WHITE PAPER

Intel, Kaloom Create P4-programmable Network Solutions

Companies create advanced network systems with the P4 programming language and Intel® Tofino family of Ethernet switch ASICs to deliver fully programmable, cloud-native networking solutions for distributed edge data centers

Many core elements of networks are becoming programmable with software control thanks to software-defined networking (SDN) and its ability to program the network control plane, which improves network performance and agility. The flexibility and performance of this software programmability is now taking hold in the data plane of networks with the growing popularity of the P4 programming language.

P4 got its start in a SIGCOMM conference paper in 2014 titled, "Programming Protocol-Independent Packet Processors"[1], which described a language specifically developed for programming the data plane of network devices. The paper led to the creation of the open-source P4 programming language and the development of P4.org, which is now a project of the Open Networking Foundation with more than 100 industry and academia members. P4 is becoming more mainstream with industry-leading and startup switching companies launching their own P4-based products.

P4 has come to be regarded as a networking standard with its domain-specific language based on a simple, declarative nature. The simplicity of programming in P4 facilitates the description of network requirements in terms of the exact processing expectations in the data plane via a technical, yet humanly understandable format.

P4 is a protocol-independent programming language where new network protocols can be defined and implemented while ensuring the vendor maintains control of its intellectual property. This makes P4 an enabler for innovation as it facilitates product differentiation and value creation.

While there are other relatively high-performance platforms for packet processing, such as network processing units (NPUs) that are programmable with various languages from assembly to C or C derived languages (e.g., PacketC), these platforms don't address a fundamental principle of "modularity based on abstraction", which is critical to the design and specification of programming languages.[2] This abstraction allows network specialists to focus on the solution and be more innovative without the need to understand the underlying layers, since the hardware details of the target platform are completely abstracted for the developer.

P4 provides well-defined, high-level data plane abstractions, such as headers, parser and deparser modules, match-action tables, and externs like counters, meters, etc. When given the right abstractions, a P4 network programmer can easily implement an L2 switch or an L3 router as described by well-known protocols like Ethernet, IPv4 or IPv6 specifications.

Table of Contents

How P4 Networks Work 2
P4 Runtime 2
Match-Action 2
Intel® Tofino and Tofino (2) P4 Switch Description 3
How Kaloom™ Uses P4 3
P4's Key Enablers Laying the Foundation for Kaloom's Adoption 3
Intel Tofino and Tofino 2 System Interoperability 4
P4 in Kaloom's Product and Solution Offerings 4
Security and Network Slicing..... 5
Conclusion...................... 5

P4 provides complete control of the data plane functions implemented versus a solution based on a fixed-function ASIC, in which only a fraction of the supported features gets used. Moreover, the bugs in a fixed-function ASIC remain in the data path, while in P4, they can be addressed immediately once found.

P4 programmability offers three significant advantages for next-generation networks:

- **Protocol flexibility:** With P4, new protocols can be implemented within hours to days. This approach contrasts with fixed function ASICs, where it can take several years for the ratification of a new networking protocol to be available in a fixed network switch ASIC. With a P4-programmable switch ASIC, that protocol can be operational in the network as soon as it's needed. Other protocols not in use can be eliminated from the P4 ASIC for lower power consumption and deterministic performance.
- **Faster network functions:** P4 devices can run virtualized network functionality such as load balancing, firewalling, and others. This allows packets to be processed at the network ingress and frees up downstream CPU compute cycles for other virtual network functions.
- **In-band Network Telemetry (INT):** Network congestion, dropped packets and other performance issues have a significant impact on multi-terabit network throughput. INT enables monitoring of performance from a very large sample of packets, which identifies network challenges as soon as they appear and allows for rapid remediation.

P4 programming enables enterprises, data centers and service providers to build new innovation, functionality and performance into their networks. Many networks have adopted P4. This paper will provide an introduction and history of P4 and will discuss how it has been implemented in real-world networks.

How P4 Networks Work

P4 Language is a higher-level programming language designed specifically to configure how a switch data plane will process and forward an IP packet. It has been described as a top-down networking model because it tells the switch what to do, rather than a bottom-up approach of having the switch chip describe the fixed and limited set of things it can do.

