Carrier Grade Ethernet

“Carrier grade” Ethernet

Ethernet started out as a *LAN* technology

LAN networks are relatively small and operated by consumer

hence there are usually no management problems

As Ethernet technologies advances out of the LAN environment

new mechanisms are needed

The **MEF forum** and **ITU-T** defined such mechanisms, e.g.

- OAM
- deterministic (Connection-Oriented) connections
- synchronization

What has remained constant in all Ethernet forms

is the basic *frame format*

MEF forum definitions

MEF focuses on Ethernet as a carrier-grade service to a customer

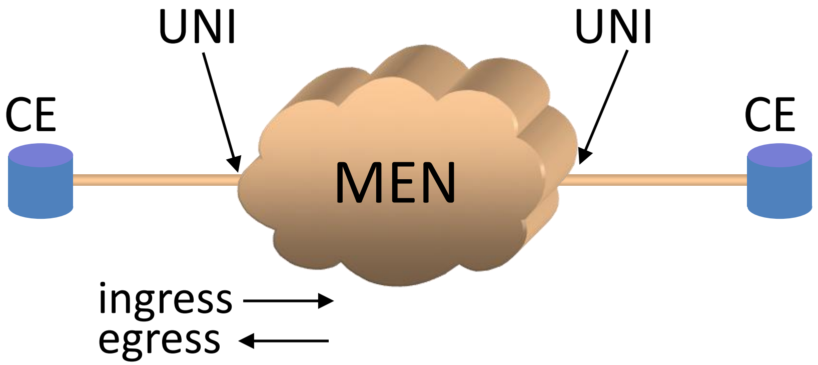
The service is *seen* by the Customer Edge

The UNI is the demarcation point between customer and MEN

Each UNI serves a single customer

- presents a standard Ethernet interface
  - at the UNI CE and MEN exchanged service (MAC) frames

Connection between UNIs called an **E**thernet **V**irtual **C**onnection



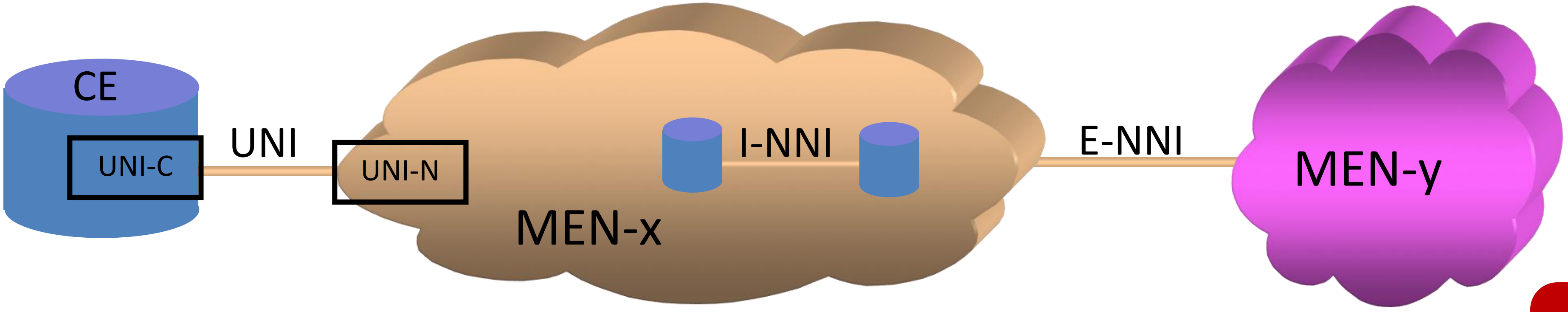
Other reference points

The UNI stands between the CE and **M**EF **E**thernet **N**etwork

The processing functions needed at the CE to connect to the MEN are called UNI-C

The processing functions needed at the MEN to connect to the CE are called UNI-N

Between networks elements of a MEN we have I-NNI interfaces while between different MENs we have E-NNI interfaces

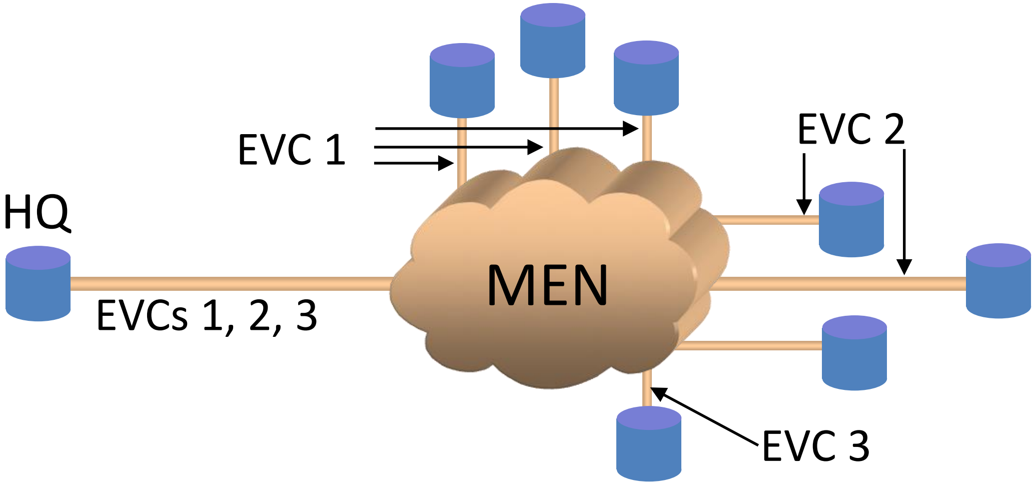


EVCs

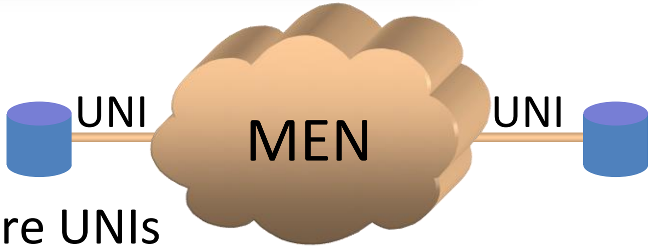
A public MEN can not behave like a shared LAN
- since ingress frames must not be delivered to incorrect customers

