



Membership Library in DPDK 17.11

Sameh Gobriel & Charlie Tai - Intel

DPDK US Summit - San Jose - 2017

Contributors



- ▶ Yipeng Wang yipeng1.wang@intel.com
- ▶ Ren Wang ren.wang@intel.com
- ▶ John Mcnamara john.mcnamara@intel.com
- ▶ Pablo De Lara Guarch pablo.de.lara.guarch@intel.com
- ▶ Antonio Fischetti antonio.fischetti@intel.com

Agenda

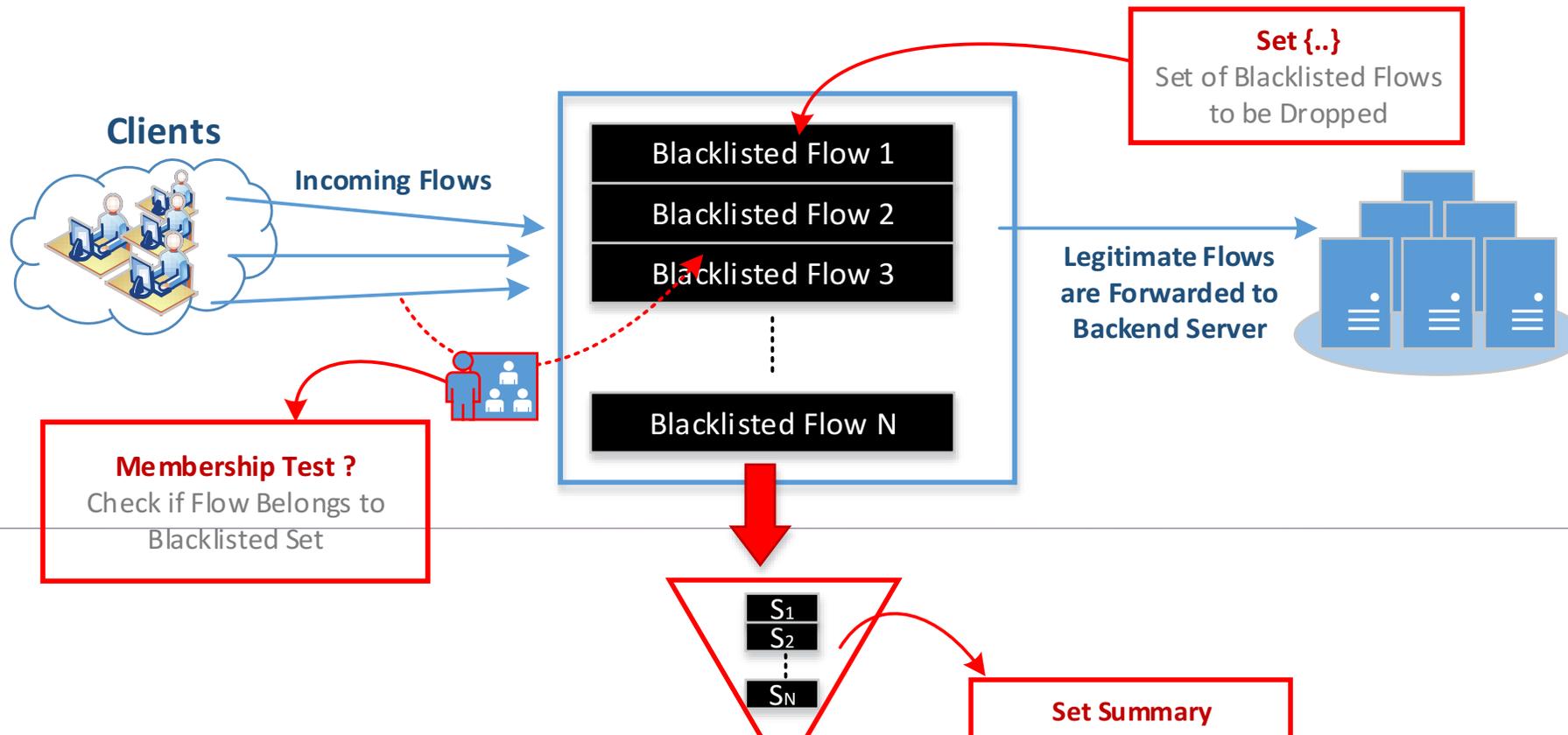


- ▶ Membership Library in DPDK 17.11
 - ▶ Membership Library Usages
 - ▶ API Overview
 - ▶ Research Proof of Concept: using Membership Library with OVS
-

