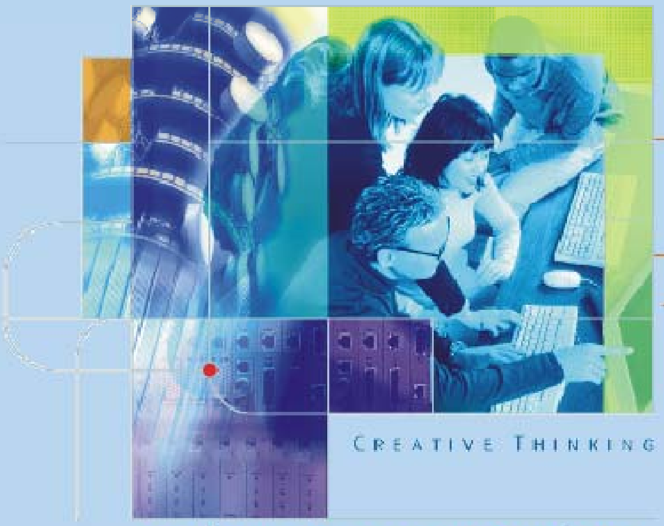
RAD

data communications

www.rad.com

pseudowires

a *short* introduction



Unique Access Solutions

Yaakov (J) Stein
Chief Scientist
RAD Data Communications

July 2010

Contents

- pseudowires
- PW encapsulations
- TDM PWs
- Ethernet PWs
- L2VPNs
- OAM for PWs
- PWE control protocol

Pseudowires

Pseudowire (PW): A mechanism that emulates the essential attributes of a native service while transporting over a packet switched network (PSN)

Pseudowires

Packet Switched Network (PSN)

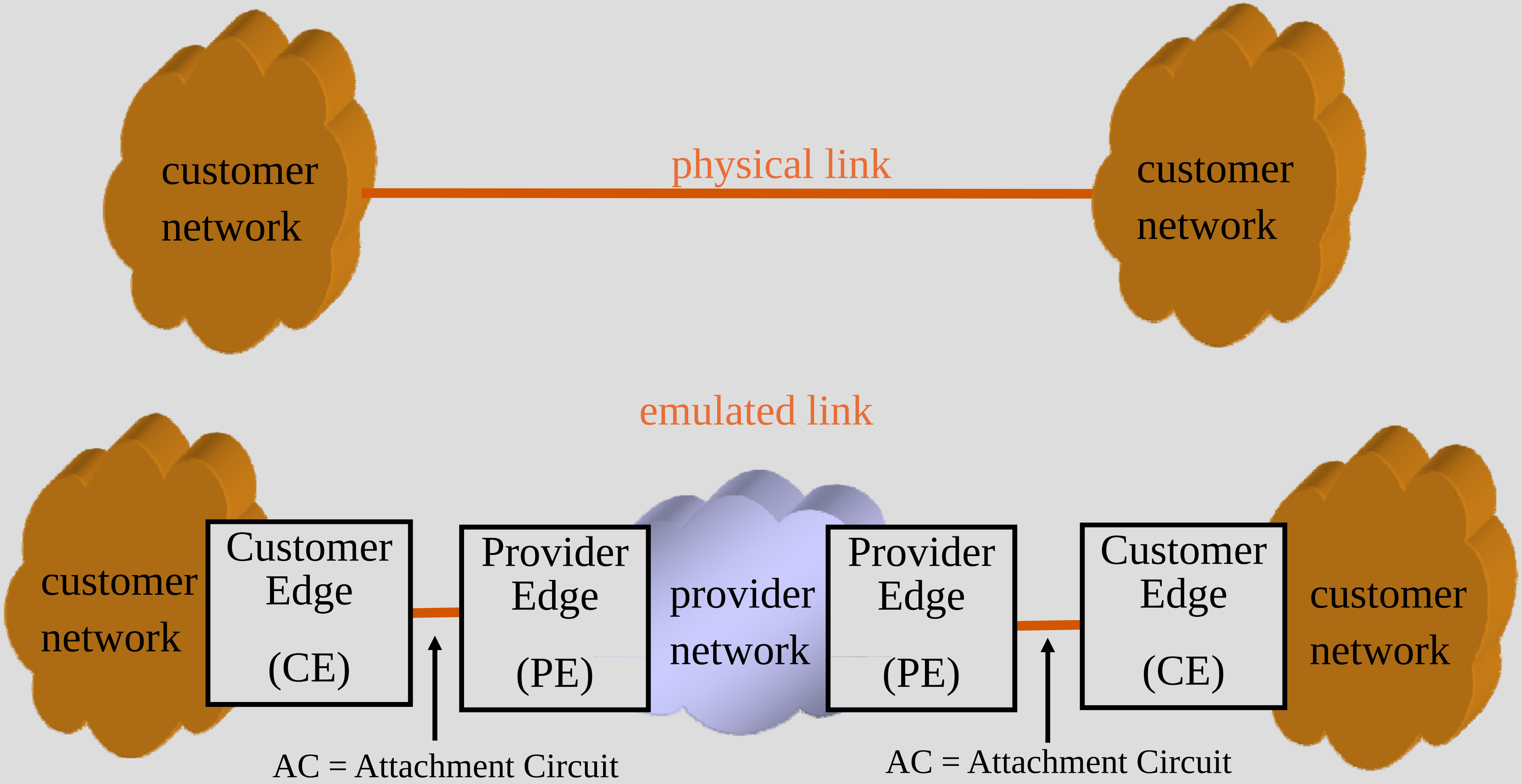
- a network that forwards packets
- IPv4, IPv6, MPLS, Ethernet

a pseudowire (PW) is a mechanism to tunnel traffic through a PSN

PWs are usually bidirectional (unlike MPLS LSPs)

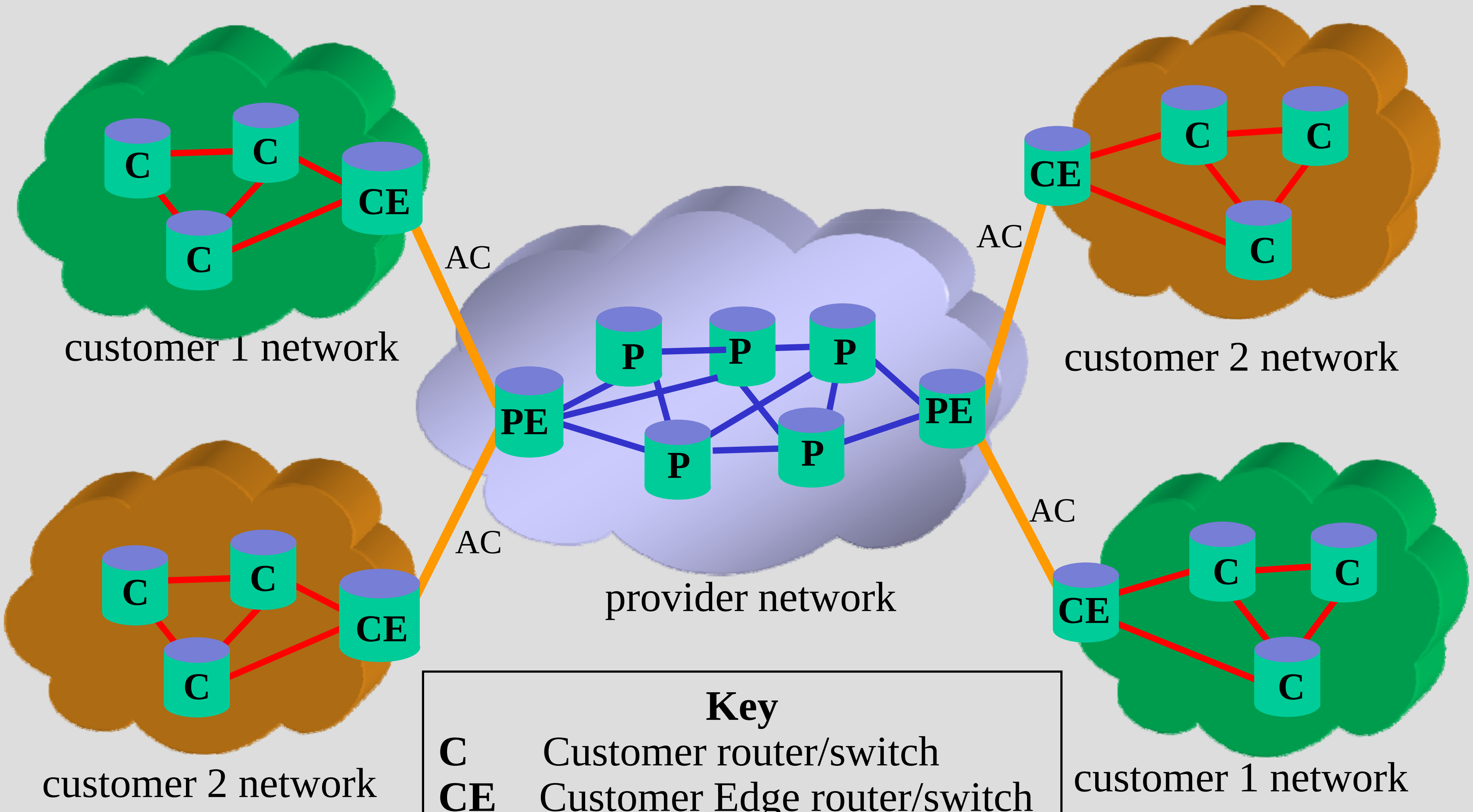
PW architecture is an extension of VPN architecture

