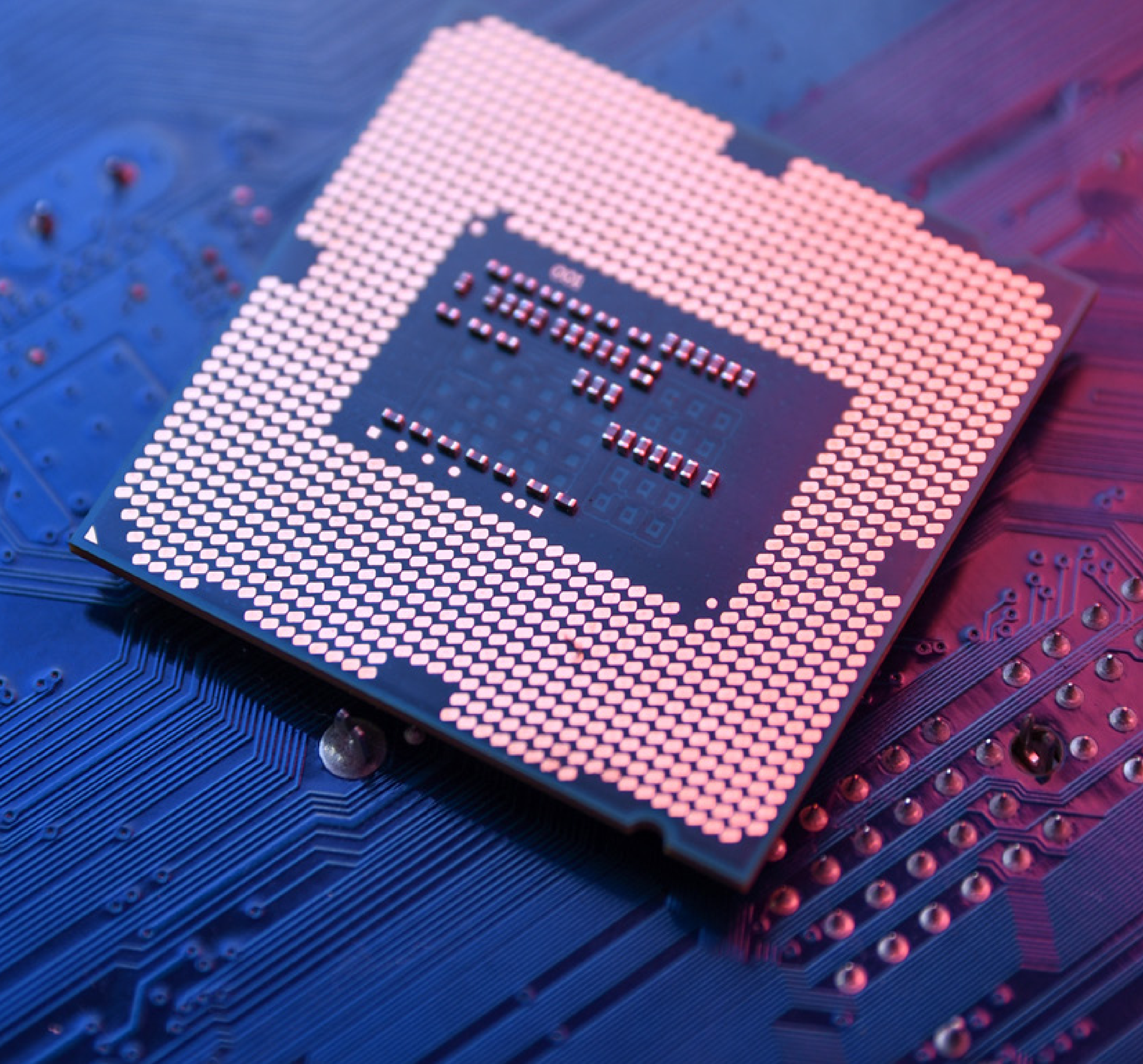
HCLSoftware | intel®

HCLSoftware accelerates Viettel High Technologies 5G User Plane Function

with Intel® FPGA | SmartNIC N6000-PL Platform



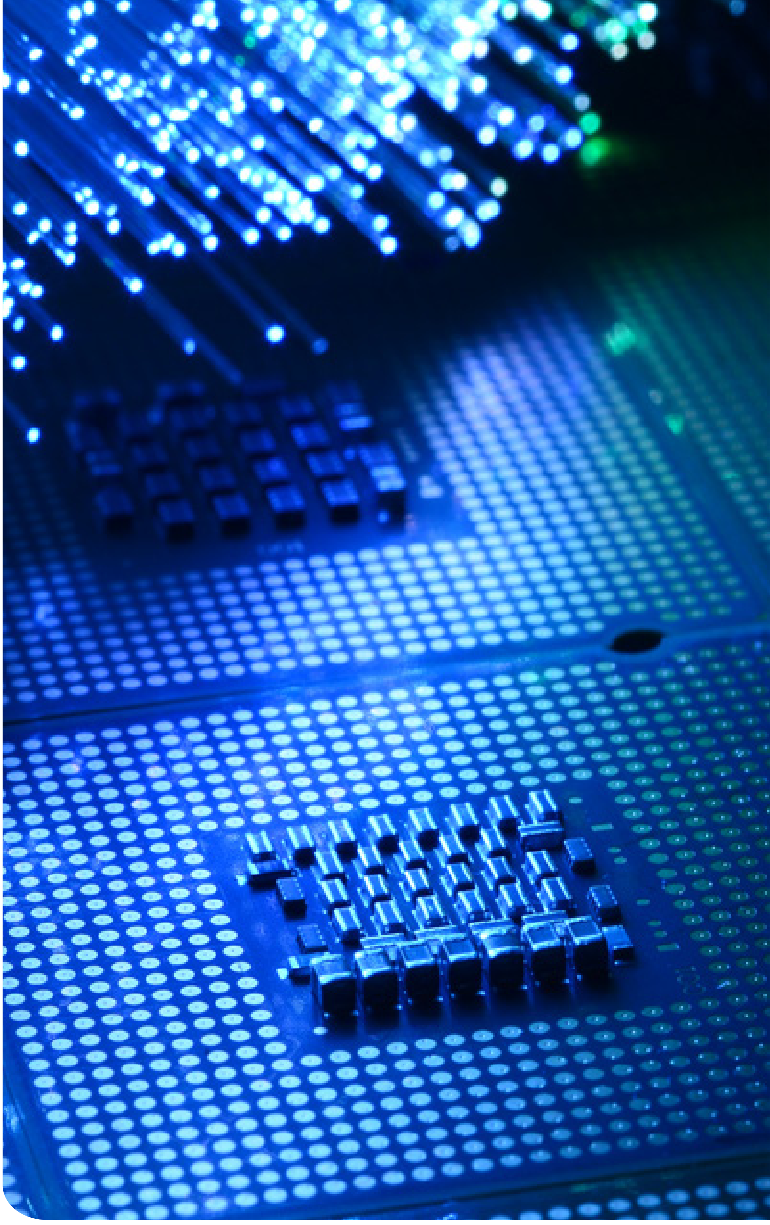
Introduction

In today's fast-paced era of 5G technology, the demand for high-performance networking solutions has become increasingly critical for businesses worldwide.

User Plane Function (UPF) is a data plane function responsible for operations like mobility (hides mobility to external network), routing and forwarding, packet inspection and policy enforcement, buffering and QoS enforcement, provide charging data and traffic statistics to Session Management Function (SMF). UPF performances in terms of achievable throughput and reliable/predictable latencies are critical to optimize operators' networks as well as customer experience.

In Figure 1, the reference 5G architecture with all Network Functions (NF) and interface definitions are reported: UPF is connected to RAN through N3 interface which is based on protocol stack reported in Figure 2. GPRS Tunneling Protocol (GTP-U, interchangeability referred as GTP) is used for both N3 and N9 interfaces (N9 being the interface between two different UPF functions). N6 is the interface towards Data Network (DN), and it is usually pure IP based interface (either IPv4 or IPv6 or both for dual stacked interfaces). All these are data plane interfaces.

N4 is the UPF interface towards Session Management Function (SMF) which is responsible for selecting and instructing UPF on how to handle Protocol Data Unit (PDU) Session.