An association of 2 or more UNIs is called an **EVC**
- ingress frames must be delivered only to UNI(s) in the same EVC
- when several UNIs frames may be flooded to all or selectively forwarded
- frames with FCS errors must be dropped in the MEN (to avoid incorrect delivery)

A single UNI may belong to several EVCs (differentiated by port and/or VLAN ID)



EVC types



A point-to-point EVC associates **exactly** 2 UNIs

- the service provided is called E-LINE [MEF-6]

A multipoint-to-multipoint EVC connects 2 or more UNIs

Note: MP2MP w/ 2 UNIs is different from P2P (new UNIs can be added)

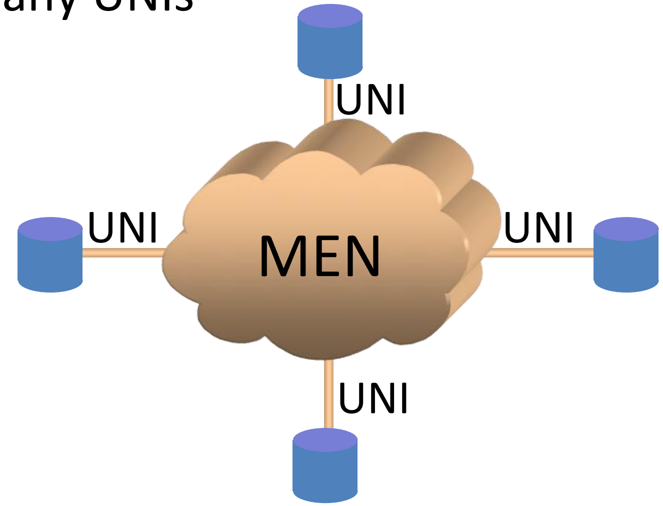
unicast frames may flooded or selectively forwarded

broadcast/multicast frames are replicated and sent to all UNIs in the EVC

- the service provided is called E-LAN [MEF-6]

A tree-topology EVC connects one UNI to many UNIs

- the service provided is called E-TREE



Ethernet services

we previously defined

- E-LINE point-to-point layer 2 service
- E-LAN multipoint-to-multipoint Ethernet service

but MEF and ITU have gone a step further

MEF 6 splits E-LINE into EPL and EVPL

ITU followed - Recommendations: G.8011.1 and G.8011.2

and E-LAN can be split into EPLAN and EVPLAN

these distinctions are made in order to live up to SLAs

i.e. provide defined service attributes

EVCs revisited

in our previous discussion of EVCs we didn’t mention VLANs

we now realize customer EVCs can be distinguished by VLAN IDs

if the transport infrastructure is ETH, there may be an SVID

if the customer wants to have several EVCs, there will be a CVID
(here we simply mean the customer’s 802.1Q VLAN ID)

the provider may promise “VLAN preservation”

i.e. not change CVIDs (untagged remain untagged)

at the UNI-N there will be a ***CVID to EVC map*** (see MEF 10.1)

there can be three types of maps:

- all to one
- one to one (not MEF 10 term)
- arbitrary (not MEF 10 term)

All to one bundling map

all frames (independent of CVID) are mapped to the same EVC

VLAN preservation

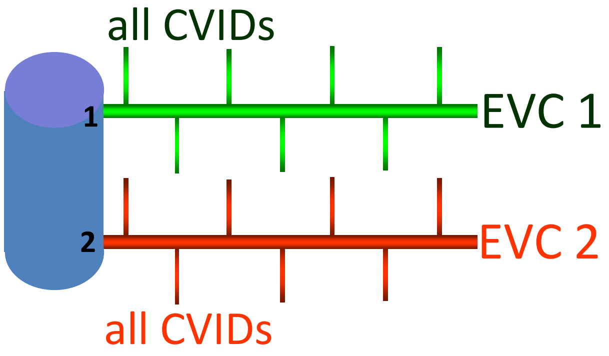
no need for customer-provider coordination

