Troubleshooting WAN Protocols

Skill Level Assumptions

- Have protocol level knowledge of:
  Frame Relay, PPP and/or ATM
- Familiar with Cisco IOS® WAN configuration of:
  T1, T3 and SONET controllers
  HDLC, PPP, Frame Relay and WAN ATM
Session Goal

- Troubleshooting process
  - Check your physical layer first
  - Use your debugs
  - Use your show commands

Session Topics:

- Cabling and Lead States
- T1 and T3 Transmission Lines
- Point-to-Point SONET
- Frame Relay
- HDLC
- PPP
- WAN ATM
Physical Layer Topics

- The physical layer topics covered in this presentation are targeted toward the North American market

T1 and T3 Transmission Lines
Troubleshooting T1 or Digital Leased Lines

- Verify CSU/DSU settings
  - Framing, line code, clocking, timeslots, speed, interface type
  - SCTE used to compensate for clock phase shift
- Loopbacks are your friends

T1: Meaning of Controller Counters

Data in current interval (615 seconds elapsed):
- 0 Line Code Violations, 0 Path Code Violations
- 0 Slip Secs, 0 Fr Loss Secs, 0 Line Err Secs, 0 Degraded Mins
- 0 Errored Secs, 0 Bursty Err Secs, 0 Severely Err Secs
- 0 Unavail Secs, 0 Stuffed Secs

- Line Code Violations (LCV)—number of bipolar violations or excessive zeros
- Path Code Violations (PCV)—number of frame synchronization bit errors in SF and “E1-noCRC” frame formats or CRC errors in ESF and “E1-CRC” frames
- Slip Secs—number of seconds containing one or more controlled slips
- Fr Loss Secs—number of seconds with at least one Out of Frame (OOF) error
- Line Err Secs—number of seconds with at least one line code violation
- Degraded Mins—minutes since controller has been up and available that the error rate per minute exceeded 0.000001 but was less than 0.001
T1: Meaning of Controller Counters

Data in current interval (615 seconds elapsed):
- 0 Line Code Violations, 0 Path Code Violations
- 0 Slip Secs, 0 Fr Loss Secs, 0 Line Err Secs, 0 Degraded Mins
- 0 Errored Secs, 0 Bursty Err Secs, 0 Severely Err Secs
- 0 Unavail Secs, 0 Stuffed Secs

- **Errored Secs**—total seconds containing one or more PCVs, OOF conditions, AIS alarms or controlled slips for ESF and “E1-CRC” formats; SF and “E1-noCRC” formats also count seconds containing one or more LCVs
- **Bursty Err Secs**—number of seconds containing one to 320 PCVs
- **Severely Err Secs**—for ESF format, the total number of seconds containing more than 320 PCVs, at least one OOF condition or an AIS detection; for E1-CRC format, the total number of seconds with more than 832 PCVs or at least one OOF condition; for E1-noCRC format, the total number of seconds with more than 2048 LCVs; for SF/D4 format, the total number of seconds with more than 1544 LCVs or at least one OOF condition

- **Unavail Secs**—total seconds that the E1 or T1 controller cannot pass traffic; unavailable seconds begin when ten consecutive severely errored seconds occur; the unavailable seconds end when ten contiguous seconds elapse with no severely errored seconds; during an unavailable second none of the other E1 or T1 controller counters increment

- **Stuffed Secs**—total seconds where non-data bit patterns must be transmitted due to TX buffer under runs
**T1: Red Alarm**

- Occurs when T1 receiver has Loss of Signal (LOS)
- LOS defined in RFC 1406 as:
  
  "For T1, the Loss of Signal failure is declared upon observing 175 +/- 75 contiguous pulse positions with no pulses of either positive or negative polarity"
T1: Definition of Yellow Alarm

• For SF links, the far end alarm failure is declared when bit 6 of all of the channels has been zero for at least 335 ms; the failure is cleared when bit 6 of at least one channel is not zero for a period usually less than one second and always less than 5 seconds; the far end alarm failure is not declared for SF links when a Loss of Signal is detected.

• For ESF links, the far end alarm failure is declared if the yellow alarm signal pattern occurs in at least seven out of ten contiguous 16-bit pattern intervals; the failure is cleared if the Yellow alarm signal pattern does not occur in ten contiguous 16-bit signal pattern intervals.

T1: ESF Yellow Alarm Pattern

• ESF yellow alarm is transmitted as a repeating 16 bit pattern of 8 one bits followed by 8 zero bits.

• This pattern is transmitted in the FDL overhead bits.

• Pattern must be transmitted for a minimum of 1 second.
T1/T3: Alarm States

- A T1 device’s current alarm state is based on the signal pattern or lack of signal present at the T1 device’s receiver.
- The transmitted signal pattern sent by the T1 device’s transmitter does not determine the alarm state of the device.