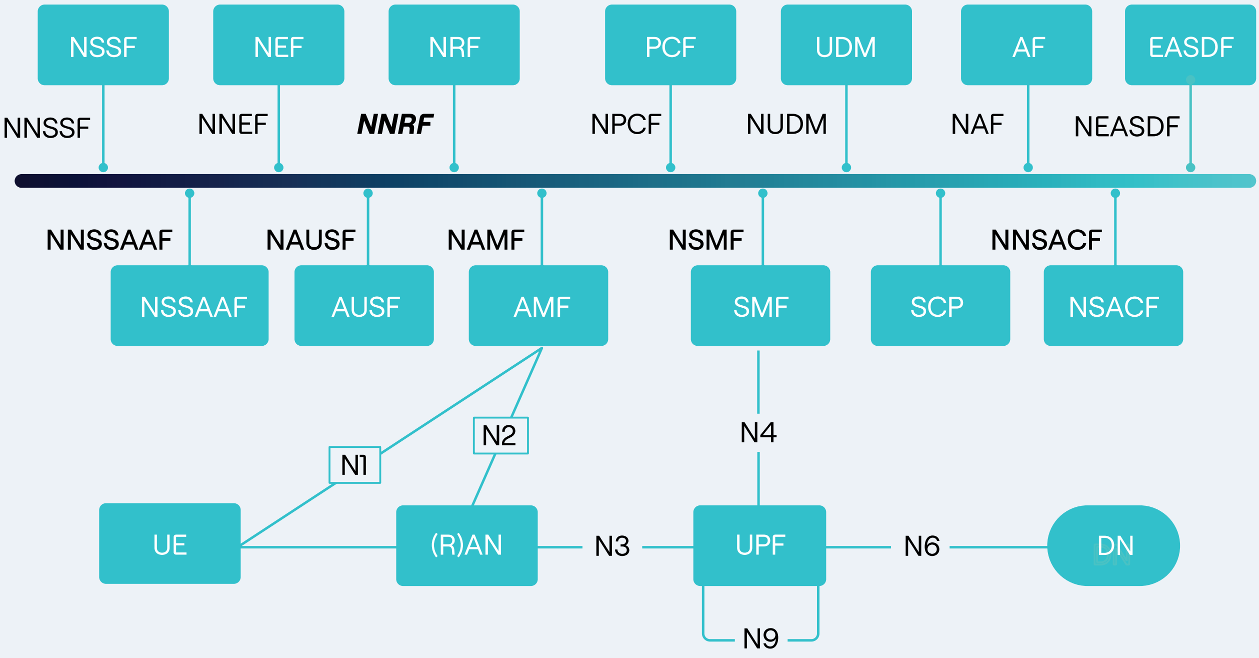


Figure 1. 5G Reference Architecture

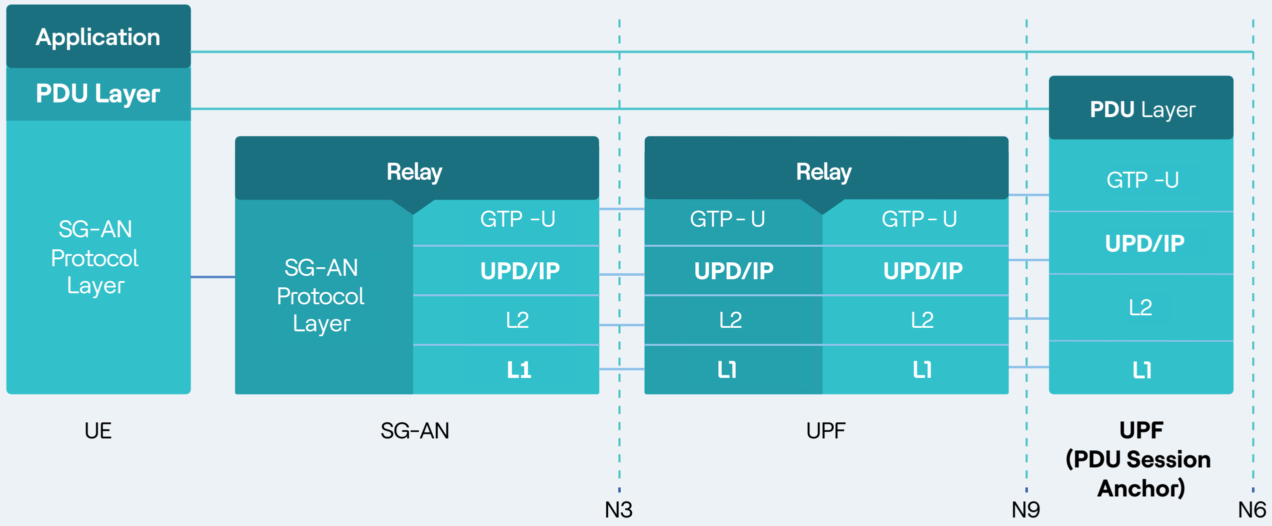


Figure 2. 3GPP Protocol Stack for N3 and N9 Interfaces

UPF being the main data plane function in 5G core network, its performance is crucial to operators when it comes to achievable performances vs price point as well as for customer experience. When moving UPF into Cloud Native environment, the above requirements require careful attention. Several solutions have been analyzed and discussed in the past.

01

Data Plane Development Kit (DPDK):

This solution improves packet processing operations performances on x86 **architectures** compared to Kernel based option. However, DPDK still required CPU cores to be dedicated to packet forwarding and being a software- oriented solution, latencies are not directly controllable but are dependent on core utilization and other processes running on the same x86 platform.

02

Dynamic Device Personalization (DDP):

This NIC feature allows optimal data distribution over DMA queues towards the host device by performing deep packet parsing and use of suitable hashing function.

Smart Network Interface Cards (SmartNICs) can perform computationally intensive data plane operations thus freeing up CPU resources on host servers which can be dedicated to customer and business-related applications. A SmartNIC-based solution holds promise for enhancing performance to levels comparable to dedicated hardware, all while enabling seamless orchestration of the entire system in the cloud, without the necessity for specialized hardware or external devices. SmartNICs can be seamlessly integrated into existing platform in cloud native environment improving the scalability and flexibility of the overall solution. Also, by being programmable, a SmartNIC allows customization and can address unique requirements from different operator networks and services.

UPF is a very complex function which requires huge resources in terms of counters for charging function, policers for traffic limiting and other routing function to be able to provide granular data traffic control at flow level. This can be implemented in FPGA based SmartNIC. The flow defined as 5-tuple combination of inner N3/N9 packets or N6 packets is a basic entity that defines offload function. The FPGA has enough capabilities to make a full offload of UPF function possible by storing millions of flows in DDR memory available on SmartNIC card.

The SmartNIC card referred in this paper provides two 100G Ethernet interfaces and it has peripheral component interconnect express (PCIe) form factor FHHL with on-board Intel Ethernet Controller E810 providing standard host interface based on SR-IOV (Single Root Input/Output Virtualization). The E810 helps to enable the best features of traditional networking like DPDK and DDP along with FPGA providing additional flexibility, all with a single SmartNIC.

HCLSoftware, the software business division of HCLTech collaborated with Viettel High Technologies (VHT) a subsidiary of Viettel group, a top Telecom Infrastructure R&D company, a leading mobile network operator in Vietnam, to deploy a 5G User Plane Function offloading solution utilizing the Intel FPGA SmartNIC N6000-PL Platform. This SmartNIC platform, based on Intel Agilex® 7 FPGA technology, offers 2x100GbE connectivity and is designed to support hardware-programmable acceleration for Network Functions Virtualization (NFV) environments. By leveraging Intel board capabilities, HCLSoftware solution can improve efficiency and scalability, maximize throughput while minimizing latencies and optimize Total Cost of Ownership.

Extensive testing of Viettel specific network scenarios have demonstrated that the maximum overall performance can reach 100 Gbps, supporting approximately 500,000 user subscribers (UE) with almost no server CPU core consumption.

This document presents a comprehensive overview of the solution, performances in the reference test environment, associated benefits, and potential future implications.

Solution overview

This section provides an overview of VHT core network and the logical architecture of the User Plane Function (UPF) and dives deeper into the technical implementation of the HCL 5G User Plane Function Offload solution that leverages the Intel FPGA SmartNIC N6000-PL Platform.

The integration of HCL Offload solution with VHT 5G core network UPF will be described, along with the technical details of the solution. This comprehensive explanation aims to provide readers with a thorough understanding of the solution and its various components.

