AVB for low latency / industrial networks:

Media redundancy for fault tolerance and AVB

Oliver Kleineberg, Hirschmann Automation & Control
Franz-Josef Goetz, Siemens AG Industrial Automation

IEEE 802 Plenary Meeting, November 2010, Dallas

Media redundancy and AVB

- **Aims of this Presentation:**

- Give an overview of common network topologies, application and redundancy use cases from low latency/industrial networks
- Give ideas on how to combine „common“ Layer 2 redundancy and AVB mechanisms to achieve determinisic failover times
- Is it possible to achieve protocol-neutral interoperability between arbitrary redundancy control protocols and AVB ? → Trigger discussions for future work items.

=> **The total network reconfiguration time in AVB** enabled low latency networks needs to be **pre-determinable**. It is an important requirement (for e.g. industrial communication.) This puts **additional requirements on interoperability** between AVB components and media redundancy protocols.

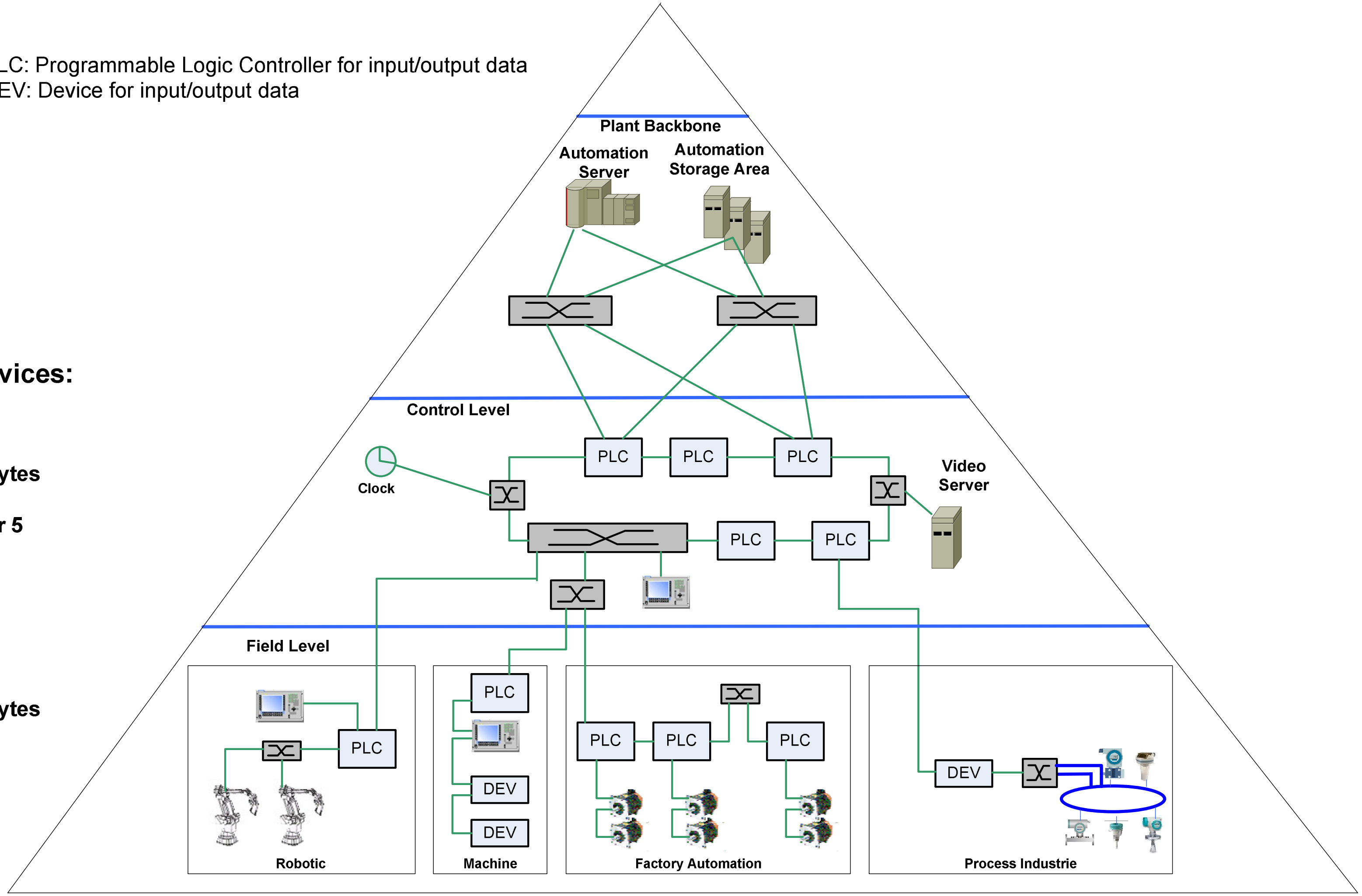
Agenda

- Use cases / „Insight into industrial automation“
  - Network topologies / latency and applications
  - Network topologies / fault tolerance
- Examples of fault tolerant topologies
- Media redundancy classification
- Short flashback: Network reconfiguration time
- Challenges for AVB and media redundancy

Insight into industrial automation: Latency, topologies and applications

Hierarchical network for industrial communication

PLC: Programmable Logic Controller for input/output data
DEV: Device for input/output data



Requirments for periodic real-time services:

Control Level

- **CyleTime: 1ms – 10ms**
- **Data Size / Node > 300 bytes**
- **Max Latency: < 2ms**
- **Hop Count: Ring 24, Star 5**

Field Level

- **CyleTime: 50µs – 1ms**
- **Data Size / Node < 150 bytes**
- **Max Latency: < 500us**
- **Max Hop Count: Line 32**