Membership Test Usage (example)

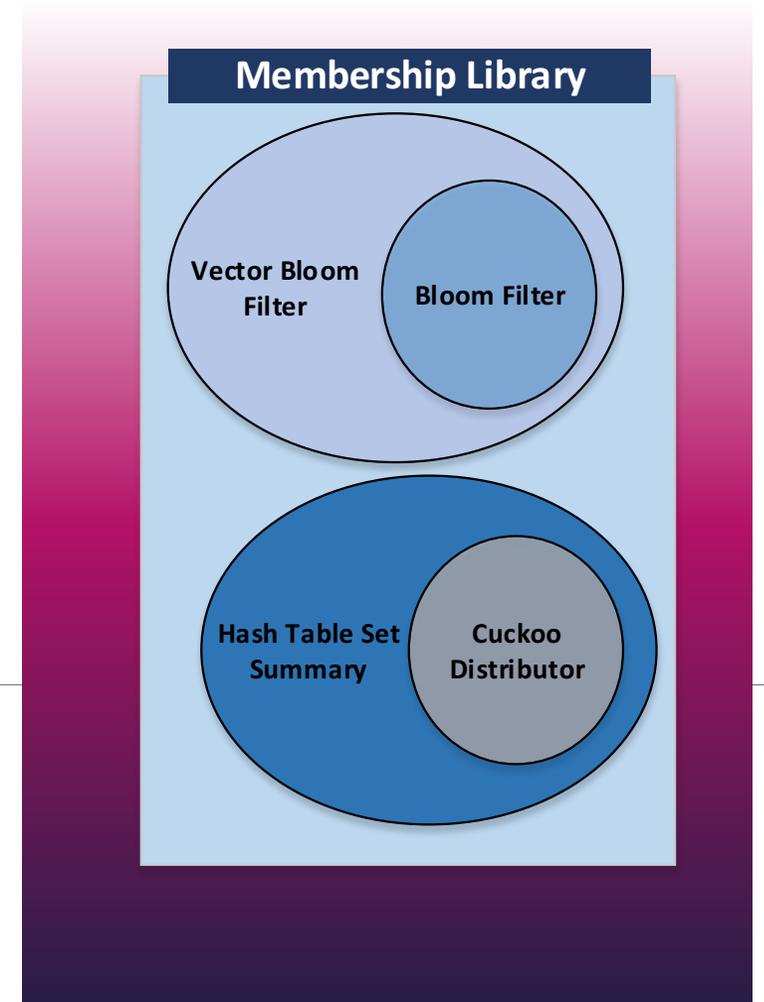
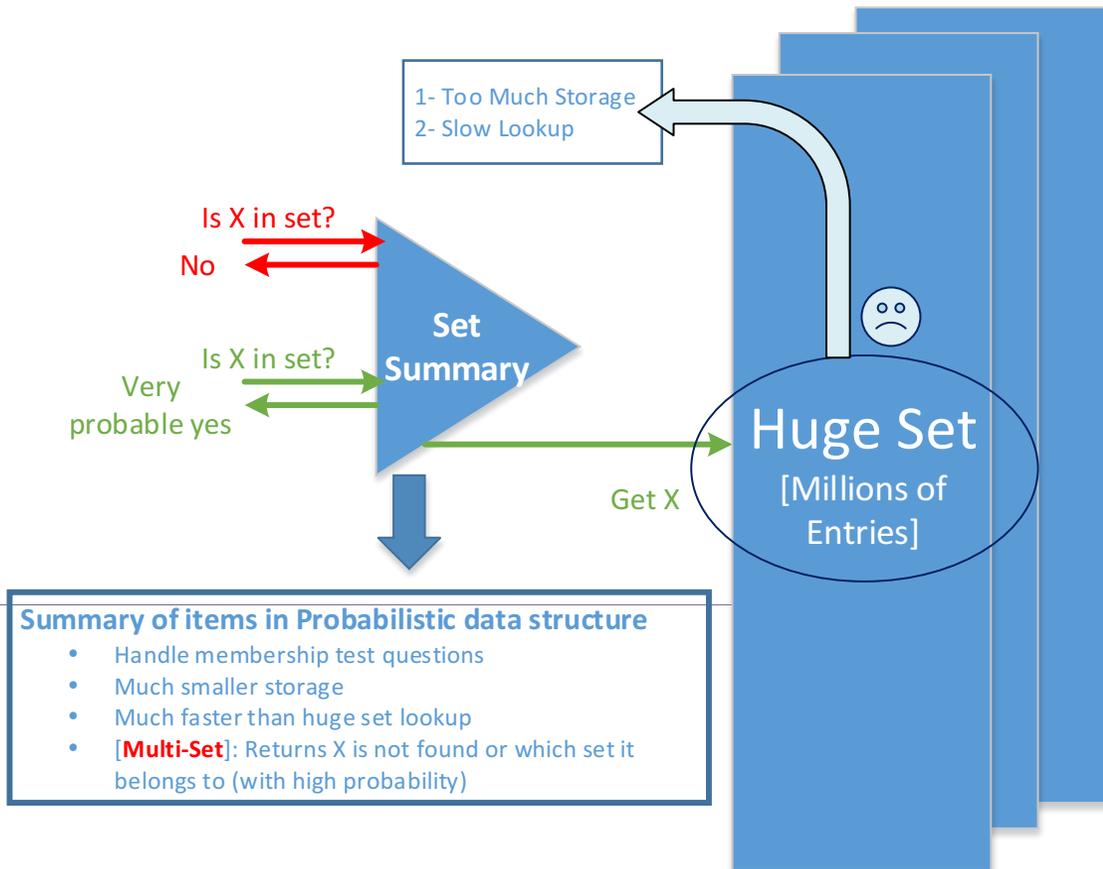


