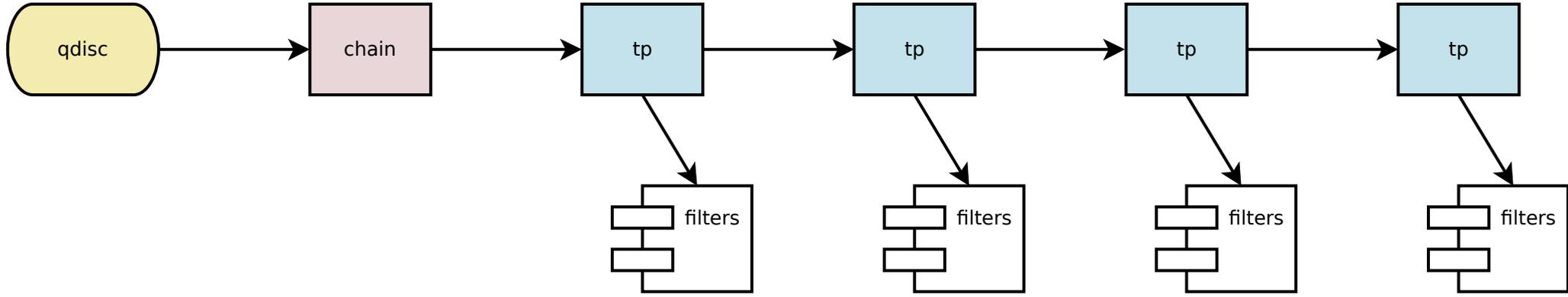


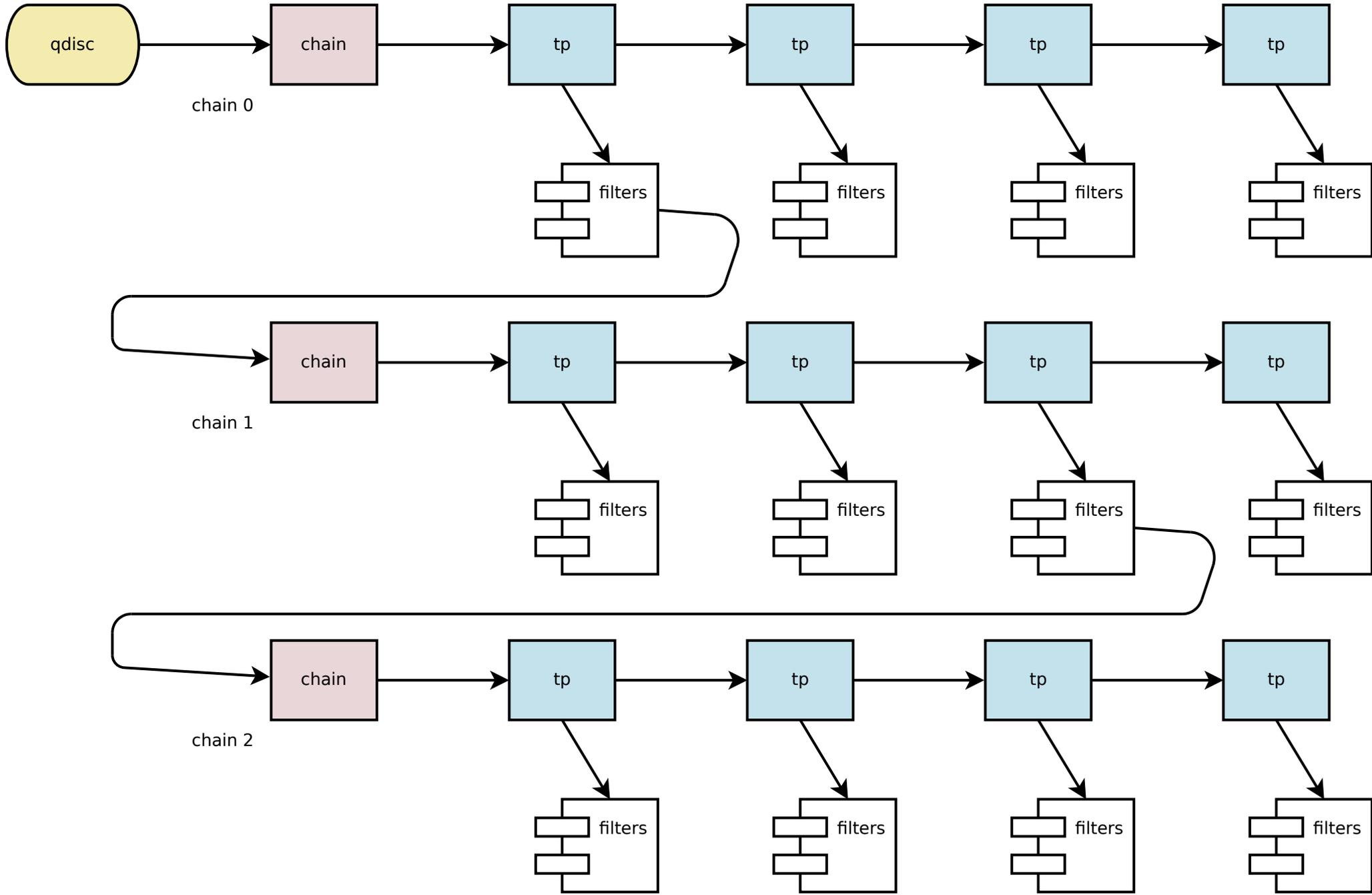
TC classifier-action subsystem  
Update since netdevconf 2.1  
WIP, future work