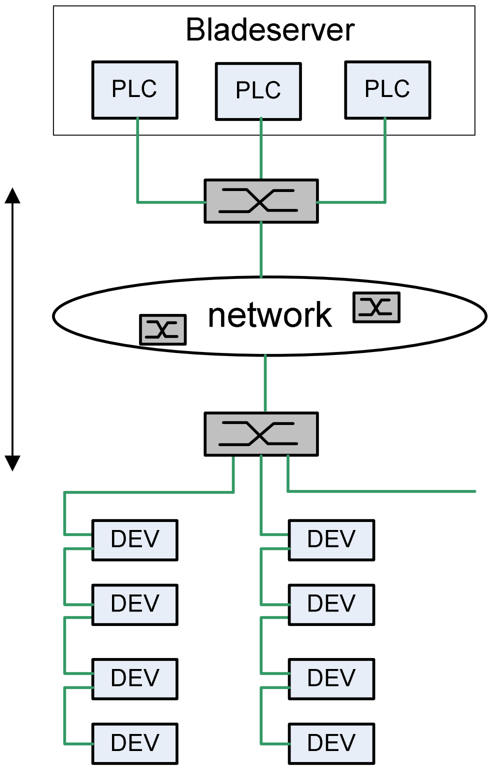
Total

- **Max Hop Count: Line 66**
- **Max Latency < 2,5 ms**

Industrial communication integrated in plant network

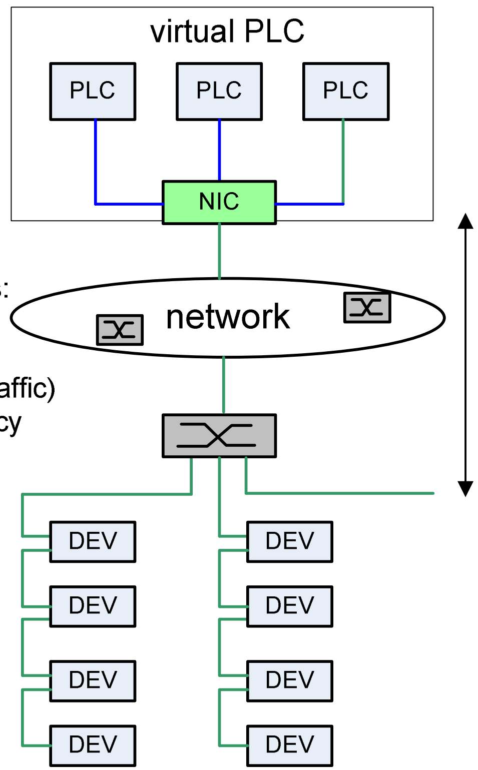
- Integration of real time communication in existing network
- Multiple real time protocols within one network

Network is used by multiple services



Network Requirements:

- Garantied CoS
  (Streams & legacy Traffic)
- Garantied Low Latency
- Protocol neutral
- Synchronization



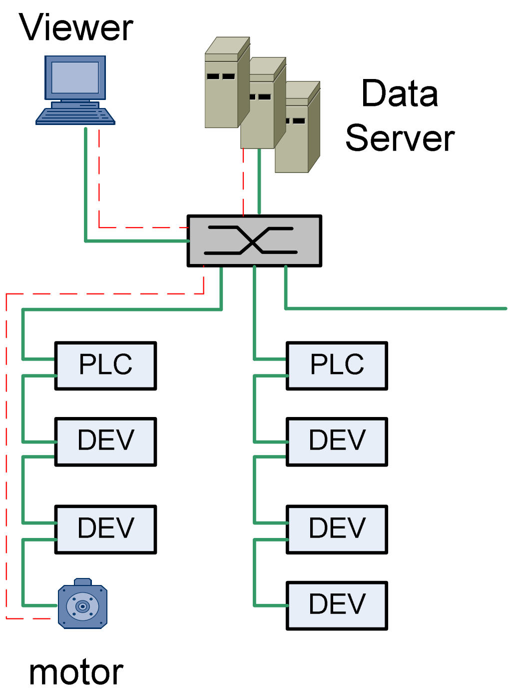
Reliable behaviour for RT streams in „shared“ networks is needed!

PLC: Programmable Logic Controller for input/output data
DEV: Device for input/output data

Convergent network for realtime, measurement and video

- **Multiple independent streaming protocols within one network**

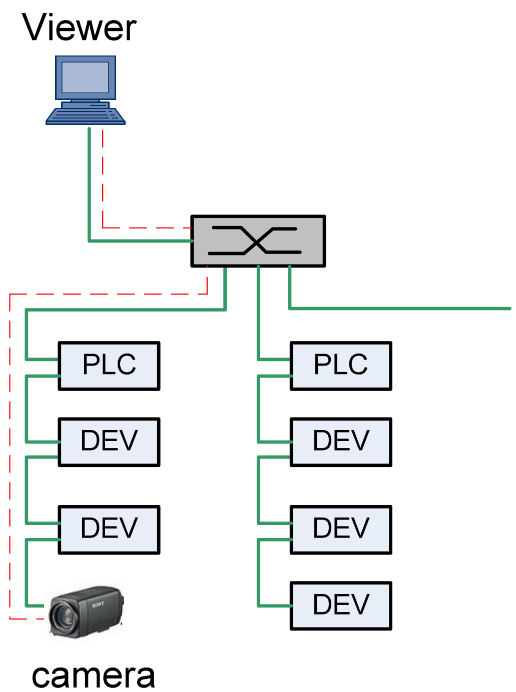
Measurement
in parallel to
industrial communication



Network Requirements:

- Garantied CoS
  (Streams & legacy Traffic)
- Garantied Low Latency
- Protocol neutral
- Synchronization

Video
in parallel to
industrial communication

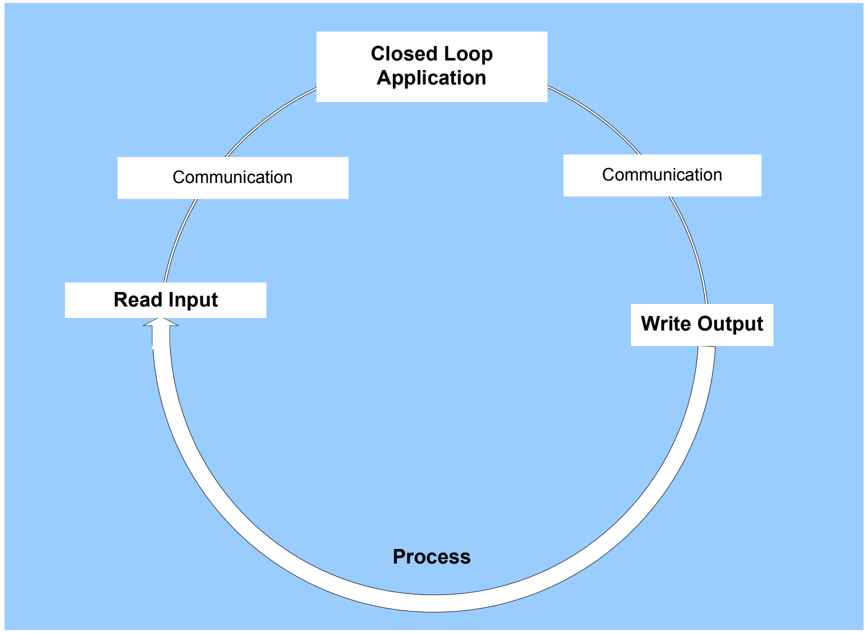


PLC: Programmable Logic Controller for input/output data
DEV: Device for input/output data