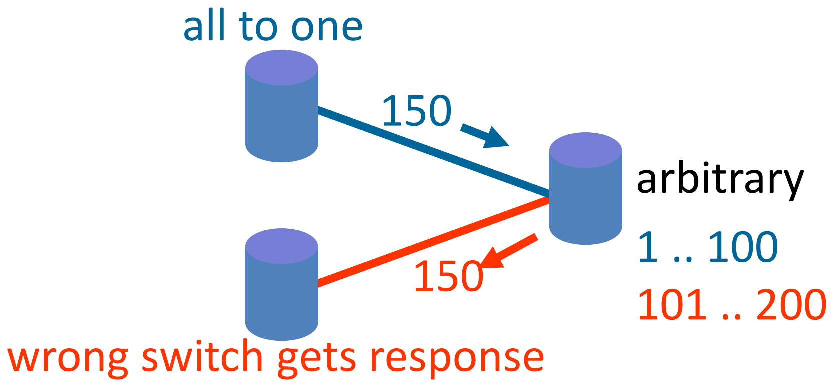
when p2p: similar to leased line

when mp2mp: similar to TLS

support multiple customer EVCs by different switch ports

can't mix all to one switch with other map types in single EVC





One to one bundling map

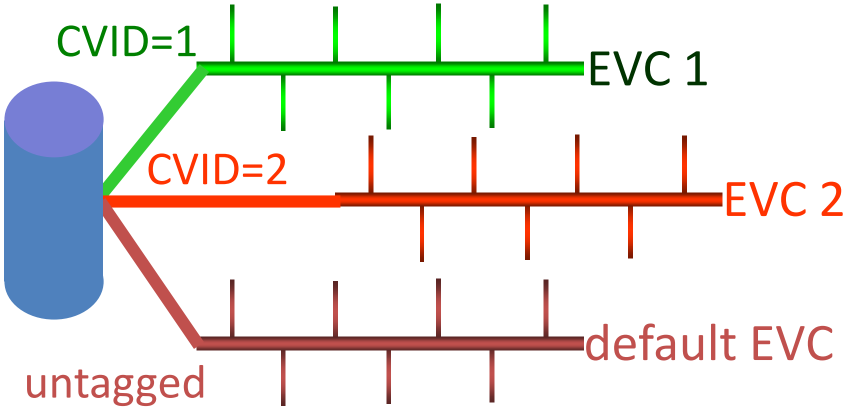
each CVID mapped to a different EVC

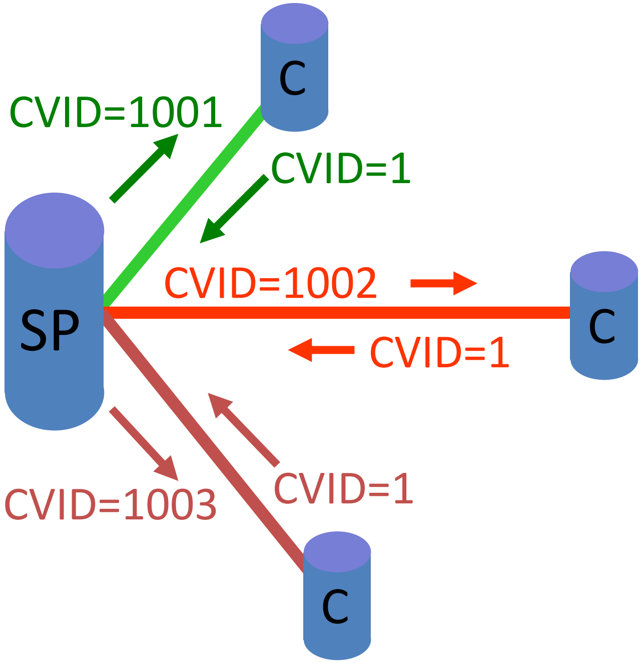
untagged (and priority tagged) mapped to default EVC

support multiple EVCs from a single switch port

no VLAN preservation can makes customer configuration easier

similar to frame relay (DLCI identifies PVC)





Arbitrary bundling map

multiple CVIDs per EVC

but multiple EVCs per port

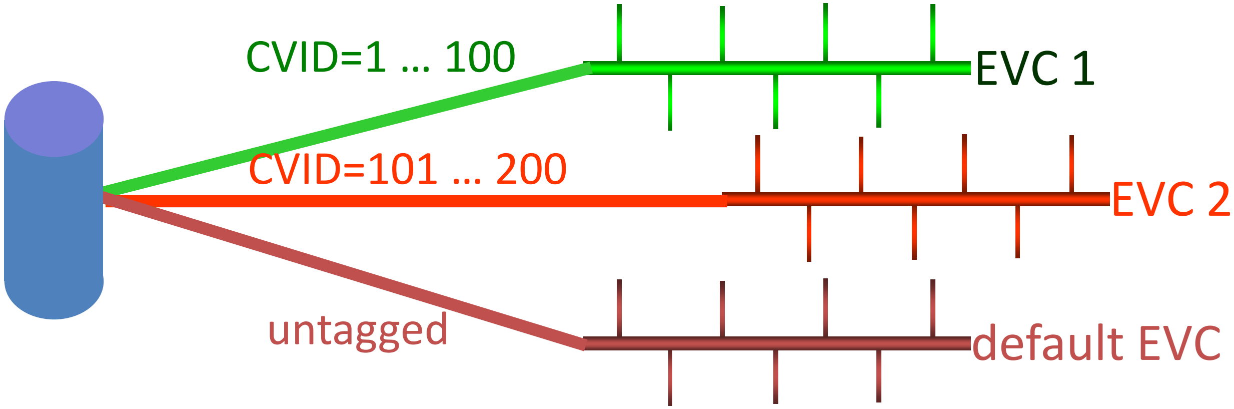
VLAN preservation

can ensure customer traffic only goes to sites where VLAN is needed

more efficient BW use

need customer and SP coordination as to CVID – EVC map

can coexist with one to one switches on same EVC

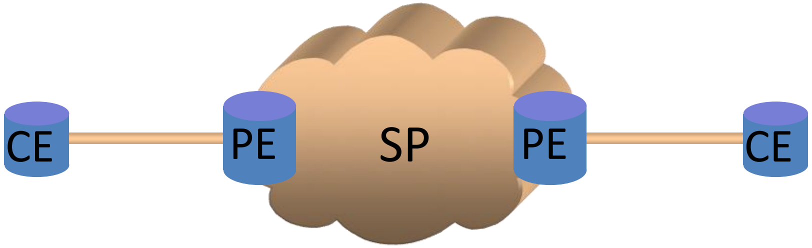


EPL

Ethernet Private Line is a dedicated-BW E-LINE (p2p) service

transport network seems to be a transparent cable

no frame loss unless FCS errors



transport layer may be

- native Ethernet – dedicated to single EVC (e.g. 100 Mb/s)
- TDM or SONET/SDH timeslot (e.g. X.86, GFP)
- VPWS service in TE tunnel

ITU further divides EPL into

- type 1 – terminate ETY, transport MAC frame over server (SDH/GFP-F, MPLS)
- type 2 – transparent transport (e.g. GFP-T)
- native (special case for 10GBASE-W)