Basic (L2,L3)VPN model



transported traffic may be **L3** (e.g. IP) or **L2** (e.g. Ethernet)

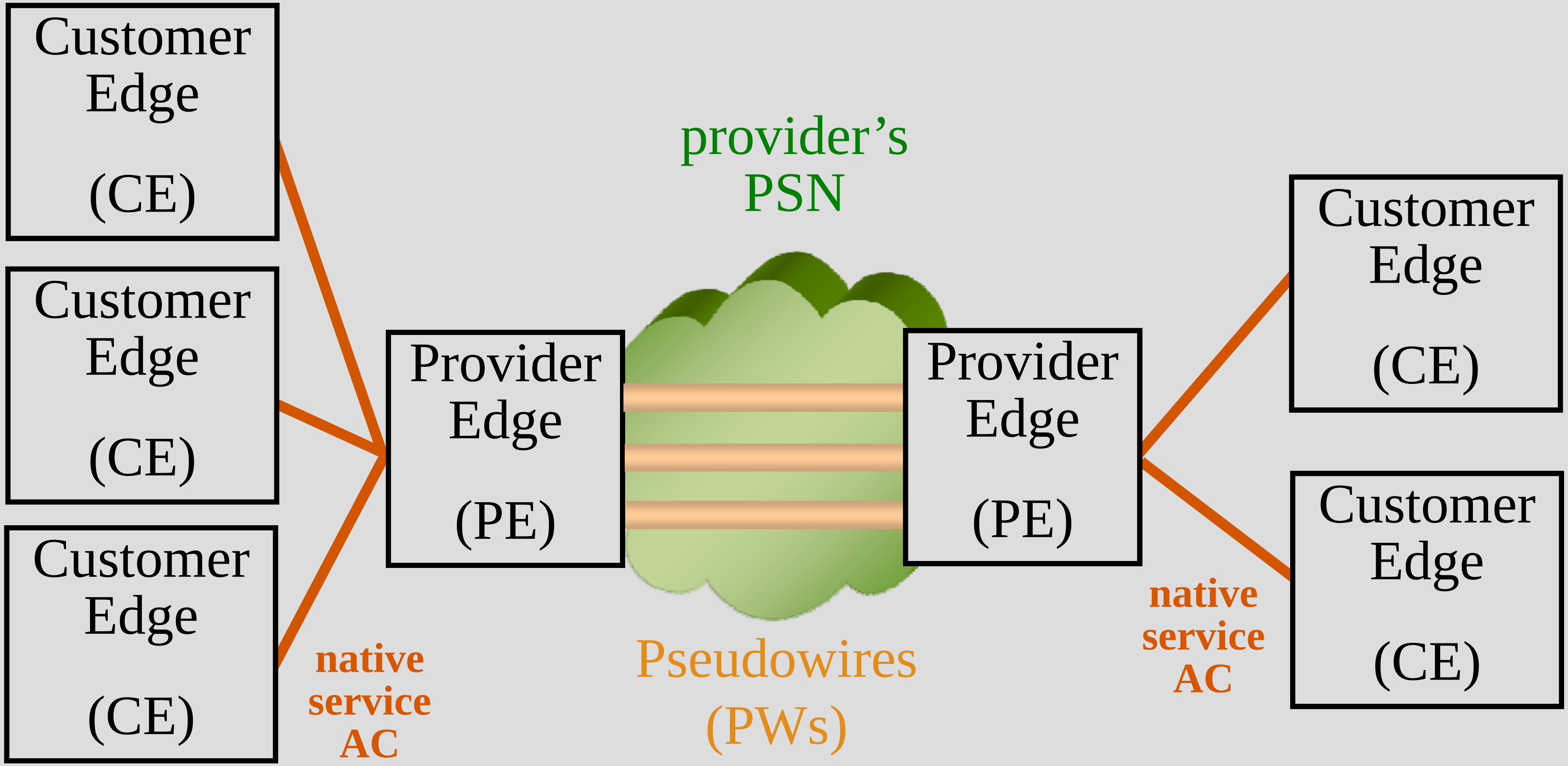
(L2,L3)VPN in more detail



Key

C Customer router/switch
CE Customer Edge router/switch
P Provider router/switch
PE Provider Edge router/switch

Pseudowire Emulation (provider) Edge to Edge PWE^3



Native services defined in IETF PWE3

The PWE3 Working Group in the IETF
has defined the following native services :

- ATM (port mode, cell mode, AAL5-specific modes) RFC 4717, 4816
- Frame Relay RFC 4619
- HDLC/PPP RFC 4618
- TDM (E1, T1, E3, T3) RFC 4553, 5086, 5087
- SONET/SDH (CEP) RFC 4842
- Fiber channel
- Multiprotocol packet service
- Ethernet (raw, VLAN-aware) RFC 4448

Note that most are *legacy* services
but the most interesting service today is *Ethernet*

What else ?

PWs emulate the native service –
but may not completely reproduce it (applicability statement)

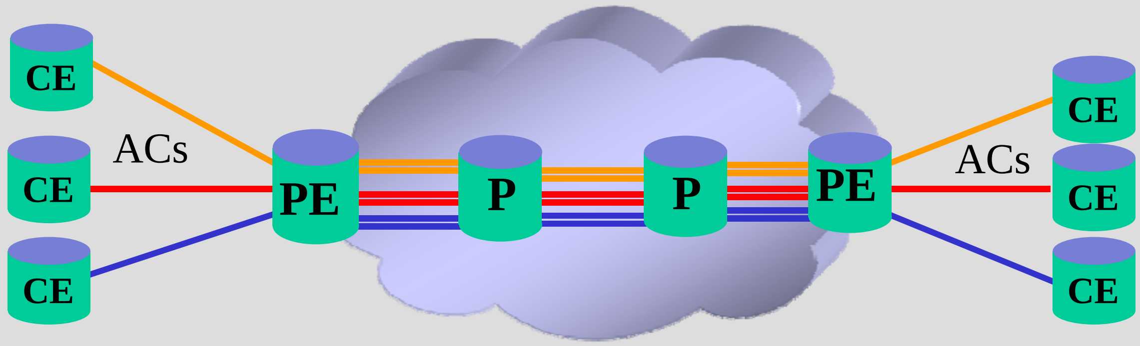
PW packets are not self-describing (like MPLS, unlike IP or Ethernet)

An demultiplexing identifier is provided to uniquely identify PWs

We may also need :

- Native Service Processing (NSPs)
- PW-layer OAM (at least Continuity Check)
- PW control protocol
- Load balancing
- Protection (redundancy) mechanism
- Multisegment PWs (MS-PWs)

Simplistic MPLS solution



each customer network mapped to pair of (unidirectional) LSPs

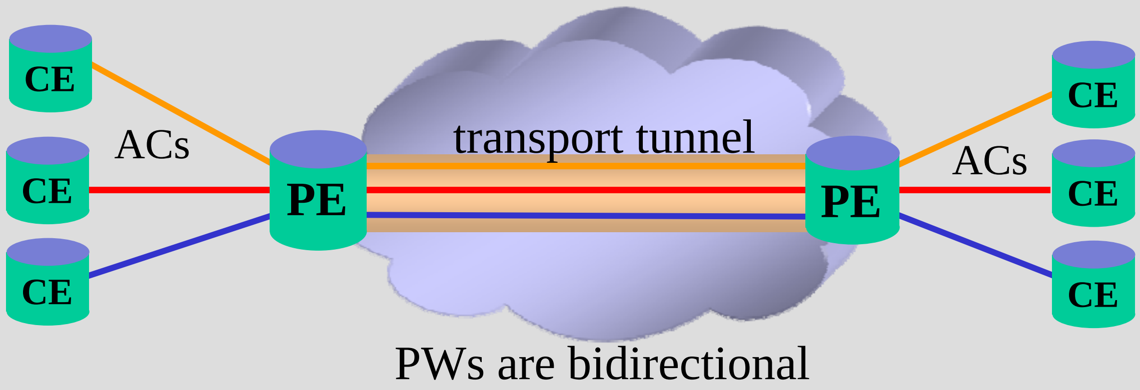
supports various AC technologies

each native packet/frame encapsulated with MPLS label

scaling problem:

- requires large number of LSPs
- P-routers need to be aware of customer networks

