

# UNROLLING THE LOOP

## ELASTIC END-TO-END REDUNDANCY (EE2ER)

*#OneStepAhead*

# UNROLLING THE LOOP

## TABLE OF CONTENT

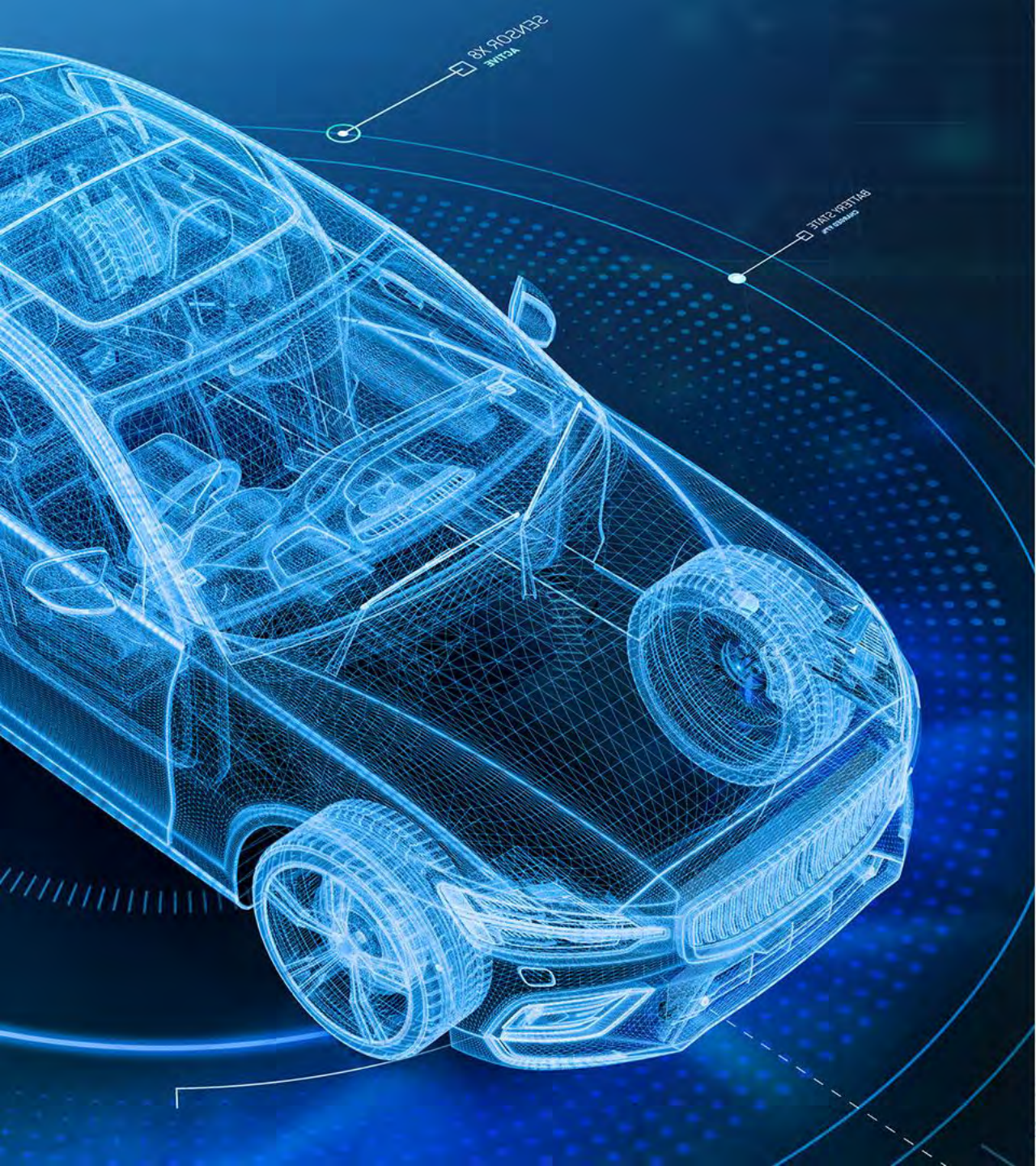
#1 | INTRODUCTION

#2 | SOLUTION

#3 | IMPLEMENTATION

#4 | SUMMARY AND NEXT STEPS





#1

UNROLLING  
THE LOOP

INTRODUCTION



# INTRODUCTION

## WHY IS EVERYONE TALKING ABOUT LOOPS?

- Safety Use cases with high availability goals exist:
  - Break-by-wire, steer-by-wire, autonomous driving, etc.
  - A system-level approach for redundancy is taken.
  - Often mixed communication technology.
- Ethernet-based redundancy can have benefits:
  - High bandwidth and lower latency allows more flexibility for SDV.
- Many approaches to redundancy on Ethernet exist already!
  - But do they solve the problem?

# INTRODUCTION

## WHAT IS THE STATE OF THE ART?

|   | Strategy? | Reaction in? | Redundancy? | Detection of degradation? | Single Point of Failure | Cost Add |
|---|-----------|--------------|-------------|---------------------------|-------------------------|----------|
| Spanning Tree Protocol (STP, RSTP, ...) | Reactive  | s – ms       | Limited     | Missing                   | Yes                     | Low      |
| Ring protocols (e.g., ERPS)             | Reactive  | ms           | Limited     | Missing                   | Yes                     | Low      |
| FRER (802.1CB)                          | Proactive | 0            | Limited     | Missing                   | Yes                     | Low      |
| Two independent Ethernet networks       | Proactive | 0            | Full        | Implicit                  | No                      | High     |

- Can you guarantee a reactive solution in critical scenarios?
- Redundancy needs to be “end-to-end” for cost-effective safety.
- Applications need to know the current state of redundancy.
- Running two Ethernet networks is too expensive!
- Can we create a more cost-effective solution?



# INTRODUCTION

## WISH LIST FOR ELASTIC END-TO-END REDUNDANCY (EE2ER)

- What is needed for EE2ER as end-to-end redundancy solution:
  - Scalable: allowing “no single point of failure” in designs, if needed.
  - Proactive redundancy with no reaction time.
  - Monitoring of current state via E2E protection as proven mechanism.
  - Switches and intermediate systems with no ASIL requirements.
  - No changes to today’s semiconductors necessary.
  - Only minimal added cost.
- Ethernet ring between HPCs and Zonals?
- Can we achieve this?