Build



Membership Library is a DPDK Library to Provide Users the Functionality to Create Different Types of Set-Summaries

Overview of DPDK Membership Library



Library Usages?



Safe Browsing

TCP Connection Tracker

Database Semi-join

Signature Matching
and packet

Distributed web caching

Set intersections and keyword
searches

P2P Overlay Networks =
Object Indexing

Network Statistics and
summaries

Detecting loops in unicast and
multicast routes

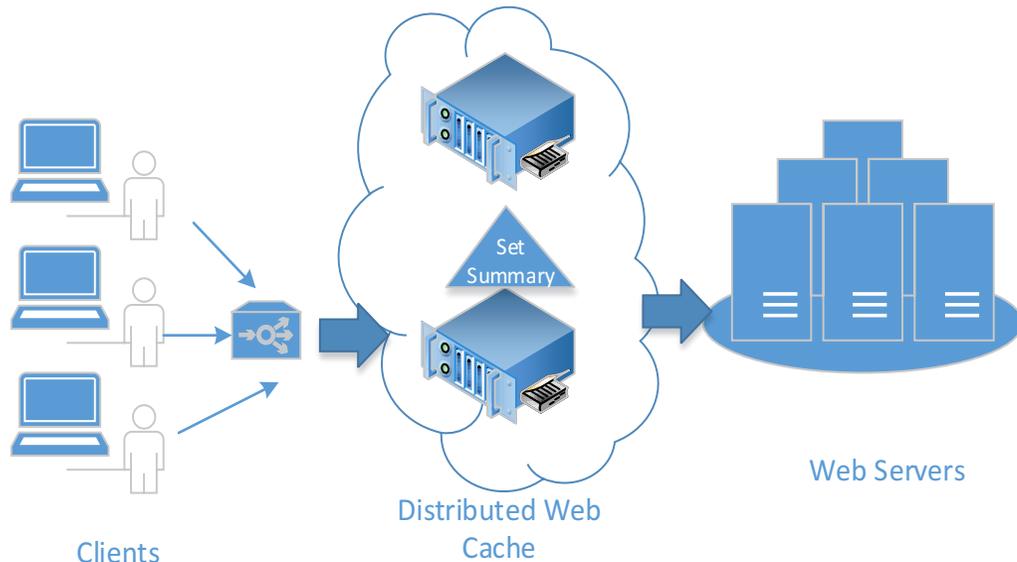
Wildcard Classification

Heavy Hitters Flows Detection

100's of usages for Membership Library in Wide Range of Applications

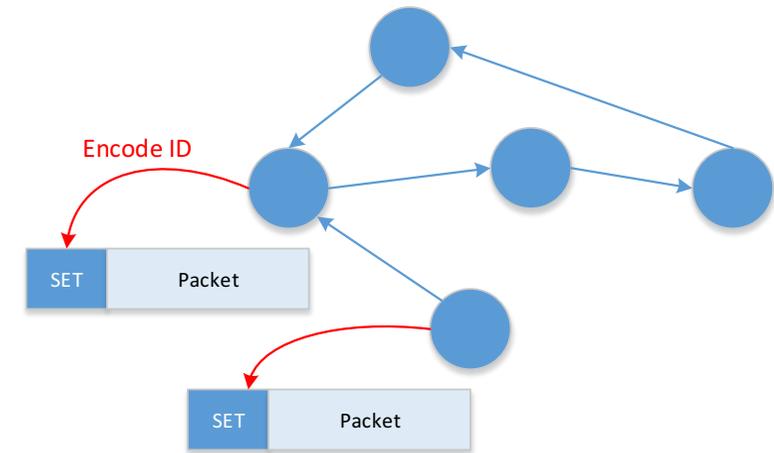
Library Usages? 100's too many to list

Distributed Web Cache



- Web proxies consult set summaries for each http request.
- Element membership in the set-summary will determine response location.
- For element hits, requests directed to a near cache and misses are forwarded to backend web servers.

Routing Loop Detection and/or Network Statistics

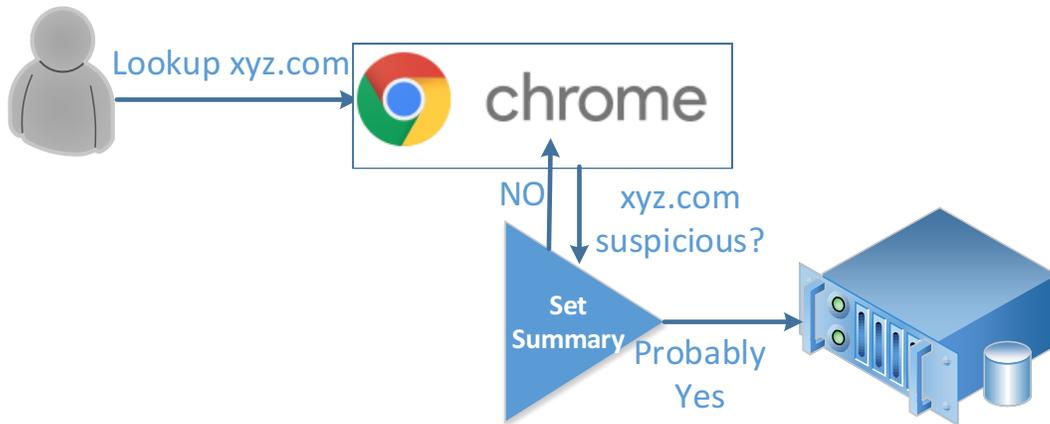


- Node ID's encoded in embedded set-summaries in the packet header
- Instead of waiting for slow TTL, node checks membership in set-summaries. Misses indicate loop-free routing.
- Idea can be generalized (for e.g. heavy hitters detection, ..etc.) to wide range of network stats.