Communiation latency has impact on reaction time

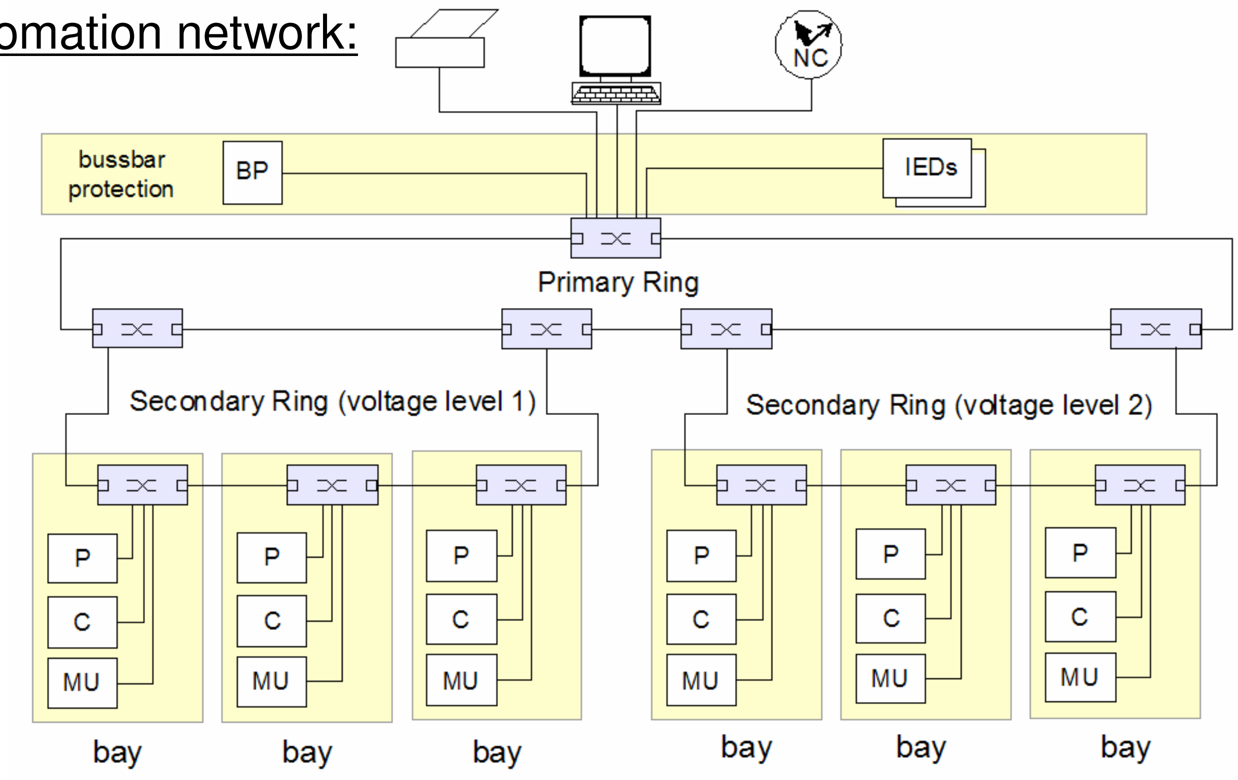
- **Event Based Application**
  - Building Automation
  - Logistic Automation
  - Manufactoring Industry (Production Line)
  - …

- **Closed Loop Application**
  - Short control cycle (< 250us)
    - Motion Control
    - Wind Energy
    - Frequency Converter
    - Solar Power
    - …
  - Long control cycle (> 2ms)
    - Process Industry
    - Energy / Substation
    - …



Low latency + redundancy in substation automation systems

Use case – substation automation network:



IED = Intelligent Electronic Device

MU = Merging Unit

P = Protection Relay

BP = Busbar Protection Relay

C = Bay Controller

NC = Network Clock

timing requirements:

event trip messages (GOOSE):
approx. 12ms end-to-end delay (incl. network recovery)

cyclic traffic (Sampled Values):

Approx. <2ms end-to-end delay, no two consecutive frames may be lost

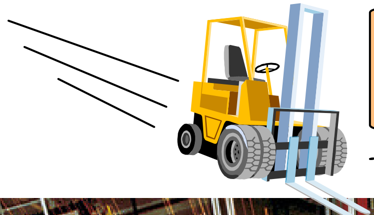
- A primary ring network connects a busbar protection and SCADA/network clock to the individual bays of each voltage level of an electric substation
- Different timing requirements on primary and secondary rings

Pictures taken from TR IEC 61850-90-4 draft

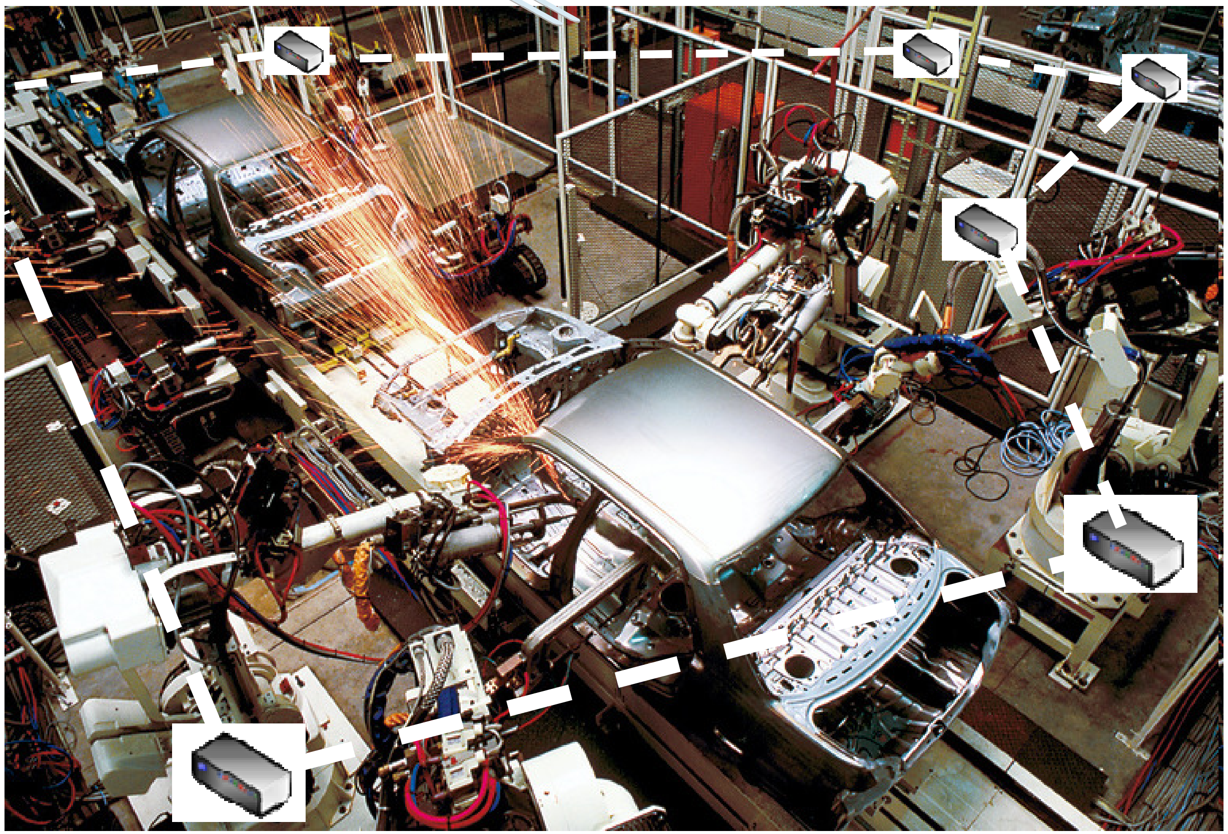
Insight into industrial automation: how is media redundancy used?

