Protocol Independent Forwarding (PIF) Project Position Paper

Scope/charter of the effort

The group started work in September 2014, downstream from the publication of ONF TR-505, "OF-PI: A Protocol Independent Layer", together with Nick McKeown's keynote "How to tell your plumbing what to do: Protocol Independent Forwarding" at the ONF member workdays. It was set up as an open source software driven project ahead of the OSSDN umbrella being created by ONF in 2015.

The PIF project concerns establishing an Intermediate Representation (IR) to be used by compilers from high-level languages for describing packet forwarding datapaths (e.g., P4, PIFL, PX, PacketC) to diverse target technologies (e.g., ASIC, FPGA, NPU, CPU).

It has been organized to have three main threads of activity:

- Experimentation with IR features and capabilities: to lead to a final IR proposal;
- Investigating diverse use cases for the IR: to ensure apt coverage;
- Runtime API for the IR: to feed into next-generation OpenFlow.

The group pioneered the notion of "software first, leading to specification if appropriate" within ONF, and so it seamlessly morphed to become an OSSDN project in 2015. As the threads indicate, the primary aim was experimentation with intermediate representations through development of software allowing the simulation of IR features, followed by a secondary aim of devising runtime software APIs for an evolved intermediate representation.

The OSSDN home for the project is at:

http://opensourcesdn.org/projects/pif-open_source_intermediate_representation_for_datapaths

and the GitHub for the actual software development, and archival of documents, presentations, and meetings, is at:

https://github.com/OpenNetworkingFoundation/PIF-Open-Intermediate-Representation

Although PIF became an OSSDN project, open to all, in 2015 (with currently 92 people following the email list), the small number of core contributors continued to be drawn from ONF member organizations.

The outputs from the group to date have largely been software concerning the first thread above, experimentation with IR features and capabilities, in two main open source releases:

- Q4 2014: Meta-IR infrastructure for experimenting with different IRs, and **AIR** – an initial strawman IR strongly influenced by P4 as a source and RMT as a target.
- Q1 2016: Improved meta-IR infrastructure for experimenting with different IRs, and **BIR** – a second strawman IR influenced by a wider variety of sources and targets.

The BIR deliverable illustrated the "Python to PDF" model in action: code first for experimentation, and then a specification (initially in English, later accompanied by a machine-readable BNF grammar).

Alignment with ONF's vision and goals

The PIF activity aligns well with the goal of the 'new ONF' to make it "so that programming the network is as easy and commonplace as programming applications." The starting point was to facilitate the compilation of diverse high-level network programming languages to diverse targets, to turn an "m*n" compiler problem into an "m+n" compiler problem. However, in searching for a neutral intermediate representation of programmable datapaths for compiler use, which is a relatively niche area, the group has realized that this representation is offering a model of programmable datapaths that has wider applicability, for example in TTP datapath schema generation and in runtime control API definition. This generalized view has fed into thinking on SDN ecosystem evolution, as published in ONF TR-535, "ONF SDN Evolution". Importantly, the PIF work process pioneered the "software first" approach within ONF.

Evidence of the effort's relevance and potential impact

The hands-on effort and the impact has not been as great as had been hoped for. A small set of less than ten people has been driving the group, and there have been changes in the context surrounding the activity. One notable trend has been a growing view that P4 is the only high-level language worth putting effort into, which makes a project around devising a neutral intermediate representation seem less useful. Another fact has been the complete absence of network operators in the discussions, which has deprived the project of practical use cases for programmable datapaths to motivate possible capabilities for the intermediate representation.

The AIR intermediate representation has seen practical use in the Netronome P4 compiler to NFP, including some extensions for missing capabilities in that initial strawman IR. The BIR intermediate representation has seen practical use in the Cornell P4FPGA P4 compiler to FPGA, and a variant is in use for the Xilinx P4 compiler to FPGA. These use cases have shown that the two IRs can form a viable basis for compiling P4 to both NPU and FPGA.

The group has also listed and categorized the requirements for annotations for the intermediate representation, to ensure that useful information in high-level language descriptions is not lost during the compilation process, and also to preserve information of use to other tools such as debuggers, visualizers, and analyzers. However, these have not been incorporated into a full tool flow to date.

Assessment

The group feels that the PIF activity has probably reached the end of its natural lifetime in its current form. There are several trajectories for building on the fruits of the group:

- P4.org: the BIR programmable datapath representation might form part of the compiler flow, as an explicit IR in contrast to the current internalized data structure intermediate forms. It could potentially be a generic "hardware" counterpart to the existing eBPF generic "software" target.
- OPNFV or other: the runtime API aspect of PIF has been explored in the group, but now deserves a dedicated open source software project. This could be wrapped together with the putative OpenFlow runtime API project, and ideally also with the emergent P4 runtime API work.
- ONF ODWG: the PIF model for programmable data paths provides an important ingredient for SDN ecosystem evolution, as envisaged in ONF TR-535. The next step is to obtain realistic calibration via the involvement of network operators in the discussion.