# #2

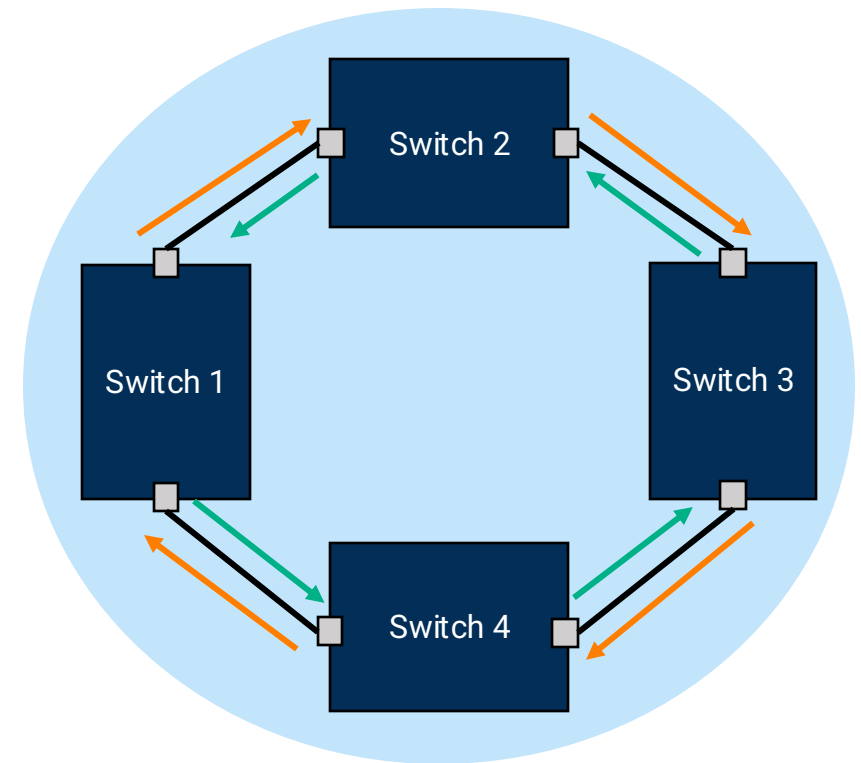
## UNROLLING THE LOOP SOLUTION



```
(groupsalloc);  
EXPORTSYMBOL(groupsalloc);  
void groups_free(struct group_info *group_info)  
{  
    void groups_free(struct group_info *group_info)  
    if (groupinfo->blocks[0] != group_info->small_block) {  
        int i;  
        if (groupinfo->blocks[0] != group_info->small_block) {  
            for (i = 0; i < group_info->nblocks; i++)  
                freepage((unsigned long)groupinfo->blocks[i]);  
            for (i = 0; i < group_info->nblocks; i++)  
                freepage((unsigned long)groupinfo->blocks[i]);  
            kfree(groupinfo);  
            kfree(groupinfo);  
        }  
    }  
    EXPORTSYMBOL(groupsfree);  
    /* export the groupinfo to a user-space array */  
    /* export the groupinfo to a user-space array */  
    static int groups_touser(gid_t_user *grouplist,  
        const struct group_info *group_info)  
    {  
        const struct group_info *group_info)  
        int i;  
        {  
            unsigned int count = groupinfo->nblocks;  
            int i;  
            unsigned int count = groupinfo->nblocks;  
            for (i = 0; i < group_info->nblocks; i++) {  
                unsigned int cpcount = min(NGROUPSPERBLOCK, count);  
                for (i = 0; i < group_info->nblocks; i++) {  
                    unsigned int len = cpcount * sizeof(*grouplist);  
                    unsigned int cpcount = min(NGROUPSPERBLOCK, count);  
                    unsigned int len = cpcount * sizeof(*grouplist);  
                    if (copyto_user(grouplist, group_info->blocks[i], len))  
                        return -EFAULT;  
                    if (copyto_user(grouplist, group_info->blocks[i], len))  
                        return -EFAULT;  
                }  
            }  
        }  
    }  
}
```

# SOLUTION

## BASIC APPROACH.

- Ring with traffic in both directions.
- Applications send data redundant and monitor all streams via E2E protection:
  - No reaction time.
  - No reaction mechanisms that can fail.
  - Proven mechanism for safety.
  - Minimal ASIL requirements (=cost) for switches.
- Assumed traffic:
  - Multicast Signal-PDUs (CAN-like).
  - Multicast SOME/IP-based.
  - Only a few messages need redundancy.



