Post 5

Thursday, January 24, 2019 11:24 PM

FRAME RELAY P1

Eric Scace: ver of X.25 protocol across ISDN int: Provided to support voice/data traffic bet LAN's over WAN

Each end usr gets private/leased line to Frame Relay node: Handles trans over freq changing path transparent to all end usrs

- **CPE: Customer Premise Equip** rtrs/other devices to comm w/SP FR switch
- SP configs FR switch: Keeps end-usr config to min

Disadvantage: Customers pay for leased lines w/fixed capacity: WAN traffic: Some capacity unused

- Each endpoint: Separate phys int on rtr: Increases equip costs: Any changes: Req site visit by carrier

Frame Relay: WAN protocol: L1/L2

- Req: Single access circuit to FR provider to comm w/other sites connected to same provider
- Capacity bet any 2 sites vary

Benefits: Less suitable for WAN today: Still locs in world that rely on FR for WAN connectivity

- Greater BW/reliability/resiliency than private/leased line

PVC: Permanent Virtual Circuit: Logical path along originating FR link through network

- Compared to phys path used by dedicated connection
- Network w/FR access? PVC defines path bet 2 endpoints

Cost Effective	• **Dedicated lines:** Customers pay for end-to-end connection: Includes local loop/network link ▪ Customers pay for local loop/BW purchased from provider ▪ Distance bet nodes not impt ▪ Dedicated line model: Provided: Increments: 64 kb/s ▪ FR customers: Can define virtual circuit needs • **FR shares BW across larger base of customers:** Provider can service 40+ 56 kb/s over 1 T1 ○ Dedicated lines: Would req more CSU/DSUs (1 for each line)/more complicated routing/switching ○ Providers save: Less equip to purchase/maintain

Flexibility	**Virtual circuit:** • SPAN wants to comm w/other SPAN? All needs is to connect to virtual circuit **End of each connection has # to ID it** • **DLCI: Data Link Connection Identifier** • Any station can connect w/other simply by stating addr of station/DLCI # of line

VC: Virtual Circuit: Connection through FR network bet 2 DTEs

- Circuits virtual b/c no direct electrical connection from end to end
- Logical: Data moves end to end w/out direct electrical circuit
- FR shares BW among multiple usrs/any site can comm w/any other w/out multiple dedicated phys lines

2 ways to establish VCs:

SVC	**Switched Virtual Circuits:** • Established dynamically by sending sig msgs to network ○ **CALL SETUP** ○ **DATA TRANSFER** ○ **IDLE** ○ **CALL TERMINATION**
PVC	**Permanent Virtual Circuits:** • Preconfig by carrier • After set up: Only **DATA TRANSFER/IDLE** modes

	• More common

VC: Created by storing input-port to output-port mapping in mem of each switch
- Links 1 switch to another until continuous path from 1 end of circuit to other ID'd
- Can pass through any # of intermediate devices (switches) loc w/in FR network
- Bidirectional comm path from 1 device to another: ID'd by DLCIs

DLCI values: Typically assigned by FR SP: Local significance:
- Values not unique in FR WAN
- DLCI ID's VC to equip at endpoint
- No significance beyond single link
- 2 devices connected by VC may use diff DLCI value to refer to same connection
- Local addressing prevents customer from running out of DLCIs as network grows

DLCIs: 0-15/1,008-1,023: Reserved **|| SP's typically assign range:** 16-1,007
Multiple VCs: FR: Statistically multiplexed: Transmits only 1 frame at time but many logical connects can co-exist on single phys line
FRAD: FR Access Device/rtr connected to FR network: May have multiple VCs connecting to various endpoints
- Multiple VCs on single phys line: Each VC has own DLCI

Encapsulation
FR: Takes packets from network layer protocol (IPv4/6):
- Encapsulates as data portion of FR frame
- Passes frame to phys layer for delivery on wire

Takes packets from network layer protocol: Wraps w/address field: Contains DLCI/checksum
- Flag fields: Added to indicate beginning/end of frame: Always same
- **Hex 7E || Bin 01111110**

After packet encapsulated: FR passes frame to phys layer for transport
- CPE rtr encapsulates each L3 packet inside FR header/trailer before sending it across VC
- Header/trailer defined by **LAPF: Link Access Procedure for FR:** ITU Q.922-A