# TC cls-act changes since netdev2.1 - multichain

net: sched: introduce multichain support for filters

```
$ tc qdisc add dev eth0 ingress
$ tc filter add dev eth0 ingress protocol ip pref 33 \
  flower dst_mac 52:54:00:3d:c7:6d action goto chain 1
$ tc filter add dev eth0 ingress protocol ip pref 22 \
  chain 1 flower dst_ip 192.168.40.1 action drop
```





# TC cls-act changes since netdev2.1 – the rest

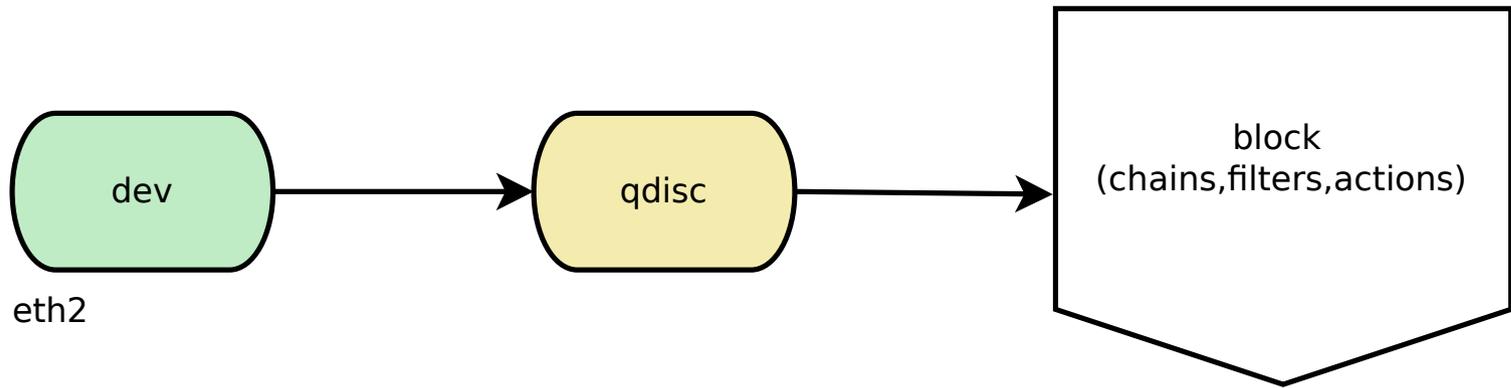