EVPL

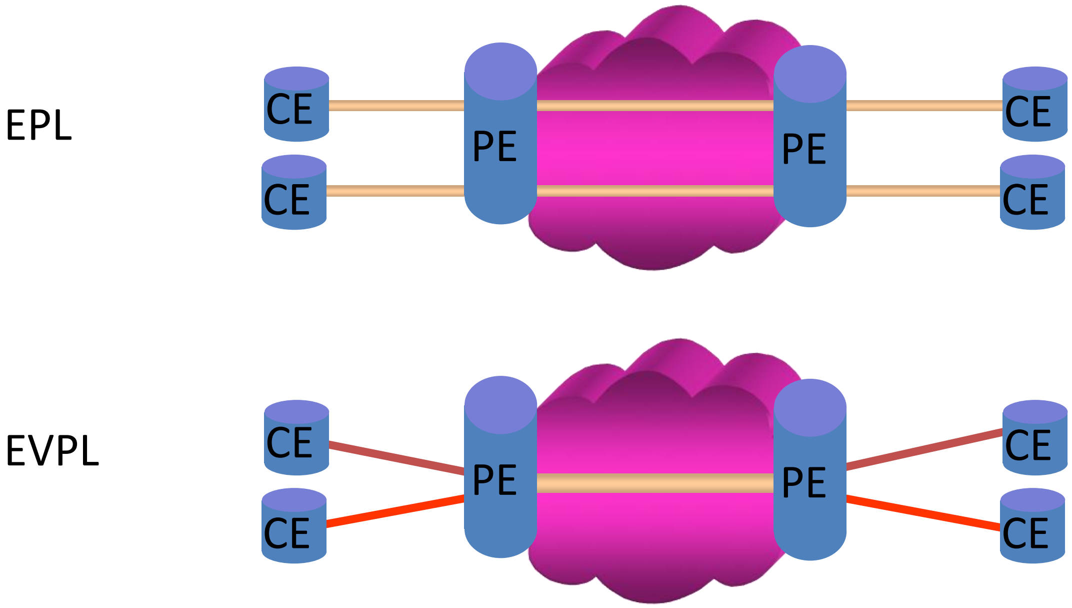
Ethernet Virtual Private Line is a shared-BW E-LINE (p2p) service

statistical multiplexing of user traffic, marked by VLAN IDs

(actually, all resources are shared – constraint may be switch fabric computation)

there may be frame loss due to congestion

normally a policing function is required at SP network ingress



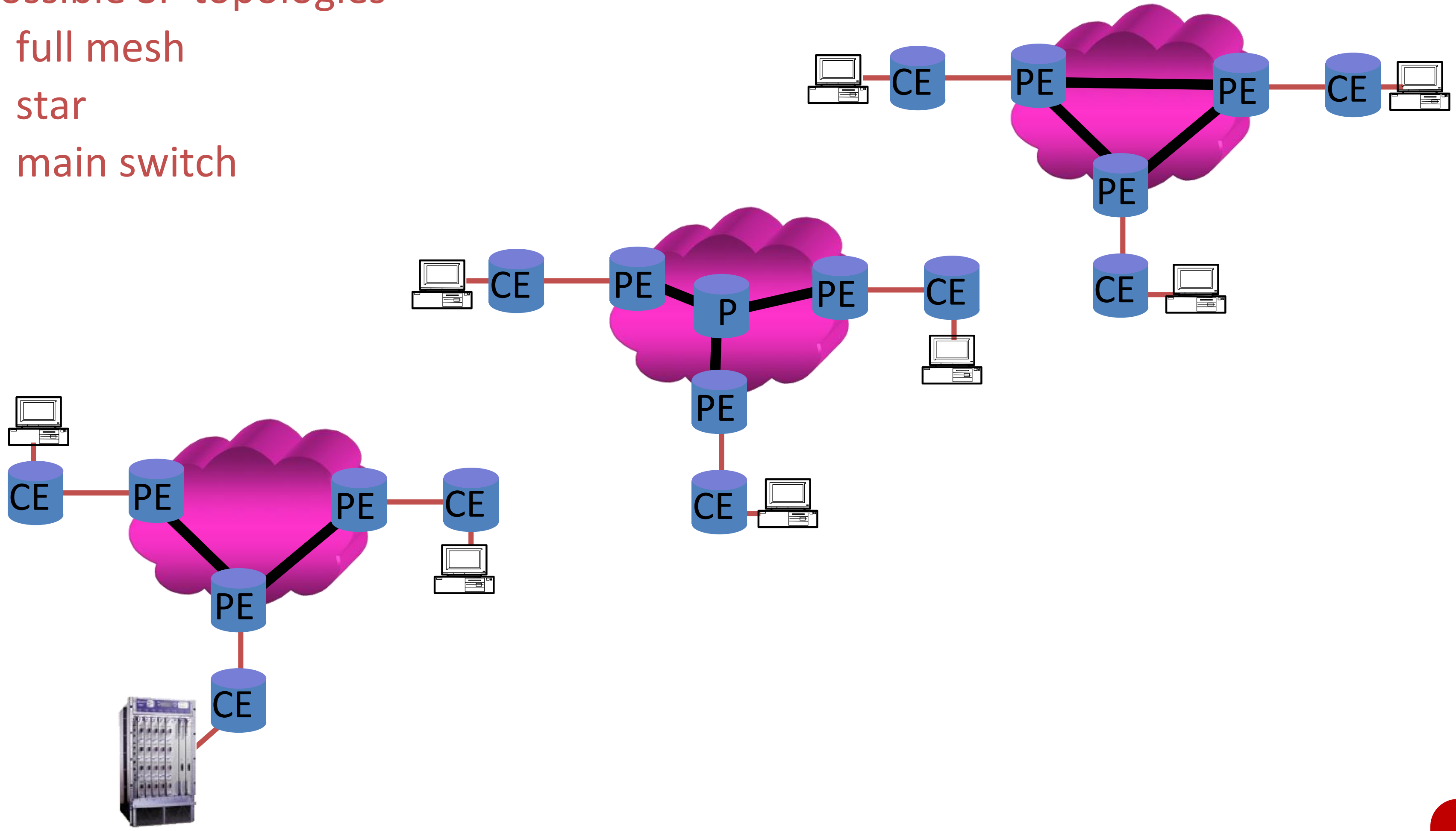
ITU terms:
spatial vs. logical
traffic separation