(Martini) Pseudowires



MPLS (outer) label
PW (inner) label
payload

transport MPLS tunnel set up between PEs

multiple PWs may be set up inside tunnel

native packet/frame encapsulated with 2 labels

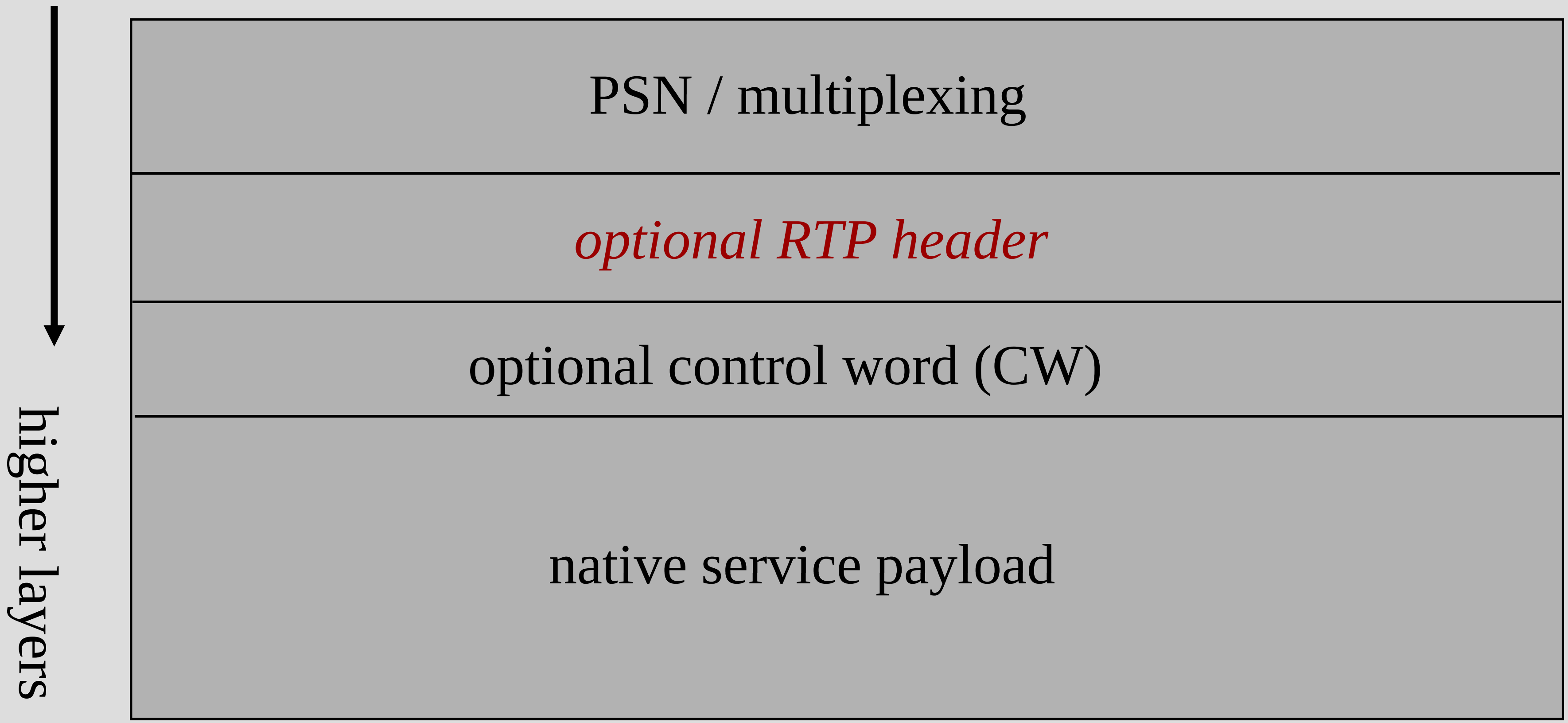
PEs contain the PW *interworking function*

P-routers are unaware of individual customer networks

Pseudowire encapsulations

Encapsulation: In order to enable transport over the PSN, native service Protocol Data Units (PDUs) must be inserted into packets of the appropriate format.
This is usually accomplished by adding headers.

Generic PWE3 packet format



We will ignore the RTP header in the following

MPLS PSN

MPLS PSN

tunnel label(s)	PW label	CW	**Payload**

MPLS over Ethernet

Ethernet MAC header (DA SA, …)
MPLS label stack
PW label (bottom of label stack S=1)
Control Word
native service payload
Ethernet FCS

IP PSN using L2TPv3

L2TPv3 – RFC 3931 (without UDP)

IP header (5*4 B) IP protocol 115
session ID (4 B)
optional cookie (4 or 8 B)
control word (4 B)
native service payload

IP PSN using UDP with PW label in destination port

UDP/IP for TDM PWs

	IP header (5*4 B)
UDP header (8B)	return PW label (2 B)
	PW label (2 B)
	UDP length and checksum (4 B)
	control word (4 B)
	native service payload

PW labels between C000 and FFFF

IP PSN using UDP with PW label in source port

UDP/IP - 5087

	IP header (5*4 B)
UDP header (8B)	PW label (2 B)
	well known port (085E) (2 B)
	UDP length and checksum (4 B)
	control word (4 B)
	native service payload

PW labels between C000 and FFFF

IP PSN using RFC 4023

MPLS over IP using RFC 4023

IP header (5*4 B) IP protocol 47(GRE) or 137(MPLS)
optional GRE header (8 B) GRE protocol 08847(MPLS Ethertype)
PW label (4 B)
control word (4 B)
native service payload

PWE Control Word (RFC 4385)

0 0 0 0	flags	FRG	Length	Sequence Number

0 0 0 0

- identifies packet as PW (not IP – which has 0100 or 0110)
- gives clue to ECMP mechanisms
- 0001 for PWE associated channel (ACh) used for OAM

Flags (4 b)

- not all encapsulation define
- used to transport native service fault indications

FRG

- may be used to indicate payload fragmentation
  - 00 = unfragmented   01 = 1st fragment
  - 10 = last fragment   11 = intermediate fragment

Length (6 b)

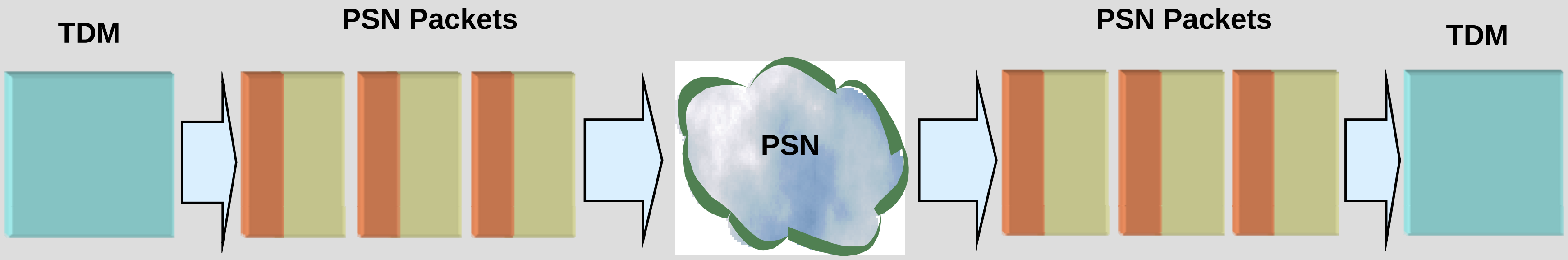
- used when packet may be padded by L2

Sequence Number (16 b)

- used to detect packet loss / misordering
- processing slightly different in TDM PWs

TDM PWs

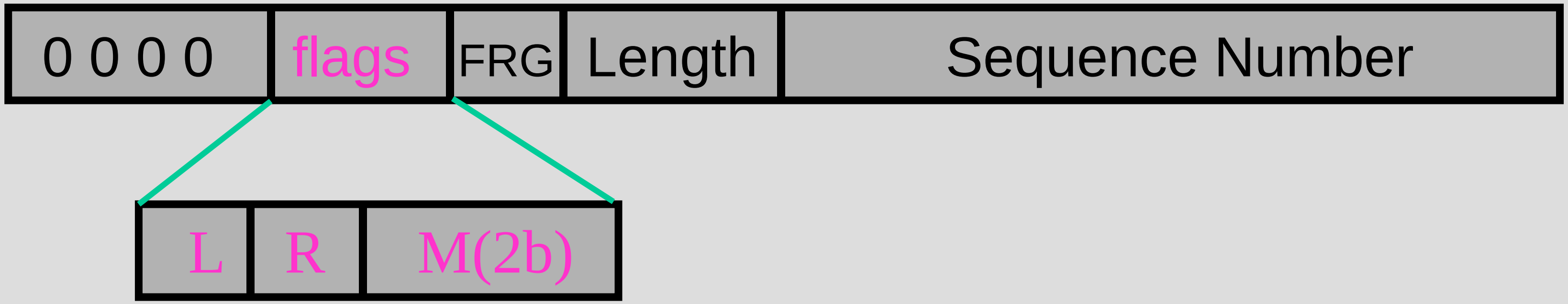
TDM PW Protocol Processing



Steps in TDM PW processing

- The synchronous bit stream is segmented
- The TDM segments may be *adapted*
- TDMoIP control word is prepended
- PSN headers are prepended (encapsulation)
- Packets are transported over PSN to destination
- PSN headers are utilized and stripped
- Control word is checked, utilized and stripped
- TDM stream is reconstituted (using adaptation) and played out

Flags



The PWE control word has 2 flags: L and R and a 2-bit field: M

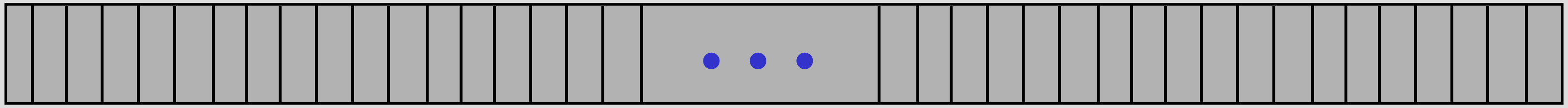
They are used in the following way :

- L is set to indicate a forward defect (AIS)
- R may be set to indicate a reverse defect (RDI)
- M can modify the meaning of the FDI