T1/T3: AIS and Blue Alarms

- An unframed “all ones” bit stream is called an Alarm Indication Signal (AIS); reception of an AIS causes a “blue alarm” state on the receiving device.
- T1 transport device sends AIS when the signal to transmit downstream is lost; i.e. there is a problem upstream.
**T1: Show Controller with Alarms**

```
show controller t1
T1 2/1 is down
   Transmitter is sending remote alarm (Yellow alarm)
   Receiver has loss of signal
   Framing is ESF, Line Code is B8ZS,
   Clock Source is Line
   Data in current interval (160 seconds elapsed):
   0 Line Code Violations, 0 Path Code Violations
   0 Slip Secs, 0 Fr Loss Secs, 0 Line Err Secs,
   0 Errored Secs, 0 Bursty Err Secs,
   0 Severely Err Secs, 160 Unavail Secs
```
T1/T3: Troubleshooting Tools

- “Extended” ping tests
- Loopbacks
  - Controller loops
  - Serial loops
  - Hard wire loops
- BERT generators

Using “extended” Ping Command to Test a T1

- You can use the IOS exec command “ping” to test T1 circuits for line code mismatches, T1 repeater problems and timing issues
- The “ping” command has an “extended commands” section that allows you to enter the 2 byte data pattern that is repeated in the payload of the ping packet
Using “extended” Ping Command to Test a T1

- Three usefully ping data patterns that expose line problems:
  - 0x0000 (line code mismatches)
  - 0xFFFF (repeater power problems)
  - 0x4040 (timing problems)

Using “extended” Ping Command to Test a T1

- Example:
  
  ```
  Router#ping
  Protocol [ip]:
  Target IP address: 13.200.2.5
  Repeat count [5]: 100
  Datagram size [100]: 1500
  Timeout in seconds [2]:
  Extended commands [n]: y
  Source address or interface:
  Type of service [0]:
  Set DF bit in IP header? [no]:
  Validate reply data? [no]:
  Data pattern [0xABCD]: 0x4040
  Loose, Strict, Record, Timestamp, Verbose[none]:
  Sweep range of sizes [n]:
  Type escape sequence to abort.
  Sending 5, 1500-byte ICMP Echos to 13.200.2.5, timeout is 2 seconds:
  Packet has data pattern 0x4040
  !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
  ```
### T1: Loopback Types

- **Line**—a loopback that cross connects the RX signal to the TX signal without correcting any LCVs or framing errors in the RX signal.

- **Line payload**—a loopback that reframes the received T1 data before transmitting it to the source; not applicable to T1 SF links.

### T1: ANSI T1.403 Loopback Codes

- **For SF framing:**
  
  Loopback codes are sent in-band; i.e. “in the timeslots,” and T1 framing is maintained.

  - **Loop activate code:** 00001 and repeat for at least five seconds.
  - **Loop deactivate code:** 001 and repeat for at least five seconds.
T1: ANSI T1.403 Loopback Codes

• For ESF framing:
  Loopback codes are sent in the “DL” a.k.a “FDL” bits of the framing overhead and T1 framing is maintained

T1: ANSI ESF Loopback Codes

<table>
<thead>
<tr>
<th>Action</th>
<th>Line Loop</th>
<th>Payload Loop</th>
</tr>
</thead>
<tbody>
<tr>
<td>Loop Up Code</td>
<td>0000111011111111</td>
<td>0010100111111111</td>
</tr>
<tr>
<td>Loop Down Code</td>
<td>0011100011111111</td>
<td>0011001011111111</td>
</tr>
</tbody>
</table>

NOTE: The Right-Most Bit Is Sent First
Loopback Tests

- Use pings to test connectivity, while in loopback mode
- Remember, success of pings, depends on correct configuration
  - HDLC: if Keep alive is enable, “sh int” => up, up (looped)
  - You can then ping yourself
  - Frame Relay: if LMI is exchanged w/switch, then you see up, up if PVC-to-IP mapping exists, you can ping yourself

Troubleshooting Tools

Loopbacks

T1 CSU/DSU Loopback Line

Router (config-if) # loopback line
Troubleshooting Tools
Loopbacks (Cont.)

T1 CSU/DSU Loopback Line Payload
Router (config-if) # loopback line payload

T1 CSU/DSU Module DTE Loop
Router (config-if) # loopback dte
Troubleshooting Tools
Loopbacks (Cont.)

Far End Remote Loop
Router(config-if) # loopback remote full other type loops—payload, smart-jack

May Require Service-module T1 Remote-loopback Set at Remote

T1: DB-15 and RJ-45 Loopback Plugs

<table>
<thead>
<tr>
<th>DB-15 Pin</th>
<th>Signal Name</th>
<th>RJ-45 Pin</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>TX Tip (T1)</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>RX Tip (T)</td>
<td>4</td>
</tr>
<tr>
<td>9</td>
<td>TX Ring (R1)</td>
<td>2</td>
</tr>
<tr>
<td>11</td>
<td>RX Ring (R)</td>
<td>1</td>
</tr>
</tbody>
</table>

- To loopback a DB-15 interface, connect pin 1 to pin 3 and connect pin 9 to pin 11
- To loopback a RJ-45 interface, connect pin 1 to pin 4 and connect pin 2 to pin 5
T3: “Show Controllers Serial”