Common network topologies in industrial/ low latency networks

<u>Why media redundancy to increase availability? A short excursion:</u>



I‘m a man on a mission…



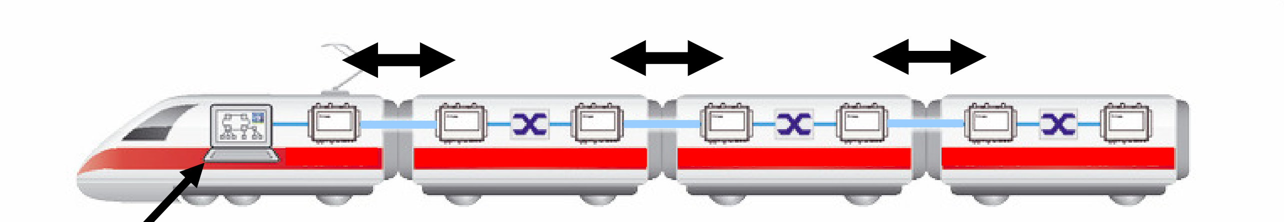
- (Physical) Interruptions of the network may be introduced because of accidents or faults due to aging or failing equipment…
- Faults that stop the production line are costly!

Common network topologies in industrial/ low latency networks

Why media redundancy to increase availability? A short excursion:

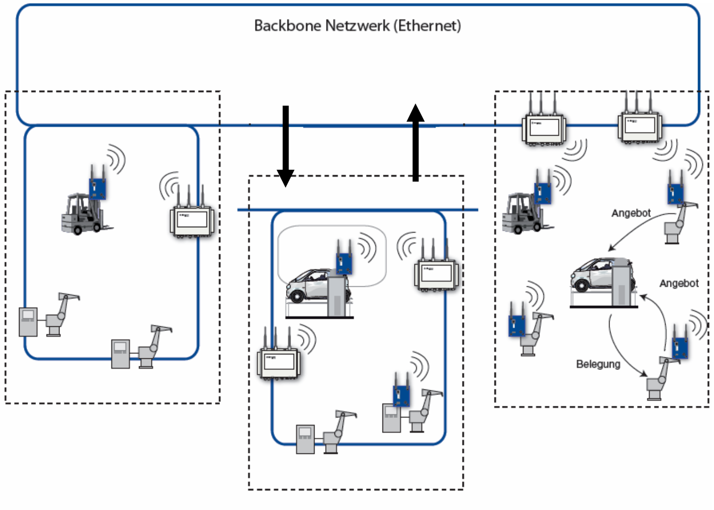
• … or may be introduced intentionally, because of regular service or redesign/ expansion

connection / disconnection of railway carriages



e.g. cabin/carriage entertainment

Expansion of an automation network, e.g. to add a new production cell or remove a cell for maintenance



Common network topologies in industrial/ low latency networks

General aims/requirements of media redundancy in industrial/low latency networks:

- Elimination of single points of failure at least on the transmission network (sometimes even to the devices on the edge ports)
- Network recovery times and behaviour must be (pre-)calculable and deterministic (total max. recovery time dependant on application)
- Media redundancy with focus on the improvement of **availability** of the transmission network, **not** with focus on an increase in **performance:**

→ e.g. RSTP and similar protocols are used where applicable but LACP use is very rare;

→ Fault tolerance is more important than load balancing

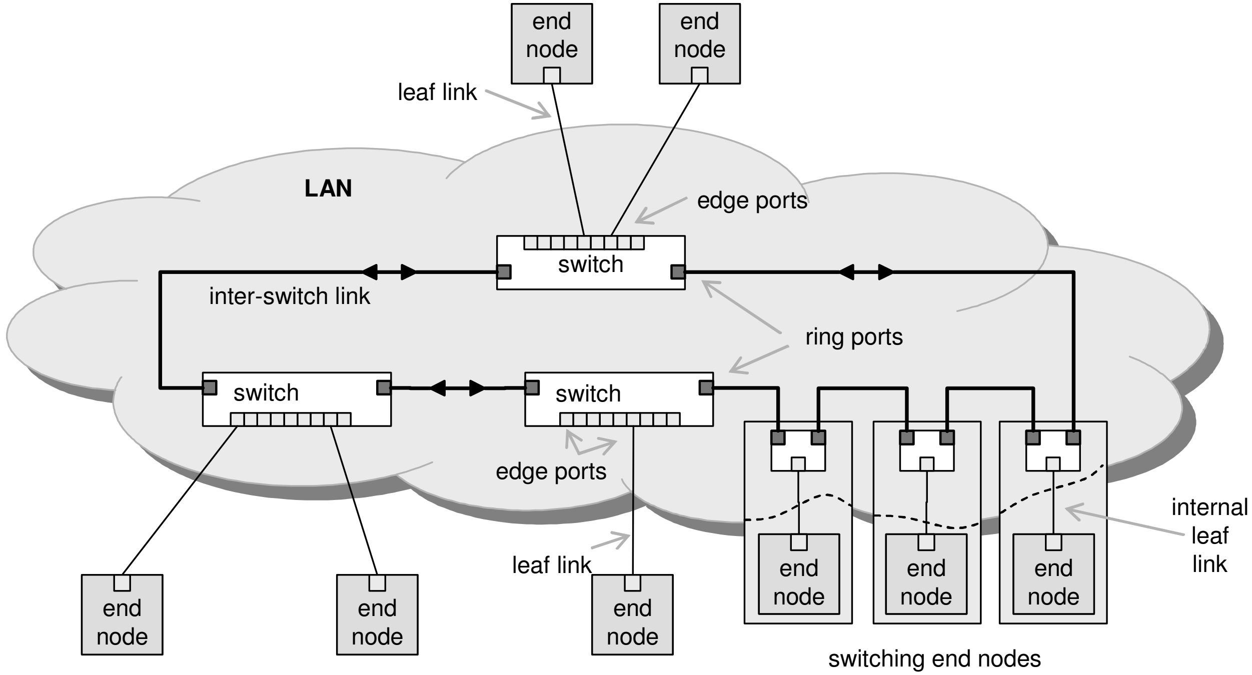
Fault tolerant topologies

Examples of physical topologies with media redundancy

Examples: fault tolerant networks

Redundant link networks:

single ring

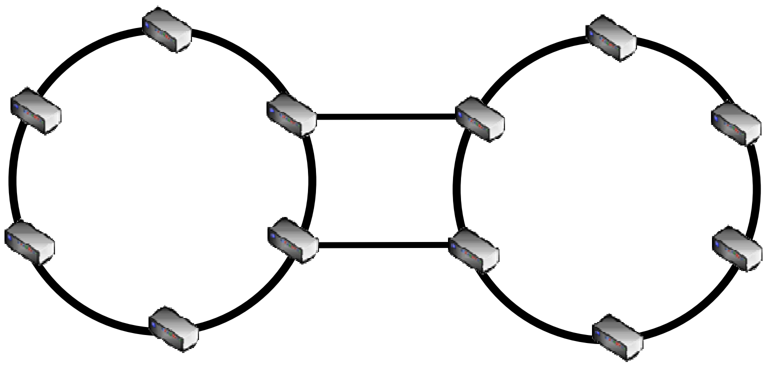


- Eliminate single points of failure by introducing multiple links
- A redundancy control protocol (like e.g. RSTP) is needed to prevent loops.
- Rings map very well to common use cases (e.g. shop floors)
- ring = closed (well-known) line structure