VHT Core Network and UPF logical architecture

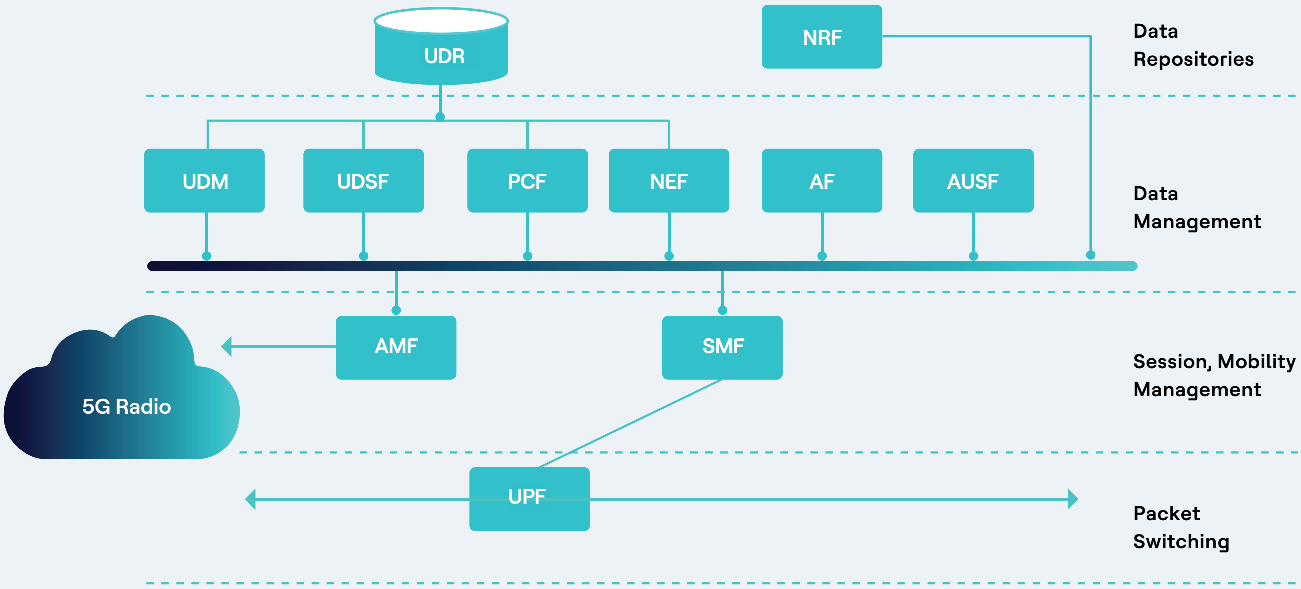


Figure 3. 5GC Service based Architecture.

The 5GC architecture (see Figure 3) is defined as service-based architecture with interaction between network functions divided into four components:

Packet switching:	Session mobility Management:	Data management:	Data repositories:
Provide data service of subscribers.	Manage mobility service of subscribers on the control plane	Manage information of subscribers	Manage information request of the subscribers.

In 5GC topology, UPF main function is packet switching. UPF acts as an anchor point with three main features:

- **Forward data between RAN and PDN**
- **Monitor for charging**
- **Apply Policy**

VHT developed its 5G core network using microservices architecture and deployed it on a Kubernetes virtualized infrastructure. This choice of infrastructure brings forth several advanced features, including automated deployment, scalability, load balancing across container groups and controlled network port exposure to external systems.

For their 5G core solutions, VHT utilizes Continuous Integration/Continuous Deployment (CI/CD) practices, open-source platform services, telco- oriented open sources, and TTCN3 for automation testing. TTCN-3 (Testing and Test Control Notation version 3) is a standardized testing language specifically designed for developing and executing test suites for communication protocols and systems. TTCN-3 plays a crucial role in guaranteeing the quality and interoperability of network protocols, services, and applications in the telecommunications industry.

By incorporating these methodologies and tools, VHT ensures efficient and streamlined development and deployment processes.

VHT 5G core network encompasses the network functions defined by the 3rd Generation Partnership Project (3GPP) for both Evolved Packet Core (EPC) and 5G Core (5GC) architectures. These functions are integrated into a cloud-native platform, enabling flexible deployment options for various network architectures such as 4G, 5G Non- Standalone (NSA), or 5G Standalone (SA) based on specific requirements. This flexibility allows VHT to adapt their network infrastructure according to the evolving needs of their customers.

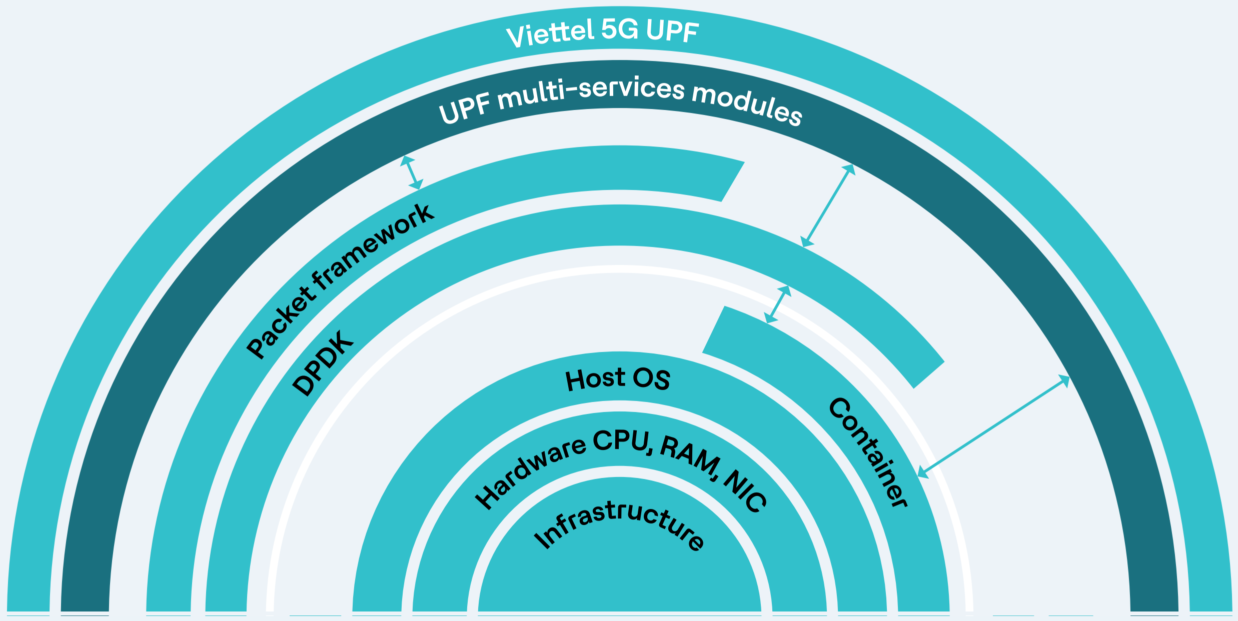


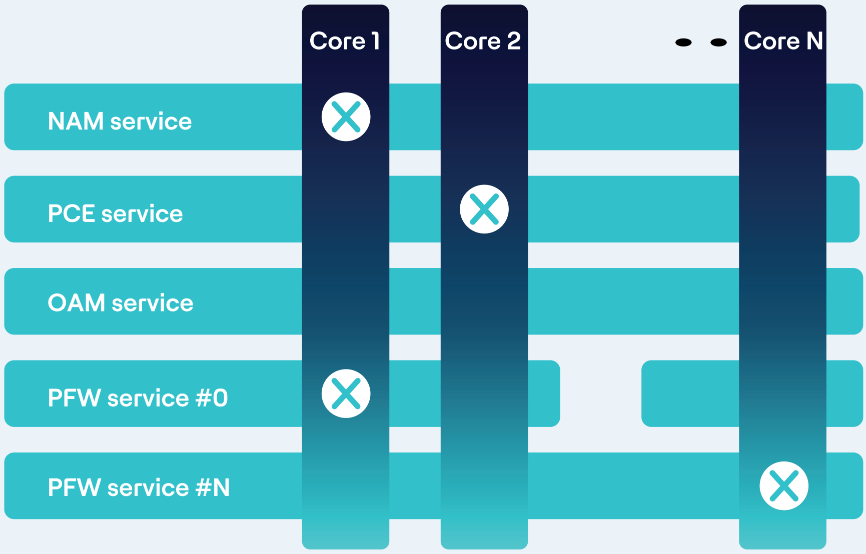
Figure 4. Viettel UPF Logical Architecture

Figure 4 illustrates the logical architecture of the User Plane Function (UPF). To optimize performance and meet the high demands of 5G networks, VHT leverages the benefits of the DPDK (Data Plane Development Kit) packet processing framework and containerization.