Router#show controllers serial 1/1/0
Serial1/1/0 -
  Mx T3+(1) HW Revision 0x3, FW Revision 3.9
  Framing is c-bit, Clock Source is Line
  Bandwidth limit is 44210, DSU mode 0, Cable length is 10
  rx FEBE since last clear counter 16094, since reset 0
  Data in current interval (565 seconds elapsed):
    0 Line Code Violations, 0 P-bit Coding Violation
    17850 C-bit Coding Violation
    0 P-bit Err Secs, 0 P-bit Sev Err Secs
    0 Sev Err Framing Secs, 7 Unavailable Secs
    70 Line Errored Secs, 70 C-bit Errored Secs, 70 C-bit Sev Err Secs
  No alarms detected

T3: “Show Controller”
Field Explanations

- **Line Code Violations**—total count of BPVs and excessive zeros conditions
- **P-bit Coding Violation**—total count of P-bit parity errors where locally calculated parity does not equal P-bit parity code in received M-frame
- **C-bit Coding Violation**—for C-bit framing, total count of CP-bit parity errors
- **P-bit Err Secs**—total seconds in which one or more P-bit coding violations, OOF conditions or received AIS occurred
- **P-bit Sev Err Secs**—total seconds in which 44 or more P-bit coding violations, one or more OOF conditions or a received AIS occurred
- **Sev Err Framing Secs**—total seconds with one or more OOF conditions or received AIS
More T3 “show controller” Field Explanations

- **Unavailable Secs**—total number of seconds the interface has been unusable; unavailable seconds are those between the time the interface experiences 10 “P-bit severely errored seconds” and time it experiences 10 seconds with no “P-bit severely errored seconds”; when the interface is “unavailable” no other interface error counters increment.

- **Line Errored Secs**—total seconds in which one or more coding violations or LOS defects occurred.

- **C-bit Errored Secs**—total seconds in which one or more C-bit coding violations, OOF conditions or received AIS occurred.

- **C-bit Sev Err Secs**—total seconds in which 44 or more C-bit coding violations, one or more OOF conditions or a received AIS occurred.

Point-to-Point SONET
SONET: Is the Receiver Overpowered?

• Optical power budgets are calculated as follows:
  
  \[ PB = PT_{\text{min}} - PR_{\text{min}} \]
  
  \[ PM = PB - LL = PT_{\text{min}} - PR_{\text{min}} - LL \]

• Where:
  
  \[ PB = \text{Power Budget} \]
  
  \[ PT_{\text{min}} = \text{minimum transmitter power} \]
  
  \[ PR_{\text{min}} = \text{minimum receiver sensitivity} \]
  
  \[ PM = \text{Power Margin} \]
  
  \[ LL = \text{Link Loss} \]

SONET: Timing

• When connecting to an ADM set clock source to line

• Rapidly incrementing “NEWPTR” counter indicates timing problems

• A DS1 transported via SONET VT1.5 can serve as timing reference if SONET transport ring has a single timing source
SONET: Error Conditions

• LOF-S—Loss of Frame Section
  LOF-S is declared when 4 consecutive A1/A2 byte pairs have incorrect byte pattern
  An “in frame” condition is declared when 2 consecutive A1/A2 byte pairs have correct byte pattern (0xF628)

SONET: Alarm Indicator Signals

• RDI—Remote Defect Indication
  Replaces former names FERF or RAI); (similar functionality as a Yellow alarm in DS3/DS1)
  Line RDI (RDI-L) is reported by the downstream LTE when it detects LOF, LOS, or AIS
  Path RDI (RDI-P) (sent in bits 5,6 and 7 of G1 byte) is reported by the downstream PTE when it detects a defect on the incoming signal
SONET: Alarm Indicator Signals

• REI—Remote Error Indicator
  
  Term now used instead of FEBE
  
  Line Remote Error Indicator (REI-L) (accumulated from the M0 or M1 byte) is reported when the downstream LTE detects BIP(B2) errors
  
  Path Remote Error Indicator (REI-P) (sent in bits –4 of the G1 byte) is reported when the downstream PTE detects BIP(B3) errors

SONET: Alarm Indicator Example

[Diagram showing the flow of alarm indicators in a SONET network]
**SONET: Alarm Hierarchy**

<table>
<thead>
<tr>
<th>Alarm</th>
<th>Priority</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOS-S</td>
<td>Priority 1 (Highest)</td>
</tr>
<tr>
<td>LOF-S</td>
<td>Priority 2</td>
</tr>
<tr>
<td>AIS-L</td>
<td>Priority 3</td>
</tr>
<tr>
<td>SF</td>
<td>Priority 4</td>
</tr>
<tr>
<td>RDI-L</td>
<td>Priority 5</td>
</tr>
<tr>
<td>AIS-P</td>
<td>Priority 6</td>
</tr>
<tr>
<td>LOP-P</td>
<td>Priority 7</td>
</tr>
<tr>
<td>RDI-P</td>
<td>Priority 8</td>
</tr>
</tbody>
</table>

**Session Goal**

- Troubleshooting process
  - Check your physical layer first
  - Use your debugs
  - Use your show commands
Cisco’s HDLC

- Its proprietary because:
  - It does not perform windowing or retransmission
  - Higher layer protocol identification method is not standardized
- Its frame format and bit stuffing technique are per the ANSI T1.618 standard
Cisco’s HDLC: Bit Stuffing Technique

- To ensure that starting and ending flags or other “pattern specific” bytes never show up in the data portion of the HDLC frame, a zero bit is inserted into the serial data stream after five consecutive one bits are transmitted.

