

DPDK Summit Userspace 2017
September 26 - 27, 2017

Day 1

Theme	Time	Title	Presenter	Description
	8:30 - 9:00	Registration		
Intro	9:00 - 9:15	Introduction	Tim O'Driscoll	Introduction, and review of the agenda and expectations for the event.
Community Updates	9:15 - 10:30	Tech Board Presentation & Panel Discussion	Tech Board	Short presentation from a Tech Board rep on: Who the Tech Board members are, what their responsibilities are, recent issues that they've addressed, and future technical priorities/challenges. Panel discussion with the attending Tech Board members. Topics to discuss include: - Generic vs specific APIs - Requirements for upstreaming a new API - CLI changes - Ethdev API offloads (how to make these consistent) - Release planning/milestones (are the intervals from proposal -> integration & integration -> release the right duration) - Questions from audience
	10:30 - 10:50	Break		
Community Updates	10:50 - 11:30	Governing Board Presentation & Panel Discussion	Governing Board	Short presentation from a Governing Board rep on: Who the Governing Board members are, what their responsibilities are, progress to date, and future priorities/challenges. Panel discussion with the attending Governing Board members. Topics to discuss include: - Project budget - Plan for 2018 events - DPDK Lab - Questions from audience
Bus & Device Mgmt	11:30 - 12:00	DPDK Bus Updates	Ferruh Yigit	DPDK bus infrastructure has been updated for the last few releases. Although these changes should not affect the user application, it's worth mentioning the changes. In this talk, I will summarize the bus changes and mentions from required modifications in drivers.
Bus & Device Mgmt	12:00 - 12:15	Ideas for adding generic HW accelerators to DPDK.	Hemant Agrawal	There are various kind of HW accelerators available with SoCs. Each of the accelerators may support different capabilities and interfaces. Many of these accelerators are programmable devices. In this talk we will discuss various ways to support such accelerators in a generic manner.
	12:15 - 1:15	Lunch		
Bus & Device Mgmt	1:15 - 1:45	FPGA bus framework and programmable device model	Jing Chen	Currently DPDK manages the fixed function devices with the prior programmed drivers, but recent NIC, FPGA and GPU make hardware programmable and make it possible to allow applications to change hardware's capabilities in running time. This talk will introduce an FPGA bus model on how to identify RTL logic and find matched drivers. It will also offer a proposal to create a new framework that allows applications to change hardware capabilities and switch to new drivers automatically on the fly.
Bus & Device Mgmt	1:45 - 2:15	Let's hot plug: use uevent mechanism to practice it in DPDK	Jia Guo	Hot plug is benefit for device online update in data centers, and meaningful for live migration development, so far, with hot plug API and fail safe in DPDK, it still not friendly use for PCIe devices. This talk propose to add a general uevent mechanism in dpdk framework, which include the uevent monitor and failure handler, to make dpdk user easy to achieve hot plug. This would be expand to other linux bus devices, so some compatibility related with recently bus rework need to be considered.
Bus & Device Mgmt	2:15 - 2:30	Proposed method for sharing a (PCI) device between multiple PMDs	Fiona Trahe	Devices on the PCI bus are found by the bus probe function. For each device, the list of registered drivers (PMDs) is searched until one (only) is found for the device. Hardware accelerators can provide multiple functions via the same PCI device, e.g. symmetric and asymmetric crypto functions. It may be desirable to present these to applications using separate PMDs. This presentation proposes a mechanism to share a pci device between multiple PMDs. It may also be extendable to non-pci devices.
Bus & Device Mgmt	2:30 - 3:00	DPDK VMBus & Azure Support	Stephen Hemminger	This is a talk about the current status and planned development of VMBus support for DPDK. There have been many requests (and several partial efforts) to support Hyper-V networking in native mode in DPDK. This talk also gives an overview of how DPDK applications are enabled on Azure Accelerated Networking using the Fail-Safe, TAP and existing drivers. It will cover some of the requirements and plans for the future. DPDK support on Azure is a joint project of Mellanox, 6Wind and Microsoft.
	3:00 - 3:20	Break		
APIs	3:20 - 4:00	rte_security: the case of IPsec offload	Boris Pismenny Declan Doherty Hemant Agrawal	Encryption in today's networks is becoming ubiquitous. However, running crypto on general purpose CPUs is costly. Moreover, the encrypted packets cannot be processed by NICs, preventing NIC offloads. In this talk we present a joint work of NXP, Intel and Mellanox on offloading protocol processing to hardware providing better utilization of host CPU for packet processing. This new feature will allow DPDK users to reach even higher packet processing rates with IPsec and other security protocols.
APIs	4:00 - 4:30	DPDK Quality of Service APIs	Cristian Dumitrescu Jasvinder Singh	This presentation focuses on the new QoS Traffic Management API for Ethernet devices that was introduced by DPDK release 17.08, as well as the new QoS Traffic Metering and Policing API planned for DPDK release 17.11. We describe the API, device drives currently supporting it and software fall-back strategy using the SoftNIC PMD.
APIs	4:30 - 5:00	Service Cores: The path to Abstracting SW/HW CPU Requirements in DPDK	Harry van Haaren	Service cores is a library since 17.08-rc2 that abstracts the platform, providing an app with a consistent environment. Service cores allows switching of SW and HW PMDs with no application threading changes. This talk introduces service-cores, and opens discussion on how to enable DPDK with service cores. Libraries and other tasks that could benefit: SW PMDs, timers, keepalive, metrics, stats, and monitoring. Discussing at Userspace would ensure community alignment on future service cores work.
APIs	5:00 - 5:20	Wireless Base Band Device (bbdev)	Amr Mokhtar	Wireless Base Band Device (bbdev) proposal for DPDK that abstracts HW accelerators based on FPGA and/or Fixed Function Accelerators that assist with LTE Physical Layer processing. Furthermore, it decouples the application from the compute-intensive wireless functions by abstracting their optimized libraries to appear as virtual bbdev devices.
APIs	5:20 - 5:40	DPDK to support InfiniBand Link Layer	Shahaf Shuler	There are many large InfiniBand clusters in the HPC market, they too would like to gain the DPDK user space high packet rate processing advantage, in addition to the RDMA capabilities. InfiniBand is based on a different addressing scheme. The standard IPv4/6 InfiniBand communication protocol is defined by IPoIB (rfc4391). I will present the basic InfiniBand and IPoIB differences from Ethernet and present results from a live POC on a 20 node cluster enabling a storage solution with DPDK using IPoIB
	5:45 - 6:00	Close		
	6:00 - 8:00	Evening Reception		

