



Networking Workloads on Intel Architecture

Communications, Storage and
Infrastructure Group
September 2014

Legal Disclaimer

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

A "Mission Critical Application" is any application in which failure of the Intel Product could result, directly or indirectly, in personal injury or death. SHOULD YOU PURCHASE OR USE INTEL'S PRODUCTS FOR ANY SUCH MISSION CRITICAL APPLICATION, YOU SHALL INDEMNIFY AND HOLD INTEL AND ITS SUBSIDIARIES, SUBCONTRACTORS AND AFFILIATES, AND THE DIRECTORS, OFFICERS, AND EMPLOYEES OF EACH, HARMLESS AGAINST ALL CLAIMS COSTS, DAMAGES, AND EXPENSES AND REASONABLE ATTORNEYS' FEES ARISING OUT OF, DIRECTLY OR INDIRECTLY, ANY CLAIM OF PRODUCT LIABILITY, PERSONAL INJURY, OR DEATH ARISING IN ANY WAY OUT OF SUCH MISSION CRITICAL APPLICATION, WHETHER OR NOT INTEL OR ITS SUBCONTRACTOR WAS NEGLIGENT IN THE DESIGN, MANUFACTURE, OR WARNING OF THE INTEL PRODUCT OR ANY OF ITS PARTS.

Intel may make changes to specifications and product descriptions at any time, without notice. Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined". Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The information here is subject to change without notice. Do not finalize a design with this information.

The products described in this document may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order.

Copies of documents which have an order number and are referenced in this document, or other Intel literature, may be obtained by calling 1-800-548-4725, or go to: <http://www.intel.com/design/literature.htm> Performance tests and ratings are measured using specific computer systems and/or components and reflect the approximate performance of Intel products as measured by those tests. Any difference in system hardware or software design or configuration may affect actual performance. Buyers should consult other sources of information to evaluate the performance of systems or components they are considering purchasing. For more information on performance tests and on the performance of Intel products, visit Intel Performance Benchmark Limitations

All products, computer systems, dates and figures specified are preliminary based on current expectations, and are subject to change without notice.

Celeron, Intel, Intel logo, Intel Core, Intel Inside, Intel Inside logo, Intel. Leap ahead., Intel. Leap ahead. logo, Intel NetBurst, Intel SpeedStep, Intel XScale, Itanium, Pentium, Pentium Inside, VTune, Xeon, and Xeon Inside are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Intel® Active Management Technology requires the platform to have an Intel® AMT-enabled chipset, network hardware and software, as well as connection with a power source and a corporate network connection. With regard to notebooks, Intel AMT may not be available or certain capabilities may be limited over a host OS-based VPN or when connecting wirelessly, on battery power, sleeping, hibernating or powered off. For more information, see <http://www.intel.com/technology/iamt>.

64-bit computing on Intel architecture requires a computer system with a processor, chipset, BIOS, operating system, device drivers and applications enabled for Intel® 64 architecture. Performance will vary depending on your hardware and software configurations. Consult with your system vendor for more information.

No computer system can provide absolute security under all conditions. Intel® Trusted Execution Technology is a security technology under development by Intel and requires for operation a computer system with Intel® Virtualization Technology, an Intel Trusted Execution Technology-enabled processor, chipset, BIOS, Authenticated Code Modules, and an Intel or other compatible measured virtual machine monitor. In addition, Intel Trusted Execution Technology requires the system to contain a TPMv1.2 as defined by the Trusted Computing Group and specific software for some uses. See <http://www.intel.com/technology/security/> for more information.

Hyper-Threading Technology (HT Technology) requires a computer system with an Intel® Pentium® 4 Processor supporting HT Technology and an HT Technology-enabled chipset, BIOS, and operating system. Performance will vary depending on the specific hardware and software you use. See www.intel.com/products/ht/hyperthreading_more.htm for more information including details on which processors support HT Technology.