Cisco’s HDLC: Why Is Line Protocol Down?

- HDLC lines transition to the “line protocol down” state when “myseq” and “mineseen” keep alive numbers differ by three or more.
- The router compares the keepalive sequence number it is going to send (“myseq”) with the returned sequence number (“mineseen”) in the most recently received keepalive; if the two sequence numbers differ by three or more then line protocol goes down.
Link Testing with HDLC

- PPP not good for link tests due to loopback testing (magic number)
- Due to DLCI addressing, Frame Relay also not good for loopback testing
- HDLC has no such protection or addressing, so line protocol will come up
- Once up/up, you can test with PING

Cisco’s HDLC: Debug Serial Interface

1 Missed Keepalive

Serial1: HDLC myseq 636127, mineseen 636127, yourseen 515040, line up
Serial1: HDLC myseq 636128, mineseen 636127, yourseen 515041, line up
Serial1: HDLC myseq 636129, mineseen 636129, yourseen 515042, line up
Serial1: HDLC myseq 636130, mineseen 636130, yourseen 515043, line up
Serial1: HDLC myseq 636131, mineseen 636130, yourseen 515044, line up
Serial1: HDLC myseq 636132, mineseen 636130, yourseen 515045, line up
Serial1: HDLC myseq 636133, mineseen 636130, yourseen 515046, line down
Serial1: HDLC myseq 636127, mineseen 636127, yourseen 515040, line up
Serial1: HDLC myseq 636128, mineseen 636128, yourseen 515041, line up
Serial1: HDLC myseq 636129, mineseen 636129, yourseen 515042, line up
Serial1: HDLC myseq 636130, mineseen 636130, yourseen 515043, line up
Serial1: HDLC myseq 636131, mineseen 636130, yourseen 515044, line up

3 Missed Keepalives So Line Goes Down and Interface Is Reset
What Does “output stuck” Mean?

%RSP-3-RESTART interface Serial8/1/3, output stuck

• “output stuck” occurs when the router cannot transmit all the packets in an interface’s process level hold queue within one minute

• When the minute elapses, the interface is automatically reset

• If automatic reset does not resolve problem, a slot specific microcode reload will

PPP
PPP: Theory

- Standard method for transporting multi-protocol datagrams over point-to-point links
- PPP is comprised of three main components that is negotiated between both sides
  - A Link Control Protocol (LCP) for establishing, configuring and testing the data-link connection
  - An authentication level for access control (optional)
  - A family of Network Control Protocols (NCPs) for establishing and configuring different network-layer protocols

PPP: Protocols (Protocol Field)

<table>
<thead>
<tr>
<th>Value (in hex):</th>
<th>Protocol Name:</th>
</tr>
</thead>
<tbody>
<tr>
<td>0001</td>
<td>Padding Protocol</td>
</tr>
<tr>
<td>0003 to 001f</td>
<td>reserved (transparency inefficient)</td>
</tr>
<tr>
<td>007d</td>
<td>:reved (Control Escape)</td>
</tr>
<tr>
<td>00cf</td>
<td>reserved (PPP NLPID)</td>
</tr>
<tr>
<td>00ff</td>
<td>reserved (compression inefficient)</td>
</tr>
<tr>
<td>8001 to 801f</td>
<td>unused [RFC1661]</td>
</tr>
<tr>
<td>8011</td>
<td>Bridging NCP</td>
</tr>
<tr>
<td>80fd</td>
<td>Compression Control Protocol (CCP) [RFC1962]</td>
</tr>
<tr>
<td>c021</td>
<td>Link Control Protocol</td>
</tr>
<tr>
<td>c023</td>
<td>Password Authentication Protocol</td>
</tr>
<tr>
<td>c025</td>
<td>Link Quality Report</td>
</tr>
<tr>
<td>c029</td>
<td>CallBack Control Protocol (CBCP)</td>
</tr>
<tr>
<td>c02b</td>
<td>BACP Bandwidth Allocation Control Protocol [RFC2125]</td>
</tr>
<tr>
<td>c02d</td>
<td>BAP [RFC2125]</td>
</tr>
<tr>
<td>c223</td>
<td>Challenge Handshake Authentication Protocol</td>
</tr>
</tbody>
</table>
PPP: Simplified Phase Diagram

- Dead
- Up
- Establish
- Opened
- Authenticate
- Success or None (No Authentication Used)
- Terminate
- Fail
- Down
- Closing
- Network

PPP: LCP (Link Control Protocol)

