OpenDataplane project

ENOG15 2018, Russia, Moscow, 4-5 June

Maxim Uvarov

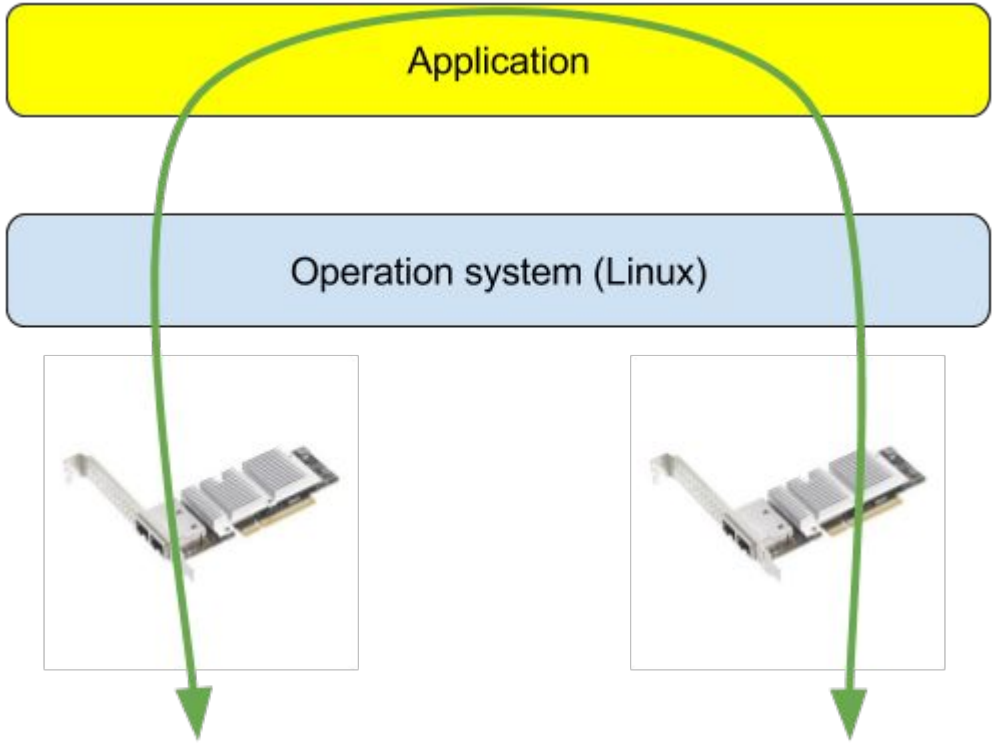
Senior Software engineer, Linaro Networking Group

maxim.uvarov@linaro.org

ENOG15, Russia, Moscow

Linaro

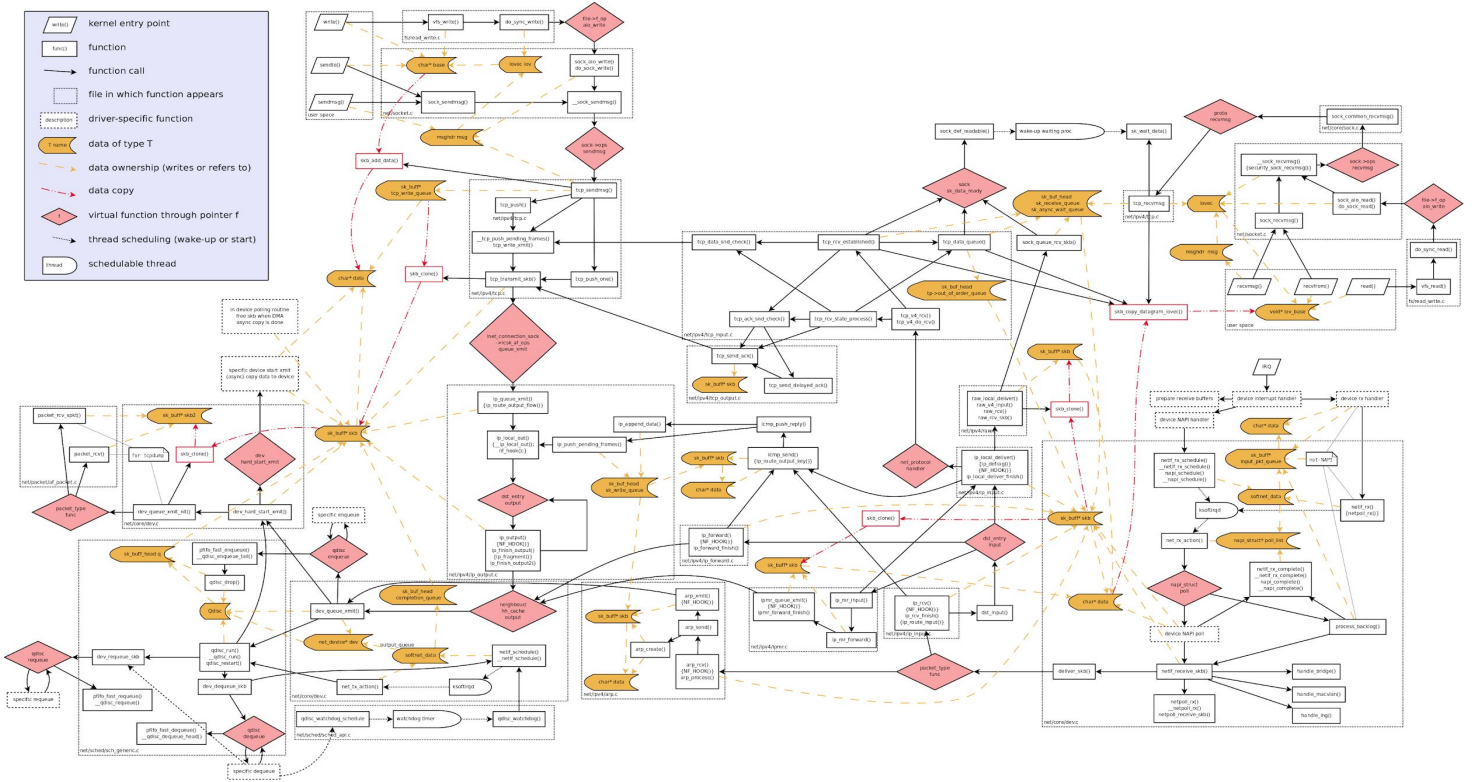
Networking looks like:



Linaro

Implementation inside linux kernel

https://wiki.linuxfoundation.org/images/1/1c/Network_data_flow_through_kernel.png



Linaro

Dataplane is...

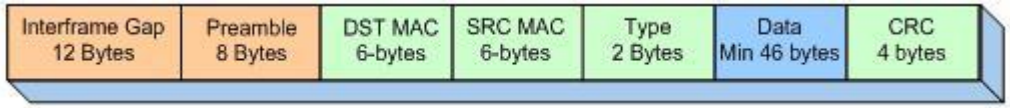
Data plane refers to all the functions and processes that forward packets/frames from one interface to another.

Control plane refers to all the functions and processes that determine which path to use. Routing protocols, spanning tree, ldp, etc are examples.

Management plane is all the functions you use to control and monitor devices.

These are mostly logical concepts but things like SDN separate them into actual devices.

Packets per nanoseconds and cpu in



Speed	bytes/second	Maximum PPS	CPU cycles @1Ghz 1 cpu cycle = 1 ns
100Mbps	12,500,000	148,810	6720
1Gbps	125,000,000	1,488,095	672
10Gbps	1,250,000,000	14,880,952	67,2
100Gbps	12,500,000,000	148,809,524	6,72 ns

Linaro

Why not linux kernel stack?

- Long and unpredictable latencies for network packets;
- Memory size for metadata for network packet (SKB metadata, TCP/IP and netfilter metadata) - few kilobytes for 64 byte packet;
- System fight for cpu time between kernel and userland, spending cpu cycles is not optimal;
- TLB entries for both userland and kernel, no Huge Pages support;
- Packet body copy, strip out vlan tags and later re insertion in user land for AF packet;
- Other interrupts involved which may delay execution;
- No hw accelerators support;
- Crash in kernel makes whole system not operable and you have to reboot. Reboot whole system vs one single application.