Intel® Virtualization Technology requires a computer system with an enabled Intel® processor, BIOS, virtual machine monitor (VMM) and, for some uses, certain platform software enabled for it. Functionality, performance or other benefits will vary depending on hardware and software configurations and may require a BIOS update. Software applications may not be compatible with all operating systems. Please check with your application vendor.

* Other names and brands may be claimed as the property of others.

Other vendors are listed by Intel as a convenience to Intel's general customer base, but Intel does not make any representations or warranties whatsoever regarding quality, reliability, functionality, or compatibility of these devices. This list and/or these devices may be subject to change without notice.

Copyright © 2013, Intel Corporation. All rights reserved.

What does the Intel DPDK team do?

Implement new features/libraries to aid networking workloads

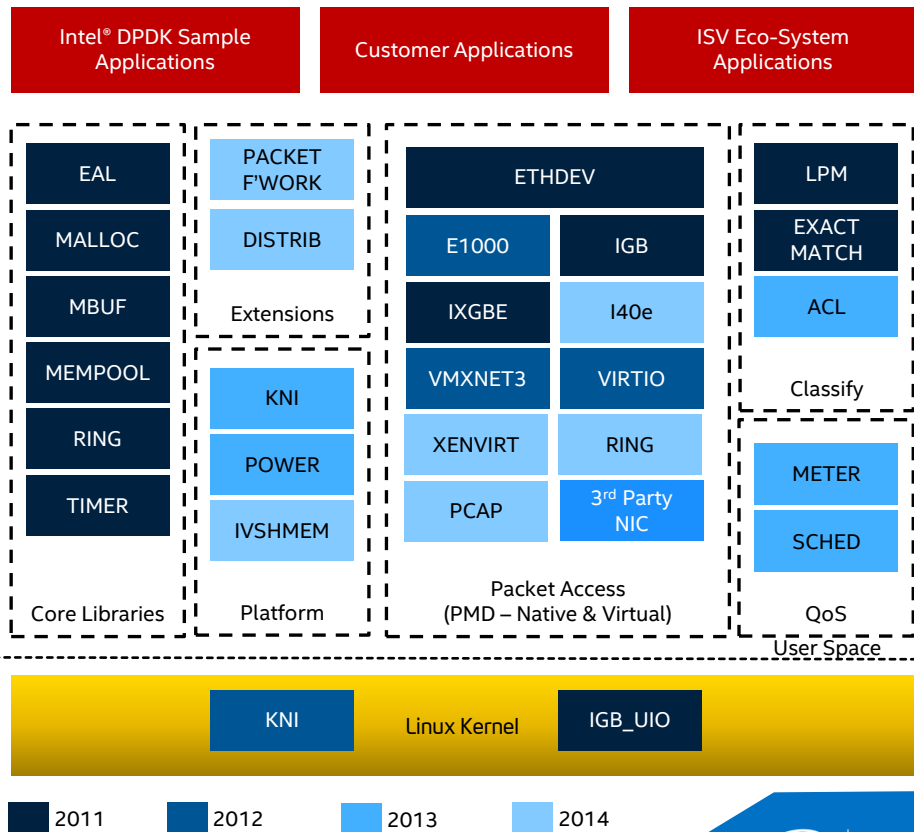
- Ongoing process as we discover bottlenecks/problems
- Customer feedback