DPDK Summit Userspace 2017
September 26 - 27, 2017

Day 2

Theme	Time	Title	Presenter	Description
	8:30 - 9:00	Registration		
CI & Process	9:00 - 9:45	Community Activity, CI & Tools	Thomas Monjalon Qian Xu	The userspace summit is a good place to make a yearly summary of community changes and interactions. First we should celebrate the participation of individuals and companies. It is also important to describe how DPDK interacts with other communities (hardware support, OS support, kernel dependencies, Qemu / OVS / fd.io interactions, etc) The last part would be about community processes (repositories, distributed CI, bugs tracking, tooling, website, mailing lists and Linux Foundation).
CI & Process	9:45 - 10:15	ABI Stability and LTS: Current state and Future	John McNamara Ian Stokes Luca Boccassi Kevin Traynor	The format will be a short 5 minute summary/viewpoint from each of the participants below. It will be followed by a panel discussion of the future direction of ABI stability & LTS/Stable releases. In particular it will look at the request for a yearly xx.11 LTS release with a 2 year duration. * John McNamara (Intel, DPDK); * Ian Stokes (Intel, OVS-DPDK); * Luca Boccassi (Brocade, Debian); * Red Hat - Kevin Traynor (OVS-DPDK)
Control Plane	10:15 - 10:30	Implementing an SR-IOV Hypervisor using DPDK.	Alex Zelezniak	In the presentation we will describe VFd, a hypervisor for SRIOV NICs jointly developed by AT&T and Intel, which uses DPDK and acts as policy enforcement software allowing advanced configuration of SR-IOV capable Network Interfaces. We will provide overview of the use cases and new DPDK API's to support them.
	10:30 - 10:50	Break		
Control Plane	10:50 - 11:15	DPDK SRIOV and control over embedded switch	Alex Rosenbaum	When working in SRIOV mode, we would prefer to let majority of the traffic to pass in HW directly from/to wire to/from VF, while the OVS-DPDK application only needs to handle exception packet flows on the PF. To support this mode we want to show a new Representor Ports model of the HW switch, which can be controlled from the DPDK framework. The control can be via PMD directly to HW NIC, or via netlink into relevant Kernel netdevice.
Control Plane	11:15 - 11:45	A framework for representation, configuration, and management of virtual function ports in DPDK	Declan Doherty	This presentation introduces a port representor framework to DPDK. The framework based around a virtual representor PMD and representor broker plugin for physical function devices, provides the infrastructure to allow SR-IOV virtual function ports to be configured, managed and monitored within a single control application as standard DPDK ethdevs using existing DPDK APIs.
VNFs	11:45 - 12:15	Improve VNF safety with Vhost-User/DPDK IOMMU support	Maxime Coquelin	This talk will cover the challenge of improving VNFs safety relying on Virtio and Vhost-user backend on host side. Maxime will first provide a brief overview of a VNF architecture relying on Virtio/Vhost-user, to take the opportunity to highlight the possible safety concerns. Then, he will talk about new developments that introduces IOMMU support to Vhost-user backend. Finally, Maxime will provide benchmarks results and discuss ways to improve both performance & safety.
	12:15 - 1:15	Lunch		
VNFs	1:15 - 1:45	Virtio Status Update and New Ring Layout	Tiwei Bie Zhihong Wang	The packed ring layout is the next generation ring layout standard for Virtio, which is designed for high performance and still in the proposal stage. We have done some prototypes in DPDK Virtio and Vhost to evaluate this new ring layout. This talk will give a quick introduction to this new ring layout definitions and summary the current status, findings, and benchmark results of the prototype in DPDK.
CPU Improvements	1:45 - 2:15	DPDK and portability: how DPDK dataplane perform on non-x86 architectures	Gergely Pongracz	In this presentation we show an optimized DPDK datapath implementing several network functions. We ran the functions on different hardware environments where we had DPDK support. The x86 environments were: Intel Atom and Intel Xeon, while the non-x86 environment was an ARM64 based state of the art Cavium CN-83xx blade with 24 ARM cores and low power consumption. For all NFs we have a performance model telling the upper bounds. The results show a clear difference between architectures.
