

Time	Title	Presenter	Description
8:30 - 9:00	Registration		
9:00 - 9:15	Introduction	Siobhan Butler Tim O'Driscoll	
9:15 - 10:15	DPDK Roadmap	Bruce Richardson	This session will be an open discussion on the DPDK roadmap (http://dpdk.org/dev/roadmap). It will cover topics such as: - What do people plan to contribute to 17.02/17.05? - What gaps exist in the roadmap?
10:15 - 10:45	DPDK Hyper-V Support	Stephen Hemminger	This presentation will cover the strategy and implications of Hyper-V support in DPDK.
10:45 - 11:15	Break		
11:15 - 11:45	NFV Use-case Enablement on DPDK and FD.io	Jasvinder Singh Cristian Dumitrescu	This presentation will discuss the rapid development of NFV use cases such as a virtualised provider edge router (vPE) using the DPDK and VPP framework on Intel multicore CPUs. Additionally, this talk will focus on enabling DPDK Hierarchical Quality of Service Scheduler (HQoS) in VPP framework. This will include the configuration of HQoS instances and their execution in VPP.
11:15 - 11:45	TBD	Hemant Agrawal Shreyansh Jain	TBD
11:45 - 12:30	Mbuf Changes	Olivier Matz	This session will be an open discussion on mbuf changes that are required, including mbuf extension for external data.
12:30 - 1:30	Lunch		
1:30 - 2:30	Cryptodev and IPsec Acceleration	Declan Doherty Damjan Marion Sergio Gonzalez Monroy	This presentation will consist of 3 parts: - A comprehensive overview of the cryptodev framework in DPDK, including its architecture, poll mode drivers, crypto application development, and details on the future roadmap. - An overview of the Vector Packet Processing (VPP) project in FD.io. - A description of the integration of cryptodev into VPP to accelerate IPsec.
2:30 - 3:00	Transport Layer Development Kit (TLDK)	Keith Wiles	This presentation describes the Transport Layer Development Kit (TLDK) project in FD.io. It will include some performance measurements of TLDK.
3:00 - 3:30	mOS Networking Stack: A Specialized Network Programming Library for Stateful Middleboxes	Prof KyoungSoo Park	Stateful middleboxes such as intrusion detection systems and stateful firewalls rely on TCP flow management to keep track of on-going network connections. Implementing complex TCP state management modules for network appliances in high-speed networks is difficult. This is especially more challenging due to the lack of a reusable networking stack that provides a development interface that monitors fine-grained flow states for stateful middleboxes. This presentation describes the middleware Operating System (mOS) which aims to address this gap.
3:30 - 4:00	Break		
4:00 - 4:15	TRex Traffic Generator	Hanoch Haim	This presentation will give high level of the capability/design of the new stateless features in Trex and how it is used by the FD.io project.
4:15 - 5:45	DPDK Project Growth	Tim O'Driscoll	As DPDK continues to expand, we need to consider what changes need to be made in order to facilitate that growth. One obvious change would be to move the project to an independent body such as the Linux Foundation.
5:45	Close		
6:00 - 8:00	Evening Reception		