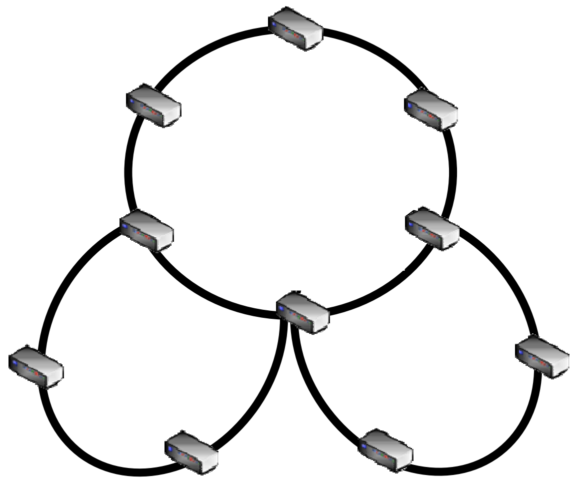
Examples: fault tolerant networks

Redundant link networks - possible combinations:

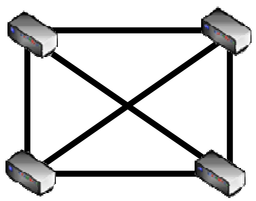
coupled single rings



ring with subrings



full mesh

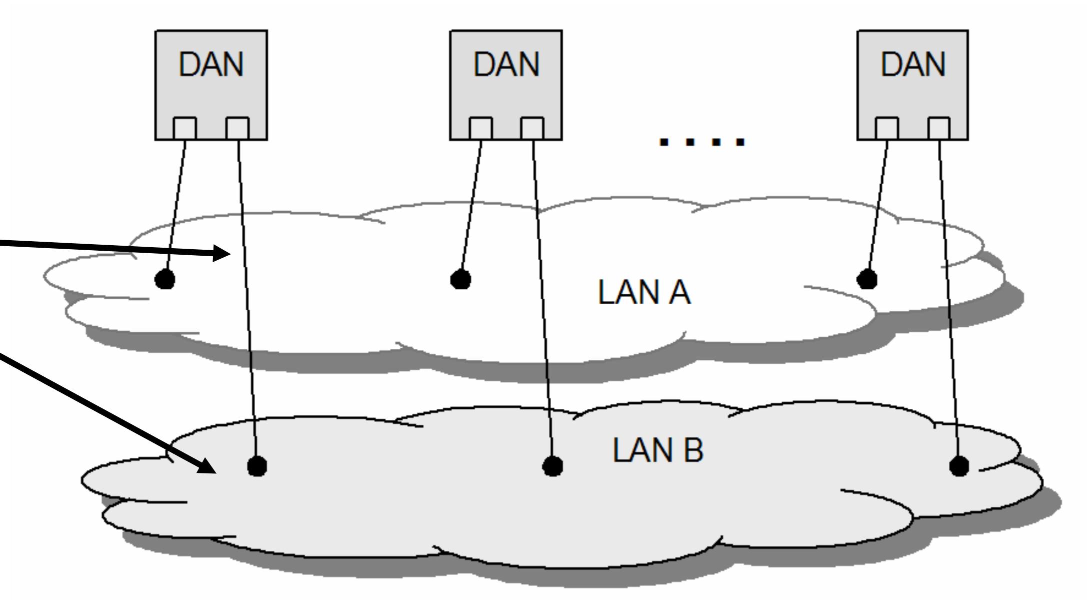


- More complex networks may be derived from the single ring structure
- Full meshes are mostly avoided due to complexity (deterministic recovery after media failure is considerably harder to achieve in a mesh than in ring)

Examples: fault tolerant networks

(Parallel) redundant networks:

May each include a line, star, ring…

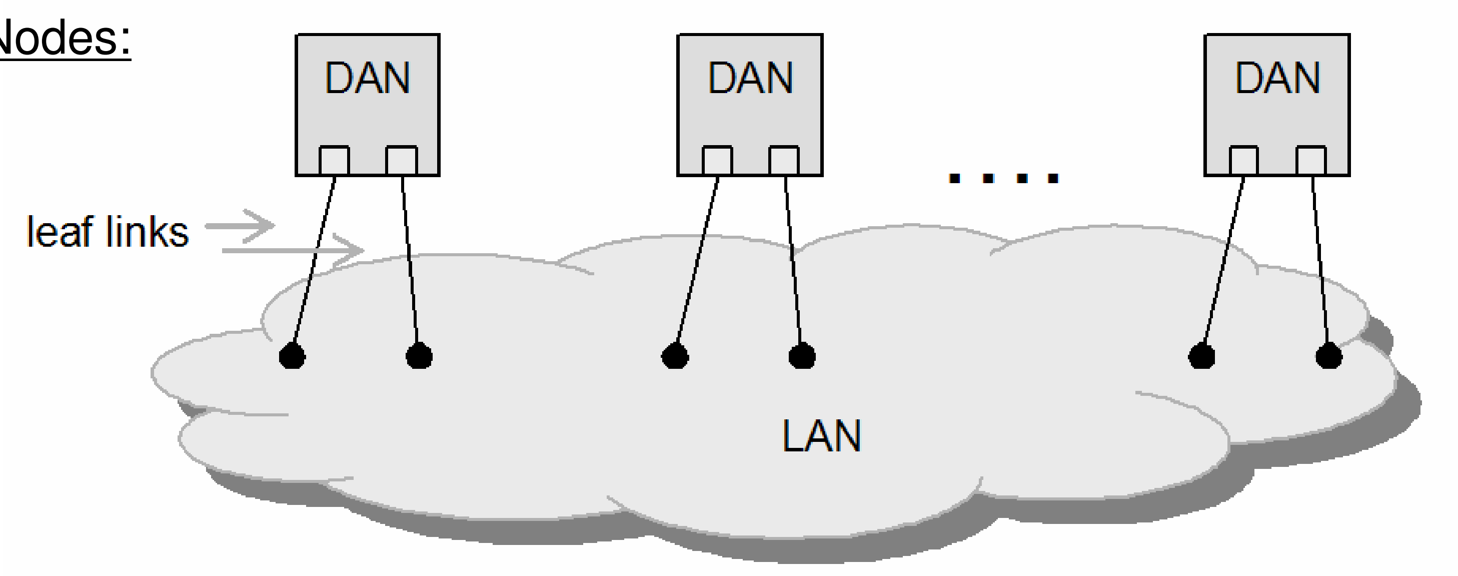


- Eliminate single points of failure by doubling network infrastructure
- Devices can be double-attached to each network (DAN = Double Attached Node) without bridging from LAN A to LAN B
- Networks are (usually) independent layer 2 broadcast domains(LAN A/B)
- Independent networks can be of any topology and may/may not make use of redundant links themselves