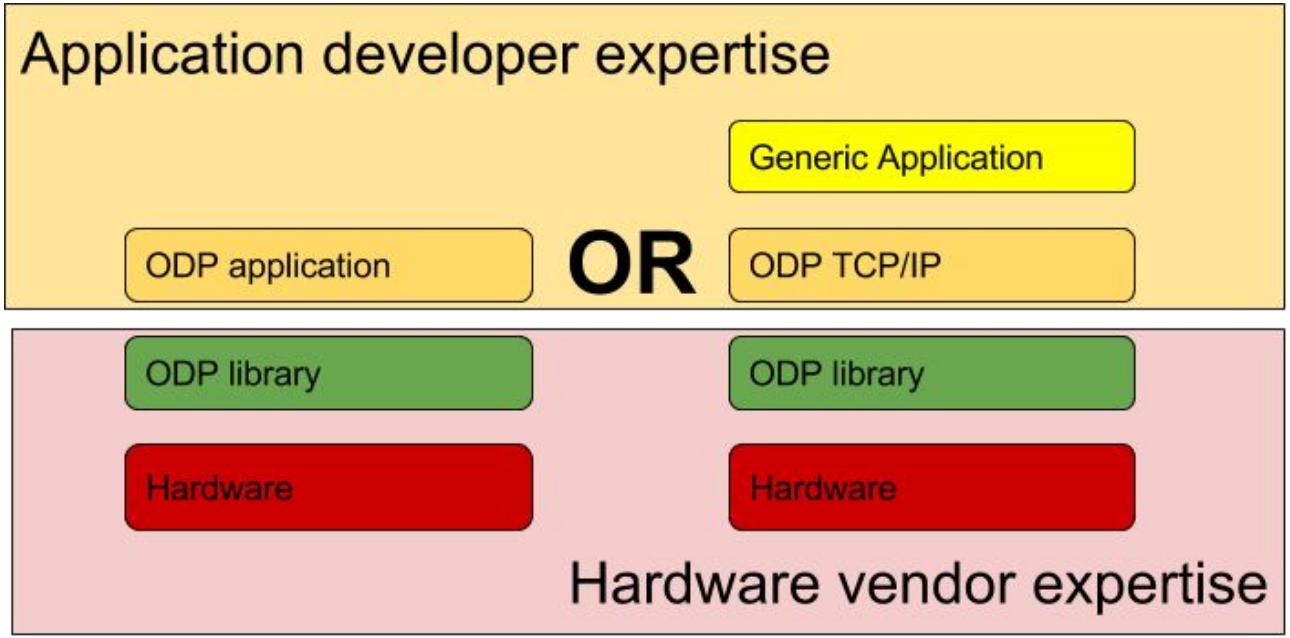
Dataplane types

- Hardware
- Software
- Software Defined

OpenDataPlane intro

The ODP project has been established to produce an open-source, cross-platform set of application programming interfaces (APIs) for the networking data plane.

- Applications for hardware agnostic API. They are portable!



- Maximum performance guaranteed by hardware vendor doing ODP implementation for specific hardware.

Linaro

Main git repo https://github.com/Linaro/odp includes:

- ODP API
- Reference implementation:
    - Static and dynamic library
    - ABI (binary) compatibility
    - Non ABI compat version uses “native types” for more speed optimization.
- API Validation and performance tests.
- Examples.
- User documentation.

ODP defines API and does not put any restrictions for implementations.

Linaro

What else?

- DPDK
- Netmap
- PFring

Are more PCI NiC specific.

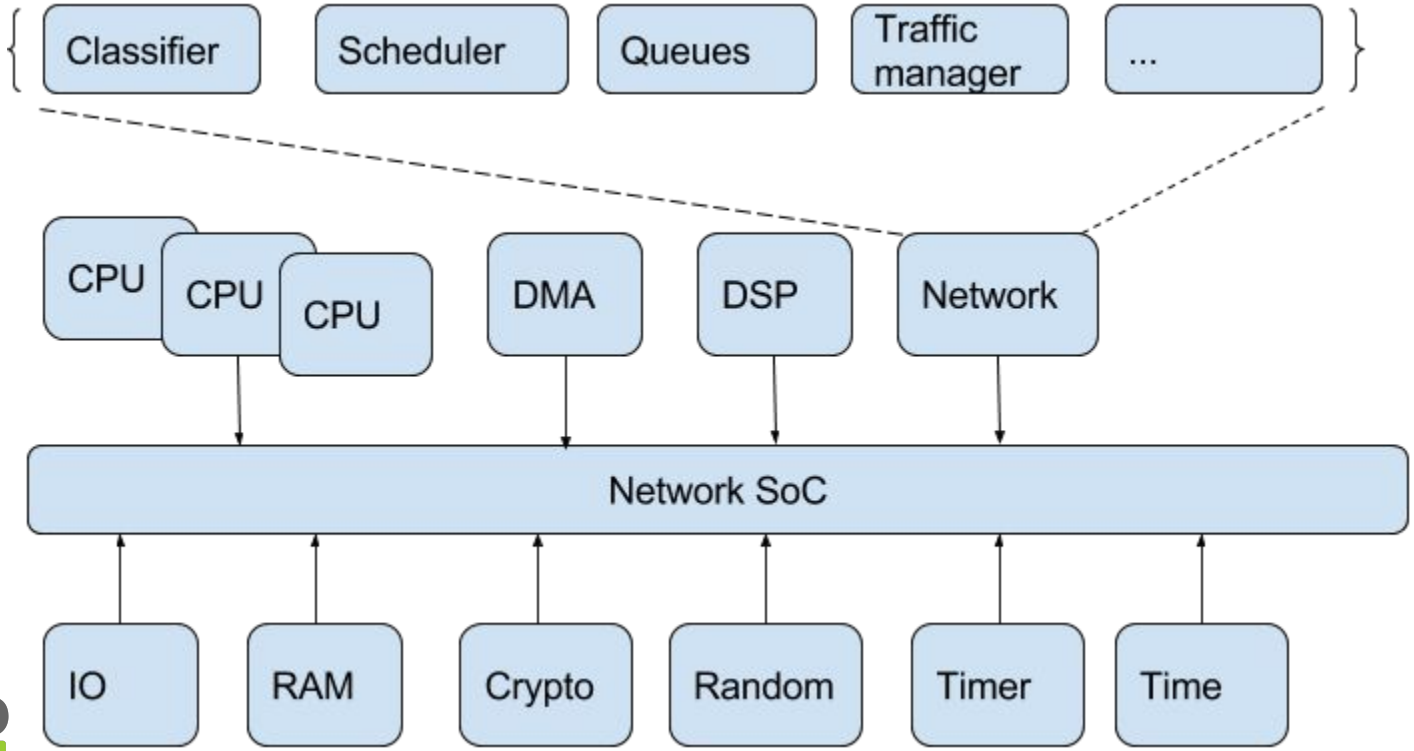
PCI express speeds:

PCI Express Version	Line Code bytes	Transfer Rate GT/s	Throughput GB/s		
			×4	×8	×16
3.0	128/130	8	3.94	7.9	15.8
4.0	128/130	16	7.9	15.8	**31.5**
5.0 (*expected in 2019*)	128/130	32 or 25	15.8 or 12.3	31.5 or 24.6	**63.0 or 49.2**

Linaro

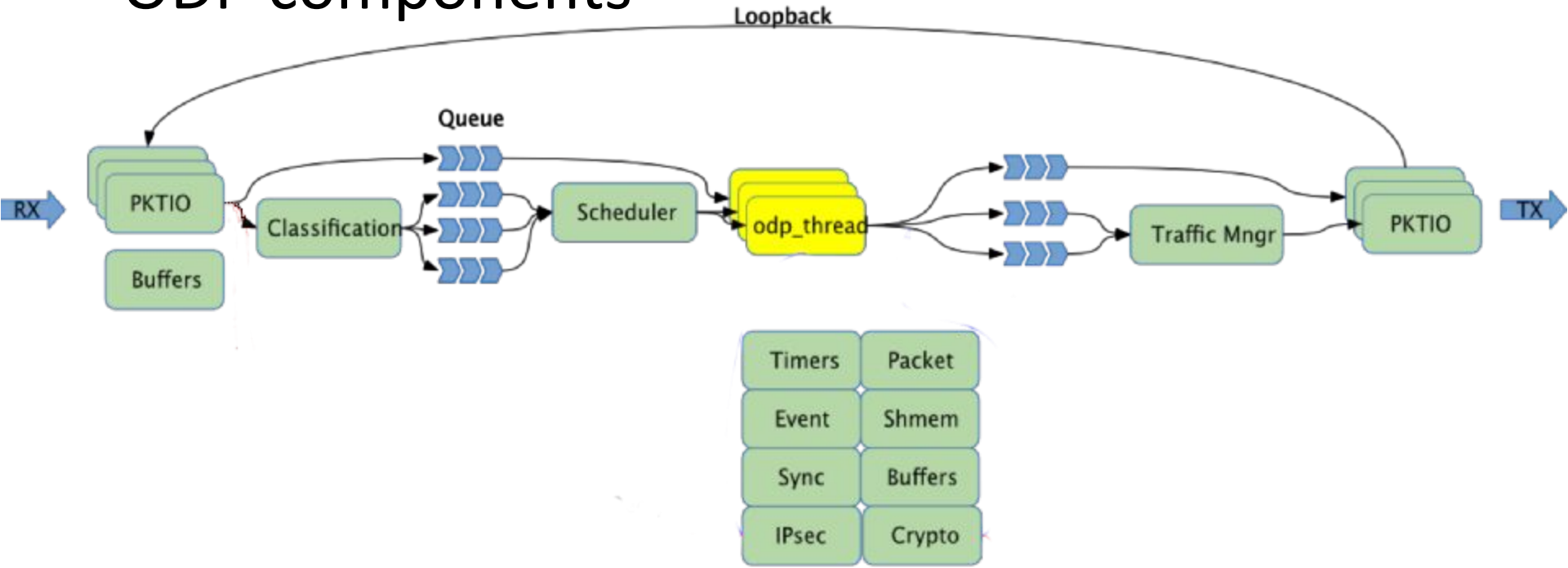
Network traffic is usually a lot of small DMA transactions which will make that results much more slower.