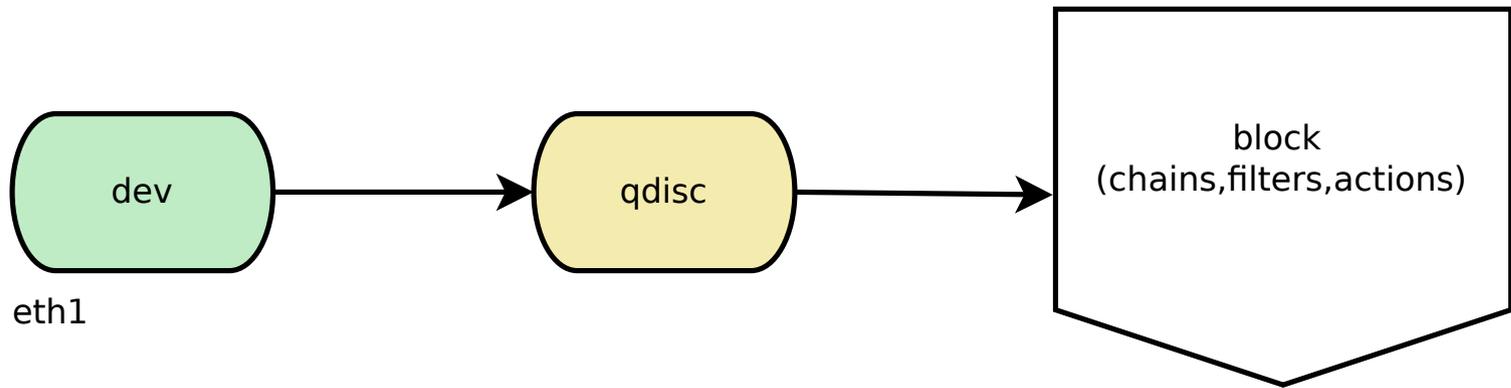
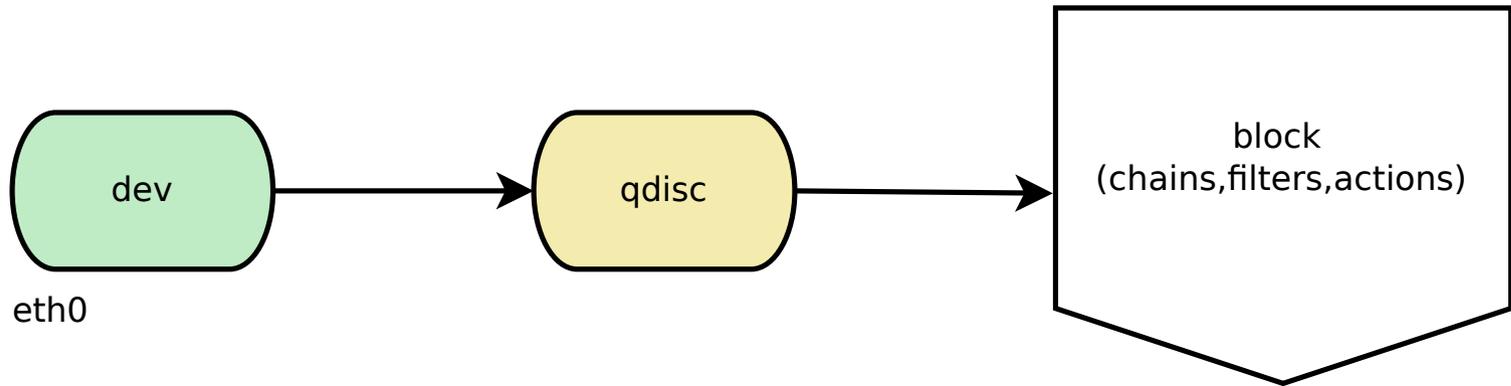
- \* cls\_flower: add support for matching MPLS fields
  - \* net/sched: flower: add support for matching on tcp flags
  - \* net/sched: cls\_flower: add support for matching on ip tos and ttl
- 
- \* net: sched: introduce a TRAP control action
    - \$ tc filter del dev eth0 ingress protocol ip pref 10 \  
flower skip\_sw dst\_ip 192.168.101.1 action trap

