TLDK overview

Konstantin Ananyev

05/08/2016

Legal Disclaimer

General Disclaimer:

Technology Disclaimer:
Intel technologies' features and benefits depend on system configuration and may require enabled hardware, software or service activation. Performance varies depending on system configuration. No computer system can be absolutely secure. Check with your system manufacturer or retailer or learn more at [intel.com].

Performance Disclaimers:
Cost reduction scenarios described are intended as examples of how a given Intel- based product, in the specified circumstances and configurations, may affect future costs and provide cost savings. Circumstances will vary. Intel does not guarantee any costs or cost reduction.
Results have been estimated or simulated using internal Intel analysis or architecture simulation or modeling, and provided to you for informational purposes. Any differences in your system hardware, software or configuration may affect your actual performance.

Transport Layer Development Kit (TLDK)

Project web-site: https://wiki.fd.io/view/TLDK

The scope of the project:

- implement a set of libraries for L4 protocol processing (UDP, TCP etc.) for both IPv4 and IPv6.

- create VPP graph nodes, plugins, etc. using those libraries to implement a host stack.

- mechanisms (netlink agents, packaging, etc.) necessary to make the resulting host stack easily usable by existing non-vpp aware software.

TLDK libraries

- The goal is – a lightweight, high performance and easily adaptable implementation for L4(UDP, TCP etc.) protocol processing.

- Built on top of DPDK.
  - Use DPDK API/features across the libraries.
  - Follow DPDK concepts (process packets in bulks, non blocking API, etc).

- The provided API is not compatible with BSD socket API.
  - Though keep similar semantics (whenever possible).
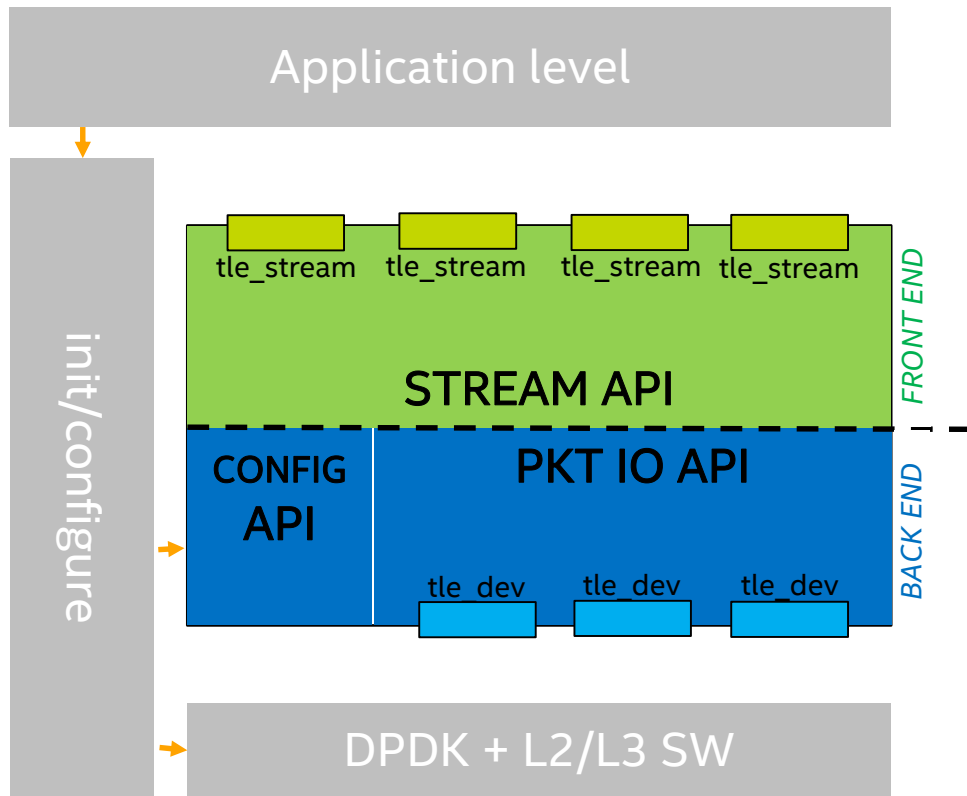
- Not a complete 'host' stack.

Current status

- libtle_udp - implementation of the UDP datagram processing.
  - Operates over both IPv4 and IPv6 packets.

- udpfwd - sample app to demonstrate and test libtle_udp usage.
  - can do simple send/recv or both over opened udp streams.
  - ability to do UDP datagram forwarding between different streams ("UDP proxy").
  - reassemble/fragment IP packets (based on DPDK librte_ip_frag).

