continuous Integration

To verify that new changes don't break TensorFlow, we run builds and tests on either [Jenkins](#) or a CI system internal to Google.

We can trigger builds and tests on updates to master or on each pull request. Contact one of the repository maintainers to trigger builds on your pull request.

### View CI Results

The Pull Request will show if the change passed or failed the checks.

From the pull request, click **Show all checks** to see the list of builds and tests. Click on **Details** to see the results from Jenkins or the internal CI system.

Results from Jenkins are displayed in the Jenkins UI. For more information, see the [Jenkins documentation](#).

Results from the internal CI system are displayed in the Build Status UI. In this UI, to see the logs for a failed build:

- Click on the **INVOCATION LOG** tab to see the invocation log.
- Click on the **ARTIFACTS** tab to see a list of all artifacts, including logs.
- Individual test logs may be available. To see these logs, from the **TARGETS** tab, click on the failed target. Then, click on the **TARGET LOG** tab to see its test log.

If you're looking at target that is sharded or a test that is flaky, then the build tool divided the target into multiple shards or ran the test multiple times. Each test log is specific to the shard, run, and attempt. To see a specific log:

1. Click on the log icon that is on the right next to the shard, run, and attempt number.
2. In the grid that appears on the right, click on the specific shard, run, and attempt to view its log.  
You can also type the desired shard, run, or attempt number in the field above its grid.

## Third party TensorFlow CI

### [Mellanox](#) TensorFlow CI

#### How to start CI

- Submit special pull request (PR) comment to trigger CI: **bot:mlx:test**
- Test session is run automatically.
- Test results and artifacts (log files) are reported via PR comments

#### CI Steps

CI includes the following steps: \* Build TensorFlow (GPU version) \* Run TensorFlow tests: \* [TF CNN benchmarks](#) (TensorFlow 1.13 and less) \* [TF models](#) (TensorFlow 2.0): ResNet, synthetic data, NCCL, multi\_worker\_mirrored distributed strategy

#### Test Environment

CI is run in the Mellanox lab on a 2-node cluster with the following parameters:

- Hardware \* IB: 1x ConnectX-6 HCA (connected to Mellanox Quantum™ HDR switch) \* GPU: 1x Nvidia Tesla K40m \* Software \* Ubuntu 16.04.6 \* Internal stable [MLNX\\_OFED](#), [HPC-X™](#) and [SHARP™](#) versions

#### Support (Mellanox)

With any questions/suggestions or in case of issues contact [Artem Ryabov](#).