DPDK provides direct access to network interfaces, allowing for efficient packet processing with minimal latency. This enables the 5G UPF to achieve high throughput and reduced processing overhead, addressing the stringent performance requirements of 5G networks.

Containerization plays a crucial role in the efficient deployment and management of UPF instances. By utilizing containers, VHT can dynamically scale the UPF based on network traffic demands. Container orchestration platforms like Kubernetes enable horizontal scaling, allowing for the addition or removal of UPF instances as needed. This flexibility ensures optimal resource utilization and scalability for the UPF.

VHT 5G User Plane Function (UPF) encompasses various functions essential for subscriber plane operations, including traffic forwarding, content filtering, and policy enforcement. To ensure optimal performance, it is crucial to enhance and optimize the User Plane to efficiently handle scalable data traffic for different 5G use cases, such as enhanced Mobile Broadband (eMBB), Ultra-Reliable Low- Latency Communications (URLLC), and massive Internet of Things (mIoT) applications.



NAM Service:
Network Access Management Service.

PCE Service:
Policy Charging Enforcement Service.

OAM Service:
Operations, Administration, and Maintenance service.

PFW Service:
Packet Forwarding Service.

Figure 5. DPDK service-core allocation

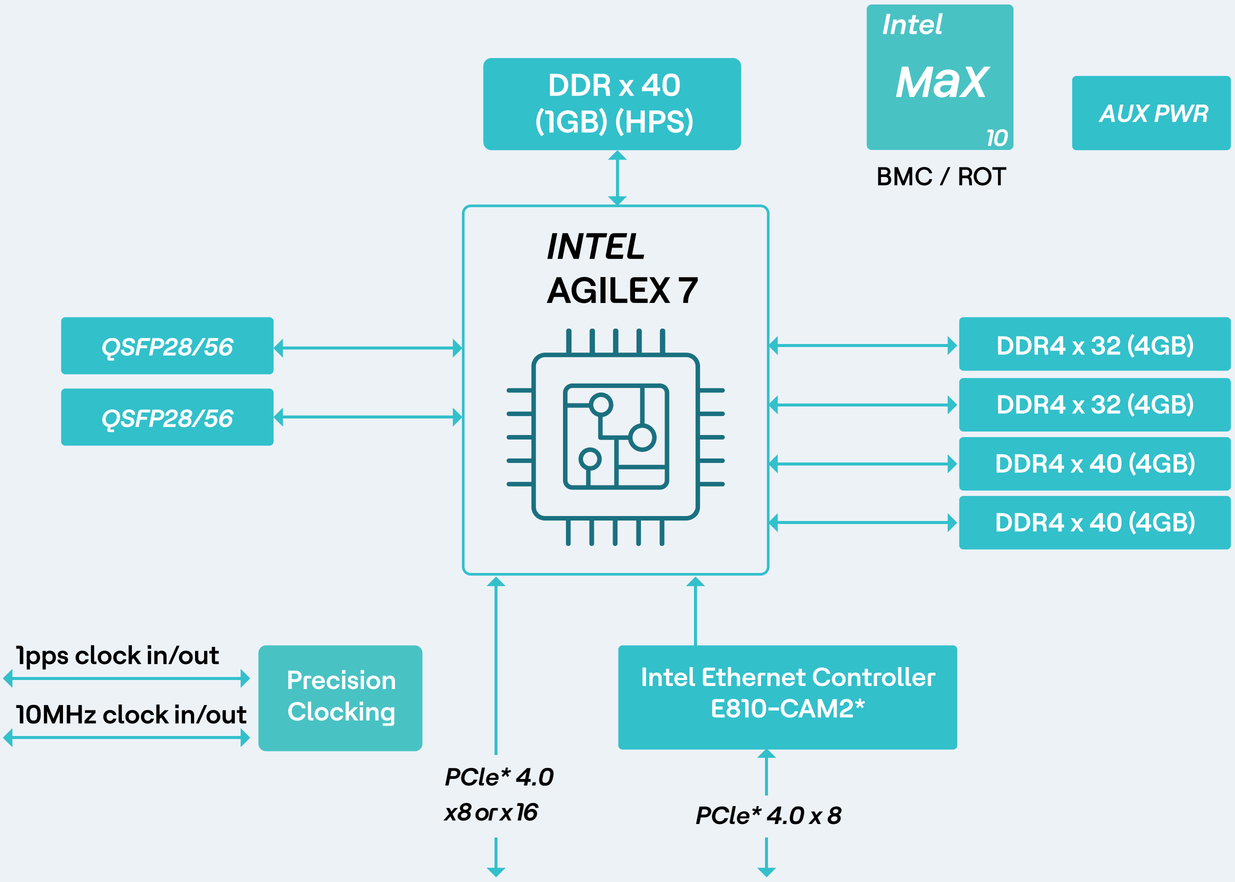
As illustrated in Figure 5, VHT has designed their 5G User Plane Function (UPF) as a modular model with multiservice core architecture. Each module within the UPF functions autonomously, performing distinct procedures. These modules are processed by one or more Data Plane Development Kit (DPDK) service-cores, which align with physical CPU cores. This approach enables efficient and simultaneous processing of tasks within the UPF, promoting enhanced scalability and performance to effectively handle a wide range of 5G workloads.

Intel FPGA SmartNIC N6000-PL Platform

The Intel® FPGA SmartNIC N6000-PL Platform is Intel 3rd generation SmartNIC providing 2x100 Gbps Ethernet connectivity. It is built using the Intel Agilex® 7 FPGA F-Series family, which utilizes advanced 10 nm SuperFin technology. This technology enables the platform to deliver approximately twice the fabric performance per watt when compared to competing 7 nm FPGAs.

Intel® FPGA SmartNIC N6000-PL Platform

PCIe from factor, Full Height, ½ Length, Single Slot



- **The N6000-based Smart NICs come in two configurations:**
- **N60000-based SmartNIC with an Intel Ethernet Controller E810 and bifurcated PCIe 4.0x8 lanes to FPGA and x8 lanes to the Ethernet controller**
- **N6001-based SmartNIC without an Ethernet controller, PCle 4.0 x 16 lanes to FPGA**

Figure 6. Intel FPGA SmartNIC N6000-PL Platform

The card uses on-board Intel Ethernet Controller E810 that could be configured as 8x10G/4x- 25G/2x100G for host applications. The NIC can support all NIC offload feature like RSS, TSO, checksum, virtualization features and DDP.

The card supports four DDR4 channels that could be used for implementing exact match lookup table with millions of entries as well as storage for millions of counters and large sets of policers used by HCL UPF offload solution. The performance of the offload design is limited by the DDR4 memory lookup.

The Intel BMC on the card is responsible for controlling, monitoring and giving low level access to board features. The Card BMC interfaces with on-board sensors, the FPGA and the flash, and controls power-on/power-off sequences, FPGA configuration and telemetry data polling. The FPGA firmware is field upgradeable over the PCIe using the remote system update (RSU) feature.

The card is enabled with SyncE & 1588v2 precision timing protocol (PTP) synchronization support for PRTC/ T-GM, T-BC, T-TC, T-TSC clocks. It has on-board OCXO providing timing accuracy and extended hold-over.

Production-ready COTS boards with reference workloads are available through ODM partners like Silicom, Winston NeWeb (WNC) and Artiza Networks. One can also accelerate their custom board design by re-using the N6000 board design and tailoring it to your unique requirements. The Intel FPGA SmartNIC N6000-PL Platform consists of board design files, pre-built FPGA Interface Manager (FIM) designs created by Open FPGA Stack (OFS), BMC design, Reference workloads, documentation and sample card. To license the Intel FPGA SmartNIC N6000-PL Platform design, contact an Intel sales representative.

More information on the Intel N6000 SmartNIC can be found at intel.com/n6000.

The HCLSoftware Solution described in this document is using Intel N6000-based SmartNIC with Intel® Ethernet Controller E810 and leverages capabilities of E810 Ethernet Controller to enable features like smart distribution of traffic towards the host system.

