

Further documentation on Flutter's build infrastructure can be found in <https://github.com/flutter/flutter/blob/master/dev/bots/README.md>.

Requirements for a Flutter/LUCI build

A general outline of the requirements that a Flutter CI test shard has:

1. On LUCI, test shards map to builders. Each test shard must have its own LUCI builder. For the Framework, these are defined in [framework_config.star](#). Generally you will need to have both a pre-submit ("try" in LUCI terminology) builder and a post-submit ("prod") builder.
2. This LUCI builder will specify a "recipe" to run. These are [starlark](#) scripts that determine the actual CI steps to run, and are defined in [flutter.googlesource.com/recipes](#). Most Framework tests use the [flutter/flutter_drone.py](#) recipe. To learn how to edit these, see <https://github.com/flutter/flutter/blob/master/dev/bots/README.md#editing-a-recipe>.
3. Builders are then added to [.ci.yaml](#). These files are read by [Flutter's build dashboard](#), and are used for scheduling builds.

Steps to add a new Framework Test Shard

It is important to land these changes in order to prevent any failing builds during the migration period:

1. Framework tests are run by a Dart test runner called [test.dart](#) that lives in the framework repository. Any new test shards must first be added to this file. Merge this framework change.
2. Next, add pre-submit ("try") and post-submit ("prod") builders to [framework_config.star](#). More detailed instructions [in the infra README](#). Ensure that the "shard" and "subshard"/"subshards" properties match what was added to test.dart in the previous step. Merge this infra change.
3. Update [.ci.yaml](#) in the Framework tree to include the newly added builder. Verify that the entry is marked as `bringup: true`. New shards should always be marked in bringup to verify they are passing on master before being able to block the tree. When opening this PR, a pre-submit check for the new test shard should be triggered. Merge this framework change.
4. Monitor the CI results of the new shard on the [Flutter build dashboard](#). After 10 consecutive passing builds, you can remove the `bringup: true` parameter from `.ci.yaml` in the Framework tree. This will allow the test to block the tree, preventing breakages.

Note: if a new post-submit target is renamed from an existing target, there is no need to follow the bringup process.