- LCP: negotiate link specific options:
  - Callback
  - Multilink (MRRU) -or- non-multilink (MRU)
  - Authentication method: PAP/CHAP (optional)
  - Magic Number (Loopback detection)
  - PFC (Protocol Field Compression)
  - ACFC (Address and Control Field Compression)
  - ACCM (Async Control Character Map)
PPP: LCP Configuration Options

- Used to modify the default link characteristics

  Common LCP Options:
  - 0x00  RESERVED [RFC2153]
  - 0x02  Maximum-Receive-Unit (a.k.a., non-multilink)
  - 0x02  ACCM
  - 0x03  Authentication-Protocol
  - 0x04  Quality-Protocol
  - 0x05  Magic-Number
  - 0x07  Protocol-Field-Compression
  - 0x08  Address-and-Control-Field-Compression
  - 0x0D  Callback [RFC1570]
  - 0x11  Multilink-MRRU [RFC1717]
  - 0x12  Multilink-Short-Sequence-Number-Header [RFC1717]
  - 0x13  Multilink-Endpoint-Discriminator [RFC1717]

PPP: LCP Negotiation

- In reality all PPP negotiations are double negotiations

  Configure-Req

  Configure-ACK

  Configure-Req

  Configure-ACK
PPP: Simplified Phase Diagram

PPP: Authentication

- Authentication follows LCP phase
- Authentication protocols (LCP option 3)
  - PAP (0xC023)*
  - CHAP (0xC223, Algorithm 0x05)*

*Prior to IOS V11.2, These Values Were Not Decoded in PPP Debugs
PPP: PAP Authentication Protocol

- Passwords sent in clear text
- Peer (client) in control of attempts
- Performed only once per session

PPP: CHAP Authentication Protocol

- Password never sent across wire
- Challenger in control of attempts
- Authenticate multiple times per connection
Simplified PPP Phase Diagram

- Dead
- Up
- Established
- Opened
- Authenticate
- Fail
- Down
- Terminate
- Closing
- Network
- Success or None (No Authentication Used)

PPP: NCP (Network Control Protocol)

- Negotiate which Layer 3 Protocol(s) to use:
  - For IP: IPCP
  - For IPX: IPXCP
  - For AppleTalk: ATCP
  - For Bridging: BridgeCP
- Each of the above have protocol-specific options that must be negotiated (protocol address, etc.)
PPP Verification

2621#show interface serial0/0
Serial0/0 is up, line protocol is up
Hardware is PQUICC with Fractional T1 CSU/DSU
Internet address is 10.1.3.1/29
MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec,
reliability 255/255, txload 1/255, rxload 1/255
Encapsulation PPP, loopback not set
Keepalive set (10 sec)
LCP Open
Open: IPCP, CDPCP
Last input 00:00:21, output 00:00:01, output hang never
Last clearing of ‘show interface’ counters 20:36:48
Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
Queueing strategy: weighted fair
Output queue: 0/1000/64/0 (size/max total/threshold/drops)
   Conversations 0/1/256 (active/max active/max total)
   Reserved Conversations 0/0 (allocated/max allocated)
   5 minute input rate 0 bits/sec, 0 packets/sec
   5 minute output rate 0 bits/sec, 0 packets/sec
   32146 packets input, 1616707 bytes, 0 no buffer
   Received 0 broadcasts, 0 runs, 0 giants, 0 throttles
   302 input errors, 44 CRC, 258 frame, 0 overrun, 0 ignored, 0 abort
   32173 packets output, 1616197 bytes, 0 underruns
   0 output errors, 0 collisions, 4 interface resets
   0 output buffer failures, 0 output buffers swapped out
   2 carrier transitions
   DCD=up DSR=up DTR=up RTS=up CTS=up

Multilink PPP (MLPPP): Caveats

- The Maximum Reassembled Receive Unit (MRRU) of a Cisco MLPPP bundle is fixed at 1524 bytes
- Maximum of 16 links per MLPPP bundle
Multilink PPP (MLPPP): Caveats

- Prior to IOS 12.2, multiple MLPPP bundles between 2 routers require CHAP or PAP authentication per bundle
- In IOS 12.2, use new command “ppp multilink endpoint” to force the use of peer router’s endpoint discriminator for bundle identification so authentication isn’t needed

MLPPP: Faster Bundling

- Commands to speed the addition of b-channels to a MLPPP bundle
  - dialer load-threshold 1
  - multilink load-threshold 1 either (for multilink interfaces)
  - load-interval 1
  - ppp timeout multilink link add 1 (post-11.3(2.1))
  - dialer wait-for-carrier 1 (pre-11.3(2.1))
PPP: Troubleshooting

- Debug ppp negotiation
- Debug ppp authentication
- Debug ppp multilink events
- Service timestamps debug date msec
- Service timestamps log date msec

PPP: Things to Look for in PPP debugs

- **LCP: state is open**
  
  LCP negotiation was successful
  
  If not, then look for options that failed during LCP negotiation by observing the CONFREQ, CONFREJ, CONFFACK, and CONFNACK activity

- **Authentication: PAP or CHAP**
  
  Check for username, passwords, etc.

