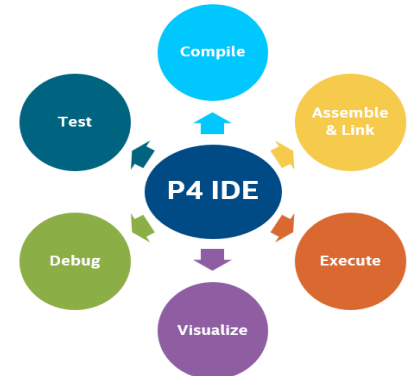


# P4 IDE - An Integrated Development Environment for P4 based data plane development

## Abstract

Software development primarily relies on Integrated Development Environment (IDE) to develop code, debug and test. Functionalities like setting up a workspace, compilation, set breakpoints, debugging etc., can be achieved in much lesser time when an IDE is used. Currently, there is no IDE available for P4 development, instead a set of CLI tools are used to compile, assemble, execute, visualize, test and debug. This adds extra responsibility for P4 developers to install and setup environment for required binaries manually using certain exact steps. With the increasing number of toolsets being developed for P4 Ecosystem, performing this setup manually is time consuming, error prone and requires domain expertise.

The proposed solution enables a GUI based single-executable which when installed in user-space handles installation of all required P4 tools along with VS Code P4 IDE extension which eases the invocation of these tools in single click. The various P4 Tools that are integrated in P4 IDE are P4 Language Server, P4 Compiler, P4 Binutils (assembler & linker), P4 Runtime (for programming rules), P4 View (Visualize resources), P4 Debugger, P4 Testgen (Generate test packets), and Hardware/Simulator.



## Proposed Solution

P4 IDE enables ease of access to all P4 Tools with simple clicks. For example, In Figure 1, When a P4 file (1) is opened, then list of icons for performing different functionalities is enabled (2). If user click at “Build” icon (3), then compilation, assembling, linking will be taken care automatically based on user given configuration and final package file (.pkg) will be generated for a given P4 program. P4 Language server provides auto-completion functionality like shown in (4). On click of “P4” icon (5), then P4 IDE walkthrough (how-to page) and documentation is displayed. Likewise, icons are provided to start data plane, start control plane, configure rules, send traffic with network packets, start visualization, start debugger and start testcase generation. Configuration of certain parameters/options (e.g., generate runtime files for compiler) for all P4 Tools are provided in an interactive manner using checkboxes and textboxes. Few advantages of P4 IDE are,

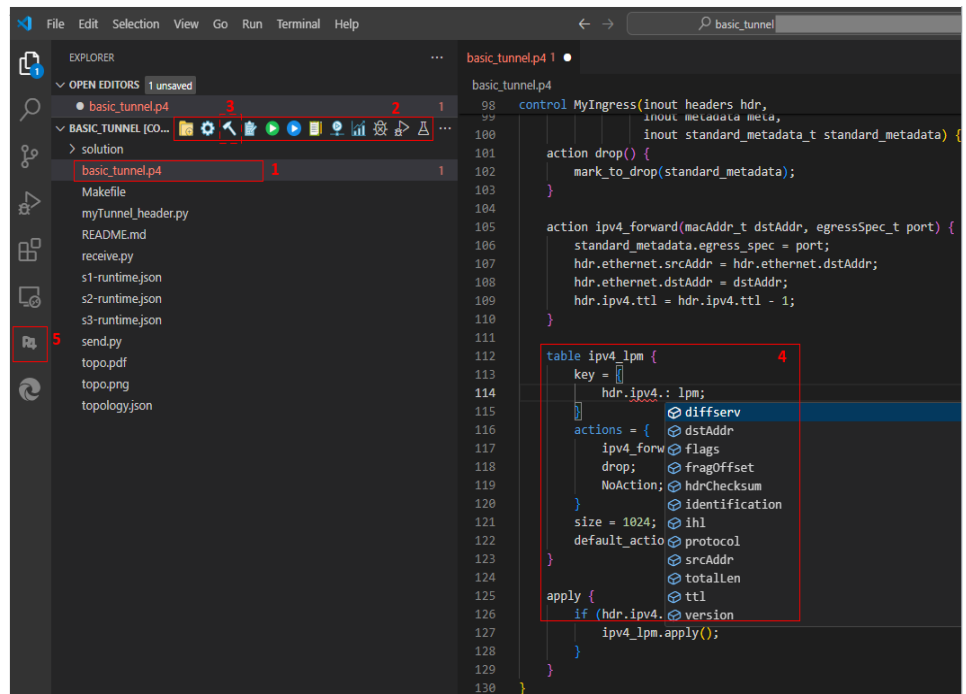


Figure 1, P4 IDE Example

Likewise, icons are provided to start data plane, start control plane, configure rules, send traffic with network packets, start visualization, start debugger and start testcase generation. Configuration of certain parameters/options (e.g., generate runtime files for compiler) for all P4 Tools are provided in an interactive manner using checkboxes and textboxes. Few advantages of P4 IDE are,

1) Ease of use

3) Provides development assistance

2) Hassle free solution

4) Multiple tools access