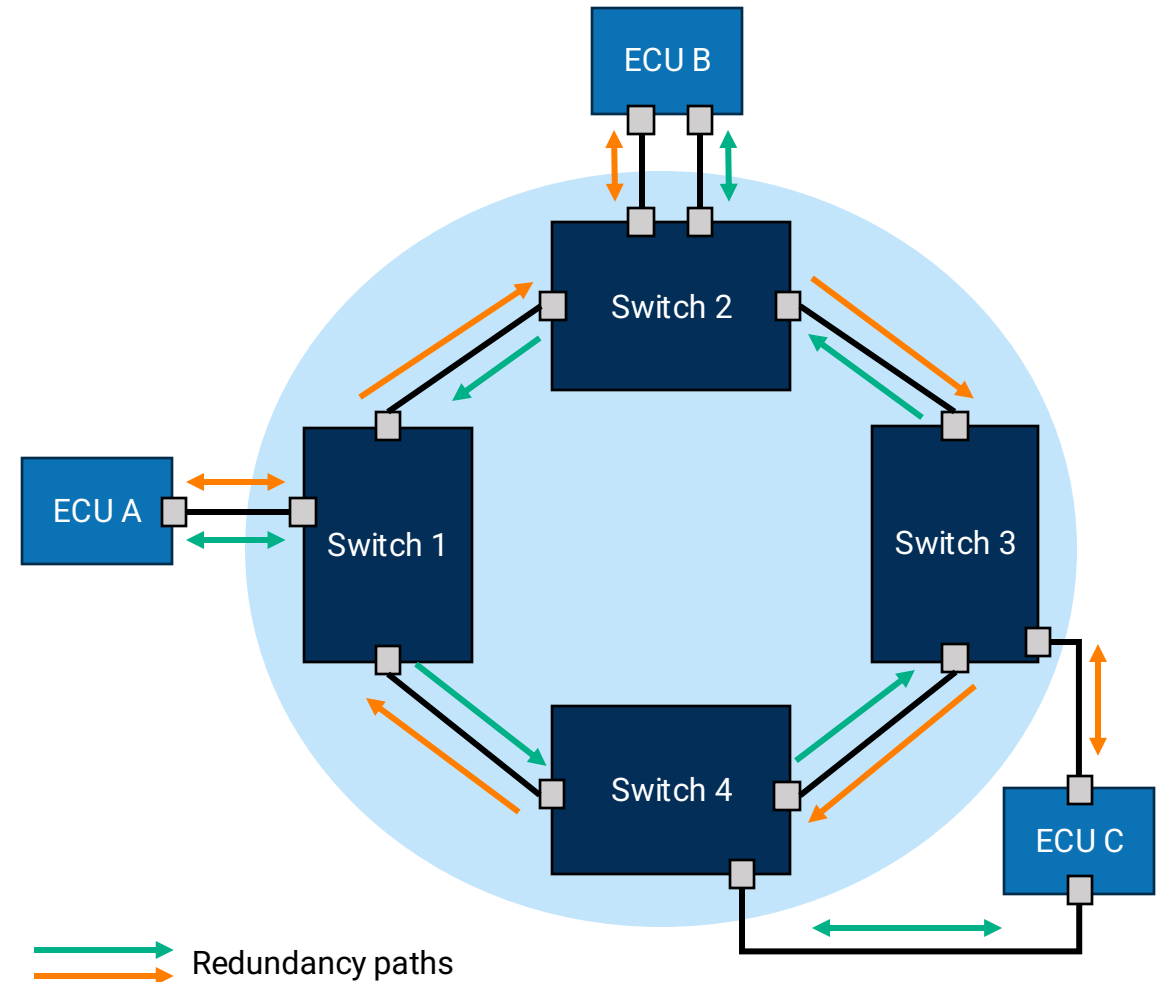
  Redundancy paths



# SOLUTION

## WHY A RING?

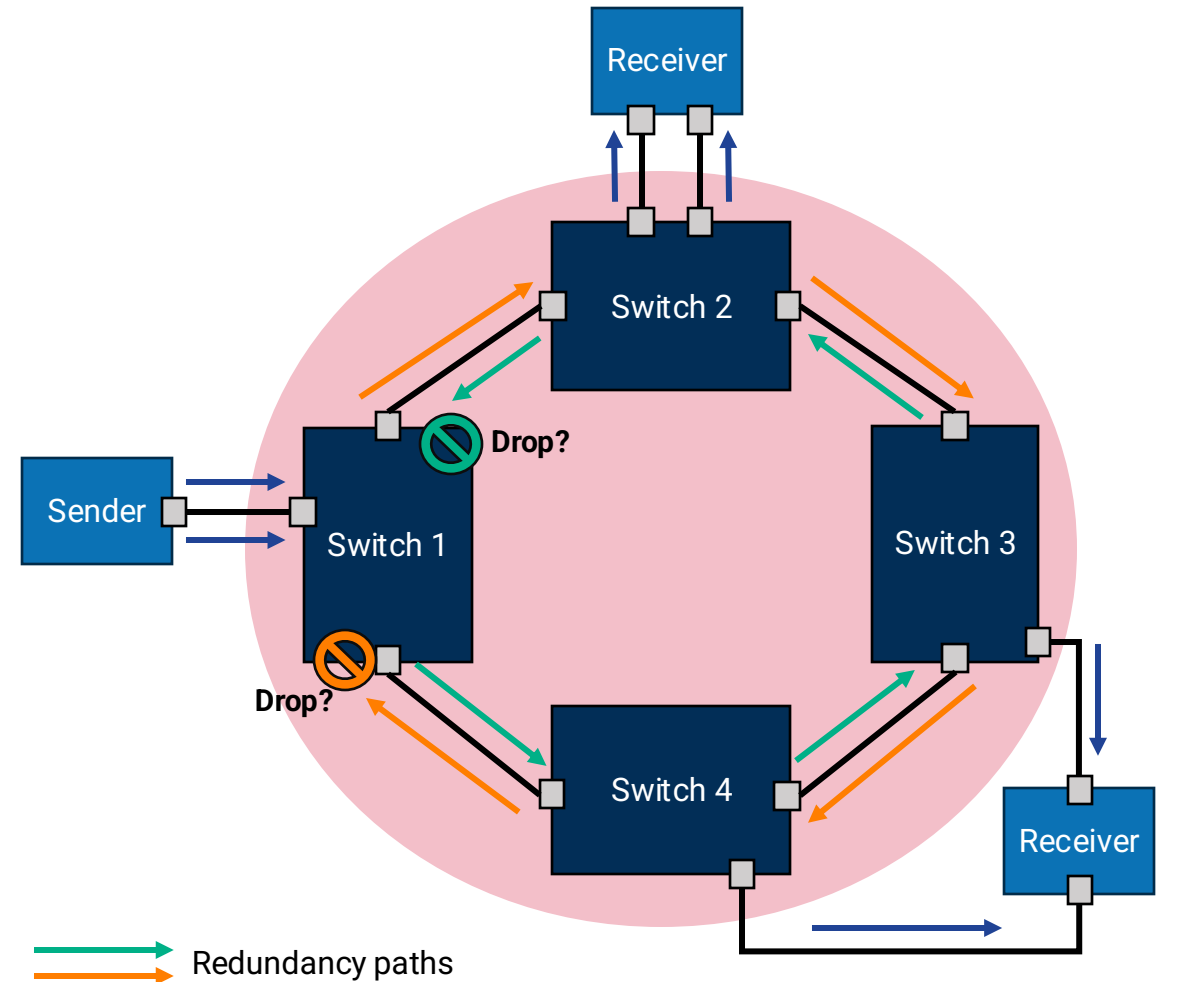
- A ring can avoid single point of failure.
- Different redundancy levels possible:
  - ECU A: attached to single Ring Switch.
  - ECU B: attached to single Ring Switch twice.
  - ECU C: attached to two Ring Switches.
- We can add regular traffic as today
  - Single direction on ring.
- Cost-effective design possible!



# SOLUTION

## ETHERNET VS. THE LOOP!

- We just created a loop!
- How to avoid the loop?
  - Just drop frames before they loop.
- Possible options:
  - Drop by source MAC Address.
  - VLAN-ID-based Hop Counter.
  - Drop by VLAN ID on specific input port.

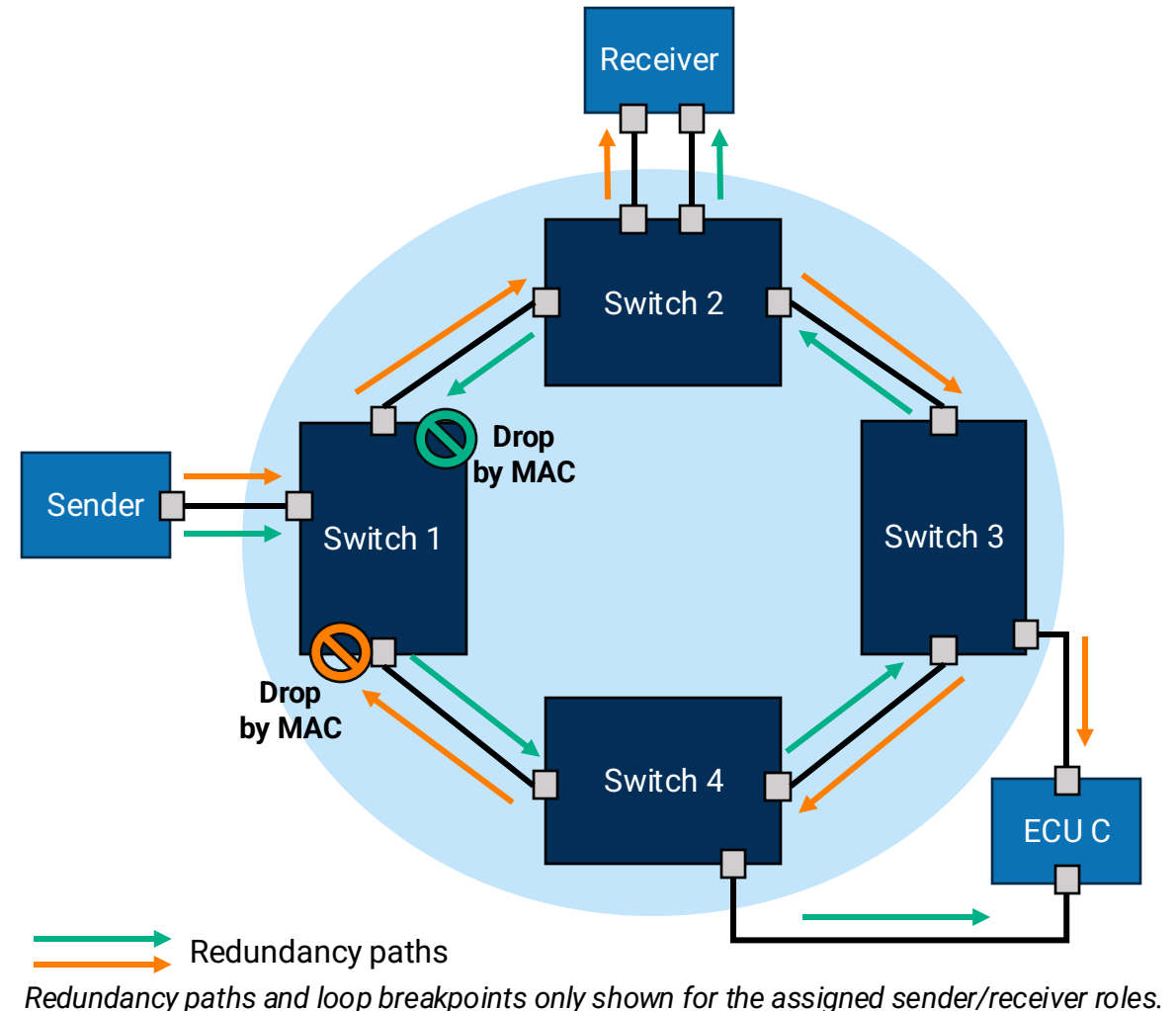


*Redundancy paths and loop breakpoints only shown for the assigned sender/receiver roles.*

# SOLUTION

## APPROACH 1: DROP BY SOURCE MAC ADDRESS.

- Approach:
  - Flood Multicast in two VLANs.
  - Drop frames via Source MAC to cut loop.
- Pros:
  - Simple configuration.
- Cons:
  - MAC addresses need to be known a-prior.
  - Large number of filtering rules.
  - Different configuration for every switch.
  - Topology change requires config change.
- A Hop Counter could solve this!?