Library Usages? 100's too many to list

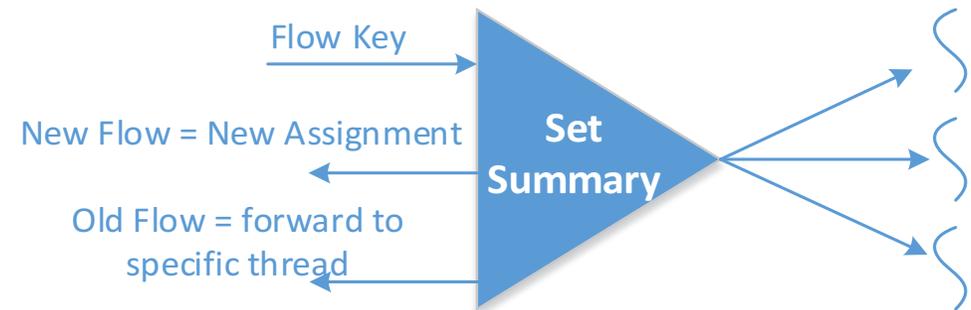


Safe Browsing and/or Signature Matching



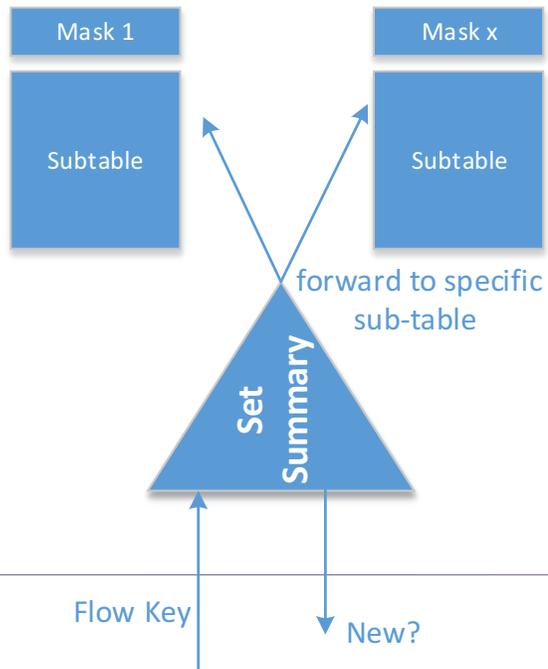
- URLs membership checked against suspicious set-summary and misses indicate safe.
- Same idea is applied in many signature matching IDS and deep packet inspection.

TCP Connection Tracker



- Flow keys membership tested and misses indicate new flows.
- Hits are forwarded to worker thread for in-order processing.

Library Usages? 100's too many to list



ACL & Wild Card Flow Classification

- **Flow keys membership results are used to optimize search for wild card match**

➔ Later Slides: Results of applying concept to OvS

Membership Library API



Set-Summary Create

```
rte_member_create(rte_member_parameters);
```

Parameters: Type, num_of_keys, key_length, number_of_sets, max_fp_rate, ..etc.

Set-Summary Element Insertion

```
rte_member_add (*set_sum, *key, set_id);
```

Insert a key into a set_summary data structure and the value is pointing to a specific set_id.

Set-Summary Element Lookup

1. *rte_member_lookup(*setsum, *key, *set_id)*
2. *rte_member_lookup_bulk(*setsum, *keys, *set_ids)*
3. *rte_member_lookup_multi(*setsum, *key, max_match_per_key, ..)*
4. *rte_member_lookup_multi_bulk(*setsum, *keys, ..);*