Continuous performance improvement of existing libraries

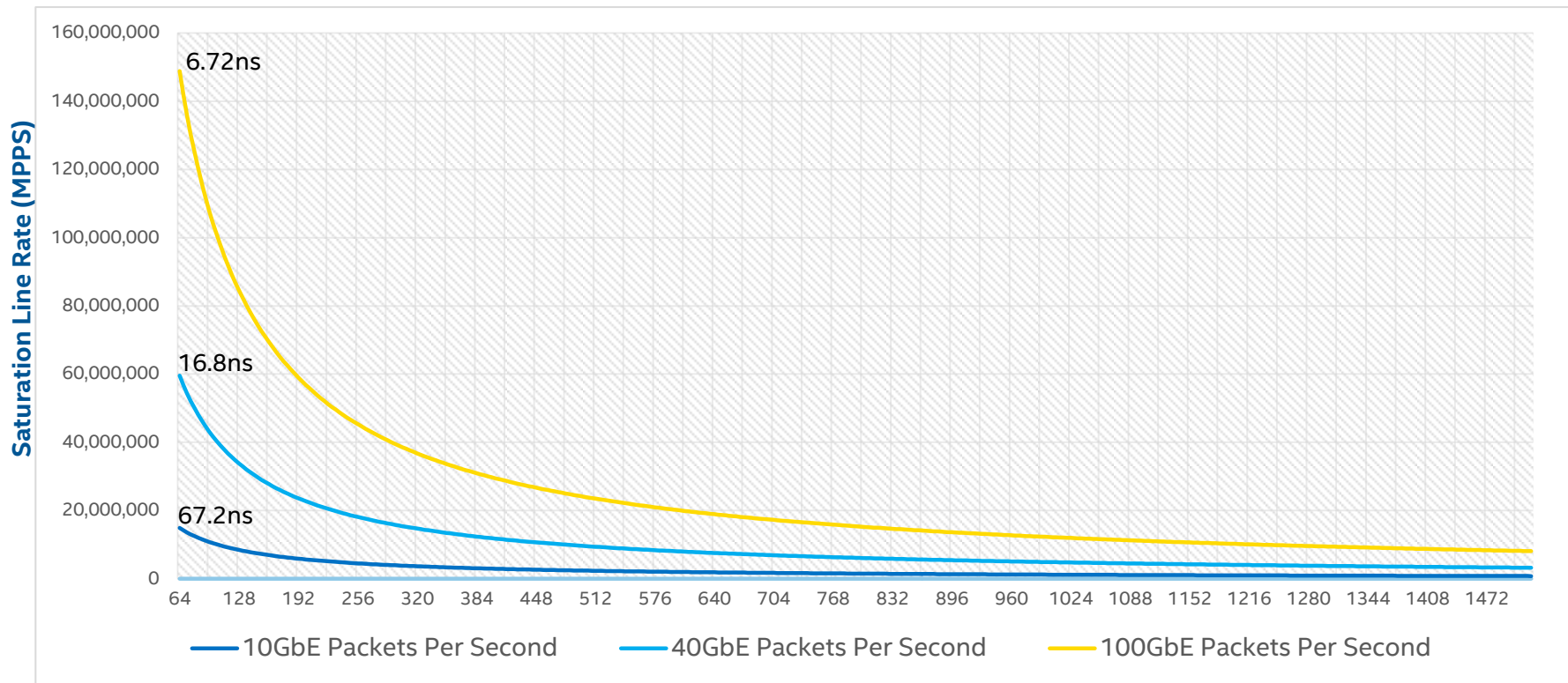
- Pure code optimizations on existing platforms
- New/improved algorithmic implementations
- With multiple CPU architectures (Intel® Xeon®, Atom)