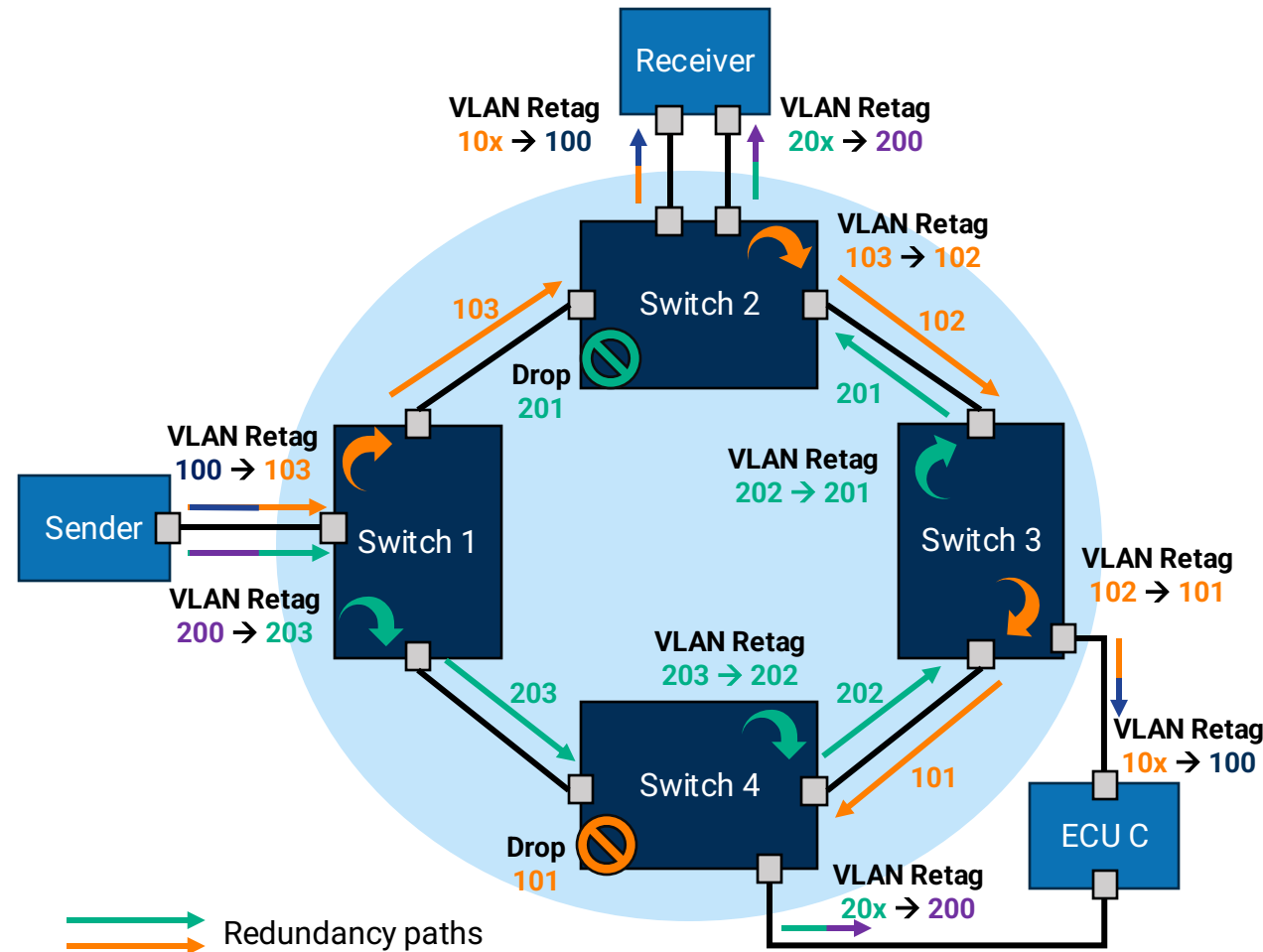




# SOLUTION

## APPROACH 2: VLAN-ID-BASED HOP COUNTER.

- Approach
  - Send with starting hop counter.
  - VLAN-ID is decremented at every hop
  - Drop, when Hop Counter would go to “0”.
- Pros
  - Fixed number of VLAN-IDs.
  - Sender MAC Addresses do not matter.
- Cons
  - Config depends on number of switches!
- Can we further improve this?