HCL UPF Offload Solution

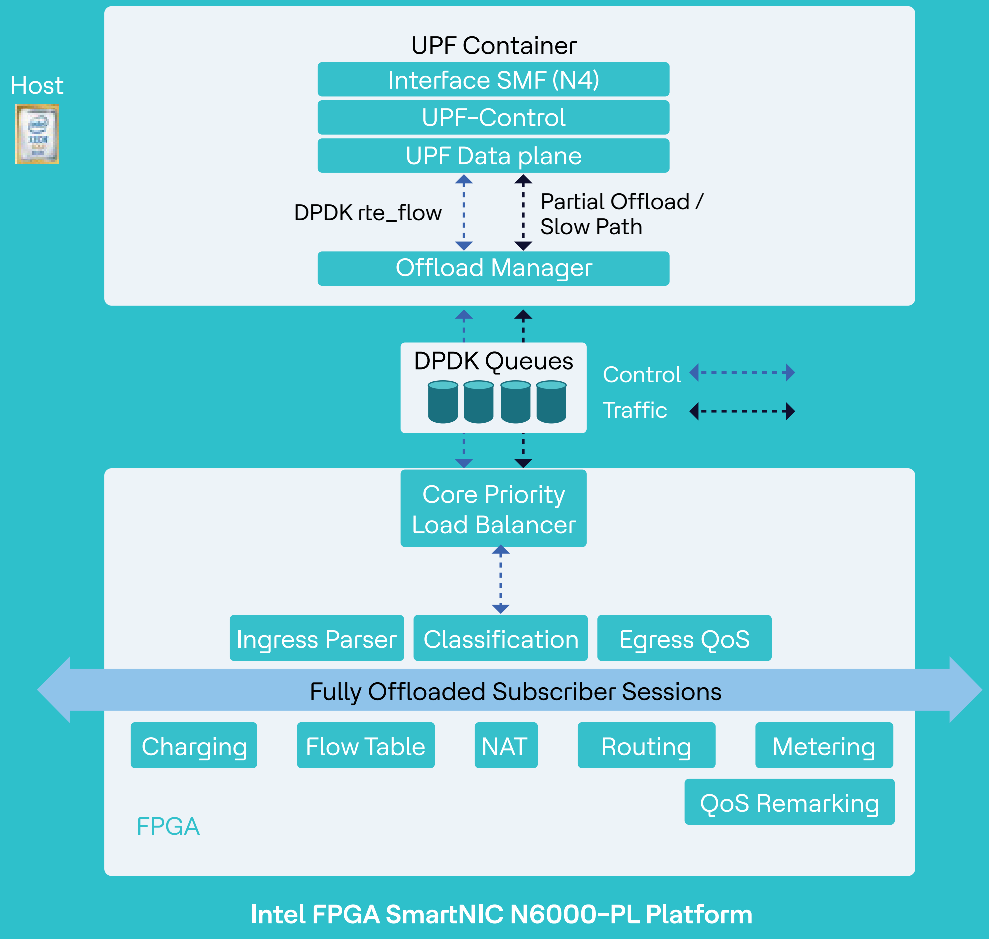


Figure 7. HCLSoftware Solution Block Diagram

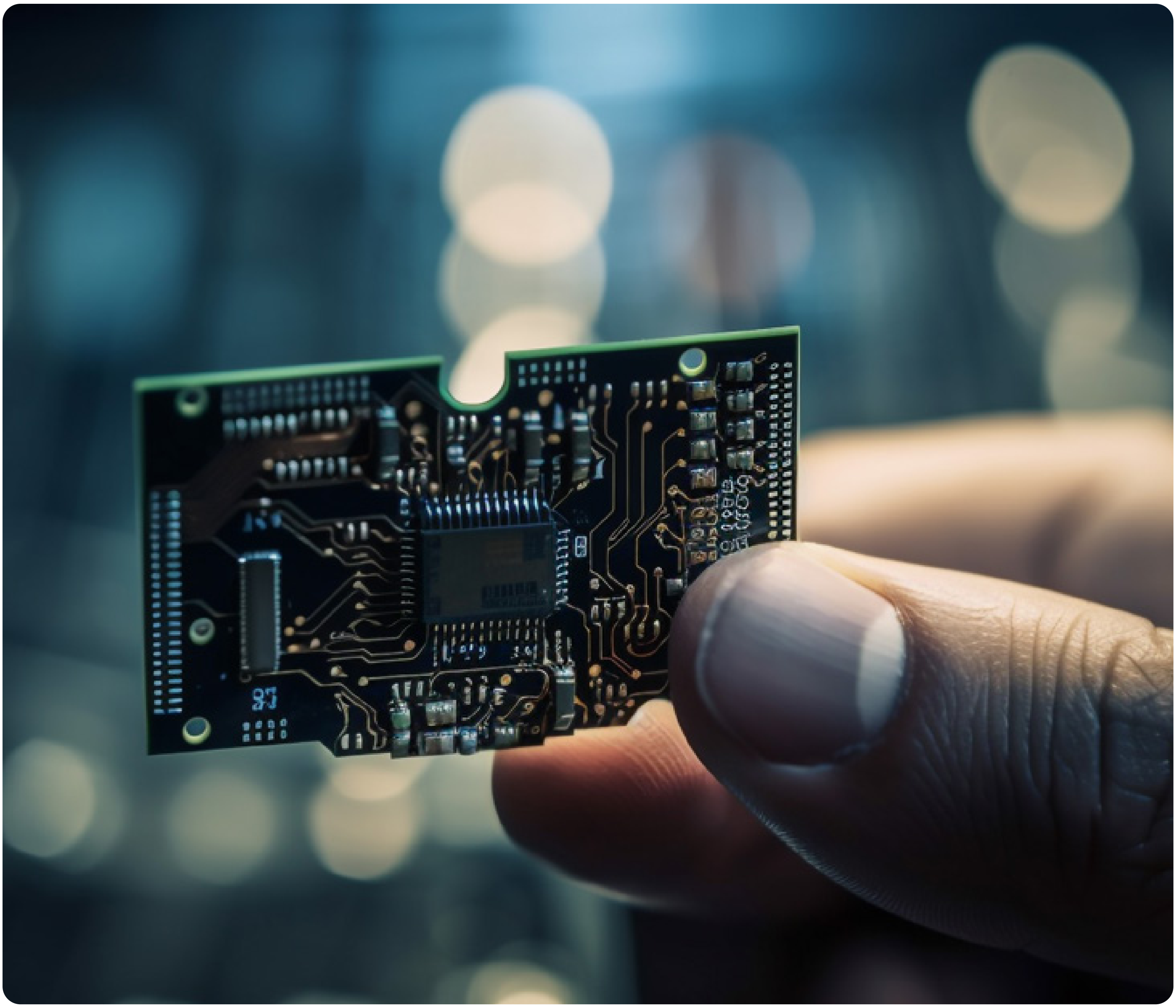
The HCLSoftware solution described in the figure above is designed for Intel SmartNICs and Infrastructure Processing Unit (IPU) platforms, taking advantage of the different characteristics of the hardware to maximize the customer TCO. The following solution describes implementation of HCLSoftware's solution, on Intel FPGA SmartNIC N6000-PL Platform however most of the concepts apply when the same solution is adopted for IPU Platforms.

When running on SmartNIC, this solution provides the UPF function with the possibility to offload subscriber flows. Offload can be either partial or full: in partial offload, the solution will execute limited processing on incoming traffic (for example GTP Encap/ Decap operations) while remaining processing is demanded from Host CPU. This way, even when processing operations are performed in host device, the SmartNIC can greatly help by preparing data for better consumption in host. As an example, Deep packet Inspection stateful engines can run in host on specific cores which will be properly addressed by SmartNIC traffic distribution. At the same time, the SmartNIC can redirect control traffic to different cores for protocol termination by assigning specific priorities to them.

Exception path or Slow path (software-based processing flow) can leverage DPDK multiple queues to allow priority load balancing to different CPU host cores, improving performances and system predictability.

This solution provides in band management protocol to configure and monitor accelerator function in hardware: this solution guarantees very high Flow Insertion rate which is desirable feature in highly dynamic mobile environments.

The RTE Flow API simplifies the development of packet processing applications by providing a standardized interface for creating and managing VLAN, flow, counter, and policer tables.



Below is the description of all main blocks which compose HCL UPF Offload Solution.

