

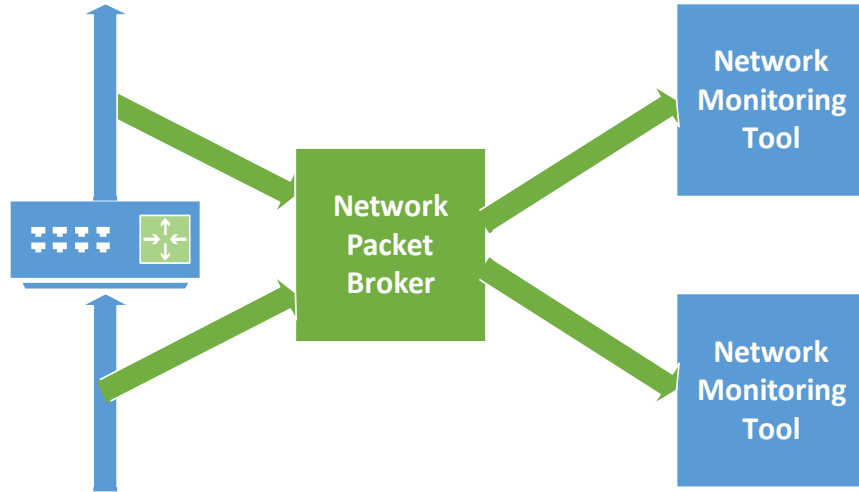


Packet Deduplication in P₄

Stefan Johansson
FPGA Architect
Keysight Technologies

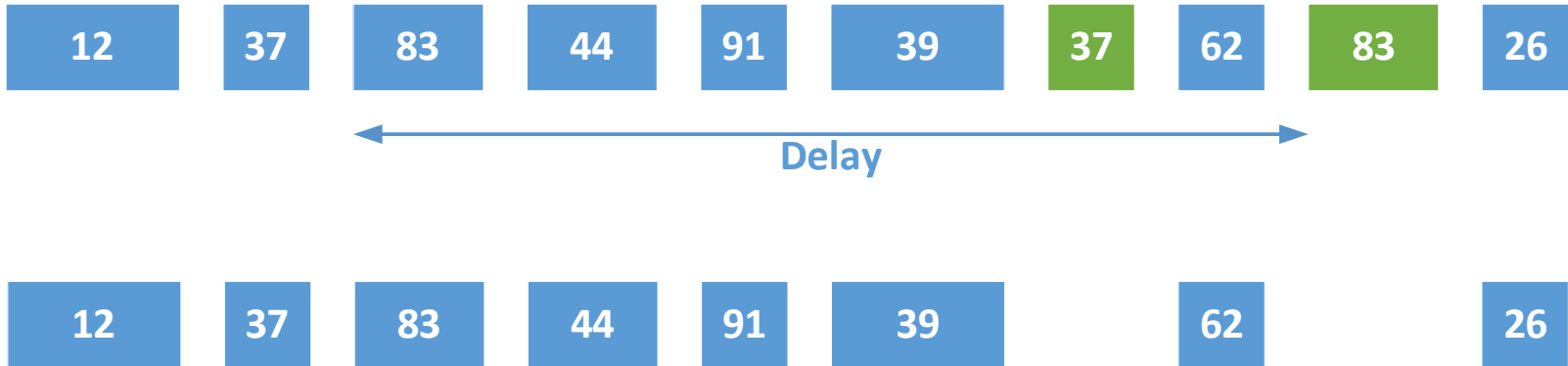
Network Packet Broker

- Forwards mirrored traffic from network to tools



Network Packet Broker

- Packet brokers often receive multiple copies of a packet
- Removing duplicates will reduce load on the tools



Requirements

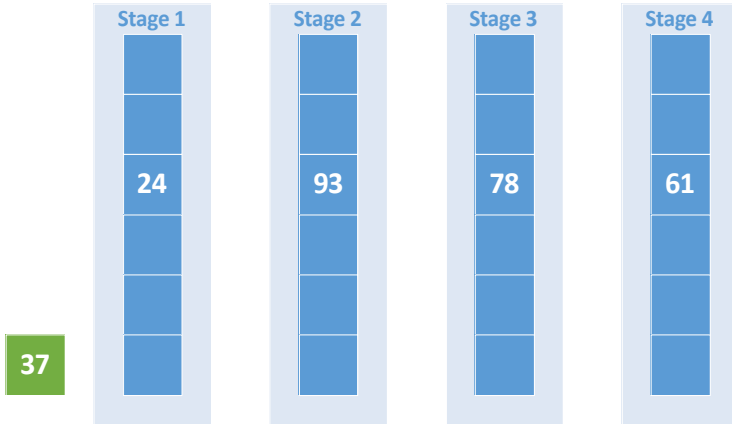
- Calculate signature for every packet
- Store sliding window of signatures
- Fast lookups
- Update table (add and remove) with every packet
- Updates done in user plane, control plane software too slow