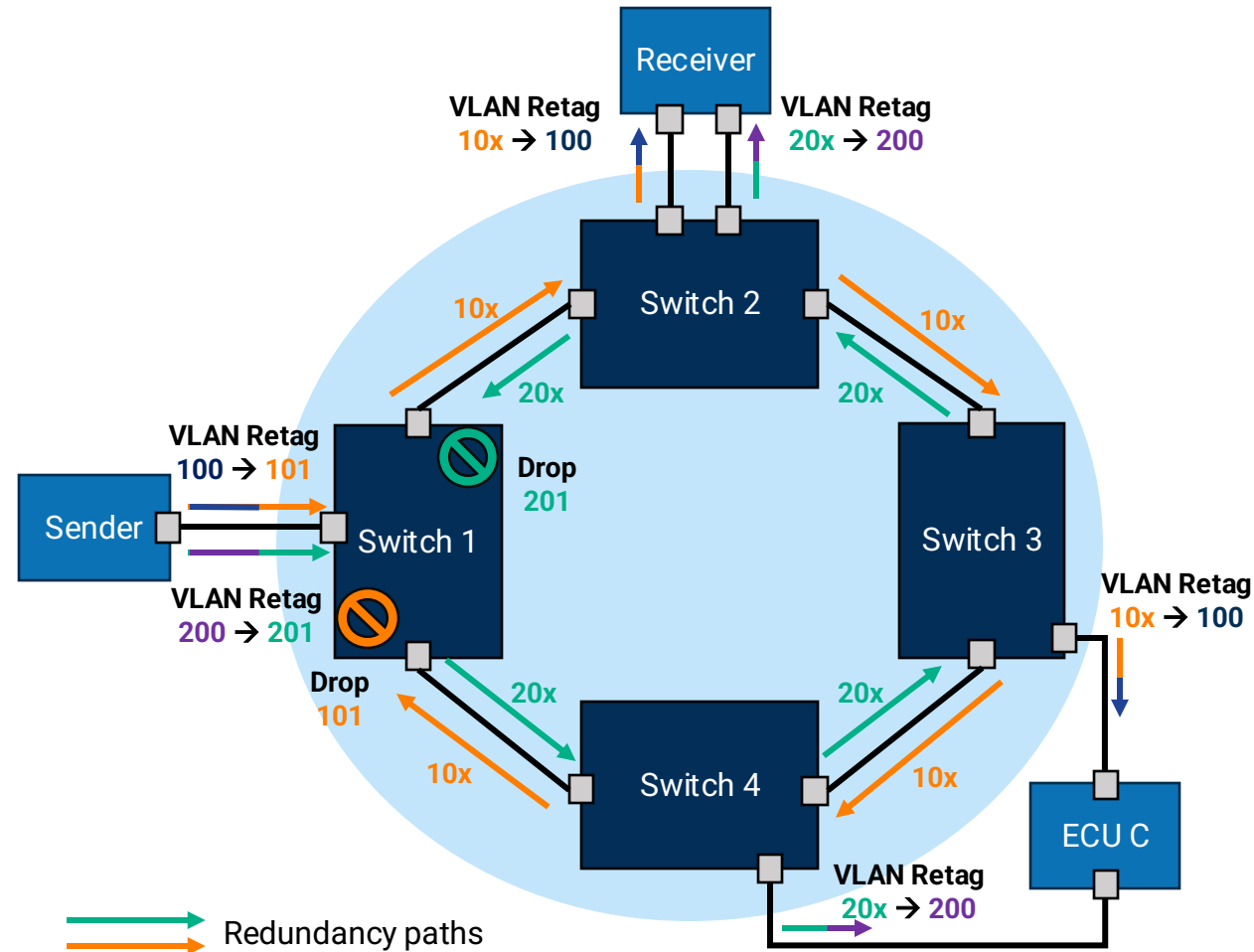


Redundancy paths and loop breakpoints only shown for the assigned sender/receiver roles.  
VLAN retagging performed always on switch egress as specified in OA TC11.

# SOLUTION

## APPROACH 3: ENCODE INPUT PORT IN VLAN ID.

- Approach
  - Virtual rings per source switch.
  - Define VLAN ranges per direction: 10x, 20x
  - Unique VLAN-ID per entrance into ring.
  - Switch drops its VLAN-ID, when receiving.
- Pros
  - Ring size can be scaled afterwards!
  - VLAN retagging only at enter/exit of ring.
- Cons
  - Different config per switch.



Redundancy paths and loop breakpoints only shown for the assigned sender/receiver roles.  
VLAN retagging performed always on switch egress as specified in OA TC11.

# SOLUTION

## CONCLUSION – SO FAR ...

- A redundant ring with today's hardware seems possible.
- We choose Approach 3 for EE2ER:
  - Allows us to build a redundancy ring.
  - Supports vehicle variant management (most flexible).
  - Best approach found.
- Two main questions remain:
  - Does this work on real hardware?
  - Can we reduce the added configuration complexity?





# #3

## UNROLLING THE LOOP

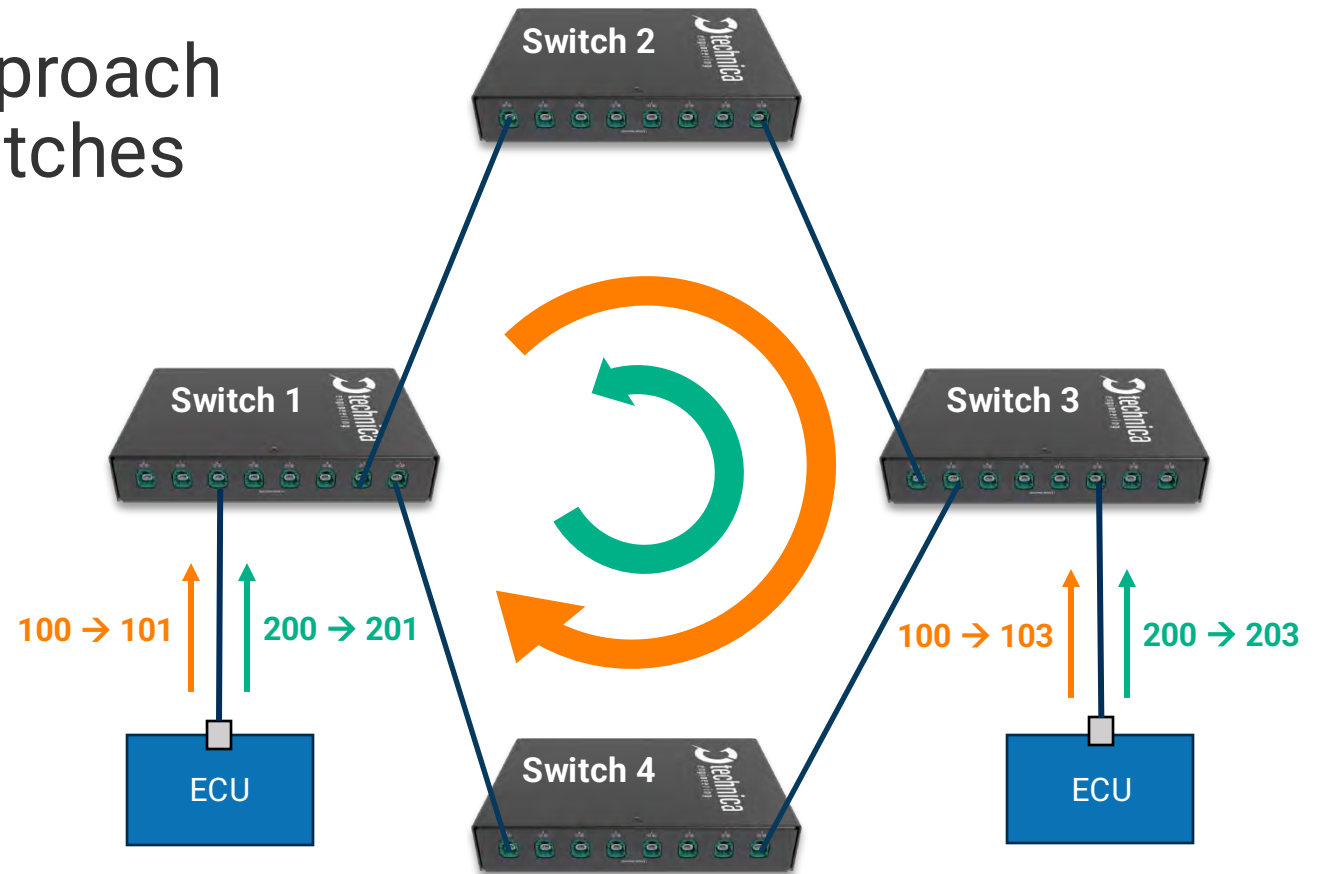
### IMPLEMENTATION

# IMPLEMENTATION

## OVERVIEW

We have implemented our approach with standard automotive switches

- No new hardware necessary!