Header contains

DLCI	**10-bit:** Impt field: Represents virtual connection bet DTE device/switch • Each virtual connection multiplexed on phys chan represented by unique DLCI • DLCI values only unique to phys chan which they reside • Devices on opposite ends can use diff DLCIs to refer to same virtual connection
C/R	Bit follows most sig DLCI byte in Addr field
EA	**Extended Address** • If value of EA field 1: Current byte determined last DLCI octet • Current FR implements use 2-octet DLCI • Allows longer DLCIs in future **8th bit of each byte of Address field indicates EA**
Congestion Control	**3 FR congestion-notification bits** **1. FECN: Fwd Explicit Congestion Notification** **2. BECN: Backward Explicit Congestion Notification** **3. Discard Eligible bits**

Phys layer typically EIA/TIA-232/449/530/V.35/X.21: FR frame subset of HDLC frame type: Delimited w/flag fields
- **1-byte flag: Bit pattern 01111110**

FCS determines whether errors in L2 addr field occurred during trans
- FCS calc prior to trans by sending node: Result: Inserted in FCS field
- At distant end: 2nd FCS value calc/compared to FCS in frame
- If results same: Frame processed || If diff: Frame discarded
  - FR doesn't notify source when frame discarded
- Error control: Left to upper layers of OSI

Topologies

Star (Hub & Spoke)	Simplest WAN topology star • Loc of hub usually chosen by lowest leased-line cost • When implementing: Each remote site has access link to FR cloud w/single VC
Full Mesh	Dedicated lines: Connects every site to every other site • Using leased-line interconnections: Addl serial ints/lines add costs

	• Can build multiple connections simply by config addl VCs on each existing link
Partial Mesh	**Large networks:** Full mesh unaffordable b/c # of links req • Full mesh: Theoretical limit of less than 1,000 VCs per link • Partial: More interconnections than req for star, but not as many as for full • Pattern dependent on data flow reqs

Address Mapping: Before Cisco rtr able to transmit data over FR:
- Needs to know which local DLCI maps to L3 address of remote dest
- Support all network layer protocols over FR: IPv4/6/IPX/AppleTalk
- Address-to-DLCI mapping can be accomplished by static/dynamic mapping

Inverse ARP: Primary tool of FR:
ARP translates L3 IPv4 addresses to L2 MACs: Inverse ARP does opposite
- Corresponding L3 IPv4 addr must be avail before VCs used

FR IPv6 uses IND: Inverse Neighbor Discovery to obtain L3 IPv6 addr from L2 DLCI
- FR rtr sends IND Solicitation msg to req L3 IPv6 addr corresponding to L2 DLCI addr of remote FR rtr
- At same time: IND Solicitation msg provides sender's L2 DLCI addr to remote FR rtr

Dynamic Mapping: Relies on Inverse ARP to resolve next-hop IPv4 addr to local DLCI
- Rtr sends I.ARP reqs on its PVC to discover protocol addr of remote device connected to FR network
- Rtr uses responses to pop. address-to-DLCI mapping table on FR rtr/access server
- Rtr builds/maintains table: Contains all resolved I.ARP reqs: Both dynamic/static mapping entries

Cisco: Inverse ARP enabled: Default: All protocols enabled on phys int
- Inverse ARP packets not sent out for protocols not enabled on int

Static FR Mapping: Override dynamic by supplying next-hop protocol addr to local DLCI
- Similar to dynamic I.ARP: Associates specified next-hop protocol addr to local FR DLCI
- Can't use I.ARP/map statement for same DLCI/protocol
- Hub & Spoke: Static mapping on spoke rtrs to provide spoke-to-spoke reachability
    - Spoke rtrs: No direct connectivity w/each other: Dynamic wouldn't work bet them

Config Static Mapping:
R1(config)# int s0/0/1
R1(config-if)# ip address 10.1.1.1 255.255.255.0
R1(config-if)# encapsulation frame-relay
R1(config-if)# no frame-relay inverse-arp
R1(config-if)# frame-relay map ip 10.1.1.2 102 broadcast cisco
R1(config-if)# no shut

 # show frame-relay map

Map bet next hop protocol addr/DLCI dest addr:
R1(config)# frame-relay map protocol *protocol-address* **dlci [broadcast] [ietf] [cisco]**
- **ietf** Connect to non-Cisco rtr

broadcast Config of OSPF simplified by adding keyword
- Specifies broadcast/multicast traffic allowed over VC: Config permits use of dynamic r-protocols

LMI: Local Mgmt Int: Provides packet-switched data transfer w/min end-to-end delays
- Original design omits anything that might contribute to delay

LMI is a keepalive mech that provides status info about FR connections bet rtr (DTE) and FR switch (DCE)
- Every 10s/so: End device polls network: Req dumb seq response/chan status info
- If network doesn't respond w/req info: Usr device may consider connection down
- When network responds w/FULL STATUS response
    - Includes status info about DLCIs allocated to that line
    - End device can use info to determine whether logical connections able to pass data

