%title: Debug Adapter Protocol %author: Brendon A. Kay %date: 2022-02-27

-> # Debug Adapter Protocol <-

-> presented using `mdp` <-

Motivation

I wanted a universal debugger service rather than relying on a bunch of different plugins for my editor.

Debug Adapter Protocol

Built by Microsoft for VSCode.

Since it's a protocol other editors and tools can implement support for it.

From the `nvim-dap` help page...

```bash A debug adapter is a facilitator between nvim-dap (the client), and a language-specific debugger:

DAP-Client ----- Debug Adapter ------- Debugger ------ Debugee
(nvim-dap)  |   (per language)  |   (per language)    (your app)
            |                   |
            |        Implementation specific communication
            |        Debug adapter and debugger could be the same process
            |
     Communication via the Debug Adapter Protocol

```

Methodology

To demonstrate the use of the DAP I will be using the Neovim plugin nvim-dap

For the debugger server I will be using the Go debugger Delve, which has a DAP option.

```
bash $ dlv -h | grep DAP dap [EXPERIMENTAL] Starts a TCP server communicating via Debug Adaptor Protocol (DAP).
```

Pros

Multiple debuggers with a single interface. Here is a list of available DAP implementations.

Can be used in many different IDEs.

Debugger control, logging, and other features.

Cons

Since it's generic it might lack the depth of proprietary debuggers.

Yet another dependency to maintain.

Disclaimer

This is not a claim to be the "best" way to debug Go services.

Many other editors have rich proprietary tooling that can be easier to configure or setup.

Even print logging is a valid method of debugging.

However, as we will see, using a DAP client provides its own set of benefits.

-> ## Demo Time ðŸŽ‰ <-

Other benefits

Using the DAP over TCP means you could connect to a debugger on another server. I have a proof-of-concept for our Docker development services in backend PR #2907

Improvements

I haven't yet tried debugpy, a DAP implementation for Python 3.

Thank you!

Questions?