In Development

- TCP processing implementation.
  - libtle_tcp.
  - sample application.

- libtle_udp/udpfwd enhancements.
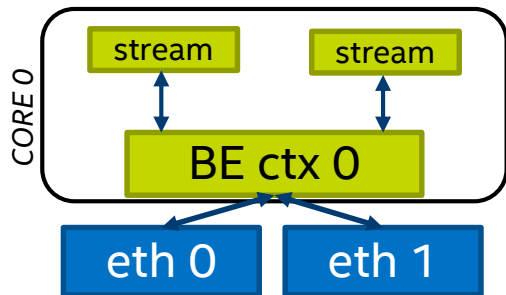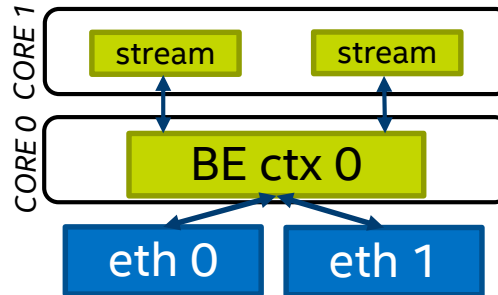 - Extra features (RSS/FD HW offloads, etc).

TLDK API overview

Application level

init/configure

tle_stream tle_stream tle_stream tle_stream

FRONT END

STREAM API

CONFIG API   PKT IO API

BACK END

tle_dev tle_dev tle_dev

DPDK + L2/L3 SW

- *Each TLDK* context operates independently.

- API can be logically divided into:
  - Back-End (BE):
    - *Config API (dev add/remove).*
    - *PKT IO (RX/TX bulk).*
  - Front-End (FE):
    - *stream control and IO (open(), close(), listen(), recv(), send(), etc.).*

- BE API is not thread-safe.

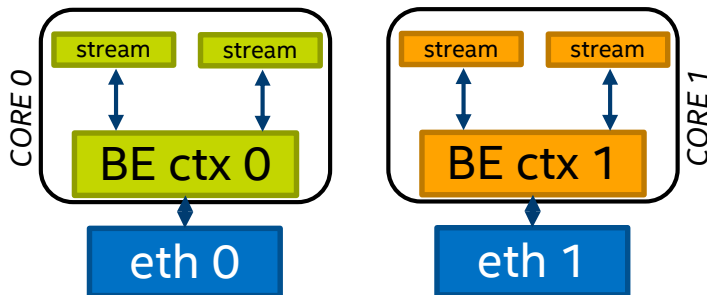- FE API is thread-safe.