Pictures taken from IEC 62439-1

Examples: fault tolerant networks

Dual Attached Nodes:



- Dual Attachment of nodes to a single network to improve availability
- Nodes either bridge from one port to the other or don‘t bridge, both types are equally used

Pictures taken from IEC 62439-1

Media redundancy

Media redundancy classification

Media redundancy classification

Media redundancy „taxonomy“:

* Network interruption = end-to-end communication experiences outage in case of fault

media redundancy

- **with** network interruption*
  - redundant links
  - redundant networks
- **without** network interruption*
  - redundant links
  - redundant networks

• RSTP, used in ring configurations (IEEE 802.1D-2004) • MRP, ring red. protocol (IEC 62439-2)	• (proprietary) „Dual Homing“ setups with multiple network interfaces	• HSR, ring redundancy protocol (IEC 62439-3)	• PRP, parallel redundancy protocol (IEC 62439-3)

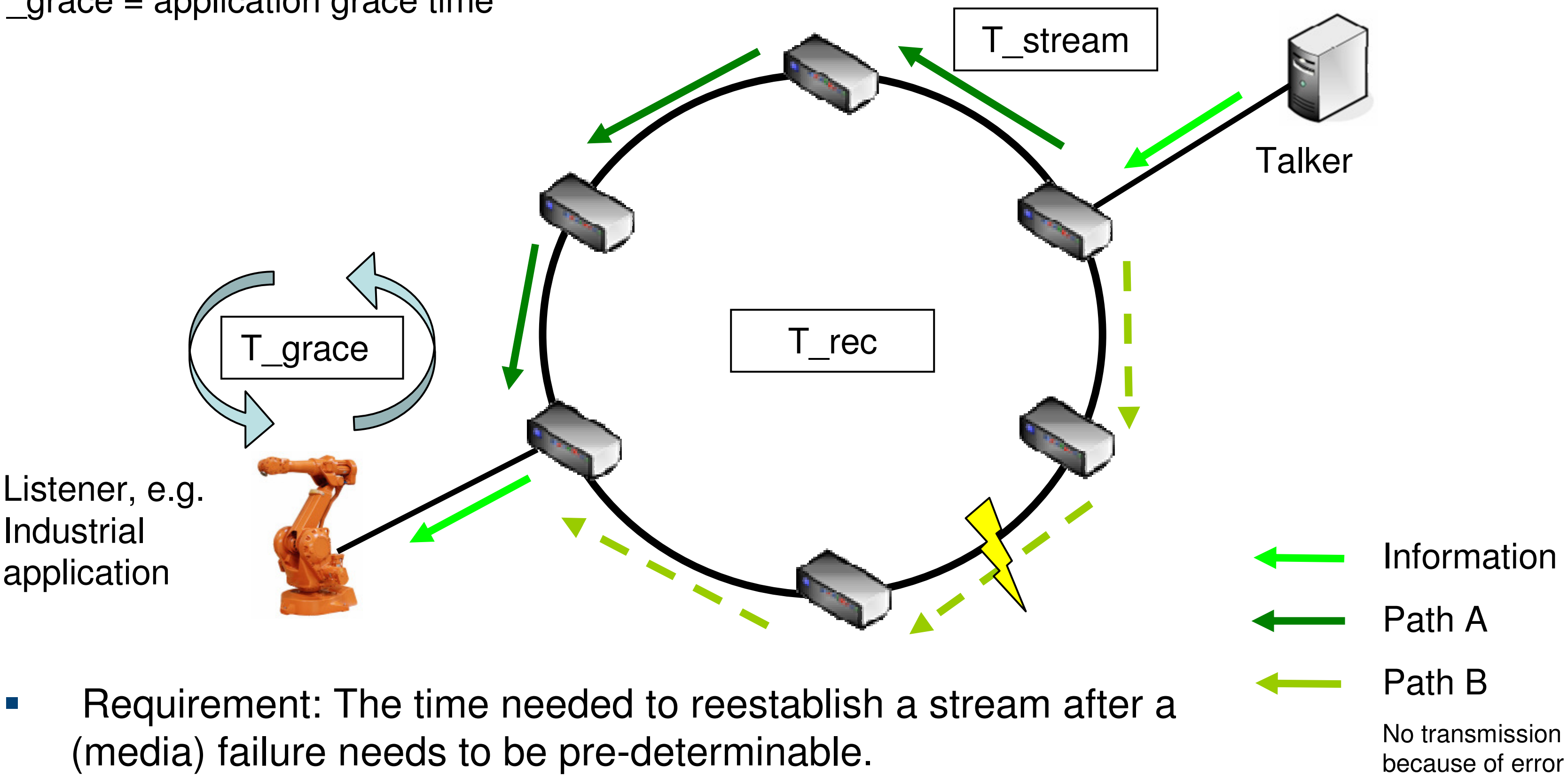
Examples

Short flashback: network reconfiguration time

flashback – network reconfiguration time

T_rec + T_stream !< T_grace

- T_rec = network reconfiguration time
- T_stream = stream reconfiguration time
- T_grace = application grace time



- Requirement: The time needed to reestablish a stream after a (media) failure needs to be pre-determinable.

Challenges and solutions?

Challenges for AVB and media redundancy

Challenges and solutions

State of the art: MSRP follows the RSTP reconfiguration path

But: For a wider audience to use AVB mechanisms in conjunction with any media redundancy protocol, a generic interface is needed to enable interoperation and optimize performance