CPU Improvements	2:15 - 2:45	Power Aware Packet Processing	Chris MacNamara Dave Hunt	A drive to deliver OPEX saving and performance where and when it's needed. Enter a new era of power optimized packet processing. The cost of combining power saving & performance is considered via critical metrics, packet loss, latency and power savings. This talk reviews new & existing DPDK extensions for policy based power control proposed in August and the associated performance benefits.
	2:45 - 3:00	Break		
User Applications	3:00 - 3:15	DPDK, VPP and pfSense 3.0	Jim Thompson	pfSense is a open source firewall/vpn appliance, based on FreeBSD, started in 2006 with over 1M active installs. With increasing use of 10Gbps to 100Gbps Ethernet, fulfilling the increased packet rate requirements becomes ever more difficult for software routers and their inherent inefficiencies. We are basing pfSense release 3.0 on FD.io's VPP, leveraging key DPDK components including cryptodev, while adding a CLI and RESTCONF layer, leveraging FRRouting and Stronswan.
User Applications	3:15 - 3:30	Hurdles & Pitfalls to port open source IPS-IDS to DPDK	Vipin Varghese	Suricata is an open source intrusion detection and prevention system for network monitoring utility used on wide variety of use cases for network traffic and data centers alike. Porting to run with DPDK removes bottlenecks (300Mbits/sec) and make use of multicore making it to run on any DPDK platform. Information shared focuses on hurdles and pitfalls made during the effort of porting open source Suricata IPD-IPS to meet various deployment scenario.
User Applications	3:30 - 3:45	Rapid prototyping of DPDK applications with libmoon	Paul Emmerich	Building a new DPDK application from scratch is a daunting task, especially for beginners. We've taught a lot of students to use DPDK, both for final theses and in a lab course and encountered multiple usability problems. This talk is about our framework libmoon (https://github.com/libmoon/libmoon), a wrapper for DPDK that makes building DPDK prototypes simple and fast. We've used it for multiple research prototypes as well as our packet generator MoonGen (presented last year here).
User Applications	3:45 - 4:00	Using DPDK with Go	TAKANARI HAYAMA	In our presentation, we share the lesson learned from our experience using DPDK with Go in order to implement a software router Lagopus2 (https://github.com/lagopus/vsw). The previous version of Lagopus is implemented in full C, but in Lagopus2 we decided to implement control plane with Go for simplicity. C is still used where we need performance, e.g. data plane. We'll explain how we carefully designed DPDK binding in Go to guarantee the type safeness and the performance at the same time.
User Applications	4:00 - 4:15	T4P4S: When P4 meets DPDK	Sándor Laki	T4P4S is a P4 compiler supporting flexible re-targetability without sacrificing high performance packet processing. To achieve this goal, it is split into hardware dependent and independent components: 1) all the hw-related functionalities are defined by a low-level library (NetHAL) to be implemented for each target; 2) the core compiler translates P4 programs to hw-independent C code using NetHAL. This talk will show the architecture of T4P4S and the design decisions made to support DPDK.
User Applications	4:15 - 4:30	Dataplane Networking journey in Containers	Kuralamudhan Ramakrishnan Gary Loughnane	Containers with their lightweight footprint, resource overhead & auto-scaling are gaining momentum within the NFV infrastructure. Container networking is suffering with performance bottlenecks, low line rate throughput and high latency due to slow overlay networking. Our advance Container Network Interface combines the benefits of containers with DPDK's ultra-low latency and fast packet processing and the results show 28x more performance with SRIOV. DPDK using Vhost-User with OVS-DPDK and VPP.
	4:30 - 5:30	Overflow/late topics		
	5:30	Close		