- **NCP: IPCP, IPXCP, ATCP state is open**
  
  Means NCP negotiation was successful
  
  If not, then monitor NCP negotiation by observing the CONFREQ, CONFREJ, CONFFACK, and CONFNACK activity
Typical Synchronous Serial MLPPP Configuration

interface Multilink2
  ppp multilink
  multilink load-threshold 1 either
  multilink-group 2
!
interface Serial1/1:2
  encapsulation ppp
  ppp multilink
  ppp multilink endpoint string group2
  multilink-group 2
!
interface Serial1/1:3
  encapsulation ppp
  ppp multilink
  ppp multilink endpoint string group2
  multilink-group 2

Debugging MLPPP Negotiation

• Notice the two-way hand shaking as well as endpoint discriminator and MRRU negotiation
Frame Relay (FR)

Frame Relay: Primer

- There are 2 types of Frame Relay frames: Data and LMI
- LMI has a Q.931 header directly after the 2 byte header and 1 byte control (0x03)
- LMI is not end to end and only occurs between the DTE and DCE
- By default the router sends out LMI Status Enquiries (i.e. keepalives) every 10 seconds that are responded to by the Frame Relay DCE with a Status Update
- The router sends a Full Status Update Enquiry every sixth Status Update in which the Frame Relay DCE responds with all the DLCI's and their status (i.e. ACTIVE, INACTIVE, DELETED, etc.)
- Cisco LMI uses DLCI 1023 for signaling while ANSI uses DLCI 0
- Data—Cisco encapsulation (proprietary) or IETF (RFC 1490) encapsulation (interoperability)
Frame Relay: LMI

- Both UNI and NNI interfaces support LMI
- UNI LMI is defined in three different standards:
  - Consortium LMI (Frame Relay lmi-type cisco)
  - ANSI T1.617 Annex D LMI (Frame Relay lmi-type ansi)
  - ITU-T Q.933 Annex A LMI (Frame Relay lmi-type q933a)
- On a UNI, the router/FRAD (UNI user-side) initiates LMI with “Status Enquiry” messages and the FR switch (UNI network-side) responds with “Status” messages; the UNI network-side never initiates UNI LMI

Frame Relay: Interface Types

- Frame Relay interfaces are multipoint by default, but can be point-to-point if a sub-interface is created (sub-interfaces can also be multipoint)
- Multipoint interfaces require an IP to DLCI mapping that can be static by using the “Frame Relay map” command or dynamic by using inverse arp (default)
- Multiple DLCI’s can be configured on one multipoint interface
- Inverse-arp can be disabled on a DLCI or a multipoint interface basis
- Point-to-point interfaces operate like other serial interfaces. There can only be one DLCI on a point-to-point interface, therefore no map or inverse arp is needed
- Advantages of multipoint—save addresses, multiple DLCI’s per interface, less configuration
- Disadvantages of multipoint—harder to troubleshoot, split-horizon issues
Frame Relay: Terms

- **DLCI—Data Link Connection Identifier**
  10 bit address for a Frame Relay connection; it only has local significance; DLCI’s 16-1007 are definable on Cisco routers

- **FECN—Forward Explicit Congestion Notification**
  This bit may be flipped by nodes in the cloud when it experiences congestion in its path to the destination remote FRAD

- **BECN—Backward Explicit Congestion Notification**
  This bit may be flipped by nodes in the cloud when packets in the opposite direction of this packet experience congestion

- **DE—Discard Eligible**
  This bit may be flipped by nodes in the cloud to indicate that packet exceeded the rate the carrier has committed to transport; packets with the DE bit set may be intentionally dropped by Frame Relay nodes in the face of congestion

Frame Relay: Debugging

```
Router#debug frame packet
Router#ping 10.5.5.5

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.5.5.5, timeout is 2 seconds:
*Mar 6 15:33:17.294: Serial0/0.102: Encaps failed--no map entry link 7(IP)
*Mar 6 15:33:17.858: Serial0/0.102(o): dlci 102(0x1861), pkt encap 0x0300 0x8000 0x0000 0x806 (ARP), datagramsize 34
*Mar 6 15:33:17.858: FR: Sending INARP Request on interface Serial0/0.102 dlci 102 for link 7(IP)
*Mar 6 15:33:17.950: broadcast dequeue
*Mar 6 15:33:17.950: Serial0/0.102(o): Pkt sent on dlci 102(0x1861), pkt encaps 0x300 0x8000 0x0 0x806 (ARP), datagramsize 34
*Mar 6 15:33:17.962: Serial0/0.102: frame relay INARP received.

Router#show frame map
Serial0/0.102 (up): ip 10.5.5.4 dlci 102(0x66,0x1860), dynamic, broadcast, CISCO, status defined, active
Serial0/0.102 (up): ip 10.5.5.5 dlci 102(0x66,0x1860), static, broadcast, status defined, active
```
Frame Relay: Debugging (Cont.)

