

The Flutter plugin for Visual Studio code has been updated to provide *experimental* support for [Visual Studio Code's multi-target debugging](#).

Requirements

- You must be on latest Flutter master channel for concurrent builds to not overwrite each other and cause build failures.

Known Issues

- The Hot Reload button on the debug toolbar does all sessions, but Hot Restart only does the active session

Setup

To debug multiple devices concurrently you should set up a launch config for each device that has the `deviceId` field set (this is the same ID you'd pass to `flutter run -d xxx`). Open the launch config by clicking `Debug -> Open Configurations`. Add a `compound` config at the bottom that will launch both (or more) configurations at the same time:

```
{
  "version": "0.2.0",
  "configurations": [
    {
      "name": "Current Device",
      "request": "launch",
      "type": "dart"
    },
    {
      "name": "Android",
      "request": "launch",
      "type": "dart",
      "deviceId": "android"
    },
    {
      "name": "iPhone",
      "request": "launch",
      "type": "dart",
      "deviceId": "iphone"
    }
  ],
  "compounds": [
    {
      "name": "All Devices",
      "configurations": ["Android", "iPhone"],
    }
  ]
}
```

Once saved, the compound configuration will show up in the drop-down at the top of the Debug side bar.

Running

Selecting the compound config on the Debug side bar and clicking `Debug -> Start Debugging` (or `Start Without Debugging`) will launch debug sessions for each device at the same time.

Note: This may be slow because there are multiple concurrent builds (it may be faster on subsequent builds). You will also see multiple progress notification windows during the builds (we can likely improve this, but right now they just run without knowledge of each other so spawn their own notifications).

When there are multiple active debug sessions, the debug toolbar will show a dropdown that lets you select the "active" debug session.

Actions on the debug toolbar will be sent only to this session (so you can step individually). The Debug Console will show a similar dropdown to switch between the output of each session.

The Variables and Watch panels on the Debug sidebar will be applied to the active debug session (being re-evaluated as you switch between them), though the Call Stack will show all sessions together, grouped by name.

All the usual functionality like hot-reload-on-save should work. Terminating an app should close only that single debug session.

Troubleshooting

The following are tips and tricks learned from connecting a large number of devices for simultaneous debugging (up to 9 at a time have been tested with these techniques).

Hardware

- Make sure to use high quality USB cables, e.g. Anker Powerline+ and/or the official Apple cables
- Make sure to use a high quality USB hub, e.g. Anker 7-Port USB 3.0 Data Hub
- If you feel the device buzzing periodically, it may be connected and disconnecting, which isn't conducive to good debugging

Chrome OS

- Make sure that the ChromeOS box is in developer mode and supports adb deployment
- Make sure the left port is used for the connection (at least for the Google Pixel family of Chrome OS devices)
- If not showing in `flutter devices`, reboot the Chrome OS device with the USB cable in place

Misc

- Changes not being replicated to the web and/or desktop app via hot reload? Clicking on the app should cause the changes to be replicated.
- Starting a debugging session always failing? Make sure that you're using the proper device name, e.g. Apple likes to use smart quotes in their device names instead of the standard apostrophe, e.g. `Bob's iPad` instead of `Bob's iPad`.
- If you're still having trouble, make sure to test each configuration on its own via the Debug Sidebar instead of choosing the device to debug from the status bar -- the one in the status bar is going to ignore your launch settings and hide your errors.