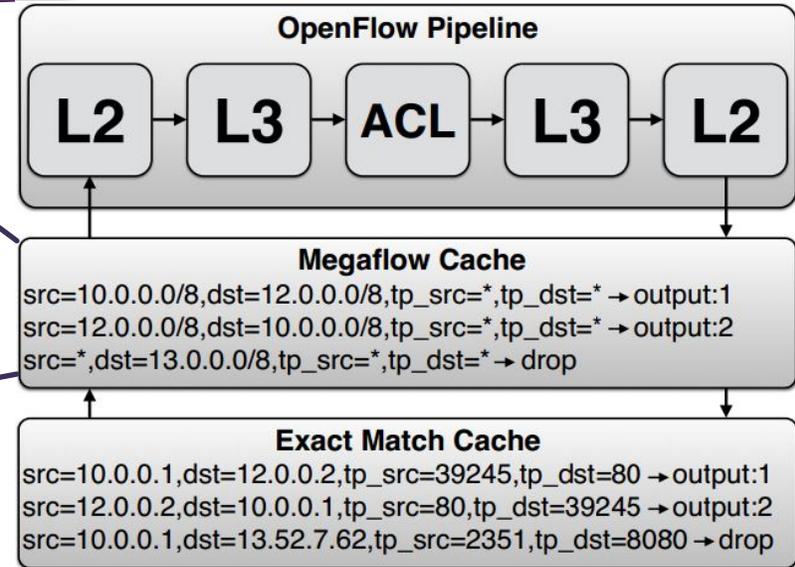
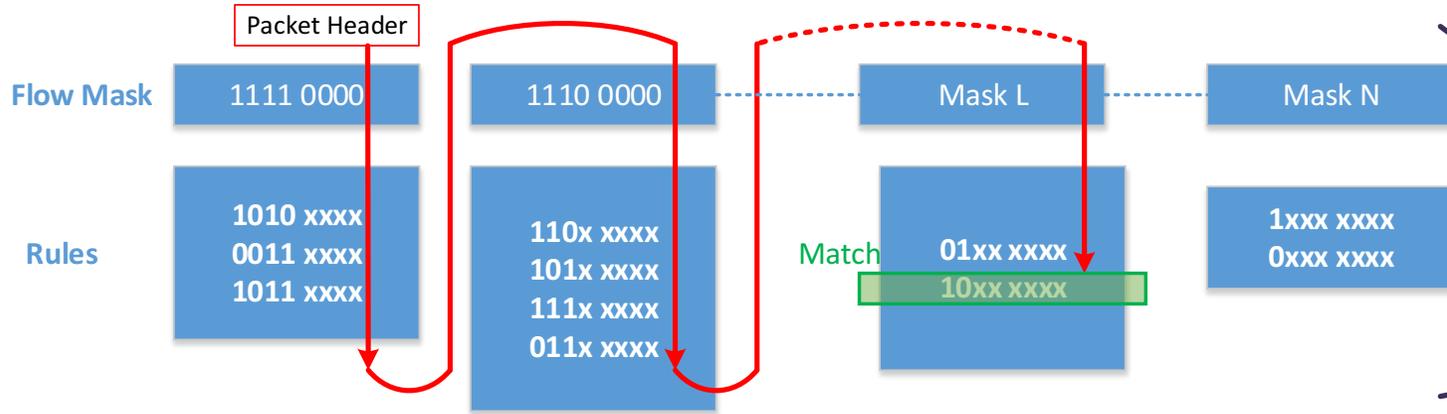
A single key or a bulk of key lookup, return the first match or up to max matches per key

Set-Summary Element Delete

```
rte_member_delete(*setsum, *key, set_id)
```

Delete a single key from a given set.
Not all modes (e.g. vBF) support delete in current implementation.

