|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Change Request** | | | | | | | |
| **Document** | **ORAN-WG6.CAD** | **ver** | **01.00.00** | **CR** | **NVD-01** | **rev** | **0** | |

|  |  |  |  |
| --- | --- | --- | --- |
| ***Title:*** | Add inline acceleation as hardware acceleation option for O-DU PHY, in Section 5.3 | | |
| ***Source to WG:*** | Nvidia | | |
| ***Target WG :*** | **WG6** | | |
| ***Category:*** | **C** | ***CR Creation Date*** | February 10, 2020 |
|  | *Use one of the following* ***categories****:* ***A*** *(mirror corresponding to a change in an earlier release)* ***B*** *(addition of feature),* ***C*** *(functional modification of feature)* ***D*** *(editorial modification)* ***F*** *(correction)*  Detailed explanations of the above categories can be found in 3GPP [TR 21.900](http://www.3gpp.org/ftp/Specs/html-info/21900.htm). | | |

|  |  |
| --- | --- |
| ***Reason for Change:*** | Add inline acceleration to Hardware Acceleration Options in section 5.3. |
| ***Summary of change:*** | New text is proposed and is provided in the marked-up text below this table. |
| ***Consequences if not aproved:*** | There is no content for inline acceleration and it is not clear as to what are the available harware acceleration options for O-DU PHY without this CR |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ***Clauses affected:*** | Section 5.3 | | | | |
|  | **Y** | **N** |  | |  |
| ***Other specs*** |  | **X** | Other core specifications: |  | |
| ***affected:*** |  | **x** | Test specifications: |  | |
| ***(show related CRs)*** |  | **X** | O&M Specifications: |  | |
| ***Supporting material:***  ***Other comments:*** | <provide file name or URL of any material supporting this CR> | | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| ***Status:*** |  | ***CR Closed Date:*** |  |
| ***Outcome:*** |  | ***Duplication:*** |  |
| ***Outcome explanation:*** |  | | |



## Hardware Acceleration Options

Cloud platforms consist of GPP CPUs, Memory, Networking I/O, and may also provide HW accelerators to offload computational-intense functions with the aim of optimizing the performance of the VNF (e.g., O-DU, O-CU-CP, O-CU-UP, RIC). There are many different types of HW accelerators: FPGA, ASIC, GPU and many different types of acceleration architectures and associated acceleration functions, such as lookaside acceleration (e.g. Low-Density Parity-Check (LDPC) Forward Error Correction (FEC) as acceleration function) and inline acceleration (e.g. end-to-end high PHY algorithms as acceleration functions) for O-DU, Wireless Cipher for O-CU, and Artificial Intelligence for RIC.

Figure 10 illustrates an example of different combinations of HW accelerator and accelerating functions for O-DU high PHY, with lookaside and inline architectures. In the example of lookaside approach, a subset of PHY layer functions are selectively accelerated, whereas in the example of inline approach, the entire PHY pipeline with all the functions in uplink and downlink high-PHY pipeline are accelerated using hardware accelerators like GPU, FPGA, or ASIC.



Figure 10: Hardware acceleration options for O-DU PHY

The combination of HW accelerator and acceleration function(s), and indeed the option to use HW acceleration, is the vendor’s choice; however, all types of HW acceleration on the cloud platform should ensure the decoupling of SW from HW. The decoupling of HW and SW implies the following key objectives:

* Multiple vendors of hardware GPP CPUs and accelerators (e.g., FGPA, DSP, or GPU) can support cloud platforms (including agreed-upon abstraction layers) from multiple vendors, which in turn can support the software providing RAN functionality.
* A given hardware and cloud platform shall support RAN software (including RIC, O-CU-CP, O-CU-UP, O-DU, and possibly O-RU functionality in the future) from multiple vendors.

### HW Acceleration Abstraction

There are different methods of abstraction that should be considered for HW acceleration on the cloud platform; these are:

* HW Accelerator Deployment model
* HW Accelerator Application APIs

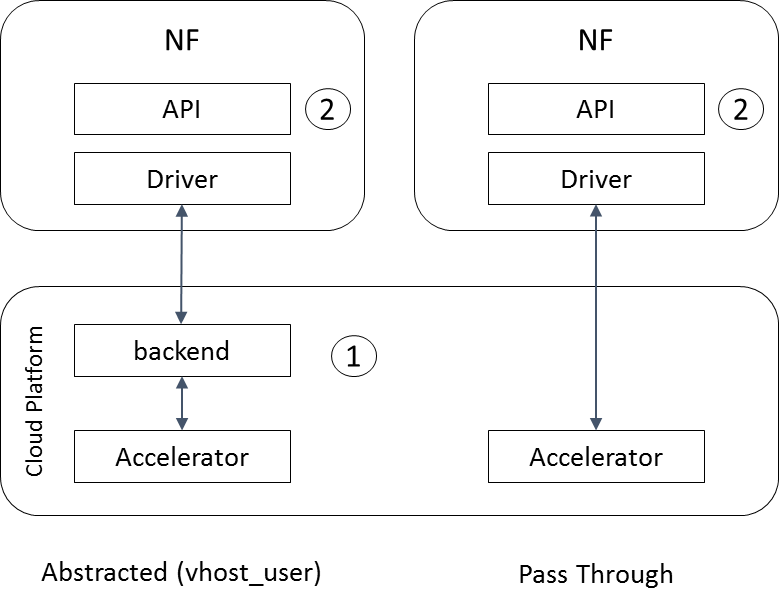


Figure : HW Abstraction Considerations

#### HW Accelerator Deployment Model

Figure 11 above presents two common HW deployment models, an abstracted implementation utilizing a vhost\_user and virtIO type deployment, and a pass-through model using SR-IOV. While the abstracted model allows a full decoupling of the Network Function (NF) from the HW accelerator, this model may not suit real-time latency sensitive NFs such as the O-DU. For low-latency HW acceleration, SR-IOV pass through may be required. The SR-IOV pass through model is also supported in container environments.

#### HW Accelerator Application APIs

To allow multiple NF vendors to utilize the same HW accelerator on the cloud platform, HW Accelerators must provide an open-sourced API. The API shall allow the NF to discover the HW capabilities assigned to it, and submit and retrieve acceleration requests/responses. Examples of open APIs include DPDK’s CryptoDev, EthDev, EventDev, and Base Band Device (BBDEV).

### HW Accelerator Management and Orchestration Considerations

The HW accelerators shall be capable of being managed and orchestrated. In particular, HW accelerators shall support feature discovery and life cycle management. Existing Open Source solutions may be leveraged for both VMs and containers as specified in O1\*. Examples include OpenStack Nova and Cyborg. An example for container deployments is seen in Kubernetes which provides a device plugin framework for vendors to advertise their device and associated resources to the Kubelet for management.