TDM Structure

handling of TDM depends on its structure

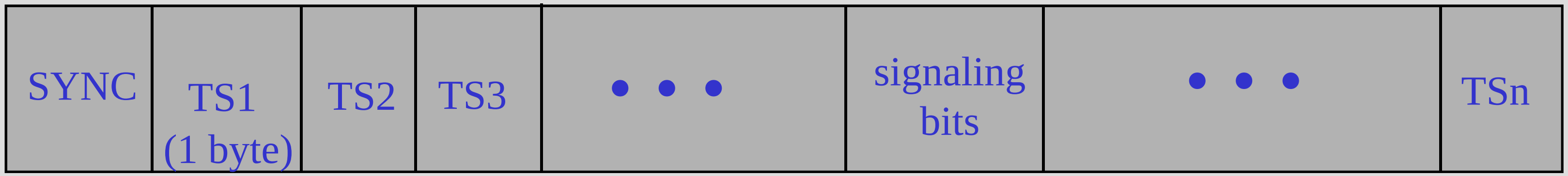
unstructured TDM (TDM = arbitrary stream of bits)



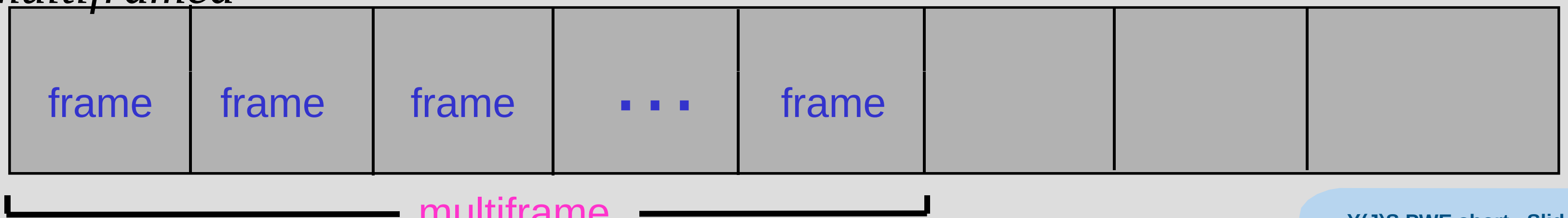
structured TDM

framed (8000 frames per second)

channelized (single byte timeslots)



multiframed



TDM transport types

Structure-agnostic transport (SAToP – RFC4553)

- for unstructured TDM
- even if there is structure, we ignore it
- simplest way of making payload
- OK if network is well-engineered

Structure-aware transport (CESoPSN – RFC 5086, TDMoIP – RFC 5087)

- take TDM structure into account
- must decide which level of structure (frame, multiframe, …)
- can overcome PSN impairments (PDV, packet loss, etc)

The Frame Alignment Signal (FAS) is maintained at PSN egress

Overhead bits *may* be transported

Structure Agnostic Transport

SAToP encapsulates N bytes of TDM in each packet
There is no TDM frame alignment !

N must be constant and preconfigured
If packets are lost, the egress knows how many TDM bytes to fill in

Default values for N :

- E1 – 256 B
- T1 – 192 B
- E3 and T3 – 1024 B

For T1 there is an optional special mode called *octet aligned mode* that adds 7 bits of padding to every 193 consecutive bits (to make 25 B)

Structure aware encapsulations

Structure-locked encapsulation (CESoPSN)

headers	TDM structure	TDM structure	TDM structure	TDM structure

Structure-indicated encapsulation (TDMoIP – AAL1 mode)

headers	AAL1 subframe	AAL1 subframe	AAL1 subframe	AAL1 subframe

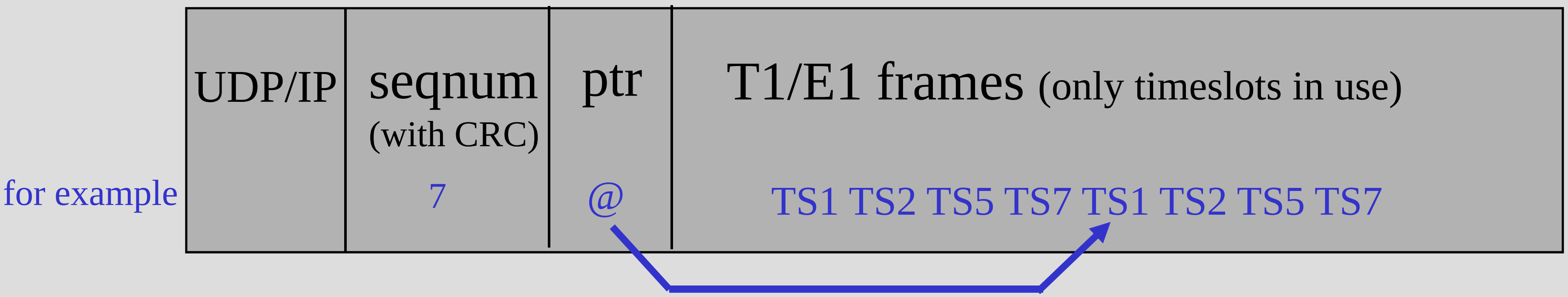
Structure-reassembled encapsulation (TDMoIP – AAL2 mode)

headers	AAL2 minicell	AAL2 minicell	AAL2 minicell	AAL2 minicell

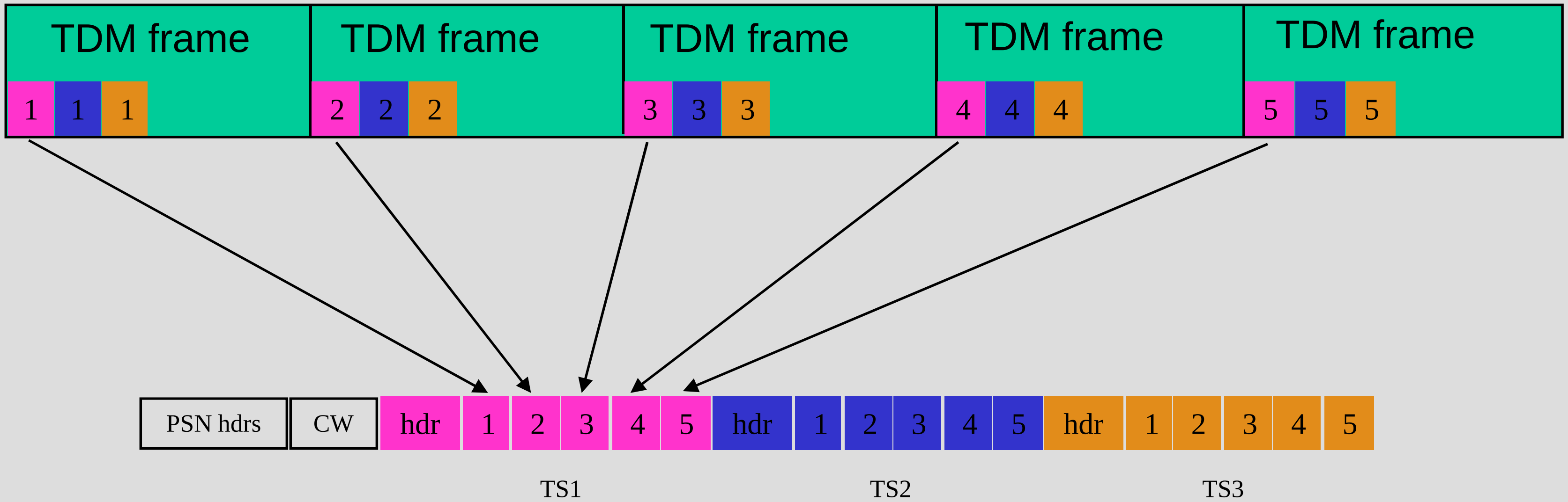
Structure indication - AAL1

For robust emulation:

- adding a packet sequence number
- adding a pointer to the next superframe boundary
- only sending timeslots in use
- allowing multiple frames per packet



Structure reassembly - AAL2



AAL1 is inefficient when timeslots are dynamically allocated

- each minicell consists of a header and buffered data
- minicell header contains:
  - CID (Channel IDentifier)
  - LI (Length Indicator) = length-1
  - UUI (User-User Indication) counter + payload type ID

CAS and CCS signaling

Channel Associated Signaling is carried in the T1/E1
(T1 uses robbed bits , E1 uses a dedicated time slot - TS16)

Unlike VoIP, TDM PWs transparently transport CAS
and may add a separate *signaling substructure* (ATM-like)
that carries the CAS signaling bits

