White Paper

Data Center and Software-Defined Networking
Intel® FPGAs and Intel® Tofino™ Intelligent Fabric Processors

Baidu Intelligent Cloud Leverages Intel® Tofino™ Expandable Architecture to Meet 10Tbps-Level Bandwidth Requirements

Gateway devices implemented using Intel® Tofino™ Expandable Architecture support extra-large tables with hundreds of millions of entries and extra-large buffers containing tens of gigabytes of data.

Authors

Zhaogeng Li
Baidu Senior Engineer

Bert Klaps
Intel Senior Systems Engineer

Yi Lei
Intel Technical Specialist

Why Baidu?

- A leading global Internet service, artificial intelligence (AI) technology, and cloud provider
- Developed universal network platform (UNP) with programmability, flexibility, and high performance for variant cloud gateway use cases
- Co-engineering with Intel on Intel Tofino Expandable Architecture innovation with FPGA to greatly enhance UNP's capabilities

Why Intel?

- Leading provider of programmable switches, FPGAs, and CPUs
- Pioneer in end-to-end P4 device programmability
- Provider of state-of-the-art IP such as Search and HQoS

Executive Summary

Baidu is a Chinese multinational technology company specializing in internet-related services and products including artificial intelligence (AI) and machine learning (ML). Headquartered in Beijing's Haidian District, Baidu is one of the largest AI/ML, internet, and cloud service provider (CSP) companies in the world.

The Baidu intelligent cloud is constantly evolving to keep pace with the explosion of applications like high-performance computing (HPC), AI, ML, data analytics, and other specialized tasks that are driving the exponential growth of data. This necessitates the ability to move vast amounts of data throughout the data center with low latency and extremely high performance.

Data centers employ special network routers to provide this data transport. In some cases, a gateway device will sit at the boundary between the data center and the outside world. However, gateway devices may also link the cloud data center and tenants' data centers, may link the cloud data center and virtual private cloud (VPC) networks, and may be used to connect VPC networks with each other.

The earliest gateway devices were implemented using dedicated fixed-function hardware devices. However, these were not flexible enough to keep up with rapidly evolving communications protocols and standards. The next generation of gateways were based on clusters of x86 servers, which are extremely flexible, but it has become increasingly difficult to handle the hundreds of times growth in bandwidth requirements using x86 server clusters.

To address this problem, Intel introduced the Intel® Tofino™ Intelligent Fabric Processor (IFP) that can be programmed using the Programming Protocol-Independent Packet Processors (P4) open-source programming language.

The Intel Tofino IFP was originally designed to act as a data center switch, but its programmability led customers like Baidu to use it for more sophisticated and demanding networking functions. Although some of these new applications can be addressed with the Intel Tofino IFP capabilities, others require additional features and resources.

The solution is to employ the Intel Tofino Expandable Architecture, which involves augmenting the Intel Tofino IFP with FPGAs to provide 100X increase in table and buffer capacity. The result is the ability to support extra-large tables with hundreds of millions of entries and extra-large buffers containing tens of gigabytes of data, thereby satisfying the extreme requirements of the Baidu intelligent cloud.

Why Intel Tofino Expandable Architecture?

- Provides up to 10Tbps of bandwidth with scalable tables and buffers
- Fully P4 programmable system provides flexibility to meet ever evolving requirements
- Architecture based on Open-Standard System Architecture (OSSA) ensures portability

Evolving Cloud Compute and Memory Architectures

Today's computational workloads are larger, distributed, more complex, and more diverse than ever before. The explosion of applications like 5G, Internet of Things (IoT), high-performance computing (HPC), AI, ML, data analytics, and other specialized tasks are driving the exponential growth of data. In turn, processing all this data requires vast amounts of computational power coupled with low latency and high bandwidth access to the data within the data center. Also required is a way to get huge amounts of data into and out of the data center with low latency and extremely high performance.