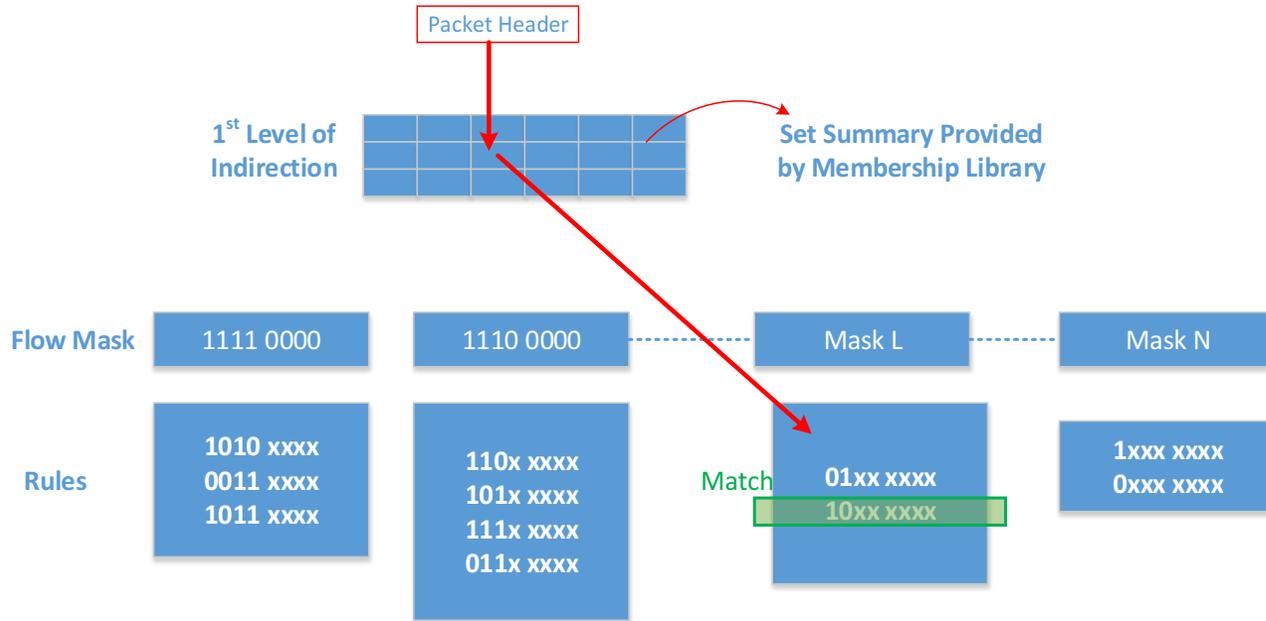
POC: Open vSwitch Flow Lookup



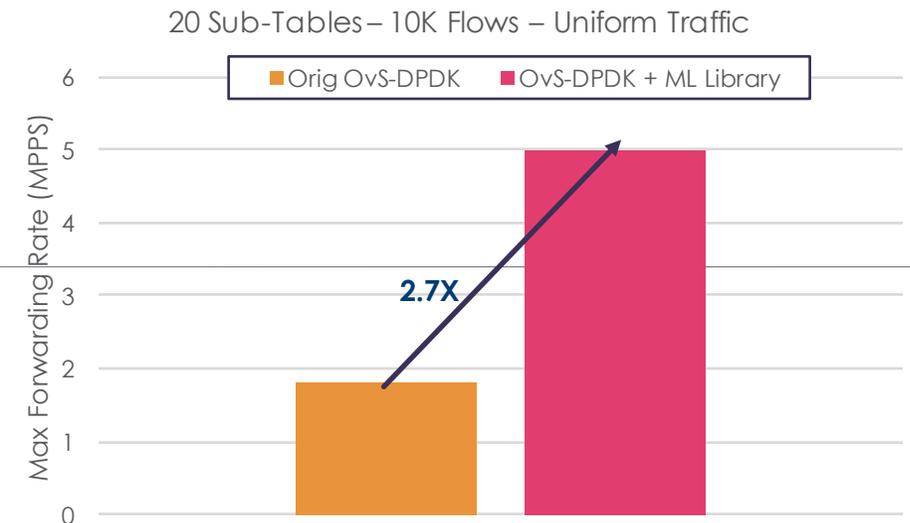
1. Set of disjoint sub-table
2. Rule is only inserted into one sub-table (lookup terminates after first match)
3. Lookup is done by sequentially search each sub-table until a match is found

▼ fast_path_processing	54.3%
▼ dpcls_lookup	53.6%
▶ netdev_flow_key_hash_in_mask	39.5%
▶ dpcls_rule_matches_key	7.1%
▶ zero_rightmost_1bit	0.0%
▶ pvector_cursor_next	0.0%

OVS with Two Layer Lookup using Membership Library



- Membership library used to create a 1st level set-summary indirection
- Flow Keys are looked up in set-summaries:
 - Hits: directs to the correct sub-table for searching (correct 97%)
 - Misses: "New" flow default sequential search & upcall if needed



Intel(R) Xeon(R) CPU E5-2699 v4 @ 2.20GHz
Hyper-Threading: disabled

2X-3X Throughput Improvement for OvS using DPDK Membership Library

- Applying Membership Library Optimization to other workloads.
 - Any Partners with huge list of Object ??
- Currently Working on ACL Library with high update rate based on tuple-search algorithm

Legal Disclaimers



No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.

Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.

This document contains information on products, services and/or processes in development. All information provided here is subject to change without notice. Contact your Intel representative to obtain the latest forecast, schedule, specifications and roadmaps.

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.

© 2017 Intel Corporation. Intel, the Intel logo, Intel. Experience What's Inside, and the Intel. Experience What's Inside logo are trademarks of Intel. Corporation in the U.S. and/or other countries.

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

Questions?

Sameh Gobriel

sameh.gobriel@intel.com

Charlie Tai

charlie.tai@intel.com