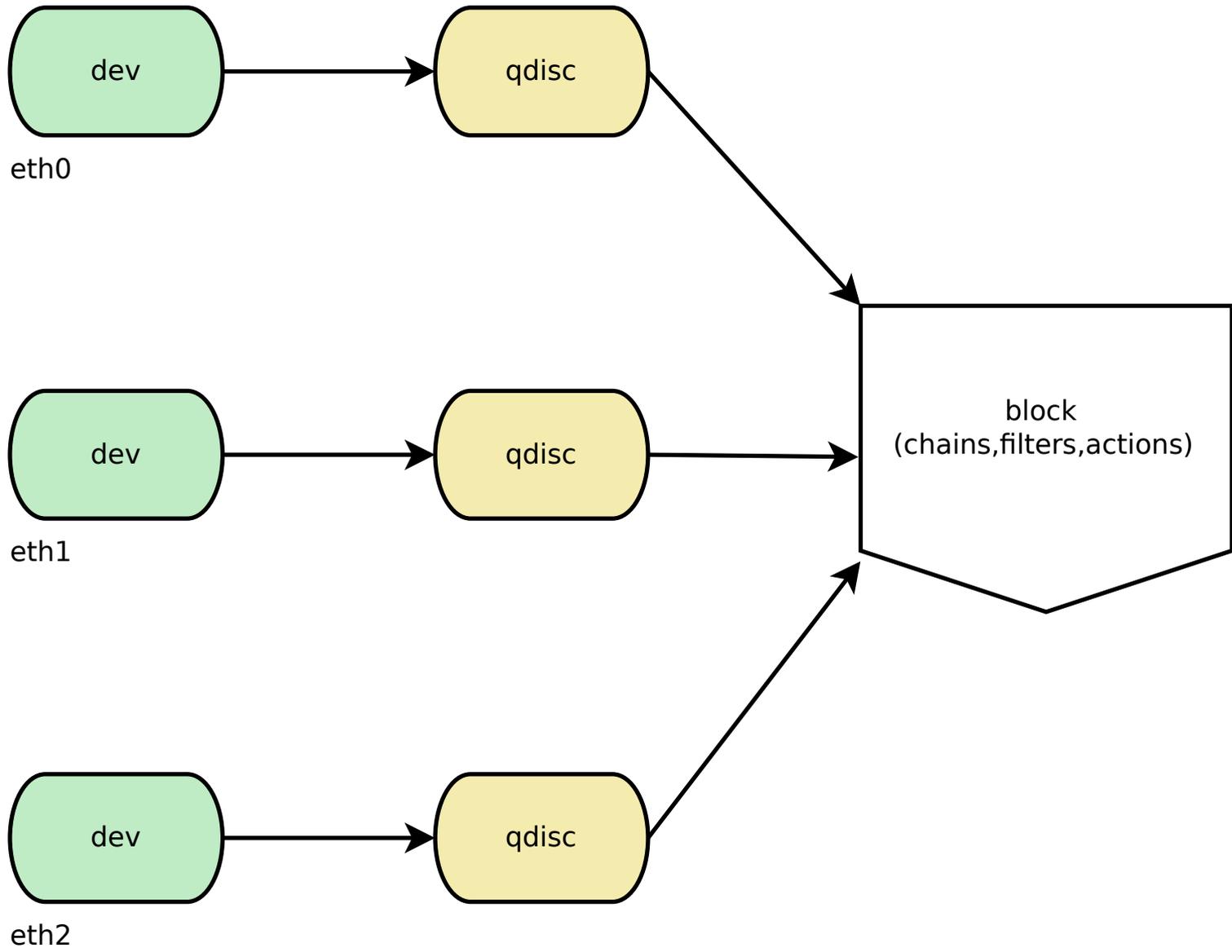
## MLXSW driver changes:

- \* mlxsw: spectrum\_flower: Add support for tcp flags
- \* spectrum\_flower: Implement gact trap TC action offload
- \* mlxsw: spectrum\_flower: Add support for ip tos
- \* mlxsw: spectrum\_flower: Add support for ip ttl
- \* mlxsw: spectrum: Offload multichain TC rules
- \* mlxsw: spectrum\_flower: Offload goto\_chain termination action
- \* mlxsw: spectrum\_flower: Offload "ok" termination action

# WIP - Shared blocks

- Purpose is to allow to share filter instances among multiple netdevices
- Why?
  - To reduce memory footprint
  - To ease-up user's job when he wants to have same ruleset (filters) on many netdevices
  - For offload use-case, to allow to optimize HW resources usage (e.g. TCAM is quite limited)
- Took a lot of effort to do all preparations, mainly tp → q removal (qdisc pointer in struct tcf\_proto)
- All preparations are merged in net-next git, one more tp → q removal needs to be done before the patchset could be merged
- mlxsw offload prepared in queue





# Shared blocks examples

```
$ tc qdisc add dev eth0 ingress block 22
$ tc qdisc add dev eth1 ingress block 22
$ tc qdisc add dev eth2 ingress block 22
```

```
$ tc qdisc
qdisc ingress ffff: dev eth0 parent ffff:fff1 block 22
qdisc ingress ffff: dev eth1 parent ffff:fff1 block 22
qdisc ingress ffff: dev eth2 parent ffff:fff1 block 22
```

```
$ tc filter add dev ens0 ingress protocol ip pref 25 \
  flower dst_ip 192.168.0.0/16 action drop
```

```
$ tc qdisc add dev eth3 clsact ingress_block 10 egress_block 11
$ tc qdisc add dev eth4 clsact ingress_block 10 egress_block 11
```

# Future?

- Hinting for offloads
  - Match keys hinting
    - During TCAM allocation, we have to know all used match keys for all possibly inserted rules (to avoid costly re-allocations)
    - User may provide a hint which keys he is going to need so the driver can prepare and optimize resource utilization
  - Filter max count hinting
    - User may know total amount of filters he is going to insert per-chain. That would again help driver to optimize resource utilization