EPLAN

Ethernet Private LAN is a dedicated-BW E-LAN service

possible SP topologies

- full mesh
- star
- main switch



EVPLAN

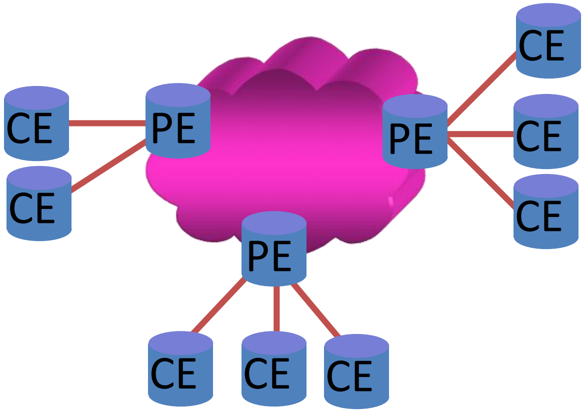
Ethernet Virtual Private LAN is a shared-BW E-LAN service

statmuxed BW and switch fabric are shared among customers

useful service, but most difficult to manage (not yet studied)

when server is MPLS, this is VPLS

best effort version is widely deployed



(MEF) Service attributes

all per EVC, per CoS

- **frame loss**

  fraction of frames that should be delivered that actually are delivered

  specified by T (time interval) and L (loss objective)

- **frame delay**

  measured UNI-N to UNI-N on delivered frames
  specified by T, P (percentage) and D (delay objective)

- **frame delay variation**

  specified by T, P, L (difference in arrival times), V (FDV objective)

- **BW profiles** (shaping/policing)

  per EVC, per CoS, per UNI

  specified by **CIR**, CBS, EIR, EBS, …

Burst size token buckets

the profile is enforced in the following way

there are two byte buckets, C of size CBS and E of size EBS

tokens are added to the buckets at rate CIR/8 and EIR/8

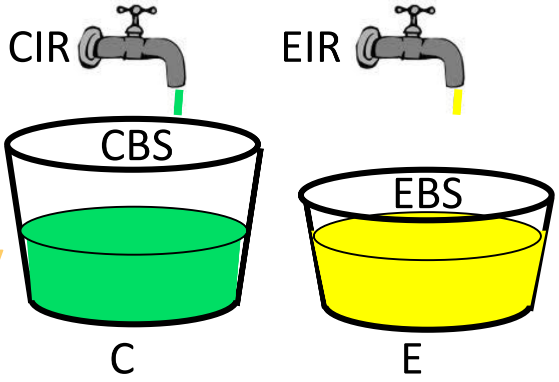
when bucket overflows tokens are lost (*use it or lose it*)

```
if ingress frame length < number of tokens in C bucket
    frame is green and its length in tokens is debited from C bucket
else if ingress frame length < number of tokens in E bucket
        frame is yellow and its length of tokens is debited from E bucket
    else frame is red
```

green frames are delivered and service objectives apply

yellow frames are delivered by service objectives don't apply

red frames are discarded



Hierarchical BW profiles

MEF 10.1 allows bandwidth profile

- per UNI (can be different at different UNIs of same multipoint EVC)
- per EVC *and* CoS

but doesn't allow a single frame to be subject to more than 1 profile

New work in the MEF is aimed at allowing

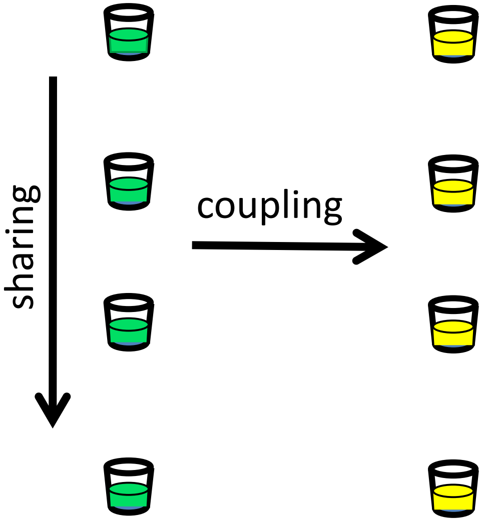
- per CoS bandwidth profile, followed by
- per EVC color-aware profile

The idea is to allow the user to use excess "paid for" bandwidth for lower priority traffic (BW *sharing*)

Thus

- frames will never be downgraded (green➔yellow, or yellow➔red)
- frames may be upgraded (red➔yellow, yellow➔green)

There are complex inter-relationships between sharing and coupling



Ethernet OAM

Analog channels and 64 kbps digital channels
did not have mechanisms to check signal validity and quality
thus

- major faults could go undetected for long periods of time
- hard to characterize and localize faults when reported
- minor defects might be unnoticed indefinitely

PDH and SDH networks evolved more and more advanced
Operations, **A**dministration and **M**aintenance (OAM) functions
including:

- monitoring for valid signal
- defect reporting
- alarm indication/inhibition

For carrier-grade Ethernet there was *clean* effort
based on everything that had been learned

OAM for PSNs

OAM is more complex and more critical for PSNs
In addition to previous problems, such as

- loss of signal
- bit errors

we have new defect types

- packets may be lost
- packets may be delayed
- packets may incorrectly delivered

OAM requirements are different for CO and CL modes

OAM remains a ***user-plane*** function
    but may influence control and management plane operations
for example

- OAM may trigger protection switching, but doesn't switch
- OAM may detect provisioned links, but doesn't provision them

Ethernet OAM functionality

- Continuity Check / Connectivity Verification
- LoopBacks
  - in-service (nonintrusive)
  - out-of service (intrusive)
  - linktrace
- defect notification / alarm inhibition
  - AIS (FDI)
  - RDI (BDI)
- performance monitoring
  - frame loss
  - one-way delay
  - round-trip delay
  - delay variation
  - throughput

Two flavors

For many years there was no OAM for Ethernet

now there are two incompatible ones!

