# Are graph databases fast enough for static P4 code analysis? [*]

**Dániel Lukács[a], Gergely Pongrácz[b], Máté Tejfel[c]**

[a]Faculty of Informatics
Eötvös Loránd University, Budapest, Hungary
dlukacs@inf.elte.hu
ORCID: 0000-0001-9738-1134

[b]Ericsson Hungary Ltd., Budapest, Hungary
Gergely.Pongracz@ericsson.com
ORCID: 0000-0002-5115-9973

[c]3in Research Group, Eötvös Loránd University, Martonvásár, Hungary
matej@inf.elte.hu
ORCID: 0000-0002-5115-9973

### Abstract

P4 is a new protocol-programming language for smart switches and virtual network functions linked up in vast SDNs. Our current work in progress is focused on analysing the execution cost of P4 programs using hierarchical control flow graphs (CFGs). We found that cost analysis of P4 requires storing and working with diverse information about various execution environments.

In theory, versatile graph backends, such as graph databases (GDBs) could provide a unifying representation in which we can store, access and traverse the CFGs and the environment information together. On the other hand, analysis efficiency is a requirement both for large-scale testing and end user application. Unfortunately, we cannot conclusively assess – without trying it out in practice – whether GDB disk and memory reads will ruin our performance or not.

In this work, we briefly detail how we realized P4 control flow analysis based on GDBs, and present our measurements to conclude if the overhead inherent in GDBs is justified for our purposes.

*Keywords:* static analysis, P4, cost analysis, graph database, software defined networking

[*] The research has been supported by the European Union, co-financed by the European Social Fund (EFOP-3.6.2-16-2017-00013, Thematic Fundamental Research Collaborations Grounding Innovation in Informatics and Infocommunications).