P4 is target-independent, so it can be used to direct packet forwarding in P4 switch ASICs, or in network processing units (NPUs), field programmable gate arrays (FPGAs), and smart network interface cards (NICs). This target agnosticism means P4 programs can run in a wide array of networks without concern that certain targets won't be able to execute the program.

P4 Runtime

Deployed P4 elements are configured and changed using P4 Runtime, the data-control plane API built into the P4 programming language. P4 Runtime is open, silicon-independent and provides a standard way to control a P4 network element, enabling a network manager to add or delete the entries of a networking element's forwarding table. P4 Runtime is designed for both remote and local control planes and can be operated with or without remote procedure call (RPC).

Match-Action

P4 switch ASICs and other programmable targets utilize a Protocol Independent Switch Architecture (PISA) to execute P4 programs. The protocol is essentially a program expressed in P4, making it easy to support newly established protocols, which can be developed and changed as needs evolve over time.

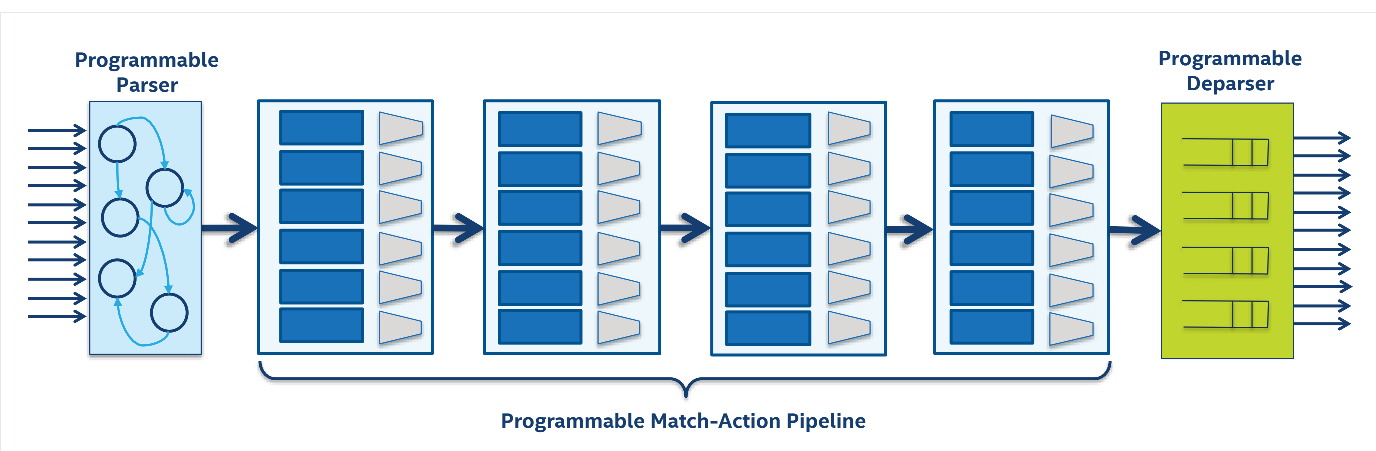


Figure 1. Match action engine in a P4 target.

As seen in Fig. 1, packets enter a PISA device, where a programmable parser examines the packet headers, identifies them, and defines their order. The packets are then forwarded through a programmable match-action pipeline that processes the packets according to the algorithm defined for that packet. As an example, A P4 switch could recognize an IPv4 packet and execute actions that are entirely different from an IPv6 packet. A deparsing function then prepares the packets for forwarding.

Intel® Tofino and Tofino (2) P4 Switch Description

The Intel Tofino and Tofino 2 Ethernet switch ASICs deliver full P4 programmability and adapt the data plane to specific protocols and applications. These switch ASICs have a non-blocking, switching capacity ranging from 1.2Tbps to 12.8Tbps while supporting 28Gbps and 56Gbps SerDes (See Figure 2).