Possible deployment scenarios

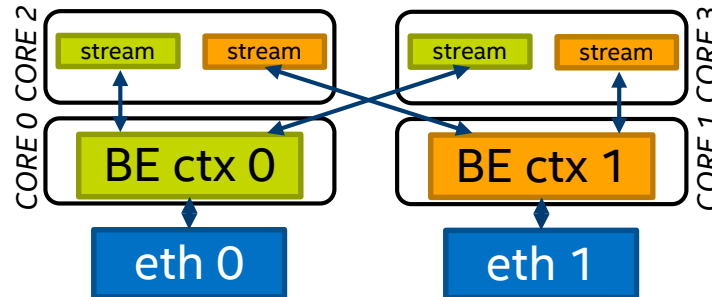1. one TLDK ctx, BE and FE on the same core
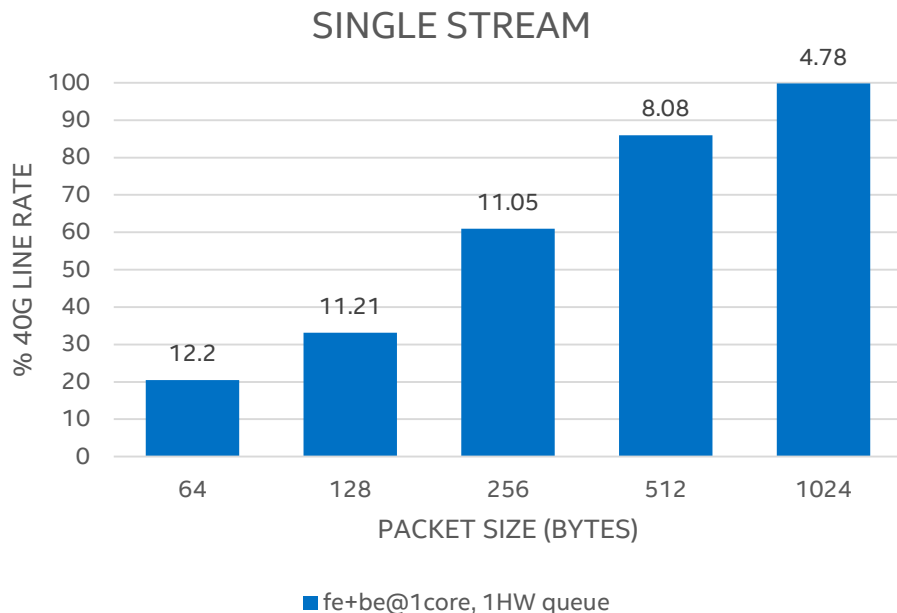2. one TLDK ctx, BE and FE on different cores
3. two TLDK ctxs, for each BE and FE on the same core
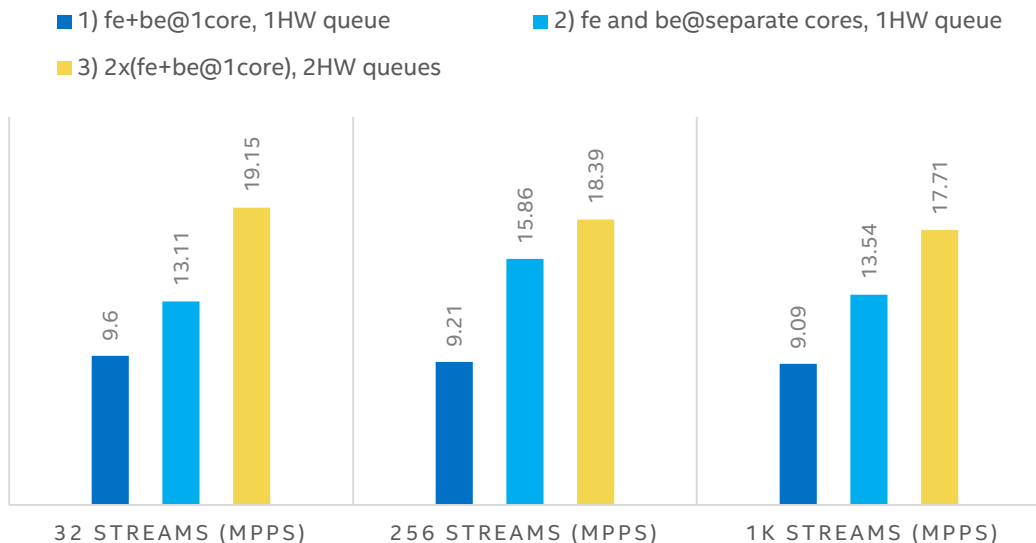4. two TLDK ctxs, BE and FE on different cores

UDPFWD peak performance numbers (echo mode, ipv4/udp)

SINGLE STREAM

% 40G LINE RATE vs PACKET SIZE (BYTES)

Packet Size	Value	Label
64	12.2	~20
128	11.21	~33
256	11.05	~61
512	8.08	~86
1024	4.78	100

■ fe+be@1core, 1HW queue

System Configuration			
Hardware			
CPU	E3-1285 v3		
Sockets	1		
Cores per Socket	4 (8 threads)		
LL CACHE	8MB		
MEMORY	DDR3 1600 MHz, 2X4GB (total 8GB), 2 Channel per Socket		
PCIe	Gen3x8		
NIC	Intel® XL710 for 40GbE QSFP+ Ethernet NIC (1x40G/card)		
NIC Mbps	40,000		
BIOS	BIOS Revision: 4.6		
Software			
OS	Fedora 22		
Kernel version	4.4.13-200		
Other	DPDK 16.07		

UDPFWD peak performance numbers (echo mode, ipv4/udp)

MULTIPLE STREAMS, 64B PKT

- ■ 1) fe+be@1core, 1HW queue
- ■ 2) fe and be@separate cores, 1HW queue
- ■ 3) 2x(fe+be@1core), 2HW queues

| 32 STREAMS (MPPS) | 256 STREAMS (MPPS) | 1K STREAMS (MPPS) |

System Configuration			
Hardware			
CPU	E3-1285 v3		
Sockets	1		
Cores per Socket	4 (8 threads)		
LL CACHE	8MB		
MEMORY	DDR3 1600 MHz, 2X4GB (total 8GB), 2 Channel per Socket		
PCIe	Gen3x8		
NIC	Intel® XL710 for 40GbE QSFP+ Ethernet NIC (1x40G/card)		
NIC Mbps	40,000		
BIOS	BIOS Revision: 4.6		
Software			
OS	Fedora 22		
Kernel version	4.4.13-200		
Other	DPDK 16.07		