Abstract network hardware, SoC



Linaro

ODP components

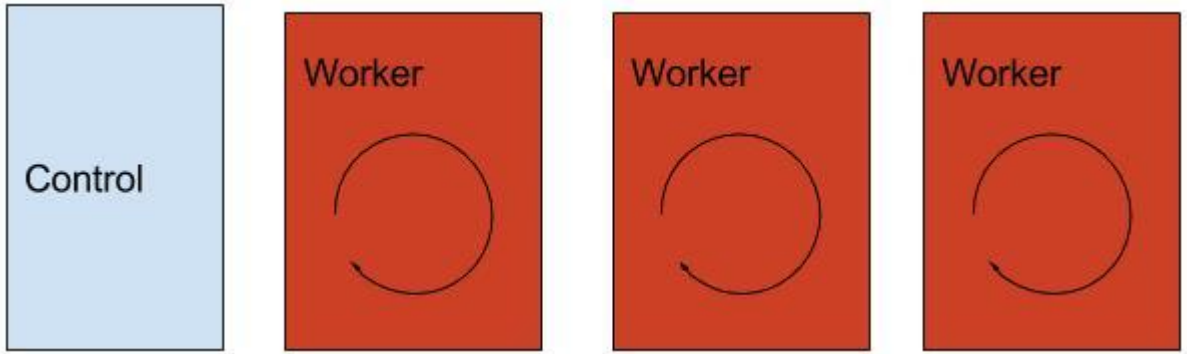


Linaro

ODP run model

- Main control thread/process to allocate memory, initialize packet i/o and configure components;

- Number of worker threads/processes owns core to process



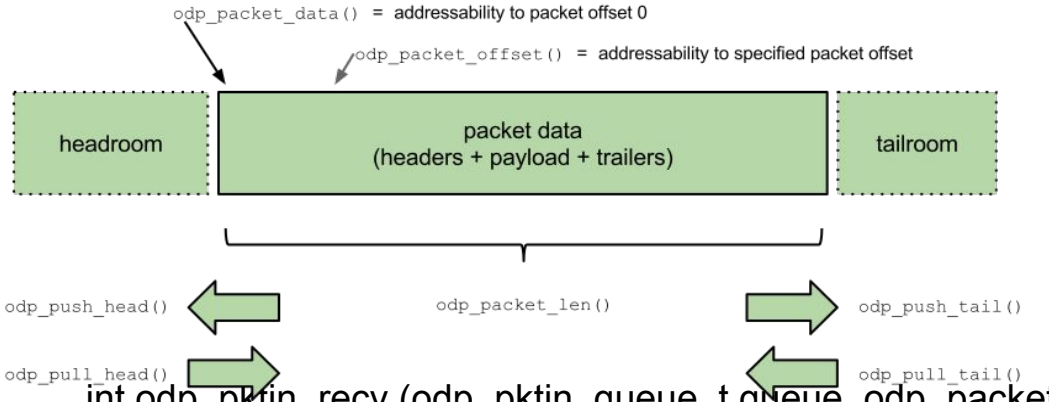
Linaro

ODP internals: packet and packet I/O

odp_pktio_t - is abstraction type of packet input/output

odp_pktio_start() , odp_pktio_stop() and etc

odp_packet_t - abstract type for network packet



int odp_pktin_recv (odp_pktin_queue_t queue, odp_packet_t packets[], int num)

Int odp_pktout_send (odp_pktout_queue_t queue, const odp_packet_t packets[], int num)

Linaro

ODP internal: queues

```
odp_queue_t odp_queue_create(const char *name,
                             const odp_queue_param_t *param)
```

- Parallel queues
- Atomic queues
- Ordered queues

Queues can be grouped to scheduler groups, have different priority and be offloaded to hardware.

Linaro

ODP internals: ODP_PKTIN_MODE_DIRECT

```
odp_pktin_queue_t    queue_in,

odp_pktout_queue_t  queue_out

odp_pktin_queue(pktio, &queue_in, 1);

odp_pktout_queue(pktio, &queue_out, 1);

while (1)  {

        pkts = odp_pktin_recv_tmo(queue_in, pkt_tbl, MAX_PKT_BURST,

                                  ODP_PKTIN_NO_WAIT);

        sent = odp_pktout_send(queue_out, pkt_tbl, pkts);

}
```