Recommend & drive enhancements

- To further packet processing solutions on Intel Architecture platforms



The Challenge



Intel® DPDK Performance

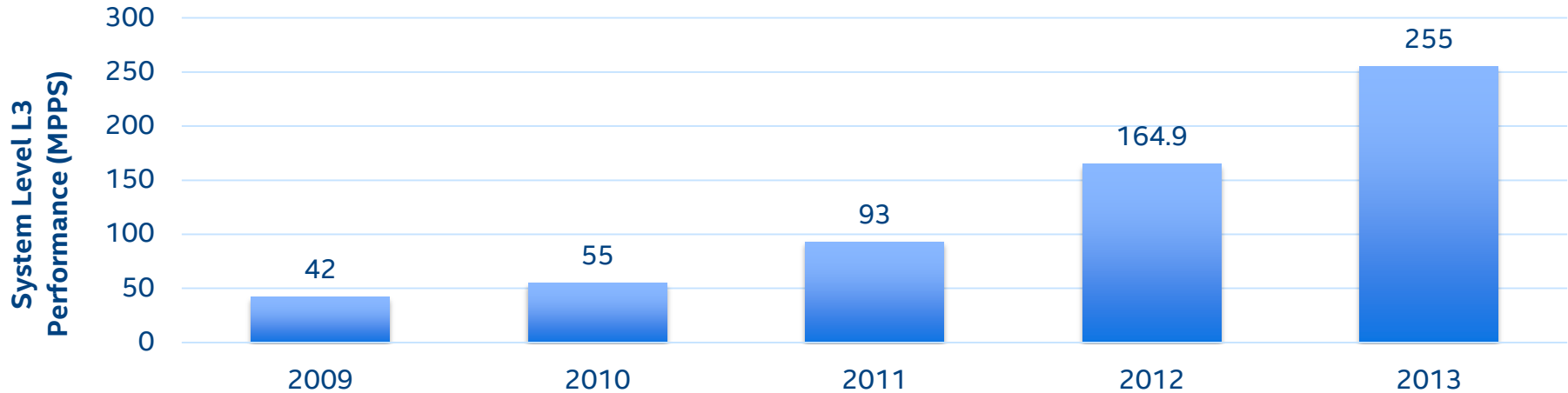
A snapshot of on different architectures

Platform
Features

Integrated Memory Controller
PCI-E Gen2

Data Direct I/O
Integrated PCI-E Gen3
AVX1 (integer)