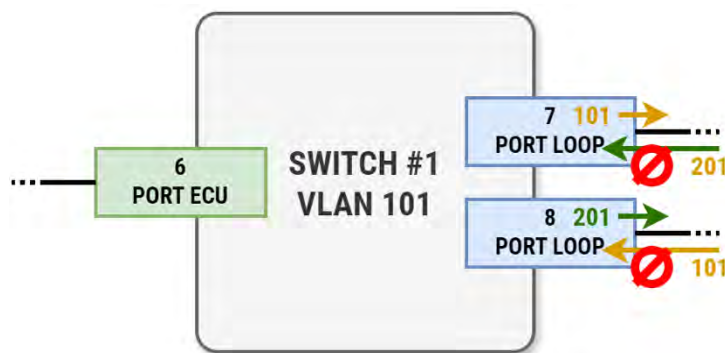


# IMPLEMENTATION

## CUTTING THE LOOP BY DROPPING VLAN – EXAMPLE ON SWITCH 1

### VLAN cuts the loop

- Drop is done via regular VLAN membership



PORT SEGMENTATION | **VLAN CONFIGURATION** | TIME SYNC | ADVANCED FILTER | TRAFFIC SHAPING | INGRESS RATE LIMITER | ADDRESS TABLE

IEEE 802.1q mode (Inner VLAN) | Double Tagging (Outer VLAN) QINQ

+ ADD NEW VLAN

| VLAN ID | BaseT1-1  | BaseT1-2  | BaseT1-3  | BaseT1-4  | BaseT1-5   | BaseT1-6   | BaseT1-7   | BaseT1-8   | SFP+ A  | SFP+ B  | Action |
|---------|---|---|---|---|--|--|--|--|---|---|--------|
|         | Ingress Policy All<br>Egress Policy Membership<br>Default VLAN ID 2 | Ingress Policy All<br>Egress Policy Membership<br>Default VLAN ID 2 | Ingress Policy All<br>Egress Policy Membership<br>Default VLAN ID 2 | Ingress Policy All<br>Egress Policy Membership<br>Default VLAN ID 2 | Ingress Policy Membership<br>Egress Policy Membership<br>Default VLAN ID 2 | Ingress Policy Membership<br>Egress Policy Membership<br>Default VLAN ID 2 | Ingress Policy All but exclusion<br>Egress Policy All but exclusion<br>Default VLAN ID 2 | Ingress Policy All but exclusion<br>Egress Policy All but exclusion<br>Default VLAN ID 2 | Ingress Policy All<br>Egress Policy Membership<br>Default VLAN ID 2 | Ingress Policy All<br>Egress Policy Membership<br>Default VLAN ID 2 |        |
| 100     |   |   |   |   |  |  |  |  |   |   |        |
| 101     |   |   |   |   |  |  |  |  |   |   |        |
| 102     |   |   |   |   |  |  |  |  |   |   |        |
| 103     |   |   |   |   |  |  |  |  |   |   |        |
| 104     |   |   |   |   |  |  |  |  |   |   |        |
| 200     |   |   |   |   |  |  |  |  |   |   |        |
| 201     |   |   |   |   |  |  |  |  |   |   |        |
| 202     |   |   |   |   |  |  |  |  |   |   |        |
| 203     |   |   |   |   |  |  |  |  |   |   |        |
| 204     |   |   |   |   |  |  |  |  |   |   |        |

# VLANs: 10

CLEAR ALL | EXPORT | IMPORT

Not Allowed | Tag Removed | Tag Normalized | Tag as Received

CLOSE | SAVE

**The VLAN policy accepts all frames except those with VLAN 201 in this port**

**The VLAN policy accepts all frames except those with VLAN 101 in this port**

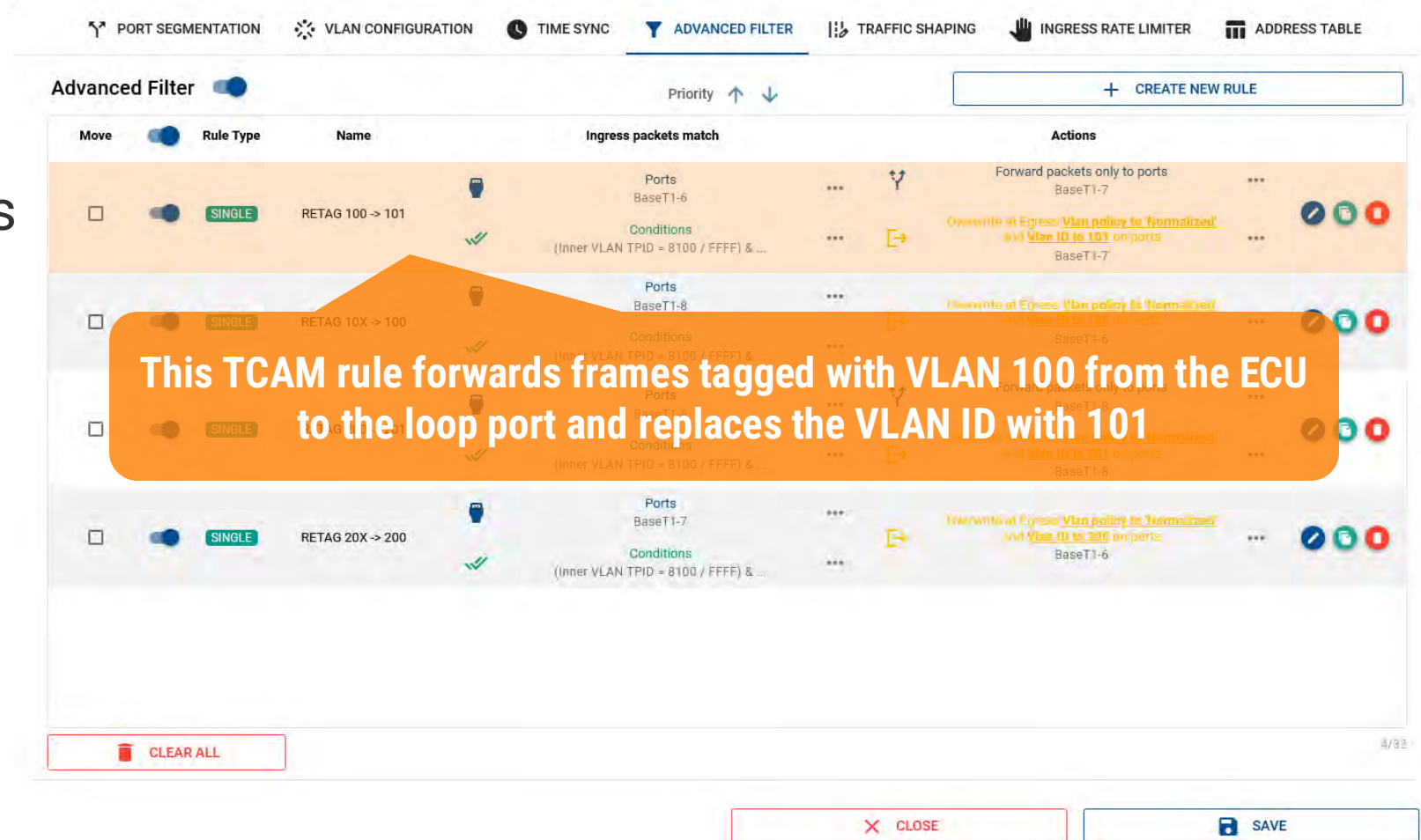
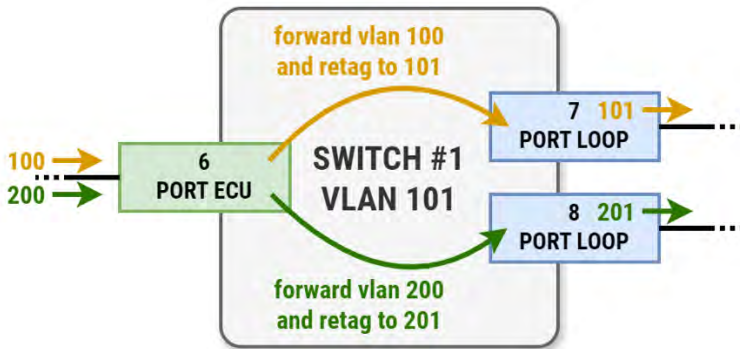