 # show frame-relay lmi Shows LMI type used by FR int/counters for LMI status exchange seq
- Includes errors like LMI timeouts

LMI: Def of msgs used bet DTE/DCE
- Encapsulation defines headers used by DTE to comm info to DTE at other end of VC
- Switch/connected rtr care about using same LMI
- Switch doesn't care about encapsulation

- Endpoint rtrs (DTEs) care about encapsulation

LMI Identifiers

VC Identifiers	VC Types
0	LMI link mgmt (ANSI/ITU)
1-15	Reserved for future use
16-991	Avail for VC endpoint assignment
992-1007	Optional L2 mgmt info
1008-1018	Reserved for future use (ANSI/ITU)
1019-1022	LMI Multicast
1023	LMI link mgmt (Cisco)

LMI Extensions

Standard FR Frame is basis for LMI frame

Flag	LMI DLCI	Un-#'d IE	Protocol Discriminator	Call Reference	LMI Msg Type	IE's	FCS	Flag
8	16	8	8	8	8	Var	16	8

In addition to FR protocol functions for transferring data: FR spec includes optional LMI extensions

Some extensions include:

VC status msgs	Provide info about PVC integrity by comm/sync bet devices • Periodically reporting existence of new PVCs/del of already existing PVCs • VC status msgs prevent data from being sent into black holes (PVCs that no longer exist)
Multicasting	Allows sender to transmit single frame delivered to multiple recipients • Supports efficient delivery of r-protocol msgs/addr resolution procedures • Typically sent to many dest simultaneously
Global addressing	Provides connection IDs w/global rather than local sig: • Allowing them to be used to ID specific int to FR network • Global addressing makes FR network resemble LAN in terms of addressing/ARPs
Simple flow control	Provides **XON/XOFF** flow control mech that applies to entire FR int • Intended for devices that can't use congestion notification bits (**FECN/BECN**) • Would be leveraged by higher layers: Still req some lvl of flow control

LMI: Used to manage FR links:

- Each LMI msg classified by DLCI appearing in LMI frame
- 10-bit DLCI field supports 1,024 VC IDs: 0-1,023
- LMI extensions reserve some of these VC IDs: Reducing # of permitted VCs
- LMI msgs exchanged between the DTE and DCE using these reserved DLCIs.

7 LMI types: Each incompatible w/others: LMI type config on rtr must match type used by SP

3 LMI Types Supported by Cisco rtrs:

CISCO	Original LMI ext
ANSI	Corresponding to ANSI standard T1.617 Annex D
Q933A	Corresponding to ITU standard Q933 Annex A

#show int [type #] Display LMI msg info/associated DLCI #'s

- Cisco uses DLCI 1023 to ID LMI msgs used for FR link mgmt

IOS 11.2+ : Default LMI autosense feature: Detects LMI type supported by directly connected FR switch

- Based on LMI status msgs from FR switch
- Rtr auto configs its int w/supported LMI type ack by FR switch

(config-if)# frame-relay lmi-type [cisco | ansi | q933a] *[int config]* Set LMI type: Disables autosense

In cases where FR switch uses non-default timeouts:

- Keepalive interval must also be config on FR int to prevent status exchange msgs from timing out

LMI status exchange msgs: Determine status of PVC connection

- Large mismatch in keepalive interval on rtr/switch can cause switch to declare rtr dead
- Impt to consult FR SP for info on how to mod keepalive setting
- Default: Keepalive 10s on Cisco serial ints

keepalive *[int config]* Can change keepalive interval w/cmd

- Status msgs help verify integrity of logical/phys links
- Info critical in routing env b/c r-protocols make decisions based on link integrity

Using LMI/I.ARP to Map Addresses

- LMI status msgs combined w/I.ARP msgs allow rtr to associate network layer/DLL addresses

I.ARP Op

- When int supporting I.ARP becomes active: Initiates I.ARP protocol/fmts Inverse ARP request for active VC
- I.ARP req includes source HW/source L3 protocol address/known target HW addr
- It fills target L3 protocol addr field w/all 0s
- It encapsulates packet for specific network/sends directly to dest device using VC
- Upon receiving I.ARP req: Dest device will use addr of source device to create its own DLCI-to-L3 map
- Will then send I.ARP response that includes its L3 addr info
- When source device receives I.ARP response: Completes DLCI-to-L3 map using provided info
- When int on Cisco rtr is config to use FR encapsulation: I.ARP enabled by default