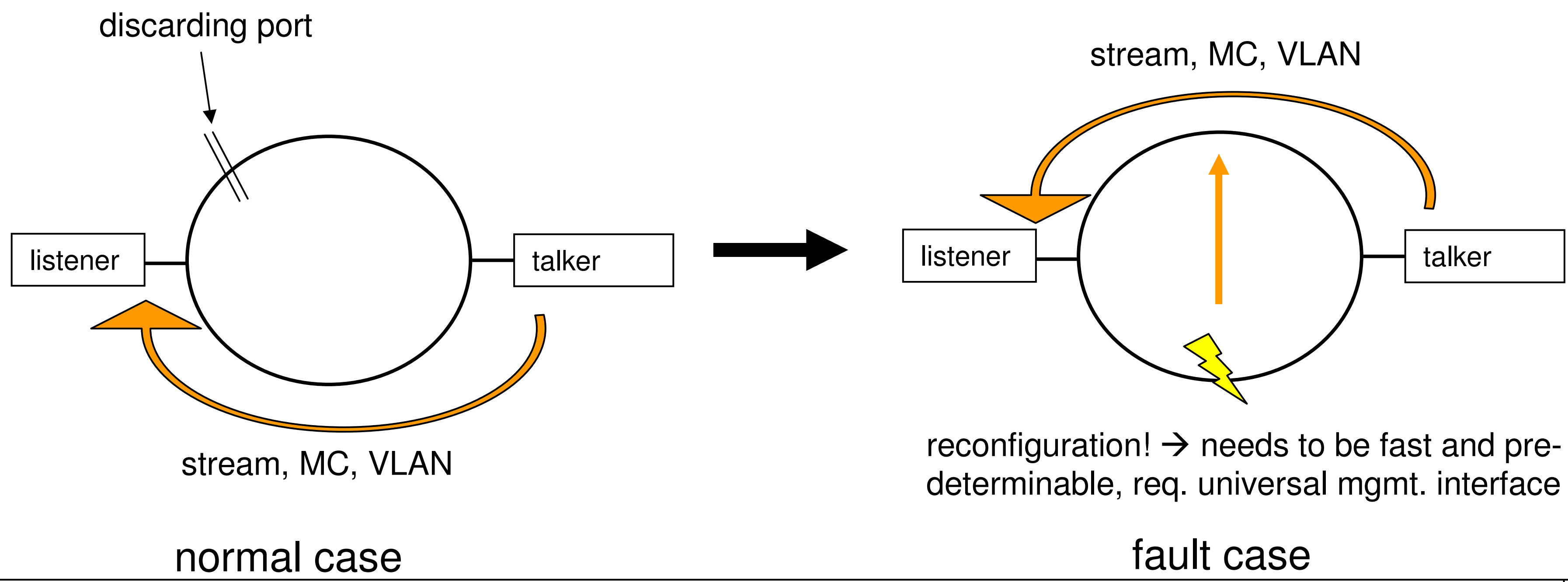
(e.g. an interface where a redundancy protocol that has finished reconfiguring the logical topology can trigger the MSRP, MMRP and MVRP reconfiguration to improve overall recovery time)

→ Since we have established a classification scheme for redundancy mechanisms, let's look at the possibilities

Challenges and solutions

Possibility 1:

	redundant links	redundant networks
with network interruption	**X**	
without network interruption		

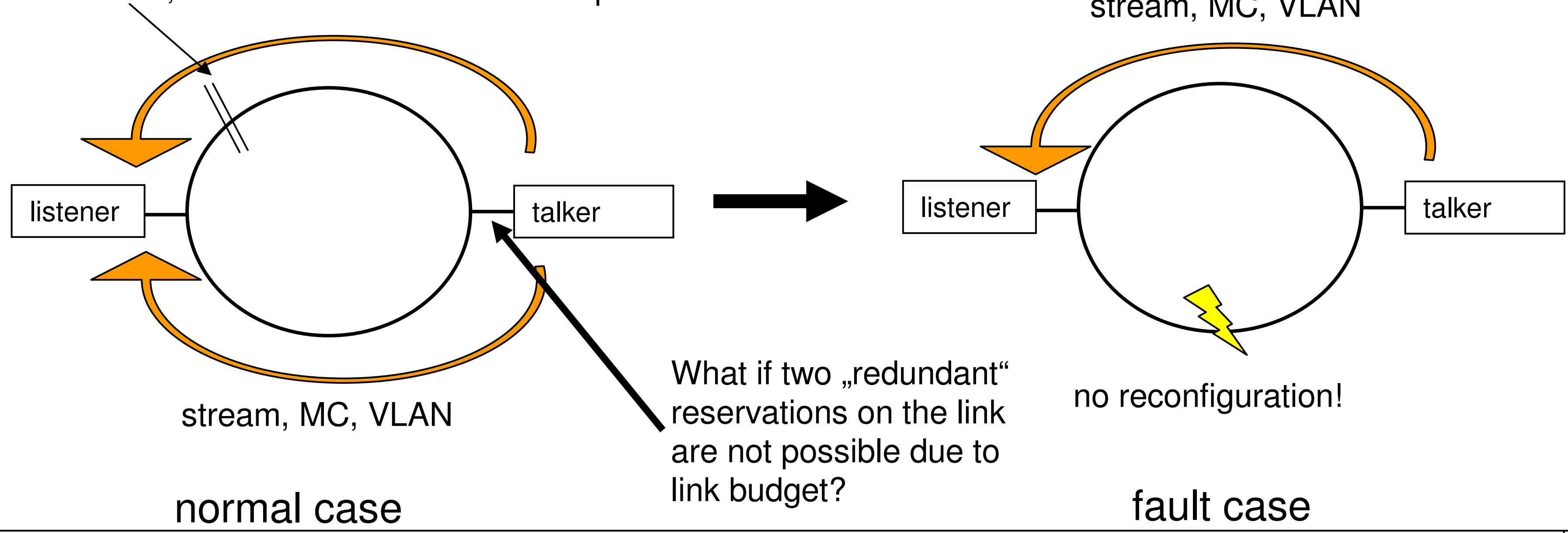


Challenges and solutions

Possibility 2:

	redundant links	redundant networks
with network interruption	X	
without network interruption	X	

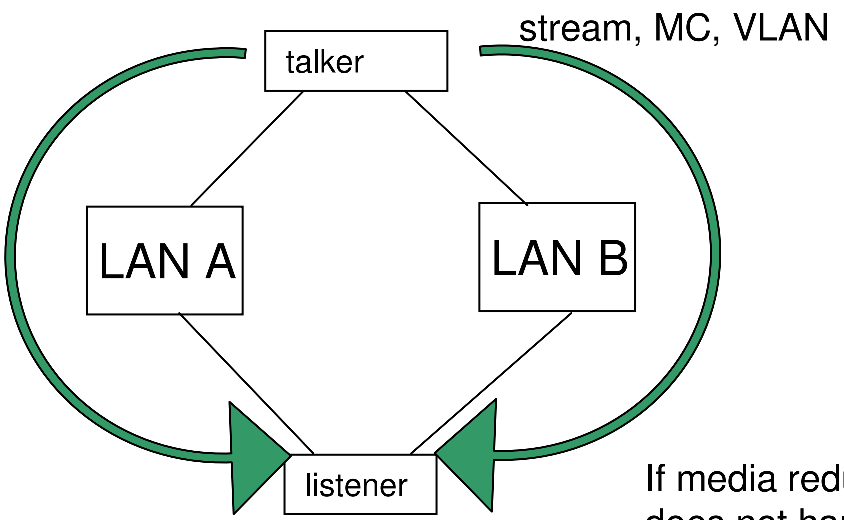
If there is a discarding port: if red. protocol is AVB unaware, will a second reservation be possible at all?