Linaro

ODP internals: ODP_PKTIN_MODE_SCHED

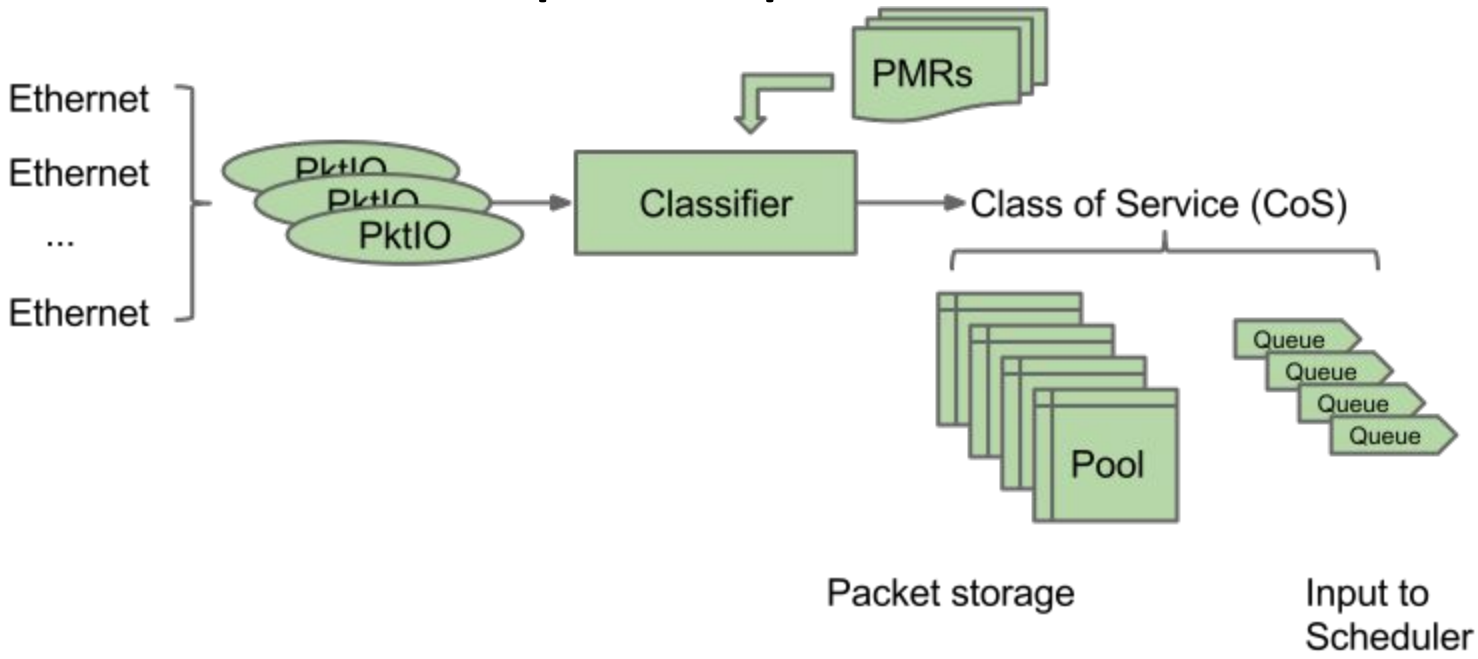
```
while (1)  {

        odp_event_t ev;

        ev = odp_schedule();

        switch (odp_event_type(ev)) {

        case ODP_EVENT_PACKET:

                pkt = odp_packet_from_event(ev);

                sent = odp_pktout_send(queue_out, pkt, 1);

        case  ODP_EVENT_TIMEOUT:

        case ODP_EVENT_CRYPTO_COMPL:
```

odp_event_t

odp_buffer_t odp_packet_t odp_timer_t

Linaro

ODP internals: pool, queues and classifier



Linaro

ODP internals: Traffic manager

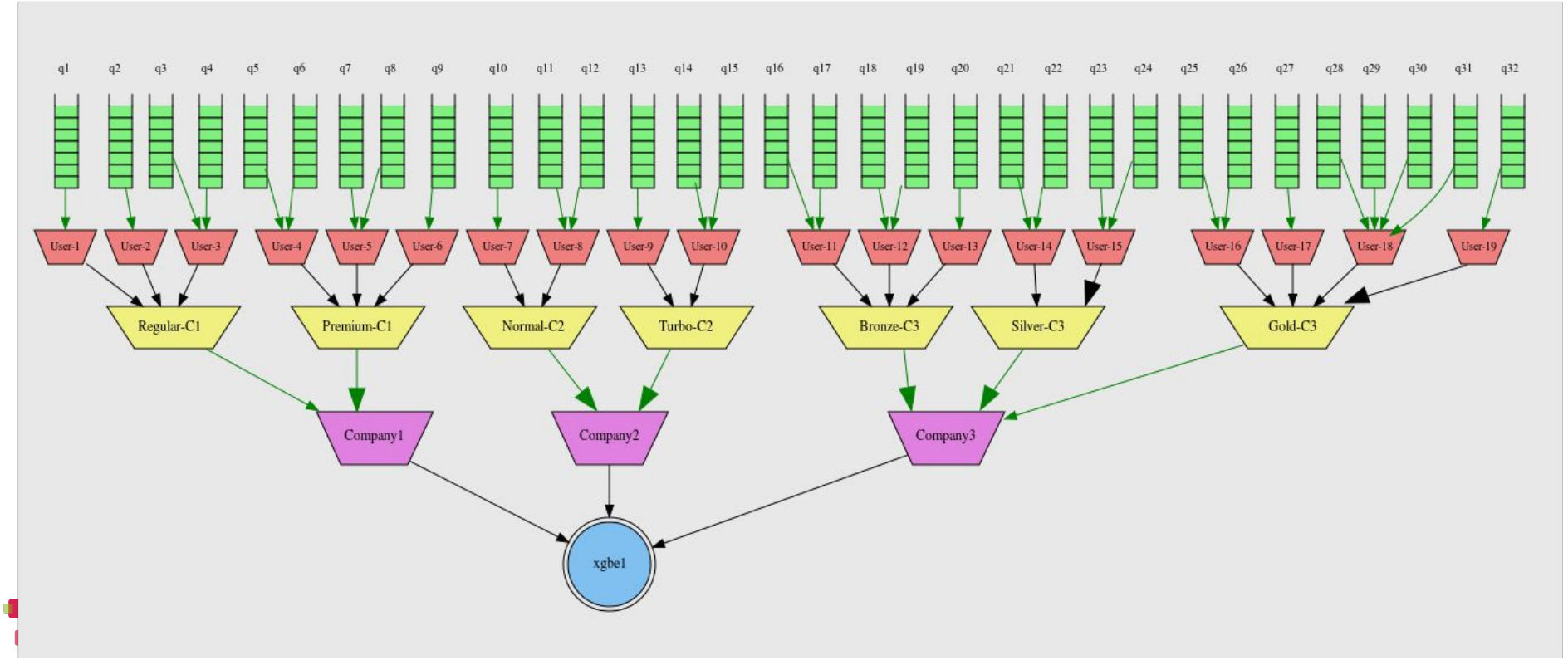
QoS packet scheduling ODP block using following algorithms:

- Strict Priority scheduling.
- Weighted Fair Queueing scheduling (WFQ).
- Bandwidth Shaping.
- Weighted Random Early Discard (WRED).

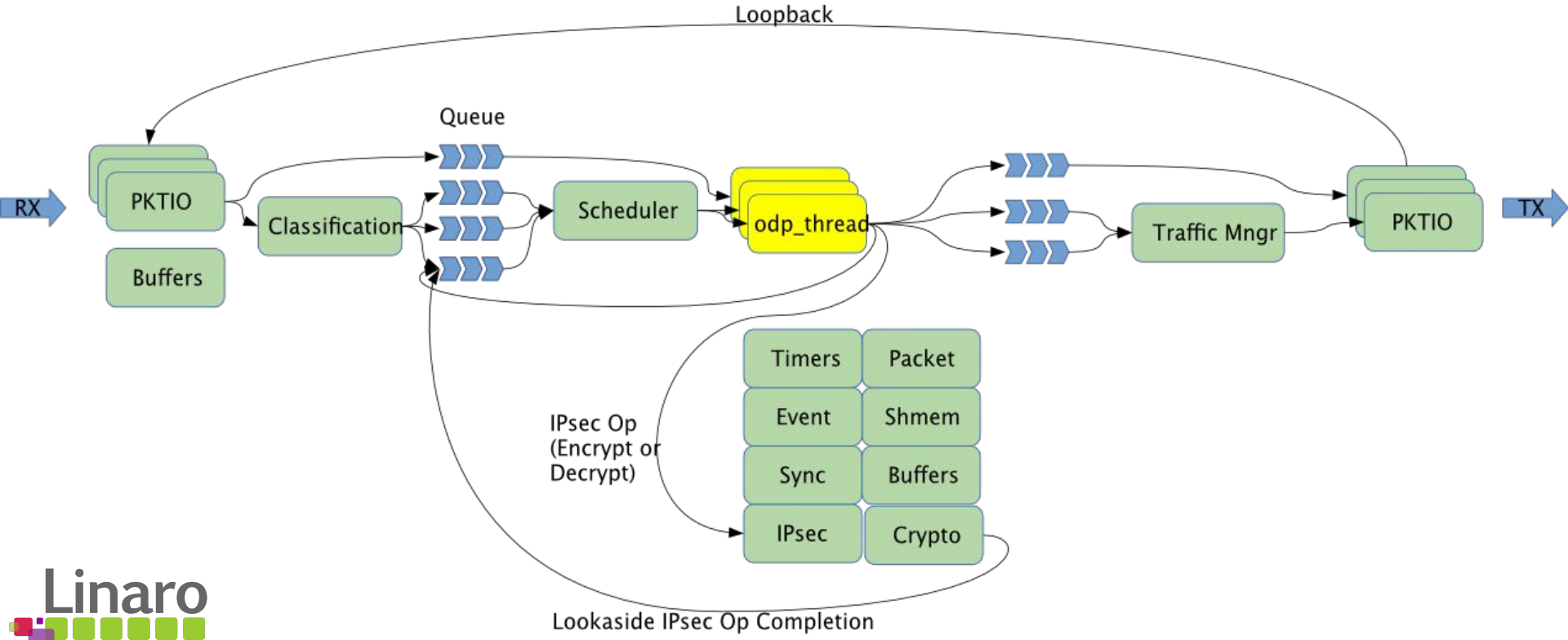
```
odp_tm_t tm = odp_tm_create("TM", &tm_params);

odp_tm_queue_t tq = odp_tm_queue_create(tm, &queue_params);

tn = odp_tm_node_create(tm, "TmNode", &node_params);

odp_tm_queue_connect(tq, tn);

sched_profile_RR = odp_tm_sched_create("SchedProfileRR", &sched_params);
```

Linaro

ODP internals: Traffic manager (Continued...)