Two of the ways in which data centers are evolving to address the computational needs of today's applications and workloads are virtualization and containerization. Virtualization allows a single physical server to be partitioned into multiple virtual machines (VMs), each of which can run its own operating system and applications while sharing the physical resources of the system (memory, storage, networks). Containerization refers to encapsulating an application in a container, which is a fully functional and portable cloud or non-cloud computing environment that includes the application along with any associated libraries and other dependencies.

Different applications and workloads require different combinations of resources. Some tasks may be GPU-centric, for example, requiring thousands of GPUs but leaving CPUs and memory underutilized. By comparison, other tasks may demand vast quantities of memory as compared to the number of CPUs and GPUs, thereby leaving large quantities of these resources sitting idle.

To address these demands, modern hyper-scale data centers—including those used by enterprise-level entities and those offered by cloud service providers (CSPs) and communications service providers (CoSPs)—are adopting the concept of disaggregation. This involves "resource pooling" in which shelves contain only CPUs, GPUs, NPUs, FPGAs, memory, or storage, and racks contain different collections of these shelves.

Disaggregation allows the concept of virtualization to be extended to encompass automatically composing collections of resources are available on-the-fly to meet the computational and memory requirements of each task on an application-specific and workload-specific basis. Once an application has completed its task, its resources are released back into their respective pools, after which they can be provisioned to future applications in different ratios.

Evolving Cloud Network Architectures and Gateways

As was previously noted, in addition to vast amounts of computational power, data centers also need to transport huge amounts of data both within the data center and with the outside world.

Special network routers (switches) called gateways are used to provide this low-latency, high-bandwidth data transport. In some cases, a gateway device will sit at the boundary between the data center and the outside world. However, gateway devices may also link the cloud data center and tenants' data centers, may link the cloud data center and virtual private cloud (VPC) networks, and may be used to connect VPC networks with each other.

The earliest gateway devices were implemented using dedicated fixed-function hardware devices. However, these were not flexible enough to keep up with rapidly evolving communications protocols and standards, and they were limited in terms of bandwidth.

As Intel x86 CPUs evolved in terms of power and performance, the next generation of gateways were based on clusters of x86 servers. However, the exponential growth of artificial intelligence and big data applications means that the amount of data traffic has increased dramatically. As a result, it has become increasingly difficult to handle the hundreds of times growth in bandwidth requirements using x86 server clusters.

In response to increasing bandwidth and infrastructure requirements, the industry started to adopt solutions based on a combination of hardware and software. A prime example of this would be an Intel Tofino IFP-based gateway that can be programmed using the Programming Protocol-Independent Packet Processors (P4) open-source programming language.

P4 is tailored for controlling packet forwarding planes in networking devices such as routers and switches. In contrast to a general-purpose language such as C or Python, P4 is a domain-specific language with a number of constructs optimized for network data forwarding. P4 is distributed as open-source, permissively licensed code, and is maintained by the P4 Language Consortium, a not-for-profit organization hosted by the Open Networking Foundation.

The combination of the Intel Tofino IFP and the P4 programming language means that a gateway based on this technology can deliver intelligence, performance, visibility, and control with program-optimized power consumption, real-time in-band network telemetry (INT), and advanced congestion control for workloads spanning the entire edge-to-cloud spectrum. Furthermore, to support the proliferation of AI, Intel Tofino IFPs offer intelligent packet processing to accelerate machine learning workloads.

In addition to the Intel Tofino IFP itself, an example Intel Tofino IFP-based gateway might include one or more x86 CPUs, where each CPU has an associated Foundational NIC (Figure 1a). The CPUs are connected to each other and to their Foundational NICs by means of Peripheral Component Interconnect Express (PCI Express or PCIe) interfaces, and the network interface cards (NICs) are connected to the Intel Tofino IFP using 100 gigabits per second (Gbps) Ethernet connections.