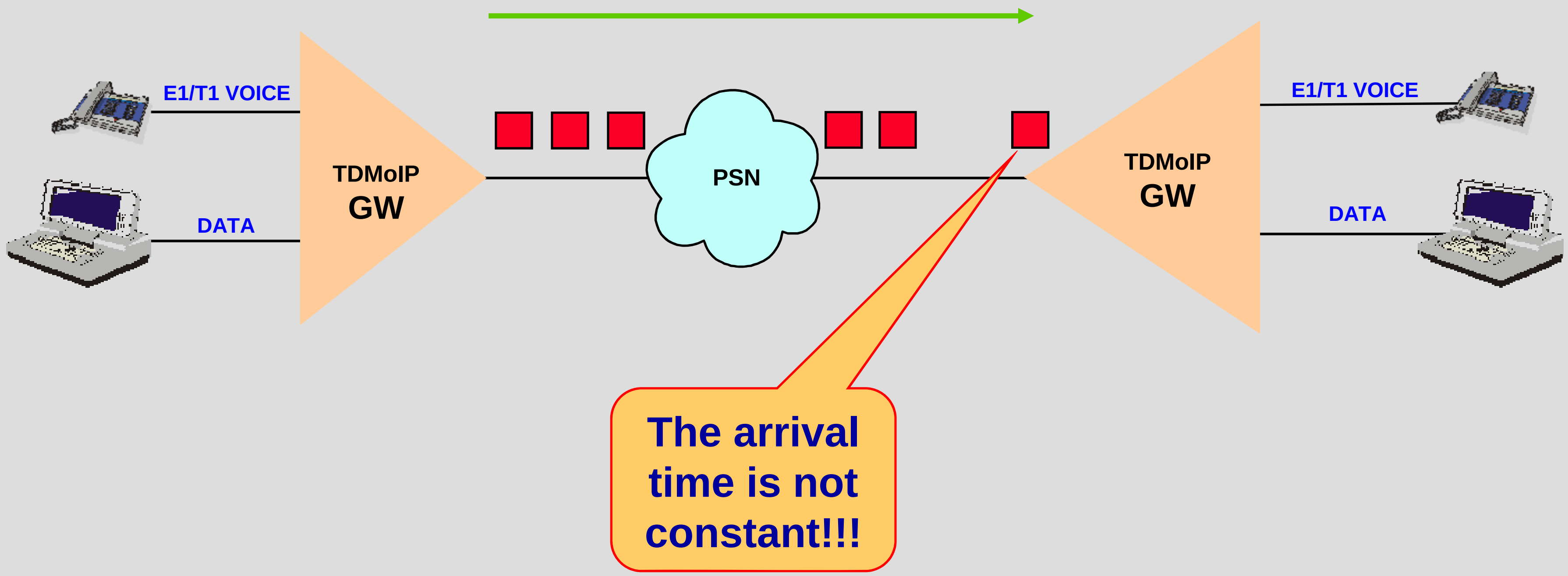
CESoPSN must respect CAS multiframe boundaries
Thus it may fragment the mutiframe (using the CW FRG bits)
and append the substructure to the last fragment

With HDLC-based trunk associated Common Chanel Signaling
(e.g., ISDN PRI signaling, SS7)

The CCS may simply be left where it is
But sometimes it is worthwhile to extract it
and transport it using a separate HDLC PW

PSN - Delay and PDV

- PSNs do not carry timing
  - *clock recovery* required for TDMoIP
- PSNs introduce *delay* and *packet delay variation* (PDV)
  - Delay degrades perceived voice quality
  - PDV makes clock recovery difficult



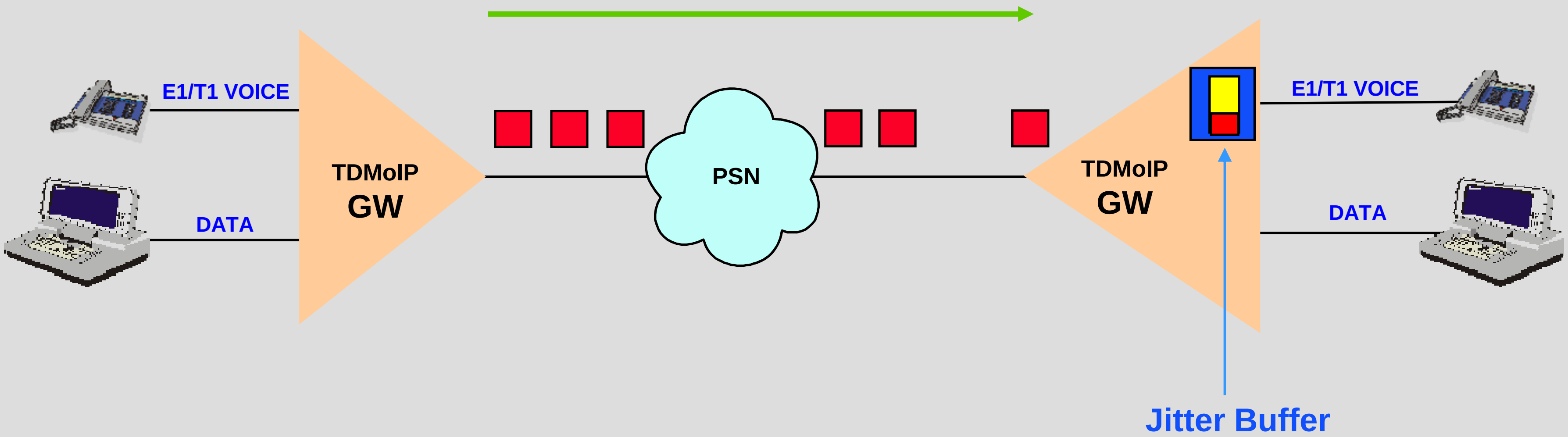
Jitter Buffer

Arriving TDMoIP packets written into *jitter buffer*

Once buffer filled 1/2 can start reading from buffer

Packets read from jitter buffer at constant rate

- How do we know the right rate?
- How do we guard against buffer overflow/underflow?



Adaptive Clock Recovery

The packets are injected into network ingress at times T_n

For TDM the source packet rate R is constant

$$T_n = n / R$$

The network delay D_n can be considered to be the sum of
typical delay d and random delay variation V_n

The packets are received at network egress at times t_n

$$t_n = T_n + D_n = T_n + d + V_n$$

By proper averaging/filtering

$$\langle t_n \rangle = T_n + d = n / R + d$$

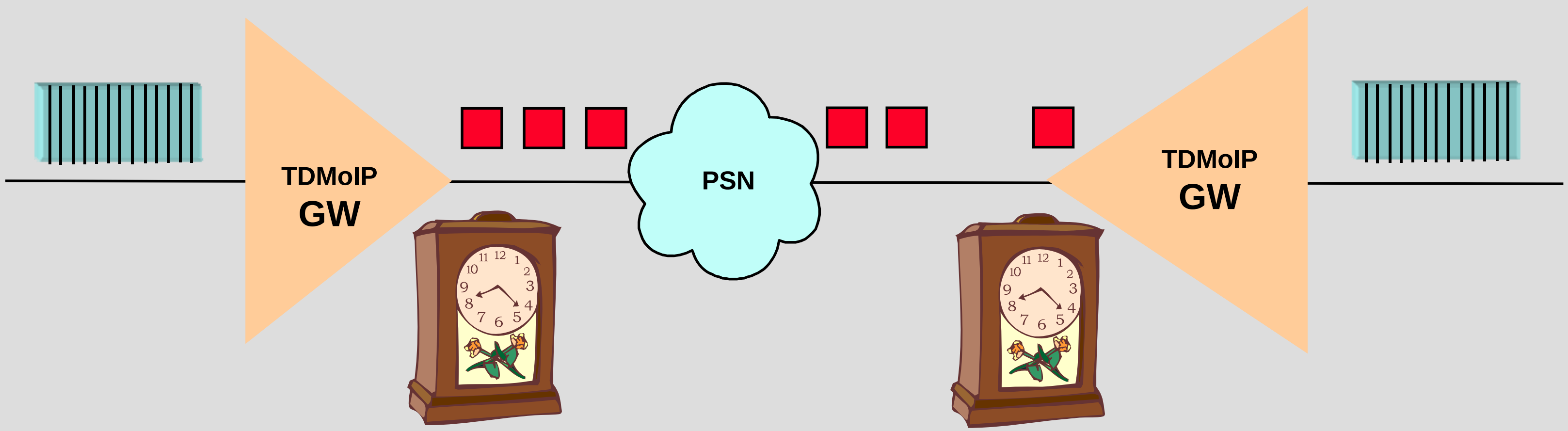
and the packet rate R has been recovered

Differential (common clock) Clock Recovery

Sometimes we have an reference clock frequency available
at both IWFs (PEs) (e.g., physical layer clock, GPS, PRCs_

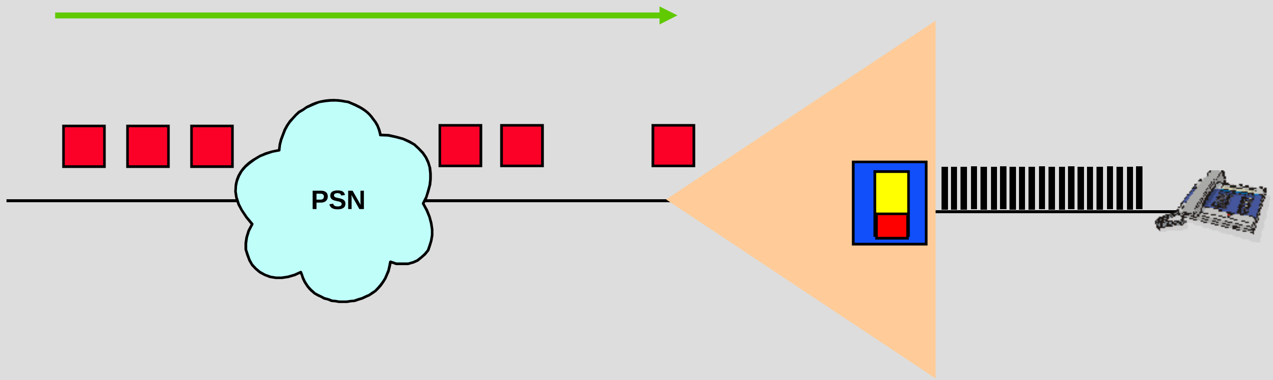
Then at ingress we can encode the frequency difference
between the TDM source frequency and the reference