# IMPLEMENTATION

## VLAN RETAGGING AT RING ENTRANCE – EXAMPLE ON SWITCH 1

### Entering the ring

- The TCAM module retags the VLAN from 0x100 to 0x101 at egress

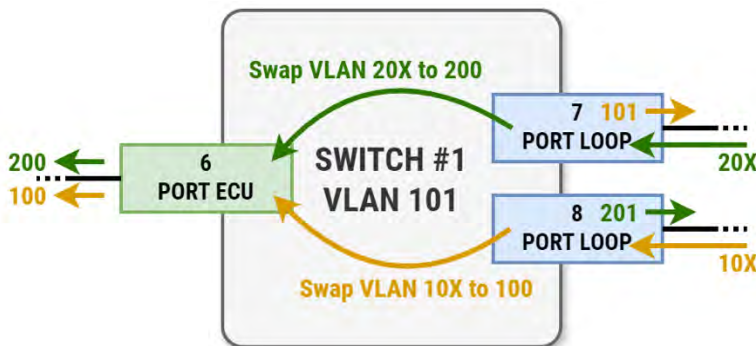


# IMPLEMENTATION

## VLAN RETAGGING AT RING EGRESS – EXAMPLE ON SWITCH 1

### Leaving the ring

- The TCAM module retags VLANs from 0x10X to 0x101 at egress
- This is done via a mask
- In real vehicles you would use binary and not decimal ranges



PORT SEGMENTATION VLAN CONFIGURATION TIME SYNC **ADVANCED FILTER** TRAFFIC SHAPING INGRESS RATE LIMITER ADDRESS TABLE

Advanced Filter ☒ Priority

| Move                     | Rule Type                                  | Name             | Ingress packets match  | Actions  |
|--------------------------|--|------------------|--|--|
| <input type="checkbox"/> | <input checked="" type="checkbox"/> SINGLE | RETAG 100 -> 101 | Ports<br>BaseT1-6<br>Conditions<br>(Inner VLAN TPID = 8100 / FFFF) & ... | Forward packets only to ports<br>BaseT1-7<br>Overwrite at Egress Vlan policy to 'Normalized' and Vlan ID to 101 on ports<br>BaseT1-7 |
| <input type="checkbox"/> | <input checked="" type="checkbox"/> SINGLE | RETAG 10X -> 100 | Ports<br>BaseT1-8<br>Conditions<br>(Inner VLAN TPID = 8100 / FFFF) & ... | Overwrite at Egress Vlan policy to 'Normalized' and Vlan ID to 100 on ports<br>BaseT1-6  |
| <input type="checkbox"/> | <input checked="" type="checkbox"/> SINGLE | RETAG 200 -> 201 | Ports<br>BaseT1-6<br>Conditions  | Forward packets only to ports<br>BaseT1-8<br>Overwrite at Egress Vlan policy to 'Normalized' and Vlan ID to 201 on ports<br>BaseT1-6 |
| <input type="checkbox"/> | <input checked="" type="checkbox"/> SINGLE | RETAG 10X -> 101 | Ports<br>BaseT1-6<br>Conditions  | Forward packets only to ports<br>BaseT1-8<br>Overwrite at Egress Vlan policy to 'Normalized' and Vlan ID to 101 on ports<br>BaseT1-6 |

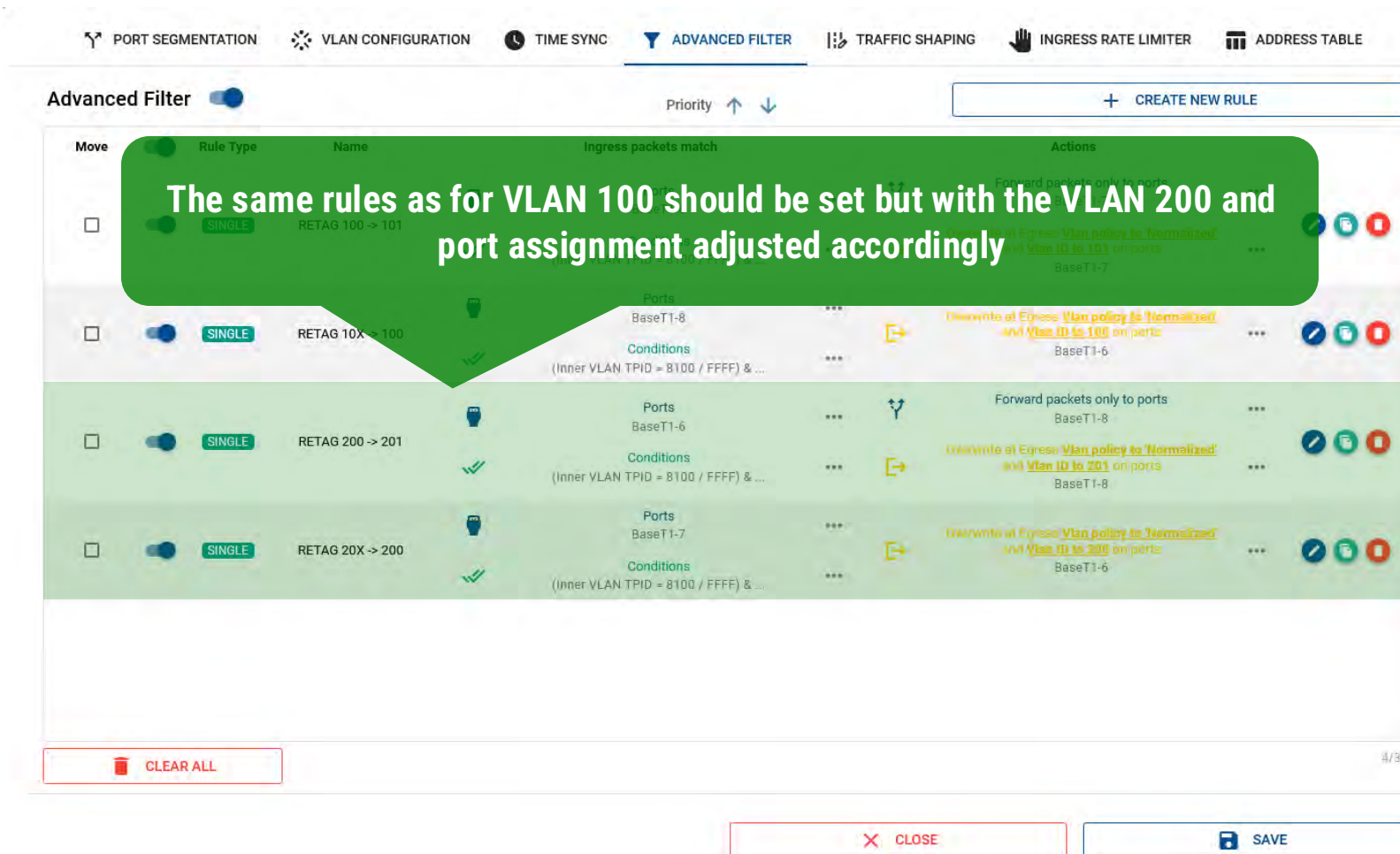
This TCAM rule retags the VLAN to 0x100 of any frame entering through the loop port with VLAN 0x10X, but only when it egresses through the port that is connected to the ECU

# IMPLEMENTATION

## ADDING THE OTHER DIRECTION TOO – EXAMPLE ON SWITCH 1

Two more rules to control the other direction of the loop

And yes, it works...