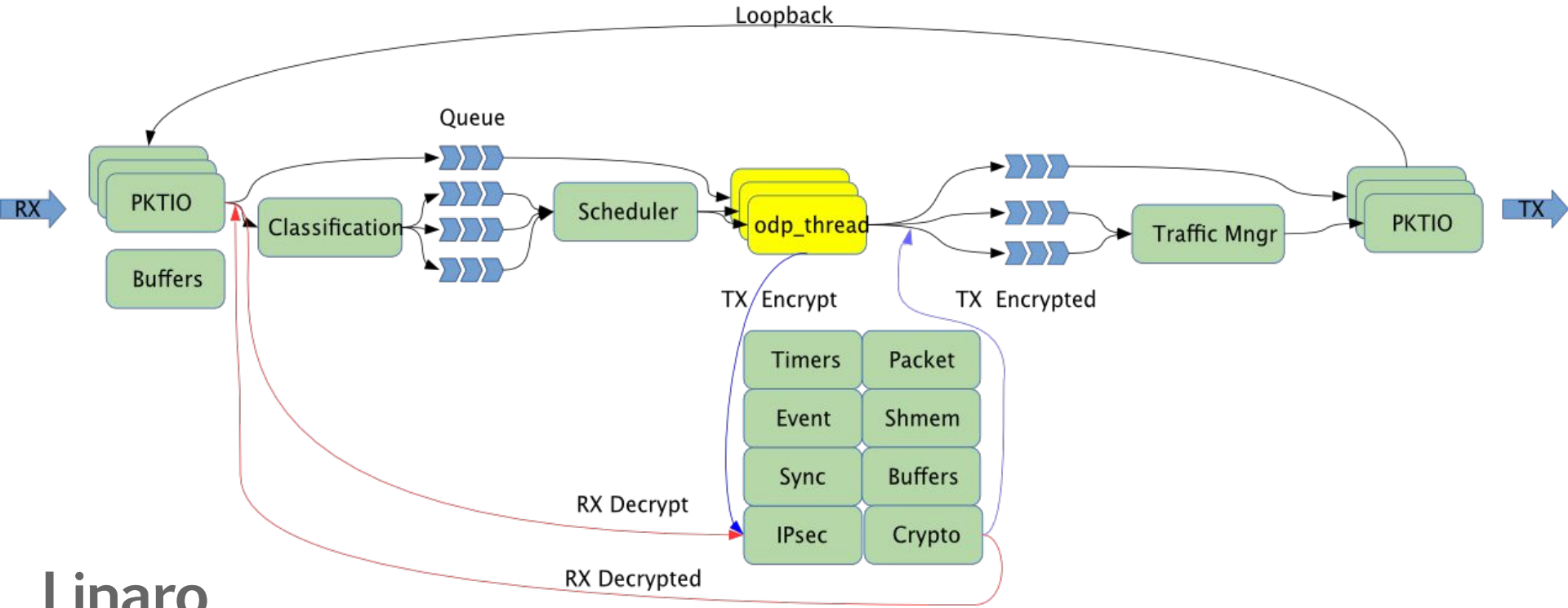


ODP internals: advanced features in short: ipsec look aside offloading



Linaro

ODP internals: advanced features in short: ipsec inline offloading



Linaro

OpenFastPath.org

- IP stack on top of OpenDataPlane
- Open Source project lead by Nokia, Enea and ARM
- http://openfastpath.org
- BSDv3 License

Linaro

Nginx + ODP + OFP

https://github.com/OpenFastPath/nginx_ofp

1) Modified nginx of OFP - performance improvement.
2) Not modified nginx - works but work in progress.

export OFP_NETWRAP_ENV="-i eth1 -f ofp_netwrap.conf"

LD_PRELOAD=libofp_netwrap_crt.so.0.0.0:libofp.so.0.0.0:libofp_netwrap_proc.so.0.0.0 **/usr/sbin/nginx**

Linaro

ODP contributors are:

Linaro

Other ODP implementations

https://www.opendataplane.org/downloads/

- Cavium (Thunder-X)
- Freescale (based on DPAA)
- Texas Instruments (Keystone II)
- Kalray (MPAA)
- Linaro (based on DPDK)

Linaro

How to contribute and get involved to ODP

- Everyone can contribute to ODP!

  lng-odp@lists.linaro.org http://opendataplane.org

  https://github.com/Linaro/odp BSDv3 License

  Public call: Tuesday's 15:00 UTC at http://meetings.opendataplane.org

- Member companies of Linaro Networking Group define strategy and priority for project and pay fees to support project maintenance and required equipment. Contact is linaro.org

Linaro

Thank you!

Questions?

OpenDataPlane
.org

Linaro

We are working on ...

- Drivers for packet I/O.
- Components Modularization.
- Hardware discovery and automatic loadings.
- Performance optimization.