

---

# git-upstream-diff(1)

## Manual Page

---

### NAME

git-upstream-diff - Print a diff of the current branch, compared to its upstream.

### SYNOPSIS

```
git upstream-diff [--wordwise] [--  
branch branch] [<extra args for git-  
diff> *]
```

### DESCRIPTION

Shows a diff between a branch and its upstream. If the --branch option is omitted or "HEAD", the tool shows the diff for the

current branch. Any additional arguments are passed through to the underlying git-diff command. This is *roughly* the same as:

```
git diff --patience -C -C  
branch@{upstream} <1> <2>
```

1. `-C -C` detects file copies/renames
2. `--patience` uses the patience-diff algorithm, which tends to produce nicer diffs in many cases.

The difference is that `branch@{upstream}` is actually the tagged merge base of your branch (See [git-rebase-update\(1\)](#)). This means that if your upstream branch was rebased, but you haven't yet rebased the current branch on top of it, you'll still see an accurate diff compared to just diffing against `@{upstream}`.

The `--wordwise` option also allows `git-diff` to do word-by-word comparison in a semi-intelligent way. However, sometimes it can produce surprising results, so it is disabled by default.

Since any additional options are passed through to `git-diff` unchanged, you can use this as a general replacement for `git diff` for things like:

```
$ # See what changed in a file
on the current branch
$ git upstream-diff
some/file.py
...

$ # Omit deleted files from the
diff
$ git upstream-diff --diff-
filter d
...

$ # etc.
```

## OPTIONS

### --wordwise

Print a colorized word-wise diff instead of a line-wise diff.

### --branch branch

Show a diff between the specified branch and its upstream, instead of using the current branch.

### <extra args for git-diff>

Extra arguments are included in the invocation of `git-diff(1)`. These can be anything that `git-diff` normally takes.

`--stat`

This is particularly useful to show *which* files have been changed in comparison to the upstream branch.

`--staged`

Diff against the staged (cached) changes, instead of against the working directory. This is *substantially* faster on large repos, but doesn't take into account unstaged changes (i.e. changes in your working copy which you haven't staged with `git add`).

`-- <filename patterns>*`

Restrict the diff to only show the diff for given files compared to the upstream.

## CONFIGURATION VARIABLES

**depot-tools.upstream-diff.default-args**

---

A list-configuration variable. Each instance of this config variable will be prepended to all

invocations of `git upstream-diff`, as if you had passed them on the command line.

## SUGGESTED ALIASES

Some common short-hand aliases. Feel free to add these to your `~/.gitconfig` file.

```
[alias]
  git udiff = upstream-diff
```

## SEE ALSO

[git-rebase-update\(1\)](#).