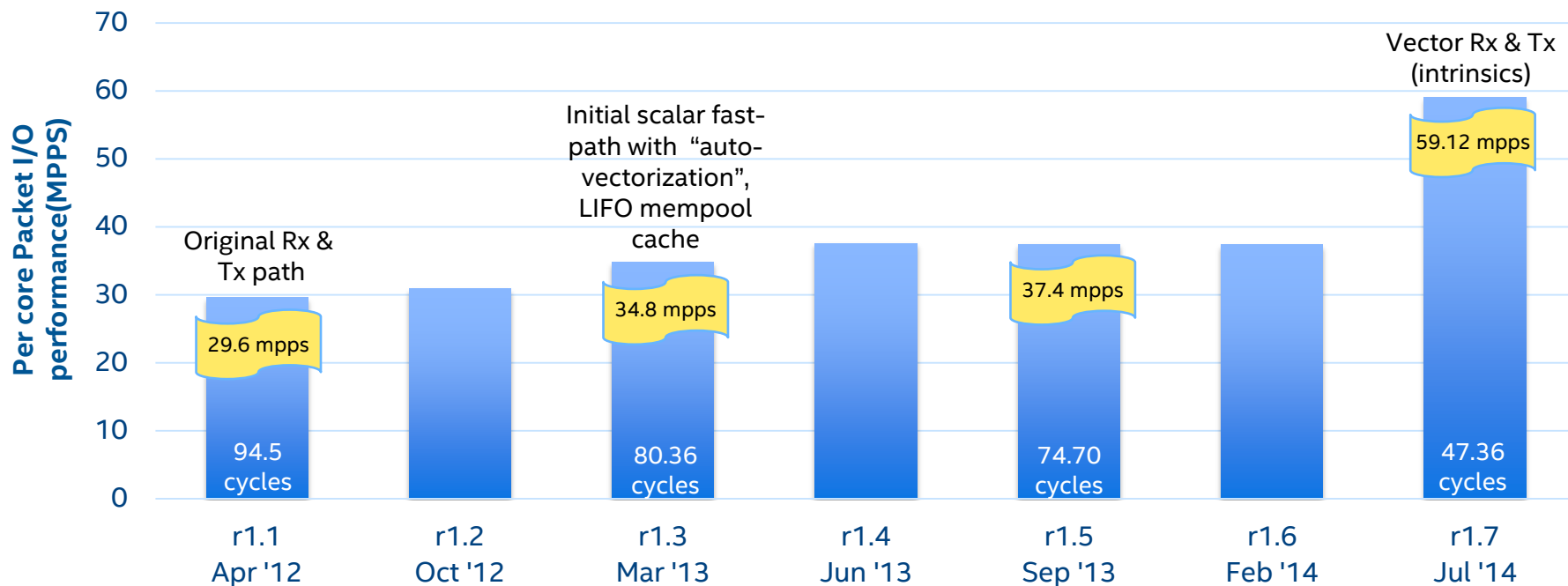
4x10 GbE NICs



Disclaimer: Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products.

Per core I/O performance ...

Performance of DPDK releases on E5-2680v2 (2.80 GHz)



Disclaimer: Software and workloads used in performance tests may have been optimized for performance only on Intel microprocessors. Performance tests, such as SYSmark and MobileMark, are measured using specific computer systems, components, software, operations and functions. Any change to any of those factors may cause the results to vary. You should consult other information and performance tests to assist you in fully evaluating your contemplated purchases, including the performance of that product when combined with other products.

Changes under consideration ...

Improving performance further ...

- Patches to rework the original 1.1 “slow” path with a faster version
- Re-organized mbuf to carry more metadata from NIC
- Investigating implementation using AVX2
- Latency ...

Performance improvements to the exact match library (rte_hash)

- Faster hash functions
- Higher flow count (16M, 32M flows)
- Characterize/limit memory bandwidth usage
- Different algorithmic implementation – cuckoo hash

DPDK and latency

L3fwd default tuning is for performance

- Coalesces packets up to 100us
- Receives and transmits at least 32 packets at a time
 - `nb_rx = rte_eth_rx_burst(portid, queueid, pkts_burst, MAX_PKT_BURST);`

Could bunch 8, 4, 2, or even 1 packet(s) and trade-off some performance

- Lower coalescing increases transmit cost per packet
- Even at ~200 cycles per packet on a CPU running at > 2 GHz software isn't a big adder to latency

Larger system-wide investigations

API versioning

Dynamic management of DPDK resources

- CPU/Threads, memory, network

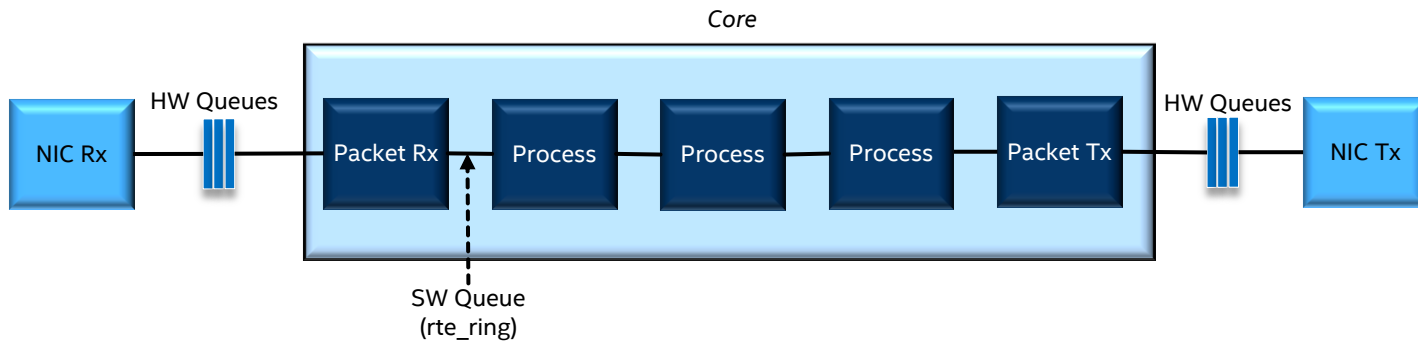
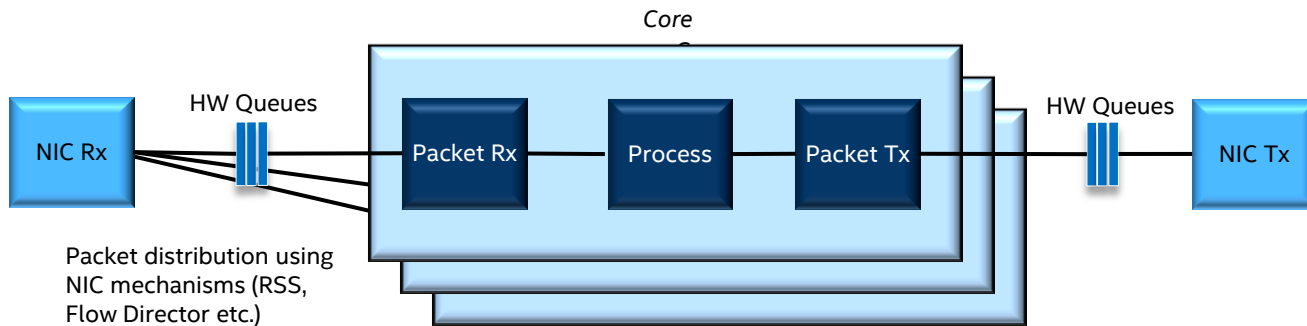
Sharing DPDK core with other pthreads