The Intel Tofino series delivers programmability while maintaining its full line-rate performance. Another benefit of Tofino's programmable pipeline and memory is a large-scale packet lookup table, alleviating the need for external lookup table memory. The design of the Tofino ASIC's lookup table capabilities is inclusive of the majority of table scale requirements.

Tofino & Tofino 2 Ethernet Switch ASIC Family

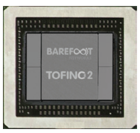
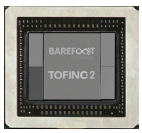
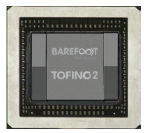
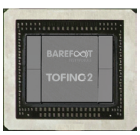
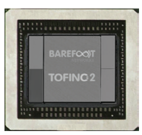
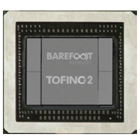
Series	12.8 Tbps	8.0 Tbps	6.4 Tbps	3.2 Tbps	2.0 Tbps	Markets
U-series						**U**ltra – Large Capacity & Highest Features
M-series						**M**ainstream – High Bandwidth & Feature Rich
H-series						**H**yperscale – Optimized for Power & Efficiency

Figure 2. Tofino and Tofino 2 switch ASIC configurations.

How Kaloom™ Uses P4

Among the early adopters of P4 has been Kaloom™, an emerging leader in fully programmable and automated cloud-native networking solutions for distributed edge data centers. Based in Montreal, Kaloom was founded in 2014 with a clear focus on delivering networking software solutions that will disrupt how distributed cloud edge and data center networks are built, managed, and operated. Its products are aimed at communication service providers, mobile operators, as well as data center and cloud service providers.

From its beginnings, Kaloom made a strategic choice to develop on P4's initial specification and use it to address how data plane components of its software architecture would be implemented. Specifically developed for programming the data plane of network devices, P4 has been regarded as a key enabler that helps solve certain restrictions existing in today's data center networks.

To further its abilities in providing contributions to the P4 community, Kaloom endowed a 5-year long Research Chair in 2019 to tackle different topics associated with P4, namely portability, modularity, and language expressiveness. Kaloom believes that code portability is key for P4's long term success. Achieving real code portability for the language across multiple platforms (FPGA, Intel architecture CPUs, Tofino switch ASICs, etc.) requires, for example, standardizing the API of the vendor-specific externs. The research chair is a joint collaboration between École Polytechnique, a local university in Montreal, and industrial partners including Intel, a strategic technology partner for Kaloom. Throughout its progression, Kaloom has stayed true to its focus on P4 and its key enablers.

P4's Key Enablers Laying the Foundation for Kaloom's Adoption

Kaloom has embraced P4 as its data plane programming language of choice in its production environment and has leveraged it to define its own internal tunneling protocol called Kaloom Normalized Format (KNF). KNF encapsulation is proprietary to Kaloom and was designed to carry packets across the company's Software Defined Fabric™ product, regardless of the original encapsulation format.

The benefits derived by the KNF specification are:

- Minimizes the number of encapsulation formats that need to be handled by the spine switches in the fabric
- Provides a lossless transformation to/from commonly used encapsulation formats
- Optimizes lookups and processing inside the fabric by reordering and rationalizing commonly used header and payload fields
- Natively supports in-band network telemetry for diagnostics, metrics, and analytics

Kaloom was able to implement KNF and maintain it internally without having to disclose it or take it into a long standardization process. The use of P4 bypasses the prolonged ASIC design phase, which can take years before technology is finally available to the market. Kaloom fundamentally believes that P4 should be used for solving networking problems. At its heart, P4 is meant for defining the data plane for P4-programmable hardware, allowing network designers to explore new solutions and applications, beyond what can be done today with a fixed-function ASIC.

Kaloom's adoption of P4 enabled them to push the technology to its limits while extending its capabilities. From the onset, when Kaloom's engineers started working with the technology, they began to envision new features and enhancements that broadened the application spectrum. To accomplish this mission, the company helped evolve the technology to support new features and elevate the next-generation of high-performance P4-capable switches.

With different aspects of P4 programming being well structured with its consortium's separate working groups overseeing language design, API, architecture, applications, and education, it was easy to engage and make contributions where possible.

