在DPDK中经常看到一个词，exception path，它是什么？

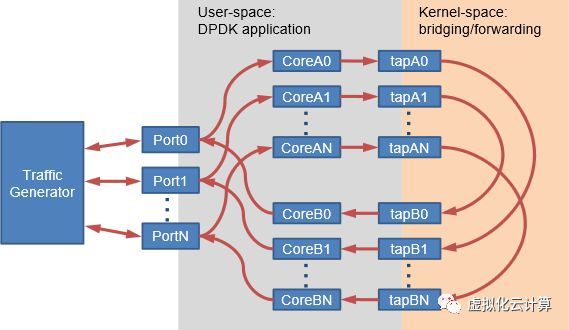
在DPDK使用环境中，物理网卡收到的报文不再进入内核协议栈，而是直接到达DPDK应用。但是在有些场景中，用户希望把报文（如控制报文）再次发送至内核协议栈进行处理。报文从用户空间再次进入内核的过程在DPDK中称为exception path。

DPDK支持几种方式让用户空间的报文重新进入内核协议栈：

**1.TAP/TUN设备**

可以使用内核提供的TAP/TUN设备，这种设备的使用需要使用系统调用，并涉及到copy\_to\_user()和copy\_from\_user()的开销。

使用virtual TAP设备实现exception path的流向图：

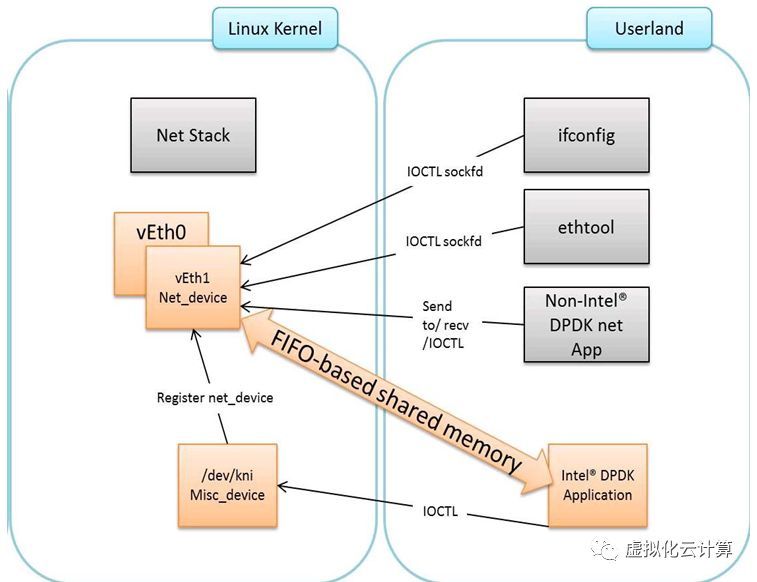


**2.DPDK Kernel NIC Interface(KNI)**

DPDK提供了KNI接口用于提高用户态和内核态之间报文的处理效率。KNI是通过内核模块构造了一个虚拟网络接口，并且通过FIFO和用户态的DPDK应用交换报文。

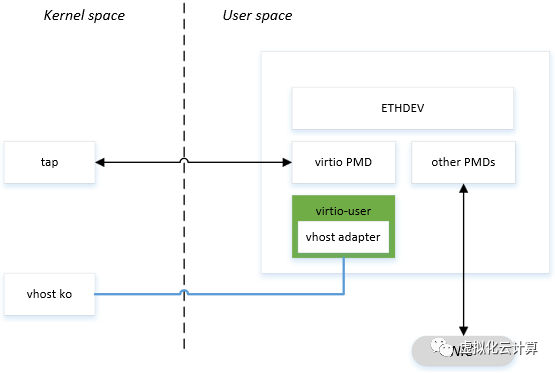
正如DPDK官方所讲，使用DPDK KNI的好处是：

* 比现有的Linux TUN / TAP（通过消除系统调用和copy\_to\_user（）/ copy\_from\_user（）操作）。
* 允许使用标准Linux网络工具（如ethtool，ifconfig和tcpdump）管理DPDK端口。
* 允许与内核网络协议栈的交互。



**3.virtio-user+vhost-net**

DPDK提供了virtio-user虚拟设备，使用它与现有的vhost-kernel方案配合来实现exception path。需要内核中vhost.ko和vhost-net.ko两个模块。



启动virtio-user，会操作/dev/vhost-net创建一个内核态的虚拟设备tap。从DPDK收到的包进入到virtio-user，通过vhost-kthread进入到tap设备，tap设备支持内核协议栈。从而实现了用户态和内核态的数据交互。