Link layer OAM – EFM 802.3ah 802.3 clause 57

single link only

limited functionality

Service OAM – Y.1731, 802.1ag (CFM)

any network configuration

full OAM functionality

In some cases may need to run both (e.g. ETH over ETY) while in others only service OAM makes sense

EFM OAM

EFM networks are mostly p2p links or p2mp PONs

- thus a link layer OAM is sufficient for EFM applications

Since EFM link is between customer and Service Provider

- EFM OAM entities are classified as active (SP) or passive (customer)
- active entity can place passive one into LB mode, but not the reverse

but link OAM may be used for any Ethernet link, not just EFM ones

EFM OAMPDUs are a slow protocol frames – not forwarded by bridges

Ethertype = 88-09 and subtype 03

messages multicast to slow protocol specific group address

OAMPDUs must be sent once per second (heartbeat)

messages are TLV-based

DA 01-80-C2-00-00-02	SA	TYPE 8809	SUB TYPE 03	FLAGS (2B)	CODE (1B)	DATA	CRC

EFM OAM capabilities

6 codes are defined

- Information (autodiscovery, heartbeat, fault notification)
- Event notification (statistics reporting)
- Variable request (active entity query passive’s configuration) (not really OAM)
- Variable response (passive entity responds to query) (not really OAM)
- Loopback control (active entity enable/disable of passive’s PHY LB mode)
- Organization specific (proprietary extensions)

Flags are in every OAMPDU

expedite notification of critical events

- link fault (RDI)
- dying gasp
- unspecified

monitor slowly degradations in performance

Y.1731 OAM

SPs want to monitor full networks, not just single links

Service layer OAM provides end-to-end integrity of the Ethernet service over arbitrary server layers

Ethernet is the hardest case for OAM

- connectionless – can’t use ATM-like connection continuity check
- MP2MP – so need full connectivity verification
- layering – need separate OAM for operator, SPs, customer
- specific ETH behaviors – flooding, multicast, etc.

Y.1731 messages

Y.1731 supports many OAM message types:

- Continuity Check proactive heartbeat with 7 possible rates
- LoopBack unicast/multicast pings with optional patterns
- Link Trace identify path taken to detect failures and loops
- AIS periodically sent when CC fails
- RDI flag set to indicate reverse defect
- Loss Measurement (synthetic and counter-based)
- Delay Measurement (1-way and 2-way)
- Client Signal Fail sent by MEP when client doesn’t support AIS
- LoCK signal inform peer entity about intentional diagnostic actions
- Test signal in-service/out-of-service tests for loss rate, etc.
- Automatic Protection Switching
- Maintenance Communications Channel remote maintenance
- EXPerimental
- Vendor SPecific

Y.1731 frame format

After DA, SA and Ethertype (8902)

Y.1731/802.1ag PDUs have the following header (may be VLAN tagged)

LEVEL (3b)	VER (5b)	OPCODE (1B)	FLAGS (1B)	TLV-OFF (1B)

if there are sequence numbers/timestamp(s), they immediately follow

then come TLVs, the “end TLV”, followed by the CRC

TLVs have 1B type and 2B length fields

there may or not be a value field

the “end-TLV” has type = zero and no length or value fields

Y.1731 PDU types

opcode	OAM Type	DA
1	CCM	M1 or U
3	LBM	M1 or U
2	LBR	U
5	LTM	M2
4	LTR	U
6-31	RES IEEE	
32-63 unused	RES ITU-T	
33	AIS	M1 or U
35	LCK	M1or U
37	TST	M1 or U
39	Linear APS	M1or U
40	Ring APS	M1or U
41	MCC	M1 or U
43	LMM	M1 or U
42	LMR	U DA
45	1DM	M1 or U
47	DMM	M1 or U
46	DMR	UA
49	EXM	
48	EXR	
51	VSM	
50	VSR	
52	CSF	M1 or U
55	SLM	U
54	SLR	U
64-255	RES IEEE	

MEPs and MIPs

Maintenance Entity (ME) – entity that requires maintenance

ME is a relationship between ME end points
because Ethernet is MP2MP, we need to define a ME Group

MEGs can be nested, but not overlapped

MEG LEVEL takes a value 0 … 7
by default - 0,1,2 operator, 3,4 SP, 5,6,7 customer

MEP = MEG end point (MEG = ME group, ME = Maintenance Entity)
(in IEEE MEG → MA = Maintenance Association)