Intel Tofino and Tofino 2 System Interoperability

Kaloom's data plane can execute on any Tofino or Tofino 2-based ODM platform, which helps facilitate Kaloom's heterogeneous device strategy and enables their customers to run networks on the equipment that best meets their needs. This strategy is based upon a cornerstone of true open source systems, where open and diverse hardware is used to enable service providers to enjoy more control and flexibility. By using hardware that best suits their network's required characteristics, service providers can innovate more rapidly.

Figure 3 provides an example of Kaloom's heterogeneous device strategy. For a container networking function (CNF), having both a control plane and data plane component, the control plane networking function can run on a general-purpose processor where portability does not present an issue. To support stateless and stateful services, Kaloom's heterogeneous hardware strategy also leverages the ability to use the optimum hardware for the required application.

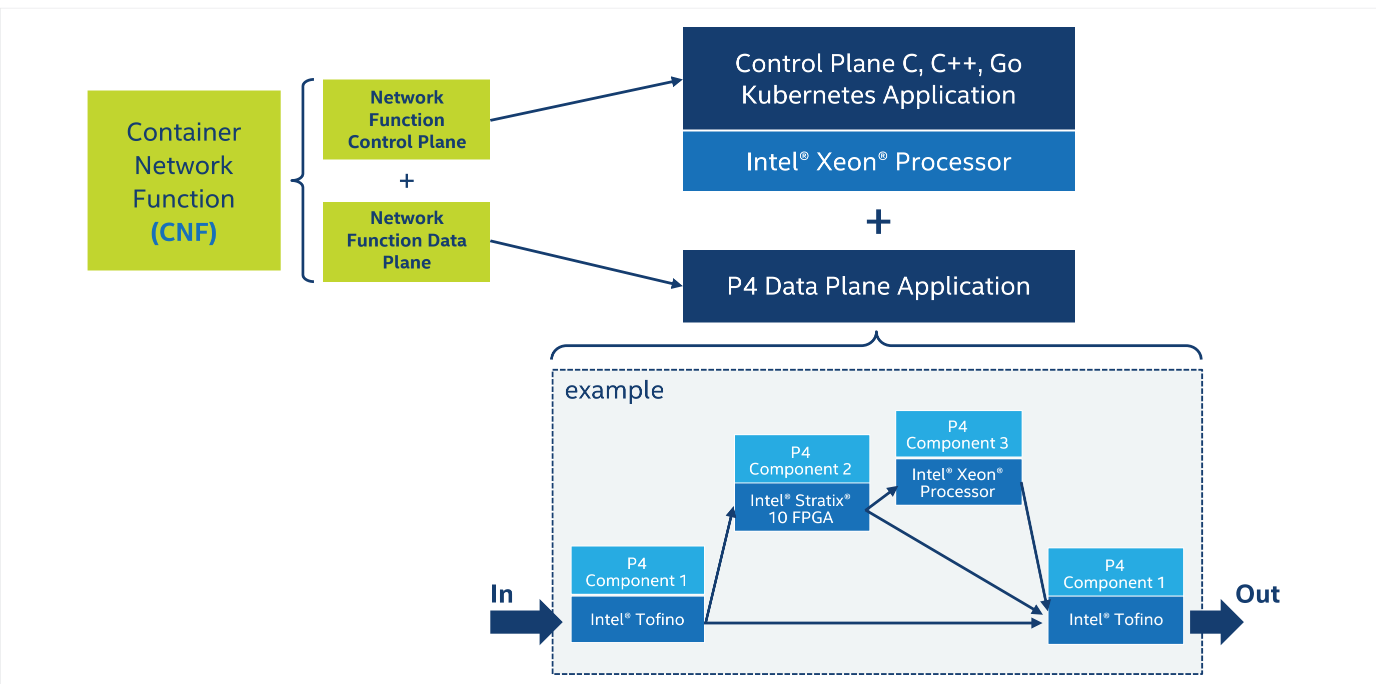


Figure 3. Example of Kaloom's heterogeneous device strategy.