P4 Implementation

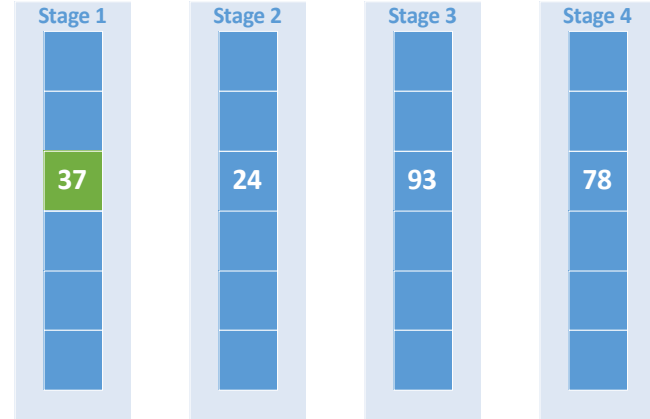
- Use stateful RAMs (registers) to implement hash table
- Multiple columns to reduce effect of hash collision
- Single lookup per stage, use one stage per column
- Sliding window achieved by shifting whole row when adding new entry

Hash Table Operation

Initial State

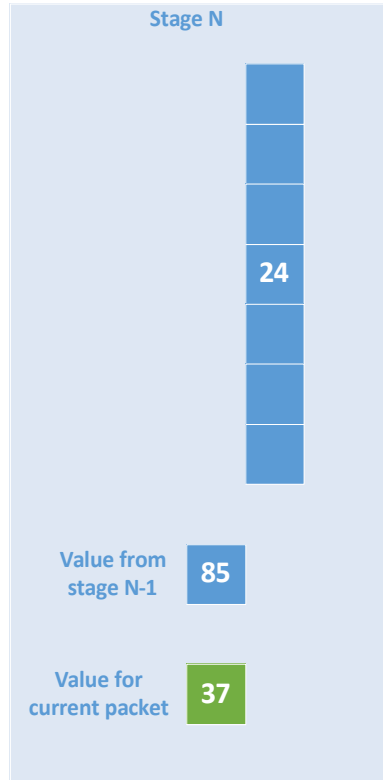


After Update

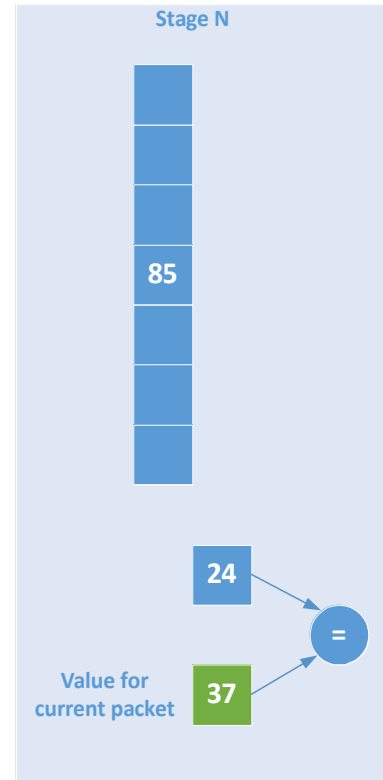


Hash Table Stage N

Initial State



Update and compare



P4 Code

- Table update

```
/* Stage 2 */
/* Hash table 32-bits x 64k entries */
Register<bit<32>, bit<32>>(65536) dedup_table_2;

RegisterAction<bit<32>, bit<32>, bit<32>>(dedup_table_2) dedup_update_2 = {
    void apply(inout bit<32> register_data, out bit<32> key_2) {
        key_2 = register_data;
        register_data = key_1;
    }
};
```

- Compare

```
/* Dedup Stage 2 */
key_2 = dedup_update_2.execute(dedup_idx);
if (key_2 == dedup_key) {
    dedup_drop = 1;
};
```


Results

- Hash table
 - 64 K rows x Four columns = 256 K entries

Packet Delay	Duplicates Removed
40,000	~ 99%
80,000	~ 95%



Thank You

Stefan Johansson
stefan.johansson@keysight.com