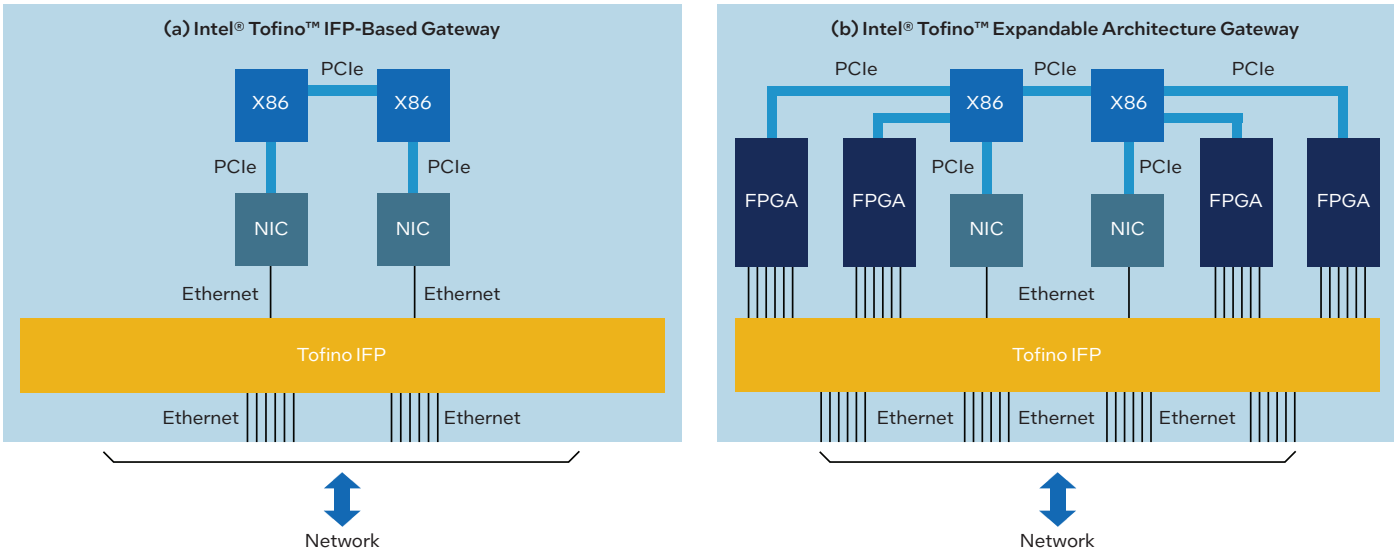

Figure 1. Intel Tofino IFP-based Gateway (left) vs. Intel Tofino Expandable Architecture Gateway (right).

In this case, the CPUs aren't using the Foundational NICs to talk to the network (that's the Intel Tofino IFP's job); instead, the NICs are being used as translators to support the legacy drivers and application programming interface (API). This makes it easy to port designs from existing x86 server clusters with minimal changes to the design because the control path remains on the CPUs while the data path is migrated to the Intel Tofino IFP.

It's important to note that the Intel Tofino IFP was originally designed to act as a data center switch, and it has more than sufficient capabilities to address these applications. However, its programmability has led customers to use the Intel Tofino IFP for more sophisticated and demanding networking functions. Although some of these new applications can be addressed with the Intel Tofino IFP's capabilities, others require additional features and resources.

The problem is that today's network traffic demands can easily reach multiple terabits-per-second (Tbps). The scale and growth of modern hyper-scale data centers makes horizontal scaling (adding more servers in the case of server clusters, even adding more Intel Tofino IFP-based gateways) unsustainable due to increased capital expenditure (hardware acquisition costs) and OPEX (maintenance, management, troubleshooting).

One of the technical challenges with respect to the Intel Tofino IFP-based gateway is limited total on-chip memory capacity for storing the Virtual Extensible LAN (VxLAN) routing table (~1M entries) and VM node controller (VM-NC) mapping table (~1M entries), while stateful tables and ultra-large table entries—such as a source network address translation (SNAT) table, which allows traffic to pass from the internal network to the internet—can easily contain 100M+ entries.