And at egress reconstruct the TDM source frequency using the reference



Handling of packet loss

In order to maintain TDM timing at egress
SOMETHING must be output
towards the TDM interface when a packet is lost



Packet Loss Concealment methods:

- fixed
- replay
- interpolation

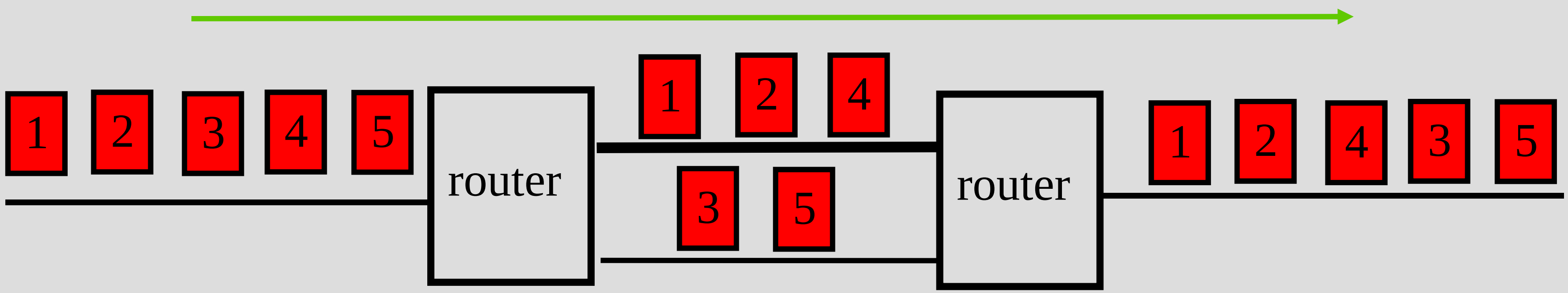
Mis-ordering

In a perfect network all packets should arrive in proper order

In real networks, some packets are delayed (or even duplicated!)

Misordering is caused by parallel paths

– aggravated by load balancing mechanisms



Misordering can be handled by

- Reordering (from jitter buffer)
- Handling as packet loss and dropping later

Ethernet PWs

Ethernet limitations

Ethernet LAN is the most popular **LAN**
but Ethernet can not be made into a **WAN**

- Ethernet is limited in distance between stations
- Ethernet is limited in number of stations on segment
- Ethernet is inefficient in finding destination address
- Ethernet only prunes network topology, does not *route*

so the architecture that has emerged is Ethernet *private networks*
connected by *public networks* of other types (e.g. IP)

Traditional WAN architecture

this model is sensible when traffic contains a given higher layer

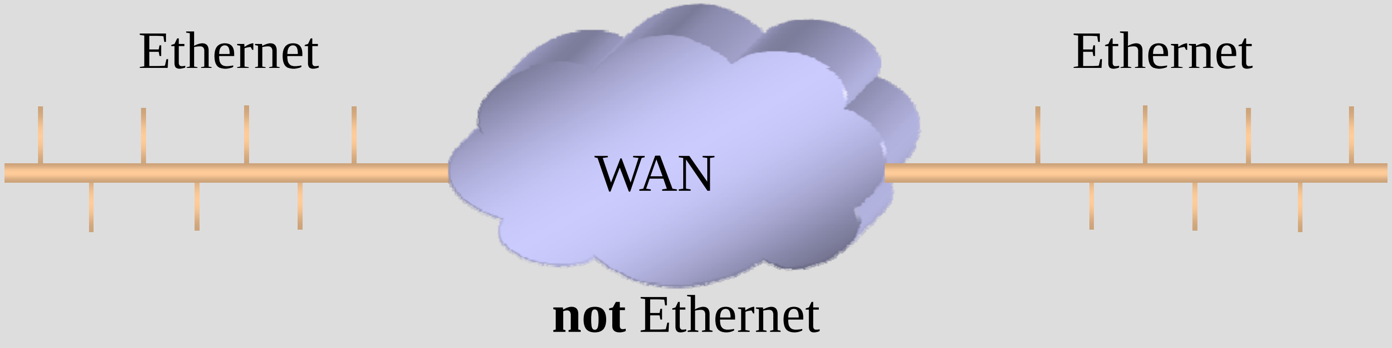
Ethernet header is removed at ingress and a new header added at egress

this model is *not* transparent Ethernet LAN interconnect

- Ethernet LANs with multiple higher layer packet types (e.g. IPv4, IPv6, IPX, SNA, CLNP, etc.) can't be interconnected
- raw L2 Ethernet frames can not be sent

the Ethernet layer is *terminated* at WAN ingress

the traffic is no longer Ethernet at all

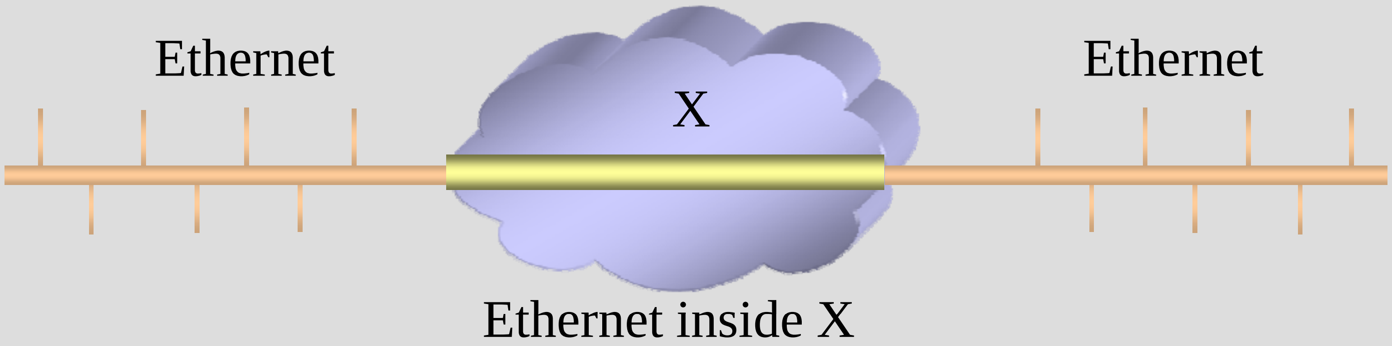


Tunneling Ethernet frames

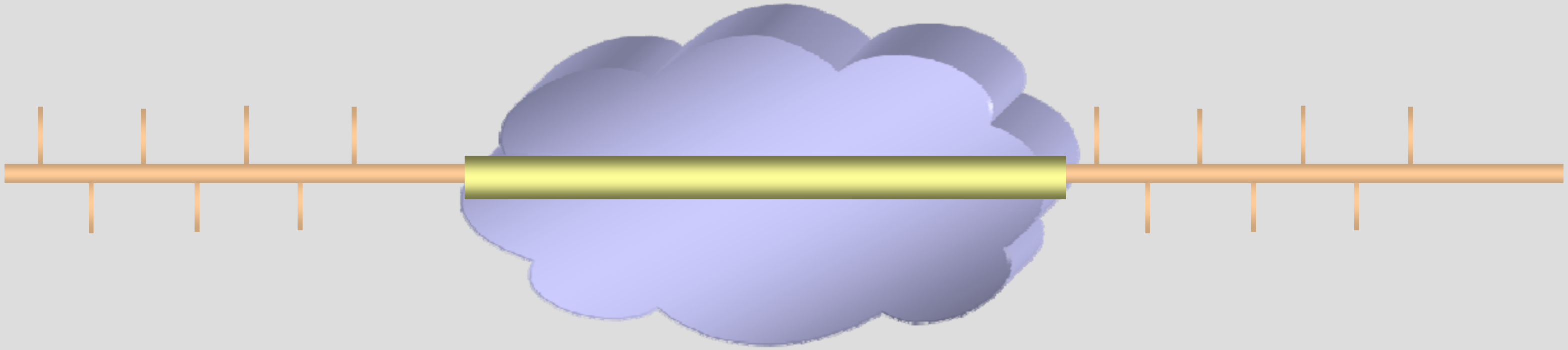
users with multiple sites want to connect their LANs
so that all locations appear to be on the same LAN

this requires *tunneling* of *all* Ethernet L2 frames (not only IP)
between one LAN and another

the entire Ethernet frame needs to be preserved
(except perhaps the FCS which can be regenerated at egress)



Ethernet over X



Ethernet frames can be carried over various WANs

HDLC: not standardized, Cisco-HDLC

FR: RFC2427 / STD0055 (ex 1490)

ATM: RFC2684 / (ex 1483), LANE

SONET/SDH/PDH: PoS (RFC 2615 ex RFC1619),
LAPS (X.85/X.86), GFP (G.7041)

PSN: Ethernet PW

Ethernet PW (RFC 4448)

can transport tagged or untagged Ethernet frames

- if tagged encapsulation can be "raw mode" or "tagged mode"
- tagged mode processes (swaps) SP tags

control word is optional

- even if control word is used, sequence number if optional

standard mode – FCS is stripped and regenerated

- FCS retention mode (RFC 4720) allows retaining FCS

Ethernet Pseudowire packet (MPLS)

tunnel label	PW label	control word	**Ethernet Frame**

Ethernet Frame usually has FCS stripped, but may retain it
SP tags may be modified