01 N6K DPDK Drivers

HCLSoftware Solution exposes a standard DPDK rte_flow API to the upper UPF layers through Intel N6000-based SmartNIC DPDK driver: the driver is responsible to translate rte_flow commands into hardware related calls thus hiding complexity of the hardware from UPF software. UPF can simply configure hardware as any other DPDK application. Rte_flow API function calls are translated into inband protocol commands to maximize flow insertion rate and statistics retrieval. Statistics can be made available to Service Assurance and Charging Modules to monitor service level and charge subscribers accordingly. By leveraging multiple queues in DPDK, the driver provides high rate for partial offload and exception path as well as prioritized handling of control traffic.

02 Ingress Parser

Pipeline logic takes input traffic from either of ethernet ports (QSFP-28 based ports) or Slow Path/ Exception Path port (Host Interface). Several configurations are possible: N3, N6, N9 interfaces can be associated to physical ports or to specific VLANs. Moreover, front ports can be statically bonded or through LACP. Active-Standby configuration (one single port active at a time) is also possible.

03 5-Tuple Match-Action Table

A flow is identified by a 5-tuple (IP source and IP destination Addresses, L4 source and Destination ports, L4 protocol) based on inner packet header in case of GTP based interface (N3 or N9) and outer packet header in case of pure IP based interface (N6). Fast lookup engine in hardware supports high speed memory access for classification process (using Match-Action Flow Table). 5-tuple Keys and corresponding results will be stored in DDR4 which allows for large number of flows to be installed in hardware for fast path forwarding.

04 Monitoring and Service Assurance

Packet counters, Byte counters per flow will be stored in DDR4. For each counter, a Watermark can be set to trigger specific event to be captured and sent to service assurance. Counters can also be associated to aggregated flows to monitor subscriber traffic or application specific sessions. Mirroring traffic to a different destination allows for advanced monitoring of specific flows.

05 Slow Path/ Exception Path Handling

In case of a Lookup miss, traffic will be sent to host as exception/slow path. Core selection will be handled here based on Receive Side Scaling (RSS) or control traffic specific mapping.

06 Protocol Handling

N3 is a GTP based tunneled interface. FPGA application will encapsulate/decapsulate.

Packets based on configured operations.

Network Address Translation and L4 Port Translation can be configured per flow for N6 interface.

07 Quality of Service Handling

Up to three level metering is available in FPGA application. Those meters can be associated to a single flow and same meter can also be associated to more flows at the same time providing aggregate policing functions, thus giving the optimal flexibility to address all possible UPF scenarios including slicing. For example, one level of policing can be associated to a given slice while other two levels can be per UE and per UE per Application. Policers are IETF compliant and are responsible for handling the QoS priority. Priority is based on policer colors.

QoS remarking in hardware provides the operator the possibility to associate 5G Quality of Service indicator to Differentiated Service Code Point (DSCP) values for seamless enable end to end QoS.

Priority Buffering in hardware can be configured to enforce QoS Policies.

The table below summarizes the main characteristics of HCL UPF Offload Solution

Key Features	Description
Interfaces	• 2x100GbE QSFP based interface. • N3/N6/N9 interfaces configurable per physical port or VLAN • Static LAG, dynamic LACP and Active-Standby options
Protocols	• Traffic types: VLAN based, IPv4, IPv6, GTPv1 • 3GPP TS 29.281 Release 17 GTP with 3GPP TS 38.415 • PDU Session container extension header • IPv4 Network Address Translation and Port Translation
Classification	• Up to 16 million 5-tuple Flows in hardware
QoS	• QFI to DSCP Remarking • Up to 1 million policers (IETF RFC2697 and IETF RFC2698) – either per flow or aggregate • Up to 3 policers per flow • 2-level DSCP based configurable priority flow load balancing towards host queues with RSS • Configurable Control Protocol priority towards host
Monitoring and Service Assurance	• Packet and Byte counters • Up to 2 counters per flow • Configurable Watermarks with threshold exceed notification. • Traffic Mirroring
Performances	• MTU size - Up to Jumbo Frames (9000B) • Throughput - 200 Gbps • Flow Insertion Rate – up to 450K flows/second • Latency - Less than 5 microseconds

Test Campaign at VHT Laboratories

Introduction

HCLSoftware together with VHT agreed to run performance tests in VHT Labs; the purpose of the tests is to measure throughput based on VHT specific test cases. These test cases use traffic dataset built up from Viettel Commercial network captures where packet size distributions have been identified for both Uplink and Downlink directions. Based on the datasets simple machine-to-machine performance measurements for UPF are run with different traffic profiles and related results are collected and presented here. VHT evaluates the feasibility (in terms of both techniques/cost) towards developing UPF on this new hardware platform with the lowest cost/resources.

Test Description

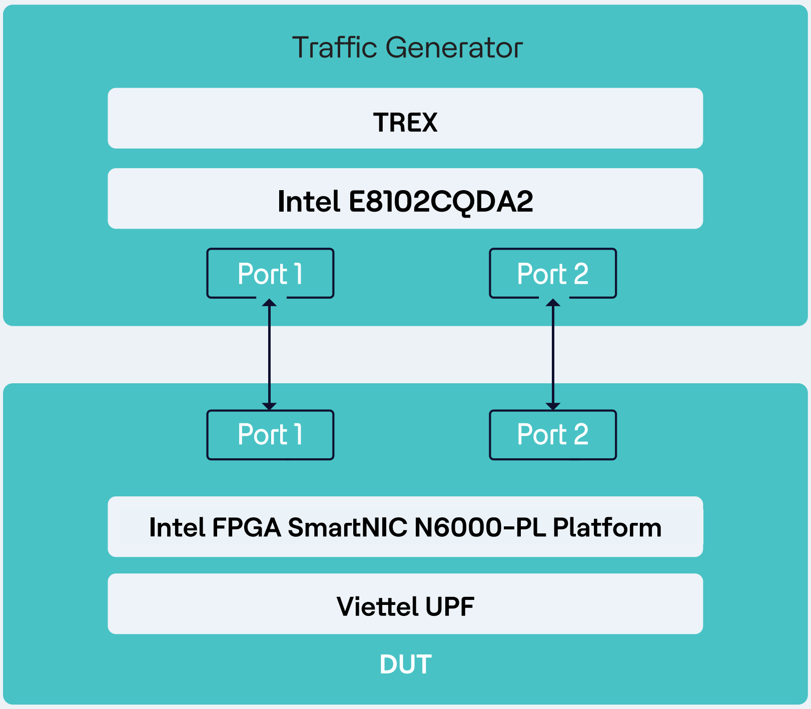
Figure 8 depicts the setup used to measure data plane performances in different conditions. The Device Under Test (DUT) system is running VHT DPDK-UPF interconnected with traffic generators through optical cables connecting QSFP-28 modules running on 2x100Gbps ports. VHT UPF uses a single logical port and traffic is identified by N3, N6 and N9 using different VLANs.

Preliminary phase of the test procedure is aiming at populating flow within the hardware. Uplink and Downlink traffic is generated by Cisco TRex Traffic Generator. As the hardware is started with empty flow table, all incoming traffic generates lookup misses. The packets are uplifted to the VHT UPF stack, which does software lookup and installs related flow in hardware for subsequent packets. At the end of this preliminary phase, the entire flow table will be populated.

Please note that VHT setup requires active standby port configuration which limit overall traffic to 100 Gbps as standby port is not carrying any traffic. Without loss of generality and simplify the deployment, the test is run on both ports with overall throughput of 100Gbps traffic sent by the Traffic Generator. To be further noted that HCL UPF Offload Solution can support up to 200Gbps throughput for typical 5G scenarios.

VHT profiles requires around 8 million Flows in hardware while HCL UPF Offload Solution can support up to 16 million flows in hardware.

Characteristics of traffic generation are reported in Table 5



Characteristics of traffic generation are reported in Table 5.