The solution is to employ the Intel Tofino Expandable Architecture, which involves augmenting the Intel Tofino IFP with FPGAs (Figure 1b) to provide 100X increase in table and buffer capacity. The result is the ability to support extra-large tables with hundreds of millions of entries and extra-large buffers containing tens of gigabytes of data (Figure 2).

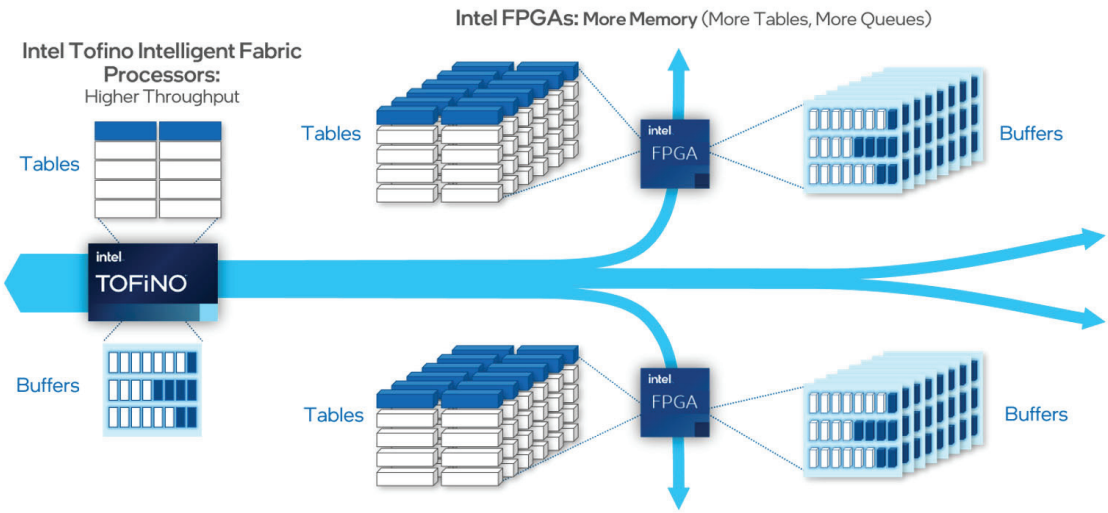

Figure 2. Complementing the Intel Tofino switch with FPGAs to provide 100X increase in table and buffer capacity.

Baidu Intelligent Cloud Leverages Intel Tofino Expandable Architecture

Baidu is a Chinese multinational technology company specializing in internet-related services and products, Cloud and AI. Headquartered in Beijing's Haidian District, Baidu is one of the largest AI, internet, and CSP companies in the world.

The Baidu intelligent cloud is constantly evolving to keep pace with the explosion of applications like HPC, AI, ML, data analytics, and other specialized tasks that are driving the exponential growth of data. As part of this evolution, Baidu has employed x86 servers, Intel Tofino IFPs, and—now—Intel Tofino Expandable Architecture as gateways.

Prior to the Intel Tofino Expandable Architecture

Before the Intel Tofino IFP became available, Baidu used only x86 server clusters as gateways. With the introduction of the Intel Tofino IFP, Baidu also started to use the Intel Tofino IFPs as gateways. The reason for using these two types is that x86 server clusters provide extreme flexibility and reasonable performance, while the Intel Tofino IFP provides extreme performance and reasonable flexibility.

In the case of relatively simple gateways, Intel Tofino IFPs can provide the best performance and total cost of ownership (TCO). A typical use case for Intel Tofino IFPs in Baidu is as virtual routers. They play an important role in the cloud by connecting different virtual private clouds (VPCs) and different subnets in one VPC.