P4 in Kaloom's Product and Solution Offerings

Kaloom's product offerings,[3] ranging from its Software Defined Fabric, Cloud Edge Fabric™, and more recently its 5G UPF solution, fully encompass the open P4 ecosystem enabling customers to have full control of their network.

Kaloom has taken a P4-based approach to the switching fabric in 5G infrastructure with its Cloud Edge Fabric, a fully automated data center network fabric with native support for network slicing, along with an embedded 5G user plane function. The Cloud Edge Fabric complements the Software Defined Fabric product, which is designed to automate and virtualize data centers.

For Kaloom, P4 enables the same code to be optimized and compiled depending on a client's chosen hardware platform and their specific network requirements. This is what Kaloom is offering in the context of its 5G user plane function (UPF), a fundamental component of 3GPP 5G architecture. UPF facilitates disaggregation of the control and data planes to deliver functionality, such as network slicing and cloud-native capabilities for service-based architectures.

Specifically, P4 code is written and exhaustively verified once, and then deployed on the right hardware for the applications' environment. This allows the use of "best of breed" hardware for a given set of required features, characteristics, and functionalities. For example, Kaloom can optimize its P4 software to run in a low-touch environment using Ethernet switch ASICs such as Intel Tofino and Tofino 2 to deliver multi-Tbps capacity. It can also be implemented in a medium-touch deployment by utilizing FPGAs and SmartNICs. In particular, running the same 5G UPF data plane P4 code on an FPGA, such as Intel® Stratix® 10 FPGA, empowered Kaloom to dramatically scale its solution from supporting hundreds of thousands of sessions to millions. Similarly, the P4 code could be instantiated for a high-touch environment (e.g., DPI) on general purpose CPUs such as Intel® Xeon® Processors.

Security and Network Slicing

Being based on well-established core underlying technologies, Kaloom's solutions can deliver military-grade security by utilizing Security-Enhanced Linux (SELinux), a security architecture initially designed by the National Security Agency (NSA). Kaloom also leverages cgroups, a Linux kernel feature that limits, accounts for, and isolates the resource usage of a collection of processes such as CPU time, system memory, network bandwidth, or combinations of these resources. Users can monitor the cgroups they configure, deny them access to certain resources, and even reconfigure them dynamically during runtime. Further, Kaloom uses network namespaces, as well as other containerization technologies provided by the Linux kernel, as lightweight mechanisms for resource isolation. Together, all of these make cloud native and P4-based solutions highly secure.

Kaloom solutions enable fully virtualizable fabric slicing with its vFabric capabilities. vFabric delivers a fully elastic and isolated network domain that is provisioned in software to deliver integrated network services. Further, each domain can be assigned different qualities for varying use cases or to meet unique SLAs. For example, they can be part of a virtual data center (vDC) for operators to offer various cloud services and each vFabric can simultaneously host millions of cloud service users (e.g., tenants). The use of a P4-based programmable fabric and networking functions can also greatly reduce the attack surface.

Conclusion

As seen in the industry support of P4.org, and the example of Kaloom, P4 is the leading domain-specific networking language and a foundational pillar for fully programmable and automated cloud-native networking solutions. P4 enables new innovation and functionality to a wide range of networking systems. P4-based networks enable customized protocol support, switch-based applications such as load balancing and firewalls, and INT for superior network control. Kaloom has embraced the technology in its Software Defined Fabric, Cloud Edge Fabric, and 5G UPF. With P4, network data planes can fully embrace the software programmability trends that are revolutionizing data center, enterprise, and telecom networks.

Notices & Disclaimers

[1] Source: P. Bosshart, D. Daly, G. Gibb, M. Izzard, N. McKeown, J. Rexford, C. Schlesinger, D. Talayco, A. Vahdat, G. Varghese, and D. Walker. P4: Programming protocol-independent packet processors. ACM SIGCOMM Computer Communications Review, July 2014.

[2] Barbara Liskov keynote address at the 2009 Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA) conference entitled "The Power of Abstraction." See also: http://www.pmg.csail.mit.edu/~liskov/turing-09-5.pdf

[3] https://www.kaloom.com/products

0820/RE/H09/PDF ♲ Please Recycle 344062-001US