Advanced Filter

Priority ↑ ↓

+ CREATE NEW RULE

The same rules as for VLAN 100 should be set but with the VLAN 200 and port assignment adjusted accordingly

| Move                     | Rule Type | Name             | Ingress packets match                               | Actions   |
|--------------------------|-----------|------------------|---|---|
| <input type="checkbox"/> | SINGLE    | RETAG 100 -> 101 | Ports<br>BaseT1-8                                   | Forward packets only to port<br>BaseT1-8  |
| <input type="checkbox"/> | SINGLE    | RETAG 10X -> 100 | Conditions<br>(Inner VLAN TPID = 8100 / FFFF) & ... | Overwrite at Egress Vlan policy to 'Normalized' and Vlan ID to 101 on ports<br>BaseT1-6 |
| <input type="checkbox"/> | SINGLE    | RETAG 200 -> 201 | Ports<br>BaseT1-6                                   | Forward packets only to ports<br>BaseT1-8   |
| <input type="checkbox"/> | SINGLE    | RETAG 20X -> 200 | Conditions<br>(Inner VLAN TPID = 8100 / FFFF) & ... | Overwrite at Egress Vlan policy to 'Normalized' and Vlan ID to 201 on ports<br>BaseT1-6 |

CLEAR ALL

CLOSE

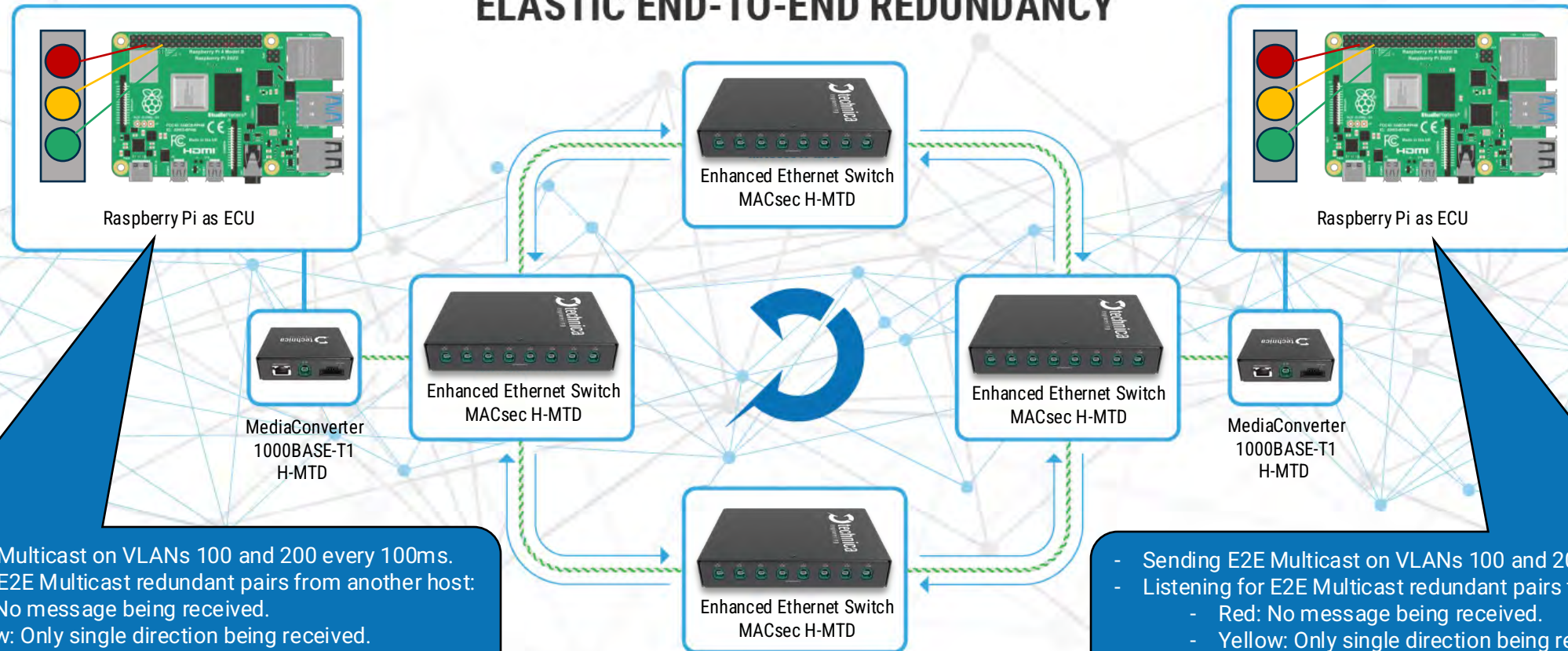
SAVE



# IMPLEMENTATION

VISIT OUR BOOTH TO EXPERIENCE EE2ER IN ACTION

## ELASTIC END-TO-END REDUNDANCY



- Sending E2E Multicast on VLANs 100 and 200 every 100ms.
- Listening for E2E Multicast redundant pairs from another host:
  - Red: No message being received.
  - Yellow: Only single direction being received.
  - Green: Redundancy active.

- Sending E2E Multicast on VLANs 100 and 200 every 100ms.
- Listening for E2E Multicast redundant pairs from another host:
  - Red: No message being received.
  - Yellow: Only single direction being received.
  - Green: Redundancy active.



# #4

## UNROLLING THE LOOP

SUMMARY / NEXT  
STEPS



# CONCLUSION AND NEXT STEPS

## SUMMARY

|   | Strategy?        | Reaction in? | Redundancy? | Detection of degradation? | Single Point of Failure | Cost Add   |
|---|------------------|--------------|-------------|---------------------------|-------------------------|------------|
| Spanning Tree Protocol (STP, RSTP, ...) | Reactive         | s – ms       | Limited     | Missing                   | Yes                     | Low        |
| Ring protocols (e.g., ERPS)             | Reactive         | ms           | Limited     | Missing                   | Yes                     | Low        |
| FRER (802.1CB)                          | Proactive        | 0            | Limited     | Missing                   | Yes                     | Low        |
| Two independent Ethernet networks       | Proactive        | 0            | Full        | Implicit                  | No                      | High       |
| <b>Elastic E2E Redundancy (EE2ER)</b>   | <b>Proactive</b> | <b>0</b>     | <b>Full</b> | <b>Via E2E</b>            | <b>No</b>               | <b>Low</b> |

- EE2ER fulfils the wish list and solves the problems of alternatives.
  - Combining the full end-to-end redundancy with minimum cost.
  - Runs on today's hardware.
- What about configuration complexity?



# CONCLUSION AND NEXT STEPS

## NEXT STEPS

- Configuration Toolchain needs to support EE2ER for ease of use:

|                           | Open Source? | Topology model? | Automotive? | SDV compatible? | EE2ER?     |
|---------------------------|--------------|-----------------|-------------|-----------------|------------|
| ARXML                     | Not allowed? | Incomplete      | Yes         | No              | TBD        |
| FIBEX                     | Possible     | Incomplete      | Yes         | No              | Unlikely   |
| YANG/NETCONF              | Yes          | Incomplete      | No          | Unclear         | Unlikely   |
| <b>Goal for toolchain</b> | <b>Yes</b>   | <b>Yes</b>      | <b>Yes</b>  | <b>Yes</b>      | <b>Yes</b> |

- Upcoming Automotive Switch Configuration Toolchain
  - Open Source and SDV compatible
  - With EE2ER support
- See the EE2ER demo at our booth and start the conversation!



## Technica Engineering GmbH

Leopoldstraße 236  
80807 Munich  
Germany

### IAGO ÁLVAREZ

Lead Engineer

[iago.alvarez@technica-engineering.de](mailto:iago.alvarez@technica-engineering.de)

### ROMÀ PAGÈS

Senior Software and Hardware Engineer

[roma.pages@technica-engineering.de](mailto:roma.pages@technica-engineering.de)

### DR. LARS VÖLKER

Technical Fellow

[lars.voelker@technica-engineering.de](mailto:lars.voelker@technica-engineering.de)