- Make sure “broadcast” keyword is on end of map statement so routing updates (and other broadcast packets) will pass

- Use “Debug frame relay lmi”:

```
*Mar 6 17:47:00.797: RT IE 1, length 1, type 1
*Mar 6 17:47:00.797: KA IE 3, length 2, yourseq 137, myseq 137
*Mar 6 17:47:00.798: PVC IE 0x7, length 0x6, dlci 100, status 0x2, bw 0
*Mar 6 17:47:00.798: PVC IE 0x7, length 0x6, dlci 101, status 0x2, bw 0
*Mar 6 17:47:00.798: PVC IE 0x7, length 0x6, dlci 102, status 0x2, bw 0
```

0x0 is added/inactive 0x4 is deleted 0xa is new/active
0x2 is added/active 0x8 is new/inactive

Frame Relay: Debugging (Cont.)

```
2600b# show Frame Relay pvc 100
PVC Statistics for interface Serial0/0 (Frame Relay DTE)
DLCI = 100, DLCI USAGE = LOCAL, PVC STATUS = ACTIVE, INTERFACE = Serial0/0.100
input pkts 19962 output pkts 25727 in bytes 297929
out bytes 2120032 dropped pkts 42 in FECN pkts 0
in BECN pkts 0 out FECN pkts 0 out BECN pkts 0
in DE pkts 0 out DE pkts 0
out bcast pkts 5775 out bcast bytes 1778682
pvc create time 4d00h, last time pvc status changed 20:02:22
```

- If DLCI is configured on the router then USAGE=LOCAL
- If DLCI is not configured on the router but switch is reporting the DLCI, then USAGE=UNUSED
- If keepalives are disabled, then STATUS=STATIC
- If DLCI is defined on the router but not switch, then STATUS=DELETED
- If DLCI is defined on switch but the pvc is not up (i.e. network failure), then STATUS=INACTIVE
- If DLCI is defined on the switch and is enabled, then STATUS=ACTIVE
Why Use Frame Relay Traffic Shaping?

![Diagram showing Max Be, Bc, and DE Domain]

What We’re Trying to Avoid

Router#show Frame Relay pvc 401

PVC Statistics for interface Serial1/0:0 (Frame Relay DTE)

DLCI = 401, DLCI USAGE = LOCAL, PVC STATUS = ACTIVE, INTERFACE = Serial1/0:0.401

input pkts 3420  output pkts 50150  in bytes 4974630
out bytes 75255200  dropped pkts 0  in FECN pkts 737
in BECN pkts 275  out FECN pkts 0  out BECN pkts 0
in DE pkts 91  out DE pkts 0
out bcast pkts 150  out bcast bytes 55200
pvc create time 02:28:23, last time pvc status changed 02:26:45
Frame Relay Traffic Shaping Facts

• Frame Relay Traffic Shaping (FRTS) takes the frame relay header and encapsulation into account
• The default value for “mincir” is CIR/2
• FRTS will activate without input BECNs if the instantaneous bit rate exceeds configured CIR

More FRTS Facts

• FRTS works on a token-bucket algorithm
• Excess burst bits (Be) above committed burst bits (Bc) are only sent in the 1st FRTS interval (Tc) after a quiet period
• As such, configure
  \[ \text{CIR} = (\text{Purchased CIR}) + \text{EIR} \]
  \[ \text{MinCIR} = (\text{Purchased CIR}) \]
  \[ \text{Bc} = \text{Bc} + \text{Be} \]
  \[ \text{Be} = 0 \]
Configuring FRTS

- Example configuration:
  ```
  interface Serial1/0:0
  encapsulation Frame Relay
  Frame Relay traffic-shaping
  !
  interface Serial1/0:0.401 point-to-point
  Frame Relay class 1536
  Frame Relay interface-dlci 401
  !
  map-class Frame Relay 1536
  Frame Relay cir 1536000
  Frame Relay bc 192000
  Frame Relay be 0
  Frame Relay mincir 768000
  Frame Relay adaptive-shaping becn
  Frame Relay fecn-adapt
  ```

Frame—ATM Interworking FRF.5

- The Frame Relay (FR) to ATM network interworking function allows FR traffic to be transported through an ATM network; FRF.5 enables two FR end stations to communicate with each other through an ATM network
Frame—ATM Interworking FRF.8

- The Frame Relay (FR) to ATM network interworking function allows FR traffic to be communication between an FR end user and an ATM end user; it is based on the Frame Relay Forum (FRF.8) implementation agreement.

![Diagram of Frame Relay to ATM interworking]

Frame—ATM Interworking Troubleshooting

- Use the same debug tools with ATM-FR interworking as you would use with a Frame Relay connection, or an ATM connection; these include “show frame pvc”, “show atm int”, “show atm vc”, and “show interfaces”.

- When in doubt, always use IETF encapsulation on the Frame Relay side; remember, Frame Relay encapsulation (Cisco) assumes you have another, Cisco encapsulated frame relay interface on the other side.