Category	Description	
Server	Vendor	Dell
	Model	PowerEdge R740
Processor	Product	Intel® Xeon® Gold 6138
	Frequency	2.00 GHz
	Number of Cores	20
Memory	Capacity	32GB
	Memory Speed	2666 MT/s, DDR4
SmartNIC	Product Name	Intel FPGA SmartNIC N6000 Platform (with E810-CAM2 Ethernet Controller)
	Number of Ports	2
	Port Type	QSFP-28/56@2x100Gbps
	HCL FW version	0x501060201020327
Host OS	Vendor/Version	CentOS Linuxrelease 8.5.2111
Kernel	Version	4.18.0-348.7.1.el8_5
DPDK	Version	20.11.3

Table 3. Device under Test Configuration

Category	Description	
Server	Vendor	Dell
	Model	PowerEdge R740
Processor	Product	Intel® Xeon® Gold 6138
	Frequency	2.00 GHz
	Number of Cores	20
Host OS	Vendor/Version	CentOS Linuxrelease 8.5.2111
NIC	Vendor	Intel
	Model	2
	Driver version	1.7.16-1
	Firmware version	2.40 MAP2.69
Kernel	Version	4.18.0-348.7.1.el8_5
DPDK	Version	20.11.3

Table 4. Traffic Generator Configuration

Category	Description
Subscribers	131,072 mobile subscribers in connected state, 2 bearers per mobile subscriber
Flows	8,388,608 total flows, 32 flows per bearer. HCL Solution can support up to 16M flows in hardware
N3 Packet Format	Eth/VLAN/IPv4/UDP/GTP-U/IPv4/UDP/Payload
N6 Packet Format	Eth/VLAN/IPv4/UDP/Payload
Interframe Gap	24 Bytes

Table 5. Test Traffic Specifications

Traffic Profiles

The performance of the system depends on packet sizes and their variability. VHT identified traffic profiles reported here are aiming at evaluating performance of the overall UPF system under similar conditions as the ones UPF will experience when receiving realistic mobile traffic data in the field.

The traffic dataset used in this analysis originated from one of Viettel commercial systems and measured the packet size distributions for both UL and DL traffic. The UL: DL ratio for bytes was 1:11, while packets were 1:1. In VHT Lab testing the distribution of packet sizes was categorized from 64 bytes to 1500 bytes.

Packet size variations impact UPF throughput. Measuring performance across different packet sizes helps identify the optimal size that maximizes throughput. Real-world network traffic consists of packets with diverse sizes. Measuring UPF performance with different packet sizes provides insights into how the UPF handles various traffic types, identifies potential bottlenecks specific to certain sizes, and ensures efficient handling of a wide range of traffic scenarios.

Packet size affects Quality of Service (QoS) parameters such as latency and jitter. Measuring UPF performance with different packet sizes helps understand the impact of size on QoS metrics. This knowledge ensures that the UPF meets performance requirements and delivers a satisfactory user experience across different traffic conditions.

Testing with different uplink (UL) to downlink (DL) packet ratios enables the simulation of realistic network traffic conditions. In real networks, UL and DL traffic volumes can vary, and different applications may generate diverse UL/DL ratios. By conducting tests with varying ratios ensures that the UPF can effectively handle the traffic mix encountered in practical scenarios. Efficient distribution of workload between UL and DL traffic is crucial for optimizing resource utilization. Evaluating the UPF's load balancing capabilities through tests with different UL/DL packet ratios allows for analysis of how the UPF manages and distributes traffic across the network, ensuring optimal resource allocation and congestion prevention.

In most network communication scenarios, users tend to consume more data in the DL direction (e.g., downloading files, streaming media, browsing websites) compared to their contributions in the UL direction (e.g., sending requests, uploading smaller amounts of data). This consumption pattern creates a greater demand for larger packet sizes in the DL direction. Certain protocols like TCP utilize techniques such as selective acknowledgments and windowing that work optimally with larger DL packet sizes, facilitating efficient data transfer and congestion control.

The selection of UL/DL packet sizes depends on factors like network architecture, protocol stack, and application requirements. VHT determines the UL/DL packet size ratio based on their actual traffic patterns and specific needs.

The following profiles were chosen for test purposes:

Category	Description
Profile 1	Profile 125% UL Traffic, 75% DL Traffic. Packet Size (Bytes): 128, 256, 512, 650, 1024, 1528.
Profile 2	50% UL Traffic, 50% DL Traffic. Packet Size (Bytes): 128, 256, 512, 650, 1024, 1528.
Profile 3	75% UL Traffic, 25% DL Traffic. Packet Size (Bytes): 128, 256, 512, 650, 1024, 1528.
Profile 4	25% UL Traffic, 75% DL Traffic. Packet Size (Bytes) in UL: 128, 256, 512, 576, 750 Packet Size (Bytes) in DL: 1408, 2816, 5632,6336, 8250
Profile 5	50% UL Traffic, 50% DL Traffic. Packet Size (Bytes) in UL: 128, 256, 512, 576, 750 Packet Size (Bytes) in DL: 1408, 2816, 5632,6336, 8250
Profile 6	75% UL Traffic, 25% DL Traffic. Packet Size (Bytes) in UL: 128, 256, 512, 576, 750 Packet Size (Bytes) in DL: 1408, 2816, 5632,6336, 8250

Table 6. Traffic Profiles

Performance Test Results

Results are presented in graphs below (Figure 9) for the different profiles highlighted in the previous section. Results are reported both in Millions of Packets Per Second (MPPS) and Gigabit per second (Gbps). Zero packet loss is the target for all performance measurements. Also, as reported previously, maximum achievable throughput is 100 Gbps in compliance with VHT requirements on redundancy.

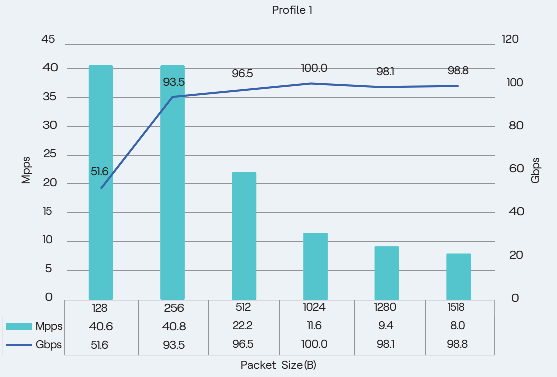
Depending on the percentage of the type of traffic transmitted by the traffic generator, whether it is from N3 (GTP-U encapsulation) or N6 (pure IP), UPF will either encapsulate or decapsulate the traffic depending on the direction, thus influencing the overall throughput in Gbps while throughput in MPPS is the same.

For Profiles 1,2 and 3, we can see from the graphs that the solution can support maximum achievable throughput when packet size is above 256 Bytes; for 128 Bytes and 256 Bytes, the results are consistent with maximum performance achievable in hardware. Differences in Gbps between the profiles are due to the different percentages of traffic between N3 and N6 based interfaces.

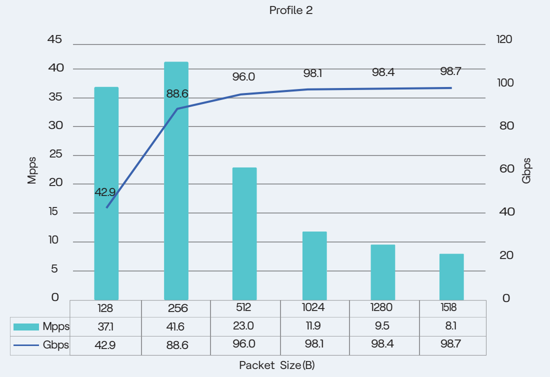
For VHT specific profiles 4,5 and 6, where a ratio of 1:11 is used between uplink and downlink packet sizes respectively. In the graphs the different packet sizes are reported as index from 1 to 5 as they are reported in Table 6.

As Profile 4 is mainly composed of Downlink traffic, maximum achievable throughput is always met for all packet sizes as the packets per seconds required to achieve that throughput is small compared to the capabilities of the accelerator in hardware.

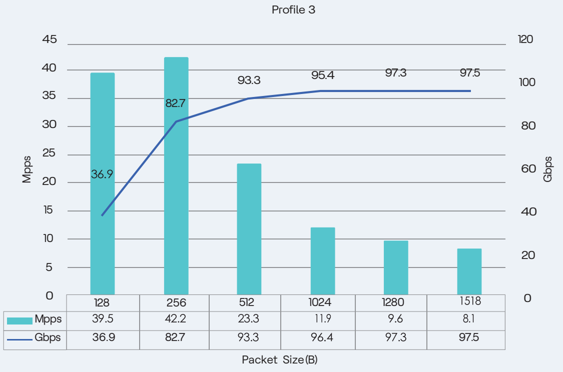
For smaller packet sizes, as the number of packets increases (as in profiles 5 and 6) the overall performance decreases; however, maximum achievable throughput is still met when packet size is above 256 Bytes.



