**T4P4S[[1]](#footnote-1): A P4 compiler
for a wide variety of targets
*Presenter: Sándor Laki, Eötvös Loránd University, Budapest, Hungary (lakis@elte.hu)***

In this talk, I will overview the recent advances of our multi-target P4 compiler called T4P4S (formerly called P4@ELTE) that currently supports various architectures like Intel x86 (using Intel DPDK), Freescale LS2085 (using Freescale’s ODP-SDK) and TP-Link TL-WR1043ND (using OpenWRT and Linux UIO). Our aim with T4P4S was to develop a high-performance P4 compiler with flexible re-targetability. To this end, we have identified the essential components of the compiler, and split them into *hw-dependent* and *hw-independent parts*. Accordingly, the hardware dependent functionalities are defined by a Networking Hardware Abstraction Library (NetHAL) that has to be implemented for each target, while the core compiler remains independent of the actual hardware. The core code and the actual target are interconnected through NetHAL. Currently, the NetHAL is implemented for Intel CPUs (Intel DPDK), Freescale LS2085 (ARM) (using the ODP SDK) and OpenWRT. For both Intel and Freescale platforms our measurements show comparable results to standard example programs like L2, L3 forwarding examples, and the system is scales well with the increasing number of cores. An early version of our compiler for Intel DPDK was presented at SIGCOMM 2016 as a demo, but this presentation will shed light on the structure of the generated switch architecture, the design decisions we made during the development of the compiler and the latest performance measurements for both Intel, Freescale and OpenWRT targets.

To analyze the performance of T4P4S, four use cases from operational OpenFlow deployments were identified, and implemented in P4 and OpenFlow. The observed performance metrics were compared to the ones resulted by OpenVSwitch (OVS), the most widely used OpenFlow software switch. In the proposed talk, we will first present the use cases to be examined, then the evaluation environment is detailed and we finally describe the obtained measurement results and characterize the performance in the various scenarios. We also show that the switch program generated by our compiler provides comparable performance to OVS with DPDK back-end and in many scenarios it even outperforms OVS.

Note that PISCES [4], a P4-enabled software switch derived from OVS, shows performance degradation caused by the P4 specific modifications in the original OVS data plane.

This presentation will focus on the recent advances of P4@ELTE, a multi-target P4 compiler that we have achieved. Though the benefits of an early version of the compiler has already been demonstrated at SIGCOMM 2016, this talk will put more emphasis on the design of the core compiler and the HAL, and the performance evaluation using various use case scenarios. Our future plans include the support of P4-16, the next generation of P4 language whose specification will be published by the end of this year, the integration of our switch into existing SDN ecosystems (e.g. interoperation with OpenDayLight controller), and the support of additional targets.

**Presenter’s biography**

**Sandor Laki** is an Assistant Professor at the Department of Information Systems, Eötvös Loránd University (ELTE), Budapest, Hungary. His research interests focus on active and passive network measurement techniques, traffic analytics, IP geolocation and algorithmic aspects of communication networks. Since 2015, he is a senior member of the Communication Networks Laboratory - a joint laboratory of Eötvös University and Ericsson Research. In the past years he has participated in various EC and EIT Digital projects, including FP7 OneLab2, FP7 NOVI, FP7 OpenLab, EIT Digital FITTING, EIT Digital Smart Ubiquitous Communication and Future Internet-PPP XIFI and FI-CORE.

1. T4P4S (pronounce as TAPAS): Translator For P4 Switch programs [↑](#footnote-ref-1)