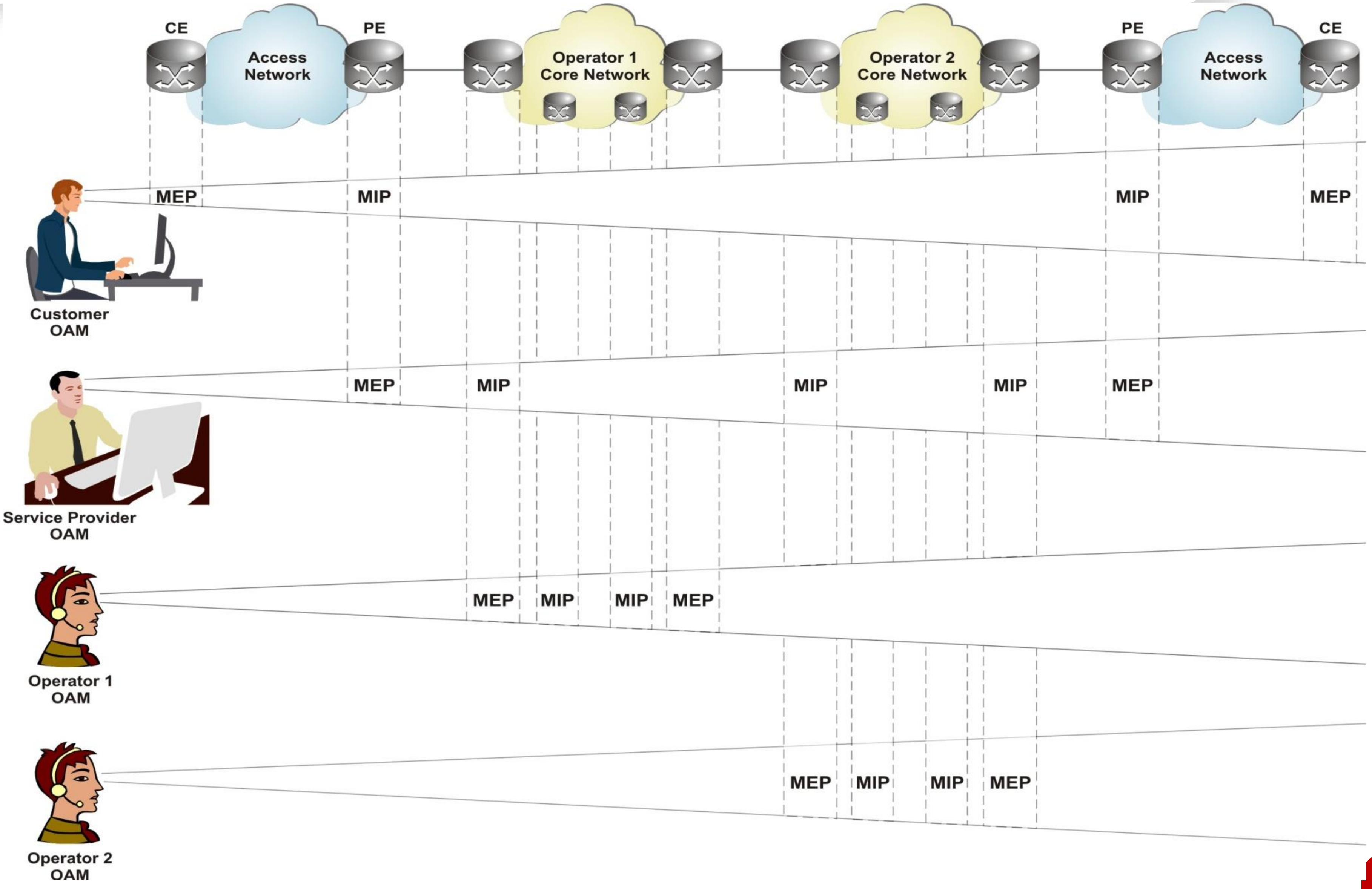
unique MEG IDs specify to which MEG we send the OAM message

MEPs responsible for OAM messages not leaking out
but transparently transfer OAM messages of higher level

MIPs = MEG Intermediate Points

- never originate OAM messages,
- process some OAM messages
- transparently transfer others

MEPs and MIPs (cont.)



How do we use all this ?

MEF-30 Service OAM FM   and MEF-xx Service OAM PM

Describe the use of OAM for Carrier Ethernet networks, such as

- which Y.1731/802.1 features should be implemented
- which Y.1731/802.1 messages should be used
- what MA names should be used
- where to put MEPs
- how to use MEG levels
- minimum number of EVCs that must be supported
- what should be reported and how

Y.1564 (ex Y.156sam)

Ethernet **S**ervice **A**ctivation Test **M**ethodology

Details how to check configuration and performance *before* EVC activation

Y.1564

Ethernet Service Activation Test Methodology

Replaces widespread proprietary use of RFC 2544 benchmarking

Tests that desired performance level can be achieved, including

- CIR, EIR (and optionally CBS and EBS for bursting)
- Traffic policing
- Rate, loss, delay, delay variation, availability (measured simultaneously)

Testing in two steps :

- Service Configuration Test – each service separately
- Service Performance Test – all services together

Performance testing may be for :

- 15 minutes (new service on operational network)
- 2 hours (single operator network)
- 24 hours (multiple operator networks)

APS

Automatic Protection Switching (APS)

is a functionality of carrier-grade transport networks

is often called resilience

since it enables service to quickly recover from failures

is required to ensure high reliability and availability

APS includes :

- detection of *failures* (signal fail or signal degrade) on a *working channel*
- switching traffic *transmission* to a *protection channel*
- selecting traffic *reception* from the protection channel
- (optionally) reverting back to the working channel once failure is repaired

Using STP and LAG

STP and **RSTP** automatically converge to a loop-free topology

RSTP converges in about the same time as STP
but can reconverge after a topology change in less than 1 second

Thus RSTP *can* be used as a protection mechanism

However, the switching time will be many tens of ms to 100s of ms

Link **AG**gregation also detects failures (using physical layer or LACP)
and automatically removes failed links

Thus LAG too can be used as a primitive protection mechanism

When used this way it is called *worker/standby* or *N+N mode*

However, LACP, which LAG uses for CC, is a slow protocol
and so is limited to 5 per second

The restoration time will be on the order of 1 second

G.8031

In 2006 Q9 of SG15 in the ITU-T produced
G.8031 Linear Ethernet Protection Switching

G.8031 uses standard Ethernet formats, but is incompatible with STP

The standard addresses

- point-to-point VLAN connections
- SNC (local) protection class
- 1+1 and 1:1 protection types
- unidirectional and bidirectional switching for 1+1
- bidirectional switching for 1:1
- revertive and nonrevertive modes
- 1-phase signaling protocol

G.8031 uses Y.1731 OAM CCM messages in order to detect failures

G.8031 defines a new OAM opcode (39) for APS signaling messages

Switching times should be under 50 ms (only holdoff timers when groups)

G.8031 signaling

The APS signaling message looks like this :

MEL (3b)	VER=0 (5b)	OPCODE=39 (1B)	FLAGS=0 (1B)	OFFSET=4 (1B)
req/state (4b)	prot. type (4b)	requested sig (1B)	bridged sig (1B)	reserved (1B)
END=0 (1B)				

- – regular APS messages are sent 1 per 5 seconds
- – after change 3 messages are sent at max rate (300 per sec)

where

- req/state identifies the message (NR, SF, WTR, SD, forced switch, etc)
- prot. type identifies the protection type (1+1, 1:1, uni/bidirectional, etc.)
- requested and bridged signal identify incoming / outgoing traffic
  since only 1+1 and 1:1 they are either null or traffic (all other values reserved)

G.8031 1:1 revertive operation

In the normal (NR) state :

- head-end and tail-end exchange CCM (at 300 per second rate) on both working and protection channels
- head-end and tail-end exchange NR APS messages on the protection channel (every 5 seconds)

When a failure appears in the working channel

- tail-end stops receiving 3 CCM messages on working channel
- tail-end enters SF state
- tail-end sends 3 SF messages at 300 per second on the APS channel
- tail-end switches selector (bi-d and bridge) to the protection channel
- head-end (receiving SF) switches bridge (bi-d and selector) to protection channel
- tail-end continues sending SF messages every 5 seconds
- head-end sends NR messages but with bridged=normal

When the failure is cleared

- tail-end leaves SF state and enters WTR state (typically 5 minutes, 5..12 min)
- tail-end sends WTR message to head-end (in nonrevertive - DNR message)
- tail-end sends WTR every 5 seconds
- when WTR expires both sides enter NR state

Ethernet rings ?