- Adding interrupt-based entry, cgroups controlled time-sharing

Sharing NIC port between DPDK and kernel

- VFIO, bifurcated driver

Building applications on top of DPDK



Run-to-completion or Pipeline?

DPDK doesn't impose a model – both are supported, as are hybrid approaches

- Each has their advantages and disadvantages
- Have tools/benchmarks to evaluate either approach

Direction should be dictated by

- Legacy code & TTM considerations
- Performance requirements
- NIC limitations (e.g. RSS is purely IP packets only)

Assessing basic performance

app/test: Implements unit-tests and micro-benchmarks

- Micro-benchmarks exist for a number of libraries – labeled *_perf_autotest – mempool, hash, ring, timer, memcpy, distributor *[adding nic in 1.8?]*

app/testpmd: NIC I/O benchmark

- Benchmarks the network I/O pipe (NIC hardware + PMD)

examples/l3fwd: An example L3 forwarder

- Increased CPU processing – NIC hardware + PMD + hash/lpm

examples/load_balance: a simple load balancer in software

examples/ip_pipeline: Using the packet framework to build a pipeline

Intel uses these micro-benchmarks to drive performance

Summary

Multiple areas of improvement in the pipeline

- Focus is on performance, functionality, usability

Looking for community input/participation

- What are the problems that we need to solve?

Let's discuss ...

Backup

The libraries/components (1)

Library	
librte_eal	Environment Abstraction Layer. Meant to hide system/OS specifics from “common” upper layers
librte_malloc	rte_malloc() - replacement for malloc(). Allows allocation of data structures backed by huge pages
librte_mempool librte_mbuf	Memory management: DPDK buffer pool management and packet buffer implementations
librte_ring	High speed ring for inter-core/process pointer passing
librte_timer	Timer routines
librte_lpm	Accelerated longest prefix match
librte_hash	Hash driven key-value exact match for tuple matching
librte_acl	Accelerated implementation of an Access Control List

The libraries/components (2)

Library	
librte_meter	Meter/mark library: Implements srTCM (RFC 2697) and trTCM RFC 2698)
librte_sched	Hierarchical traffic shaper in software
librte_pmd*	Packet Access “Poll” mode drivers
librte_ether	Generic Ethernet device abstraction – the DPDK PMD API
librte_cmdline	Command line parser library
librte_distributor	A work queue distributor
librte_power	Power management primitives
librte_ivshmem	Shared memory implementation for inter-VM communication
KNI, librte_kni	Kernel Network Interface – implements a kernel netdev for passing packets into the kernel from DPDK