Time	Title	Presenter	Description
8:30 - 9:00	Registration		
9:00 - 9:30	DPDK optimal performance everywhere (Rallying with a Formula 1)	Thomas Monjalon	This presentation will be about the fundamentals of DPDK. First of all, the DPDK userspace drivers process packets at lightning speed. And it runs on several architectures with a long list of supported devices. But the real uniqueness of DPDK is to combine optimal performance with a broad hardware support and à la carte packaging. Surprisingly, there are always some new promising optimizations to unlock the full power of the hardware. Also, after 5 years of existence, the project is still making some progress to be easier to use in more and more environments. Finally it is essential to remember that these improvements are possible thanks to an amazing and fast growing Open Source community.
9:30 - 10:00	Identifying and Fixing Performance Bottlenecks	Bruce Richardson	This presentation will focus on identifying and fixing performance bottlenecks using the Intel® VTune™ Amplifier. An example from the i40e driver will be used to illustrate the process.
10:00 - 10:30	Re-structuring the DPDK Device Driver Framework	Hemant Agrawal Shreyansh Jain	Expanding DPDK to support non-PCI devices e.g. platform bus devices in a SoC
10:30 - 11:00	Break		
11:00 - 11:15	Mellanox Bifurcated DPDK PMD	Rony Efraim	Mellanox PMD is based on Bifurcated driver and allows the kernel (netdev) and more than one PMD to run on the same PCI. If the DPDK app is not setting a rule to steer this traffic it will be processed by the kernel.
11:15 - 11:30	Intel I40E Bifurcated Driver	Jingjing Wu	DPDK is known to build the high performing data plane workload on Intel Architecture and platform, a real world packet processing workload often relies on heavily on the Linux kernel and its large stack for the control plane design and implementation. As a known limit, Linux performance is not sufficient for high speed data plane workloads. In order to combine the advantages of both, a few key technical components can be used to make DPDK and kernel stack work and evolve independently. - A high speed data path between Linux kernel and DPDK. (Kernel NIC Interface, KNI, TAP). - A high speed data traffic direction into Linux Kernel and DPDK. (Bifurcated driver, Virtualization)
11:30 - 12:15	Interworking with the Linux Kernel	Ferruh Yigit	This session will be an open discussion on the challenges of interworking with the kernel, the inability to upstream enhancements like KDP/KCP to either the kernel or DPDK, and possible next steps.
12:15 - 12:30	Monitoring your DPDK Application/Interfaces with Collectd	Emma Foley Maryam Tahhan	This presentation describes how to leverage the DPDK xstats API to monitor your interfaces with collectd (a system statistics collection daemon).
12:30 - 1:30	Lunch		
1:30 - 2:00	deb_dpdk - Challenges and Opportunities when Packaging DPDK	Christian Ehrhardt Luca Boccassi	DPDK is special in many ways which make packaging and distributing it a more interesting effort compared to many other programs or libraries. This talk is about challenges we faced, what we can learn from them and where we expect potential areas of improvement in the future. Working together on those should help DPDK distribution and growing a higher adoption rate.
2:00 - 3:00	Usability (including packaging, stable releases, LTS releases etc.)	John McNamara Christian Ehrhardt Luca Boccassi	This session will be an open discussion on usability, including topics such as packaging, stable releases, LTS releases etc.
3:00 - 3:30	Testing and Continuous Integration	Qian Xu Yuanhan Liu	Presentation from Qian on DPDK CI Enhancements. This presentation will focus on testing, performance testing and CI, and how these can be further improved.
3:30 - 4:00	Break		
4:00 - 4:15	Open vSwitch DPDK Acceleration Using HW Classification	Rony Efraim	How to use HW classification to accelerate OpenvSwitch DPDK. NIC HW can classify the packets and return flow id that can be used instead of classification the packet by software. NIC HW can classify the packets drop and count them in order to perform aging and statistics.

4:15 - 4:30	An Implementation of a P4 Software Dataplane Using Open vSwitch	Cian Ferriter	P4 is an emerging standard for programming dataplanes. Although its initial applicability is for hardware dataplanes (in particular switching ASICs), its paradigm shifting approach to programming the dataplane can equally be applied to software dataplanes with interesting consequences. The first mainstream implementation of a software P4 dataplane is likely to be via Open vSwitch and is based on DPDK. This presentation will talk about this work and what it means to the DPDK community.
4:30 - 4:45	Bridging the gap between hardware functionality in DPDK applications and vendor neutrality in the open source community	Ian Stokes Sugesh Chandran	To date there has been multiple efforts to make use of hardware features and functionality in the fast-path of OVS to improve performance e.g. Intel XL710 VxLAN tunnel optimization using flow director feature, Intel XL710/82599 packet type identification etc. However implementations to date have been hardware specific and as such are not acceptable to the Open vSwitch community due vendor neutrality. Feedback from the Open vSwitch community indicated that what is required is a common hardware API that is vendor neutral and easily consumable.
4:45 - 5:00	Testing and Benchmarking Applications with MoonGen	Paul Emmerich	MoonGen is a fully scriptable high-speed packet generator built on DPDK and LuaJIT. It can saturate a 10 Gbit/s connection with 64 byte packets on a single CPU core while executing user-provided Lua scripts for each packet. Multi-core support allows for even higher rates, we have evaluated it at rates above 100 Gbit/s at 200 Mpps.
5:00	Close	Jim St. Leger	