Ethernet has become carrier grade :

- deterministic connection-oriented forwarding
- OAM
- synchronization

The only thing missing is ring protection

However, Ethernet and ring architectures don't go together

- Ethernet has no TTL, so looped traffic will loop forever
- STP builds trees out of any architecture – no loops allowed

There are two ways to make an Ethernet ring

- open loop
  - cut the ring by blocking some link
  - when protection is required - *block the failed link*
- closed loop
  - disable STP (but avoid infinite loops in some way !)
  - when protection is required - *steer* and/or *wrap* traffic

Ethernet ring protocols

Open loop methods

- G.8032 (ERPS)
- rSTP (ex 802.1w)
- RFER (RAD)
- ERP (NSN)
- RRST (based on RSTP)
- REP (Cisco)
- RRSTP (Alcatel)
- RRPP (Huawei)
- EAPS (Extreme, RFC 3619)
- EPSR (Allied Telesis)
- PSR (Overture)

Closed loop methods

- RPR (IEEE 802.17)
- CLEER and NERT (RAD)

G.8032

Q9 of SG15 produced G.8032 between 2006 and 2008

G.8032 is similar to G.8031

- strives for 50 ms protection (< 1200 km, < 16 nodes)
  - but here this number is deceiving as MAC table is flushed
- standard Ethernet format but incompatible with STP
- uses Y.1731 CCM for failure detection
- employs Y.1731 extension for R-APS signaling (opcode=40)
- R-APS message format similar to APS of G.8031

  (but between every 2 nodes and to MAC address 01-19-A7-00-00-01)
- revertive and nonrevertive operation defined

However, G.8032 is more complex due to

- requirement to avoid loop creation under any circumstances
- need to localize failures
- need to maintain consistency between all nodes on ring
- existence of a special node (RPL owner)

RPL

G.8032 defines the **R**ing **P**rotection **L**ink (RPL)
as the link to be blocked (to avoid closing the loop) in NR state

One of the 2 nodes connected to the RPL
is designated the *RPL owner*

Unlike RAD's RFER

- there is only one RPL owner
- the RPL and owner are designated before setup
- operation is usually revertive

All ring nodes are simultaneously in 1 of 2 modes – idle or protecting

- in idle mode the RPL is blocked
- in protecting mode the failed link is blocked and RPL is unblocked
- in revertive operation
  once the failure is cleared the block link is unblocked
  and the RPL is blocked again

G.8032 revertive operation

In the idle state :

- adjacent nodes exchange CCM at 300 per second rate (including over RPL)
- exchange NR RB (RPL Blocked) messages in dedicated VLAN every 5 seconds (but *not* over RPL)
- R-APS messages are never forwarded

When a failure appears between 2 nodes

- node(s) missing CCM messages *peek twice* with holdoff time
- node(s) block failed link and flush MAC table
- node(s) send SF message (3 times @ max rate, then every 5 sec)
- node receiving SF message will check priority and unblock any blocked link
- node receiving SF message will send SF message to its other neighbor
- in stable protecting state SF messages over every unblocked link

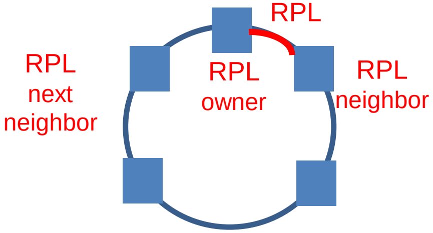
When the failure is cleared

- node(s) detect CCM and start guard timer (blocks acting on R-APS messages)
- node(s) send NR messages to neighbors (3 times @ max rate, then every 5 sec)
- RPL owner receiving NR starts WTR timer
- when WTR expires RPL owner blocks RPL, flushes table, and sends NR RB
- node receiving NR RB flushes table, unblocks any blocked ports, sends NR RB

G.8032-2010

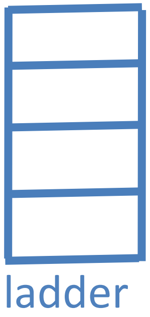
After coming out with G.8032 in 2008 (*G.8032v1*)
the ITU came out with G.8032-2010 (G.8032v2) in 2010

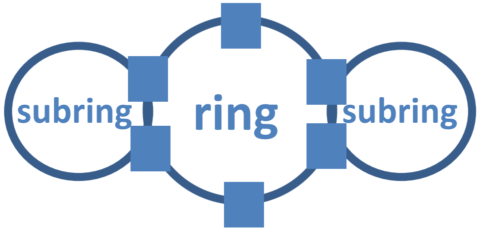
This new version is not *backwards-compatible* with v1
but a v2 node must support v1 as well (but then operation is according to v1)



Major differences :

- 2 designated nodes – *RPL owner* and *RPL neighbor node* and for optional *flush-optimization* "next neighbor node"
- significant changes to
  - state machine
  - priority logic
  - commands (forced/manual/clear) and protocol
- new **W**ait **T**o **B**lock timer
- supports more general topologies (sub-rings)
  - ladders (was *For Further Study* in v1)
  - multi-ring
- ring topology discovery
- virtual channel based on VLAN or MAC address




subring
ring
subring