Challenges and solutions

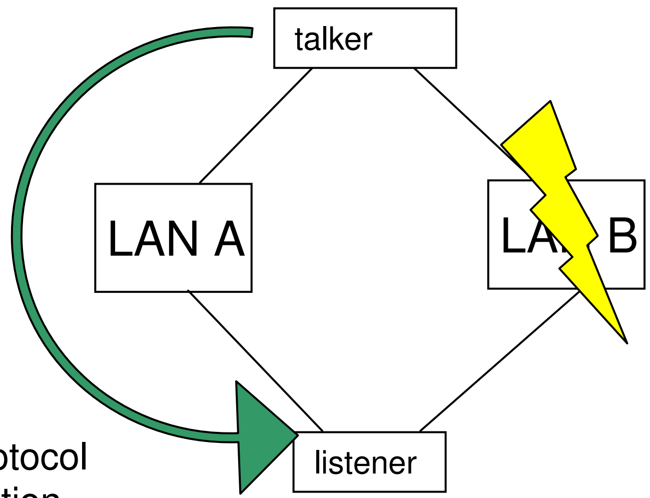
Possibility:

	redundant links	redundant networks
with network interruption		X
without network interruption		X



normal case

If media redundancy protocol does not handle duplication
→ possibility needed to register redundant streams



fault case

Challenges

It can be observed that reconfiguration time is not the major issue… the type of redundancy (redundant links or networks) is.

Note: Performance will become important after the basic concept of „how to“ is done

	redundant links	redundant networks
with network interruption	X	X
without network interruption	X	X

Requires:

•MSRP/MMRP/MVRP reconfiguration triggered by redundancy protocol via standardized interface

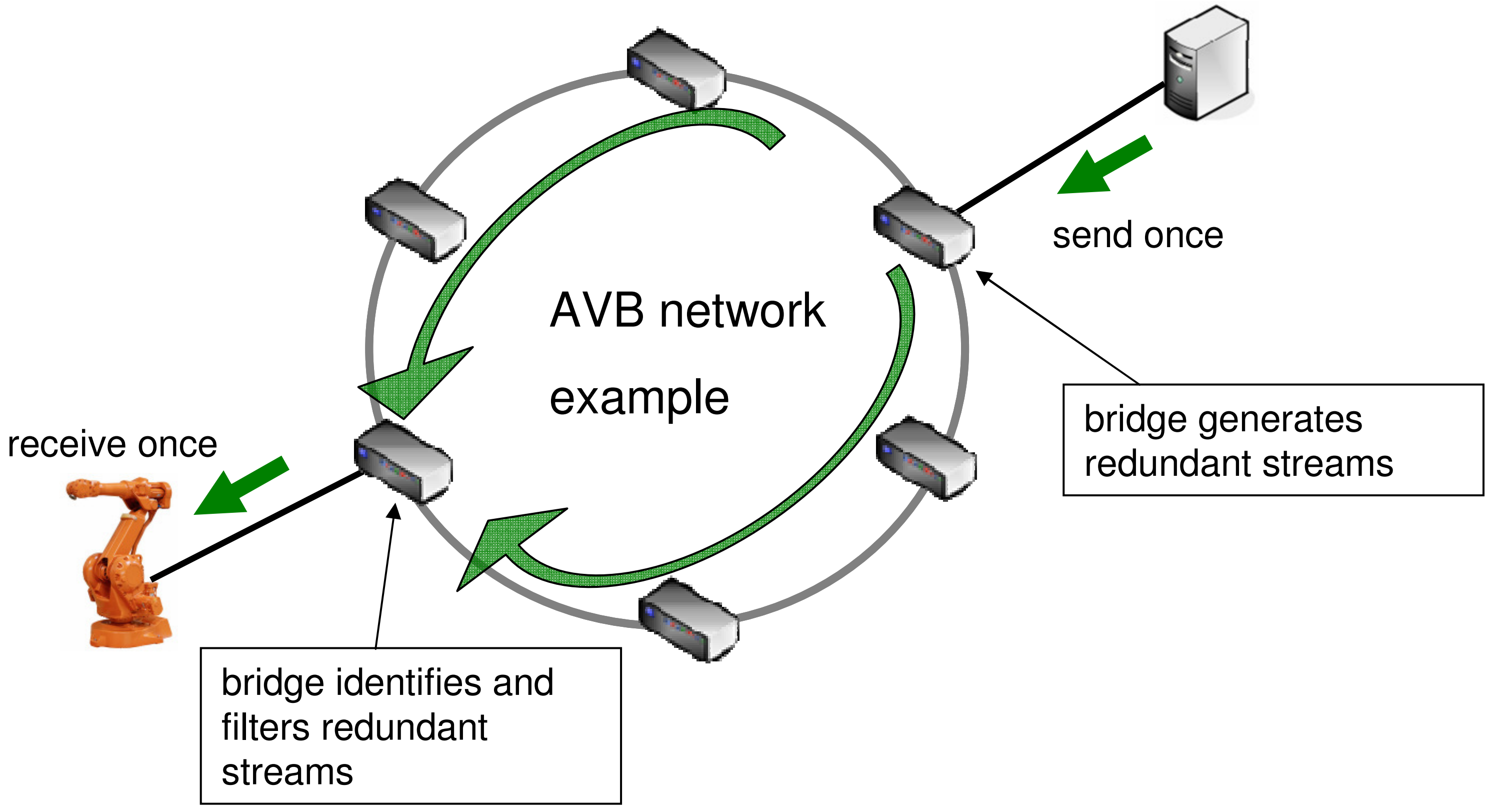
•Possibility to register redundant streams (observation on non-disjunct links)

Requires:

• Possibility to register redundant streams

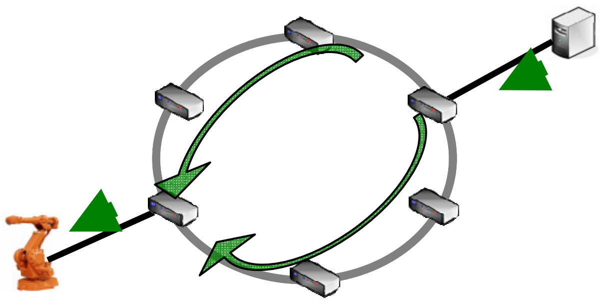
In addition to the above

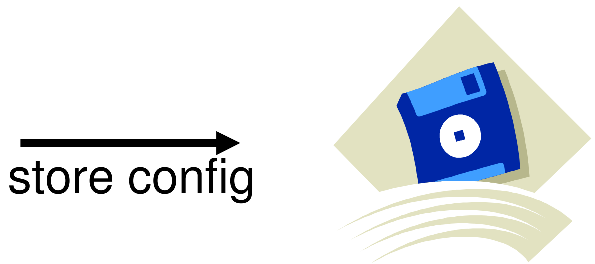
For the end devices (e.g. talker and listener), media redundancy mechanisms on the network are usually transparent:



→ It would be beneficial if bridges could generate stream redundancy and filter redundant streams. This would keep redundancy transparent to end devices

Item for further discussion





It may also be beneficial to make it possible to „store“ one or more consistent stream configuration(s):

• On network startup, all devices are immediately configured

• For an N-fold redundancy, N network configurations could be pre-configured for fast switchover

Challenges:

• How to assure consistency of such a configuration „set“ in the whole network?

• Interaction between pre-configured and non-preconfigured devices?

FIN

Thank you!