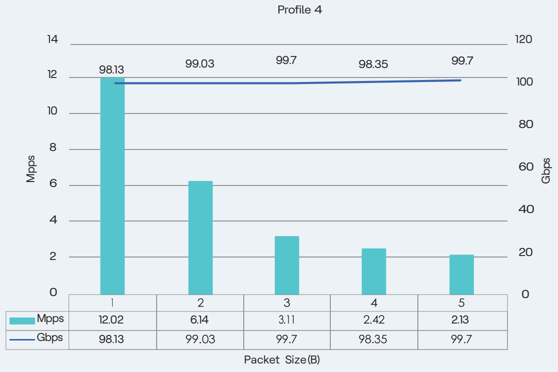
	128	256	512	1024	1280	1518
Mpps	40.6	40.8	22.2	11.6	9.4	8.0
Gbps	51.6	93.5	96.5	100.0	98.1	98.8



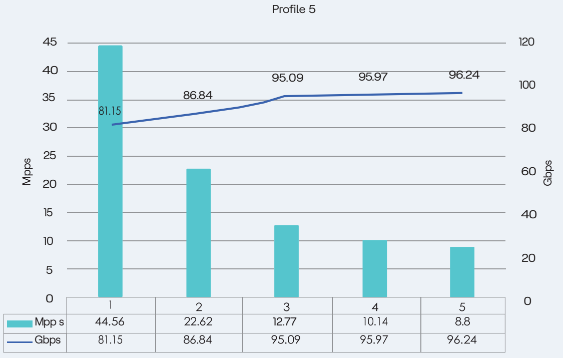
	128	256	512	1024	1280	1518
Mpps	37.1	41.6	23.0	11.9	9.5	8.1
Gbps	42.9	88.6	96.0	98.1	98.4	98.7



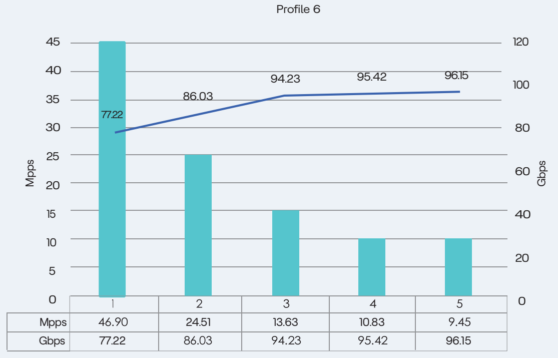
	128	256	512	1024	1280	1518
Mpps	39.5	42.2	23.3	11.9	9.6	8.1
Gbps	36.9	82.7	93.3	96.4	97.3	97.5



	1	2	3	4	5
Mpps	12.02	6.14	3.11	2.42	2.13
Gbps	98.13	99.03	99.7	98.35	99.7



	1	2	3	4	5
Mpp s	44.56	22.62	12.77	10.14	8.8
Gbps	81.15	86.84	95.09	95.97	96.24



	1	2	3	4	5
Mpps	46.90	24.51	13.63	10.83	9.45
Gbps	77.22	86.03	94.23	95.42	96.15

Conclusion

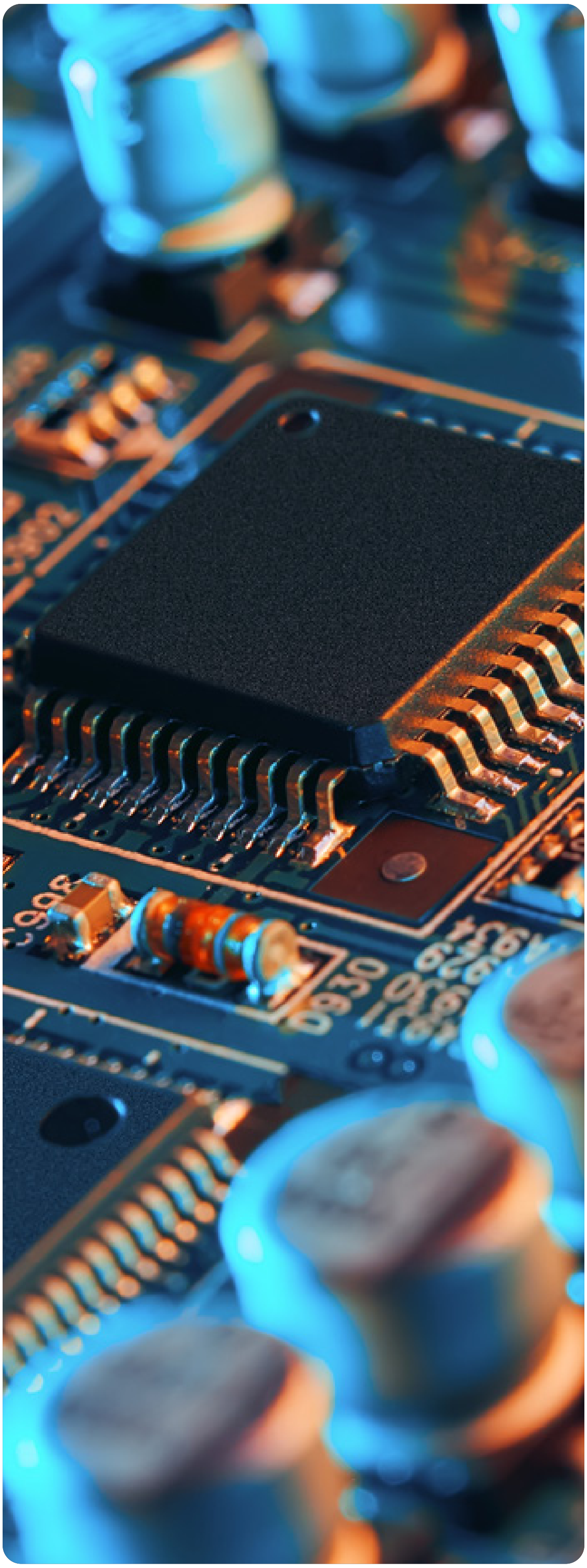
SmartNIC technology is an effective way to improve server performances in cloud while optimizing overall TCO. FPGA option allows customer to customize their application and repurpose them increasing overall system lifecycle and protecting investments. Moreover, when moving to IPU technology, complete disaggregation of UPF versus business applications can be achieved which can further improve isolation, multitenancy, and security of the overall network solution.

HCL 5G Acceleration solution on Intel FPGA SmartNIC N6000-PL Platform dramatically improves CPU utilization by using only two cores for data plane in all use case scenarios. By offloading tasks to the SmartNIC, CPU resources are freed up and can be allocated to process other tasks or business applications thus improving flexibility and scalability at the same time.

Integration of UPF stacks with HCLSoftware Solution is made easy using standard DPDK API which abstracts all hardware details from upper application layers.

Regarding the roles of the parties in the solution, Intel provides dedicated cards; HCL provides DPDK APIs & firmware for dedicated cards; Viettel High Tech provides UPF stack (software) to integrate with the card.

With exclusive partnership between Intel and HCL, we will continue to develop UPF Acceleration options on new Intel SmartNIC and IPU cards, continuing to provide different options to operators and UPF vendors based on their flexibility, scalability, and disaggregation needs.



For more information about performance and benchmark results, visit www.intel.com/benchmarks.

Test measure performance of components on a particular test, in specific systems. Differences in hardware, software, or configuration will affect performance. Consult other sources of information to evaluate performance as you consider your purchase. Intel technologies may require enabled hardware, software, or service activation. No product or component can be absolutely secure. Your costs and results may vary.

Authors

Salvatore Morsa

Solution Architect
HCLSoftware

Santosh Sethupathi

Sr. Technical Architect
HCLSoftware

Anjani Kumar Rai

Technical Architect
HCLSoftware

Quyen Vu Dinh

Technical Architect
HCLSoftware

Walukiewicz, Miroslaw

Sr. Solution Architect
Intel corporation

Vu Tuan Duc

Mgr. Software Architect
Viettel High Technology

Dinh Viet Quan

Mgr. Software Architect
Viettel High Technology

Le Ngoc Toan

Sr. System Architect
Viettel High Technology

Nguyen The Hieu

Solution Architect
Viettel High Technology

Nguyen Phu Lam

Technical Architect
Viettel High Technology

Vo Thanh Van

Technical Architect
Viettel High Technology