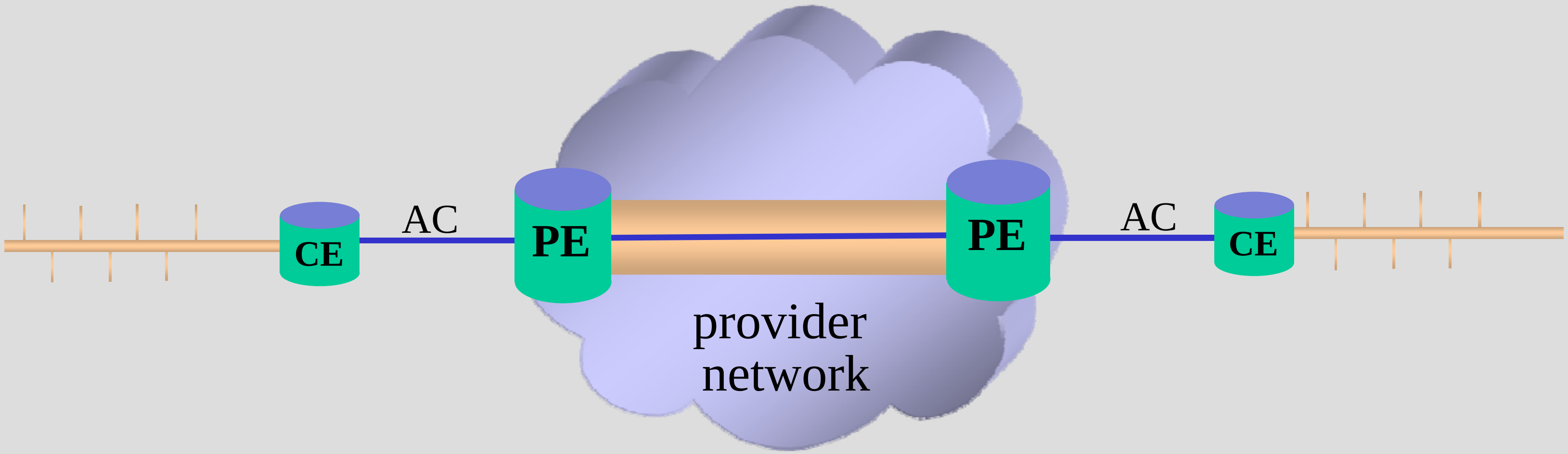
optional control word

generation and processing of sequence number is optional

0000	reserved	Sequence Number (16b)

L2VPNs

VPWS



Virtual Private Wire Service is a L2 point-to-point service

it emulates a *wire* supporting the Ethernet physical layer

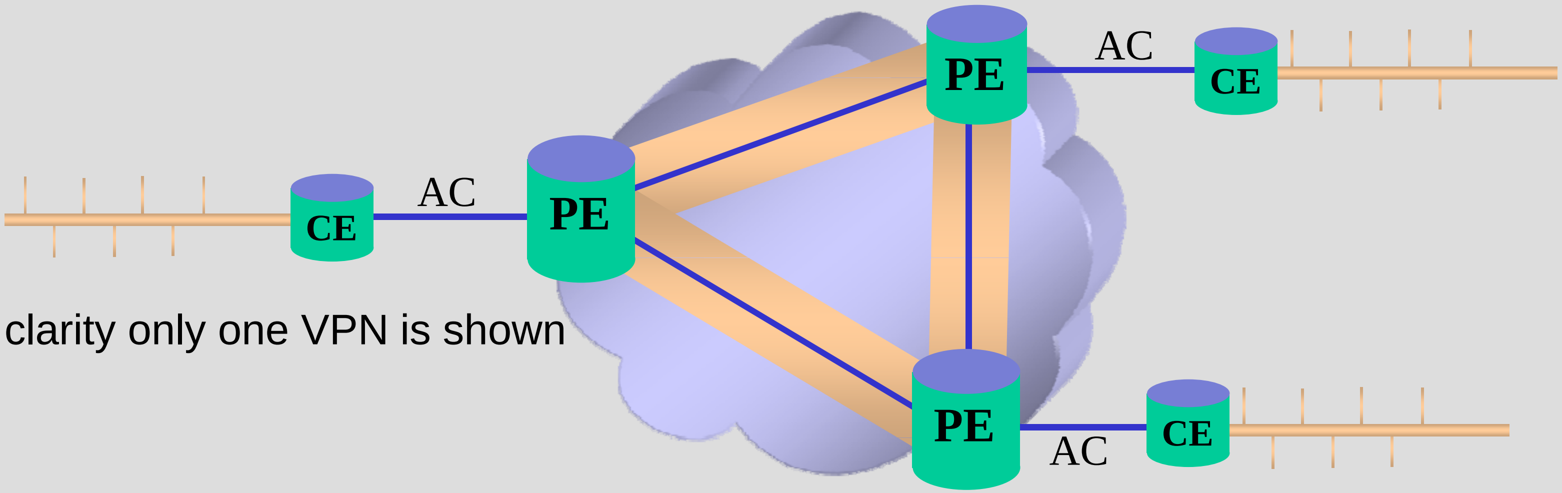
set up MPLS tunnel between PEs

set up Ethernet PW inside tunnel

CEs appear to be connected by a single L2 circuit

(can also make VPWS for ATM, FR, etc.)

VPLS



for clarity only one VPN is shown

VPLS emulates a LAN over an MPLS network

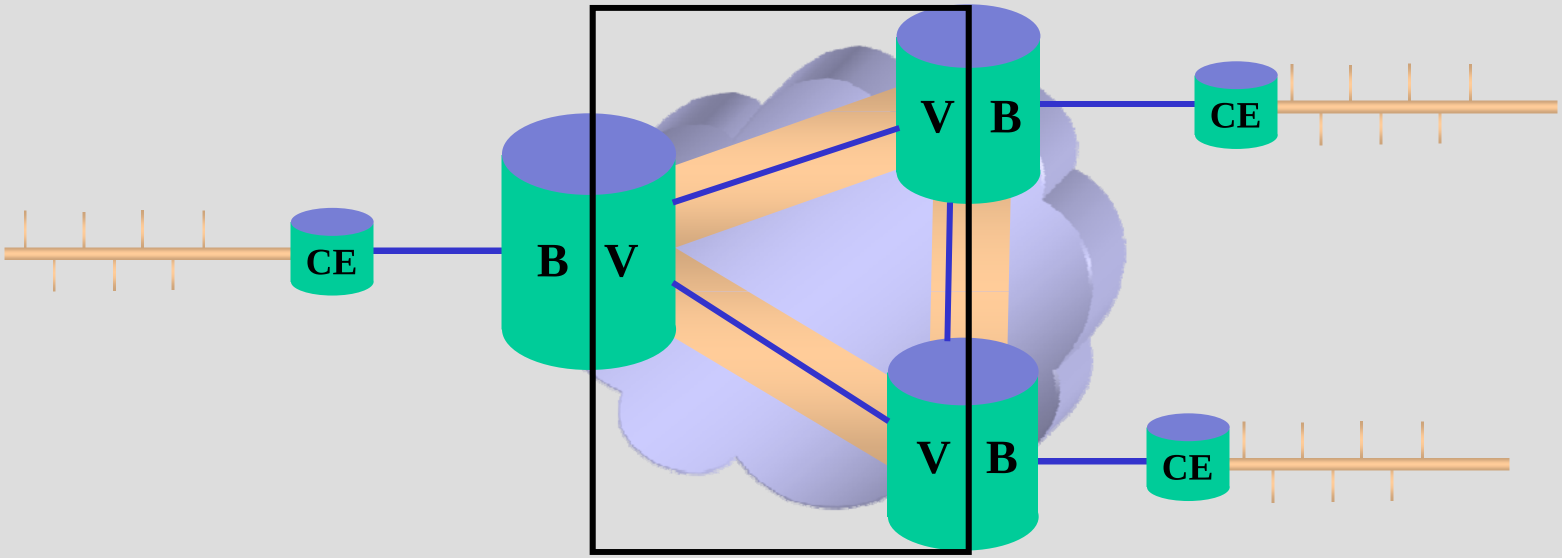
set up MPLS tunnel between every pair of PEs (full mesh)
set up Ethernet PW inside tunnels, for each VPN instance

CEs appear to be connected by a single LAN

PE must know where to send Ethernet frames …

but this is what an Ethernet bridge does

VPLS



a VPLS-enabled PE has, in addition to its MPLS functions:

- VPLS code module (IETF drafts)
- Bridging module (standard IEEE 802.1D learning bridge)

SP network (inside rectangle) looks like a single Ethernet bridge!