In addition to routing rules, which are used for forwarding packets, a virtual router also has ACL rules, which are used for filtering packets, meter rules, which are used for rate limiting, and NAT rules, which are used for address translation. All of these rules are stored in the Intel Tofino IFP's SRAM and TCAM. By introducing pipeline folding, an Intel Tofino IFP (64Q) can be used as a 1Tbps virtual router with stable low latency (about 2us). (Note that the practical 1Tbps is smaller than the theoretical 1.6Tbps because metadata is also carried by packets between folded pipelines.)

Intel Tofino IFP-based gateways satisfy the requirements of many cloud deployments. For example, a virtual router implemented by an Intel Tofino IFP can support 30,000 multi-dimension routing rules (as demonstrated in a Baidu use case). However, this may not satisfy the extreme requirements associated with super large-scale regions with super large-tenant use cases. Similarly, Intel Tofino IFPs may not be suitable for use in complex stateful gateways because of the state size and state update frequency.

Due to these limitations, x86 servers are still widely used for cloud gateways. Take server load balancing (SLB), for example. The main function of SLB is to dispatch a request for a virtual server to a real server. To realize the desired statistics and consistent dispatch (let all the packets of one or more relevant requests from one client go to the same real server), the SLB device maintains session information. Generally, there are tens of millions or even hundreds of millions of sessions in one SLB device, and this data cannot be stored by an Intel Tofino IFP. If all four pipelines are used (with pipeline folding) in one Intel Tofino IFP, as many as 200,000 sessions can be stored (as demonstrated in a Baidu use case). As good as this is, it may not be sufficient to satisfy the extreme requirements of hyper-scale data centers. In the case of the Baidu Intelligent Cloud, for example, satisfying the demands of SLB requires x86 servers as opposed to Intel Tofino IFPs.

Unfortunately, although the combination of a powerful data plane development kit (DPDK) and multi-core CPUs substantially accelerate performance on x86 servers, these software gateways still face significant challenges in modern cloud architectures as follows:

1) Performance Challenges: The average latency of a software gateway in Baidu is typically 30-100 us, and long-tail latency may even rise to 1 ms. The main reason for this latency instability is the unpredictable cache and memory accesses associated with CPUs. On the other hand, since packets in the same flow should be assigned to one specific core to avoid out-of-order processing, a single-core becomes the throughput bottleneck for its associated flow. All this means that a software gateway can support at most 10 to 15 Gbps throughput for one flow. As a result, an elephant flow (an extremely large [in terms of total bytes] continuous flow) may encounter severe packet drop while also affecting mice flows (short [in terms of total bytes] flows) as victims sharing the same core.
2) TCO Challenges: An x86 server can typically process about 100 to 400Gbps traffic in the Baidu Intelligent Cloud, depending on the specific type of gateway. Since the total bandwidth requirement for the Baidu Intelligent Cloud is in the order of 10Tbps at the time of this writing, hundreds of x86 servers are required to implement a single cluster at a single site. Baidu requires tens to hundreds of such clusters, which introduces a tremendous TCO burden.

The Intel Tofino Expandable Architecture provides an ideal solution for cloud gateways with massive demands.. The FPGAs featured in the Intel Tofino Expandable Architecture are key programmable components that can provide both great performance (like switch ASICs) and great flexibility (like x86 CPU servers) at the same time. As a result, Baidu now employs Intel Tofino Expandable Architecture as its next-generation cloud gateway platform, which is known as the Universal Network Platform (UNP).

Current Use

Currently, Baidu uses the Intel Tofino Expandable Architecture implementation illustrated in Figure 3 to accelerate SLB. A packet coming into the switch from the network is dispatched to one of the four FPGAs. If the session information is located on the FPGA's cache, then the packet will be modified accordingly and sent back to the network via the Intel Tofino IFP. If there is no related session information stored in the FPGA, this packet will be sent to the x86 CPU via the Intel Tofino IFP and NIC. The x86 CPU handles the packet just like an ordinary software SLB and creates the corresponding session information. The packet will then be sent back to the network via the NIC and Intel Tofino IFP; also, the session information will be recorded in a specific FPGA via PCIe, and this information can be looked up for subsequent packets.

