

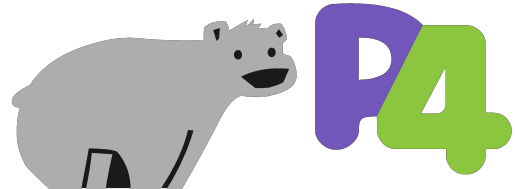
Thesis
M.Sc.

IDP

Cache Efficient Hashing-Based Data Structures in P4

Motivation

P4 [1] is a domain-specific programming language intended to describe the behavior of packet processing systems. P4 was introduced in 2014 and can define entirely new networks with new protocols that behave differently from the networks we currently use. Compiler exists for various targets (software, FPGA, SmartNIC, ASIC).



P4 programming language

`t4p4s` [2] is a P4 compiler for the software-based DPDK [3]. It uses the official P4 compiler to generate a JSON representation of the P4 program. This JSON object is then parsed by `t4p4s` using python, which in turn creates C code. This C code is platform-independent and is linked with platform-specific (DPDK) functions.

Match-Action Tables are the crucial part of the P4 pipeline for all advanced tasks. Therefore, optimizing their implementation of them is very promising. We want to evaluate the performance of a cache-efficient hash table implementation in networking scenarios. Furthermore, we plan to integrate it into the P4 pipeline to analyze whether or not this data structure is suitable for usage in upcoming P4 applications.

Tasks

- Evaluate the performance of a given hashing-based data structure
- Gain insights about possible bottlenecks
- Get familiar with P4 and the `t4p4s` toolchain and DPDK
- Integrate a given data structure into `t4p4s`
- Test functionality by providing small P4 example programs

Requirements

- Experience with C programming and Linux is required

Sources

- [1] <https://p4.org/>
- [2] <https://github.com/P4ELTE/t4p4s>
- [3] <https://core.dpdk.org/doc/>

Contact

Manuel Simon simonm@net.in.tum.de
Sebastian Gallenmüller gallenmu@net.in.tum.de