Note: if CE is a router, then PE only sees 1 MAC per customer location

VPLS bridge

PE maintains a separate bridging module for each VPN (VPLS instance)

VPLS bridging module must perform:

- MAC learning
- MAC aging
- flooding of unknown MAC frames
- replication (for unknown/multicast/broadcast frames)

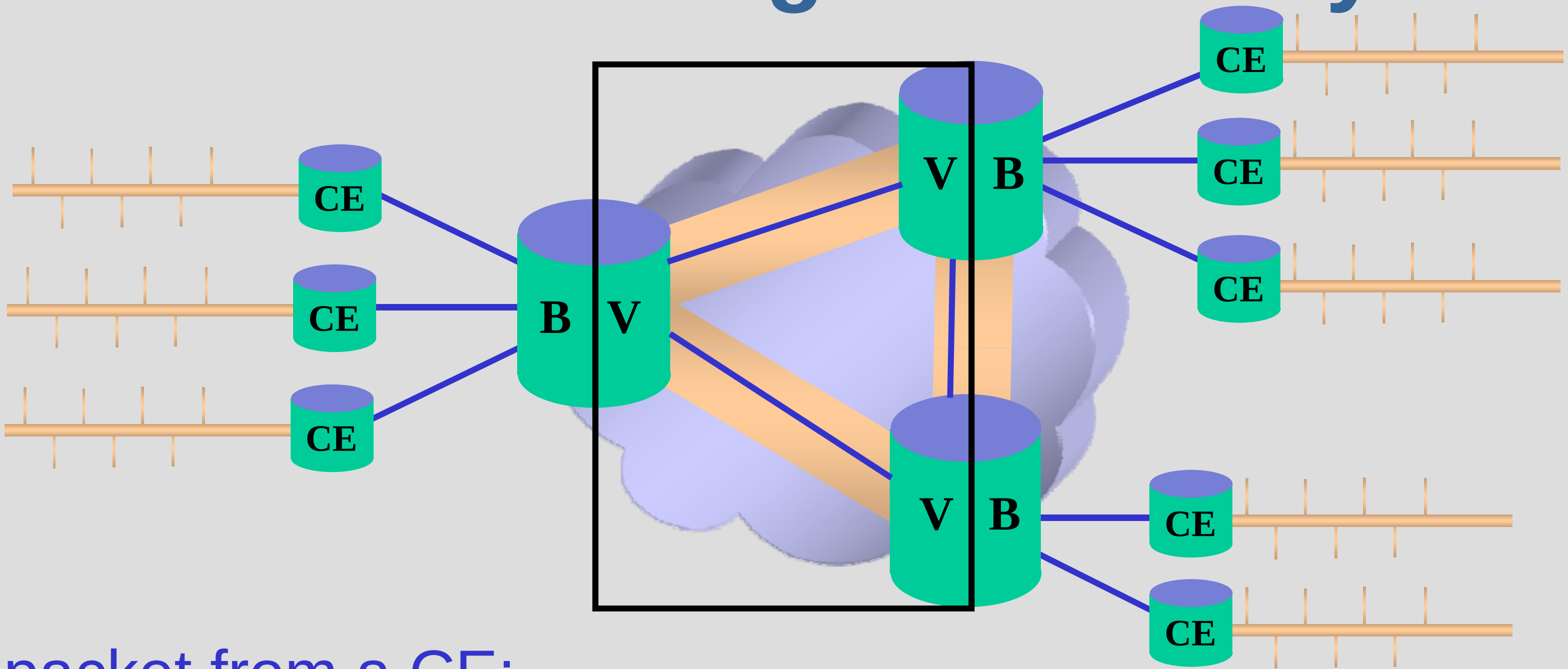
unlike true bridge, **S**panning **T**ree **P**rotocol is ***not*** used

- limited traffic engineering capabilities
- scalability limitations
- slow convergence

forwarding loops are avoided by **split horizon**

- PE never forwards packet from MPLS network to another PE
- not a limitation since there is a full mesh of PWs
  so always send directly to the right PE

Bridge - both ways



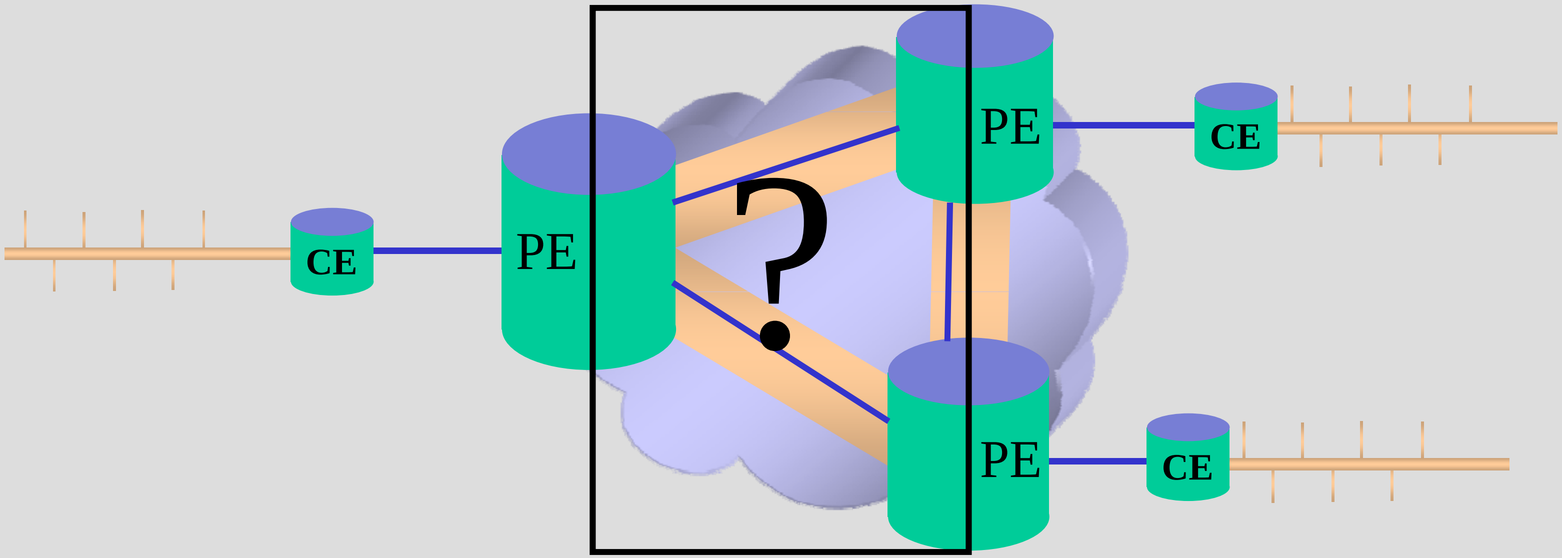
a packet from a CE:

- may be sent back to a CE
- may be sent to a PE via a PW

a packet from a PE:

- is only sent to a CE (split horizon)
- is sent to a particular CE based on 802.1D bridging

L2VPN vs. L3VPN



in L2VPN CEs appear to be connected by single L2 network

- PEs are transparent to L3 routing protocols
- CEs are routing peers

in L3VPN CE routers appear to be connected by a single L3 network

- CE is routing peer of PE, not remote CE
- PE maintains routing table for each VPN

PW OAM

PWE Associated Channel

PW associated channel fate-shares with user data
Inside the channel we can run different OAM mechanisms
The use of the Ach was extended to MPLS-TP as the GACh

ACh differentiated by control word format (RFC 4385)

0 0 0 1	VER	RES=0	Channel Type

The channel types are defined in the
Pseudowire Associated Channel Types IANA registry

1	Management Communication Channel (MCC)
2	Signaling Communication Channel (SCC)
7	BFD Control without IP/UDP Headers
021	IPv4 packet
057	IPv6 packet

VCCV

VC (old name for PW) **CV** (incorrect name for CC)

VCCV is set up by PWE control protocol, if used

VCCV can run in the ACH, but there are also other methods

VCCV enables pings, periodic CC, loopback, …

VCCV has several CV types :

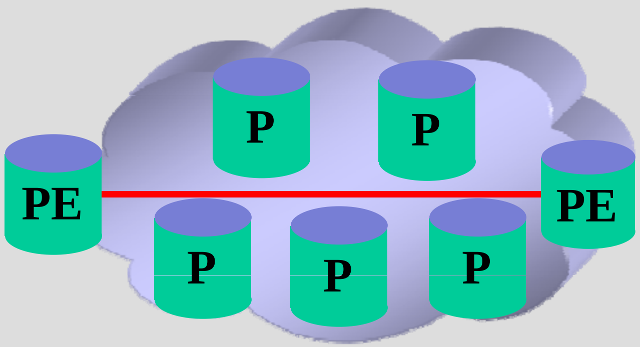
- ICMP (RFC 5085)
- LSP ping (RFC 5085)
- BFD (RFC 5885)

PWE control protocol

PWE (Martini) control protocol

- PWE control protocol (RFC 4447) used to set up / configure PWs
- used only by PW end-points (PEs in standard model)

  intermediate nodes (e.g. P routers) don't participate or see
- based on LDP
  - targeted LDP is used to communicate with remote end-point
  - 2 new FECs for PWs
  - new TLVs added for PW-specific functionality
  - associates two labels with PW



PWE control

a PW is a *bidirectional* entity (two LSPs in opposite directions)

a PW connects two *forwarders*

2 different LDP TLVs can be used

- PWid FEC (128)
- Generalized ID FEC (129)

FEC 128

- both end-points of PW must be provisioned with a unique (32b) value
- each PW end-point independently initiates LSP set up
- LSPs bound together into a single PW

FEC 129

- used when autodiscovering PW end-points
- each end-point has attachment identifier (AI) …

Generalized ID

for each *forwarder* we have a PE-unique Attachment Identifier (AI)
<PE, AI> must be globally unique

frequently useful to group a set of forwarders into a attachment group
where PWs may only be set up among members of a group

then Attachment Identifier (AI) consists of

- Attachment Group Identifier (AGI) (which is basically a VPN-id)
- Attachment Individual Identifier (AII)

the LSPs making up the (two directions of the) PW are

< PE1, (AGI, AII1), PE2, (AGI, AII2) > and
< PE2, (AGI, AII2), PE1, (AGI, AII1) >

we also need to define

- Source Attachment Identifier (SAI = AGI+SAII)
- Target Attachment Identifier (TAI = AGI+TAII)

receiving PE can map TAI uniquely to AC