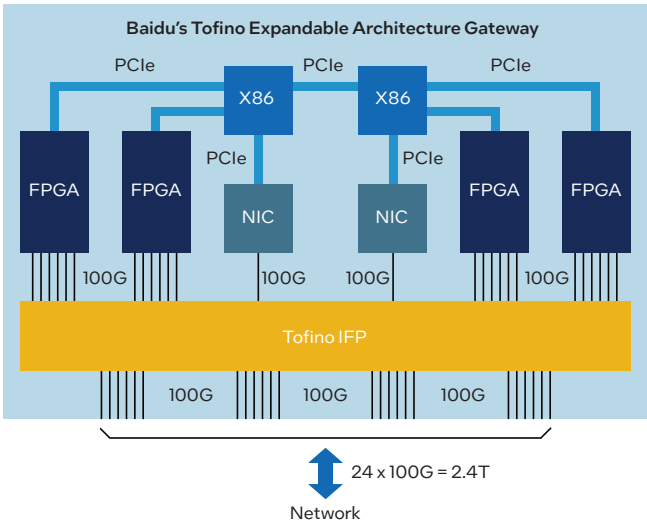

Figure 3. Baidu's Tofino Expandable Architecture Gateway.

In the Baidu use case, each FPGA connects to the Intel Tofino IFP with 6 x 100Gbps Ethernet links and provides at least 300 mega-packets per second (Mpps) processing capability, which makes the whole device capable of processing 2.4Tbps traffic. Furthermore, each FPGA can support at least 25 million sessions (according to the memory size associated with the FPGA), which makes the whole device capable of supporting at least 100 million sessions. Additionally, the processing latency of most packets can be limited around 3µs (except for the first packet of a flow), which is significantly better than a software SLB implementation.

This new SLB platform solves both performance and TCO problems. First, an elephant flow with 100Gbps throughput can be supported with no victim flows being affected, which is extremely important with respect to the load balancing of Baidu cloud services such as network file system and object storage. Second, to build a 10Tbps SLB cluster, no more than 10 Intel Tofino Expandable Architecture devices are needed, which reduces the TCO by at least 50% as compared to software solutions.

Future Use

As discussed above, the Intel Tofino Expandable Architecture is a promising new platform for gateways, and it is still evolving. Baidu plans to use such platforms for additional gateway use cases. As an example, a virtual router for a large-scale use case is under development. Since resources are easy to extend via FPGA, the router can support millions of routing, ACL, NAT, and metering rules. Security use cases such as cloud firewalls (stateful and stateless) and intrusion detection systems (IDSs) are likely to be the next step.

Summary

Before the Intel Tofino IFP became available, Baidu used x86 server clusters as gateways. With the introduction of the Intel Tofino IFP, Baidu also started to use Intel Tofino IFPs as gateways. The x86 server clusters provide extreme flexibility and reasonable performance, while Intel Tofino IFP provides extreme performance and reasonable flexibility.

The Intel Tofino IFP was originally designed to act as a data center switch, but its programmability led customers like Baidu to use it for more sophisticated and demanding applications, such as networking. Although some of these new applications can be addressed with the Intel Tofino IFP capabilities, others require additional features and resources.

To address evolving need to move vast amounts of data with low latency and extremely high performance, Baidu is migrating to gateway devices based on the Intel Tofino Expandable Architecture. This involves augmenting the Intel Tofino IFP with FPGAs to provide 100X increase in table and buffer capacity. The result is the ability to support extra-large tables with hundreds of millions of entries and extra-large buffers containing tens of gigabytes of data, thereby satisfying the extreme requirements of the Baidu intelligent cloud.

Additional Resources

Tofino Expandable Architecture White Paper

Intel.com/fabric

WP- 01322-1.0