- FRF.8 allows for OAM cells to be translated to status messages passed along by LMI status updates, therefore the following sections on OAM management should be considered for managing ATM-FR connections as well.
ATM in the WAN

ATM: Point-to-Point vs. Multipoint

• Like Frame Relay, ATM supports two types of interfaces: point-to-point and multipoint; the one you choose determines whether you need to use the configuration commands that ensure IP to ATM mappings; after configuring the PVC itself, you must tell the router which PVC to use in order to reach a specific destination
ATM: Point-to-Point

- Point-to-point subinterface—with point-to-point subinterfaces, each pair of routers has its own subnet; if you put the PVC on a point-to-point subinterface, the router assumes that there is only one point-to-point PVC configured on the subinterface; therefore, any IP packets with a destination IP address in the same subnet are forwarded on this VC; this is the simplest way to configure the mapping and is therefore the recommended method.

ATM: Multipoint

- Multipoint networks—multipoint networks have three or more routers in the same subnet; if you put the PVC in a point-to-multipoint subinterface or in the main interface (which is multipoint by default), you need to either configure a static mapping or enable inverse Address Resolution Protocol (ARP) for dynamic mapping.
ATM: Inverse ARP (InATMARP)

- With InATMARP, the ATM interface knows the layer 2 address; this is the PVC’s virtual path identifier (VPI) and/or virtual channel identifier (VCI); however, it still needs to discover which IP address is reachable at the remote end of a connection, to do this, the router sends an InATMARP request over a virtual connection for the address of the other end.

ATM: Debugging Inverse ARP

- In the example on the next slide, we have defined a point-to-point subinterface and configured an IP address and a PVC, but we have not defined static mapping; InATMARP ensures that the ping across the link is successful; to show the InATMARP negotiation, we have enabled debug atm arp.
ATM: Debugging Inverse ARP (Cont.)

Router#show running-config interface ATM 1/1/0.100
  interface ATM1/1/0.100 point-to-point
  ip address 1.1.1.1 255.255.255.0
  no ip directed-broadcast
  pvc 2/100
  shutdown
Router(config)#interface atm 1/1/0.100
Router(config-subif)#no shut
ATMARP: Sending first PVC INARP
ATMARP(ATM1/1/0.100): INARP REQ to VCD#19 2/100 for link 7(IP)
ATMARP(ATM1/1/0.100): INARP Reply VCD#19 2/100 from 1.1.1.2

ATM: Debugging Inverse ARP (Cont.)

- The show atm map command displays the dynamic mapping via inverse ARP on ATM

Router#show atm map
  Map list ATM1/1/0.100_ATM_INARP : DYNAMIC
    ip 1.1.1.2 maps to VC 19, VPI 2, VCI 100, ATM1/1/0.100
ATM: Debugging Inverse ARP (Cont.)

• Inverse ARP also is enabled on multipoint links by default; in the following example, we have created a multipoint sub-interface; by using the `debug atm arp` command, we can see that InATMARP again builds a dynamic mapping between the layer 3 IP address and the layer 2 VPI/VCI.

ATMARP: Sending first PVC INARP

ATMARP (ATM1/1/0.200)O: INARP_REQ to VCD#20 2/200 for link 7(IP)

ATMARP (ATM1/1/0.200)I: INARP Reply VCD#20 2/200 from 2.2.2.2
ATM: Debugging Inverse ARP (Cont.)

Router#show atm map
Map list ATM1/1/0.100_ATM_INARP : DYNAMIC
ip 1.1.1.2 maps to VC 19, VPI 2, VCI 100, ATM1/1/0.100
Map list ATM1/1/0.200_ATM_INARP : DYNAMIC
ip 2.2.2.2 maps to VC 20, VPI 2, VCI 200, ATM1/1/0.200

ATM: show atm pvc <vcd>

• Use the show atm pvc <vcd number> command to establish a baseline
• Ping the remote end and ensure that the router displays five-packet increments for both InPkts and OutPkts; look for the ABCD payload pattern to ensure that the packets are pings and not other packets' operation, administration and maintenance (OAM) cells
• Execute the show atm pvc <vcd number> command again and ensure that the OutPkts counter increments by at least five packets
• Compare the OutPkts value with the value you recorded before doing the ping; note that this interface still does not log any InPkts; this display suggests that the router is sending packets, but the remote device is not receiving them; a value of zero for InPkts suggests that the end-to-end path in the ATM switch cloud is not properly provisioned
ATM: show atm pvc <vcd>

Router#show atm pvc test
ATM2/IMA0.294: VCD: 5, VPI: 7, VCI: 192, Connection Name: test
VBR-NRT, PeakRate: 500, Average Rate: 500, Burst Cells: 100
AAL5-LLC/SNAP, etype:0x0, Flags: 0x20, VCmode: 0x0
*****snipped OAM*****
InARP frequency: 15 minutes(s)
Transmit priority 2
InPkts: 0, OutPkts: 5, InBytes: 0, OutBytes: 500
InFRoc: 0, OutFRoc: 6
InFast: 0, OutFast: 0, InAS: 0, OutAS: 0
InPktDrops: 0, OutPktDrops: 0
CrcErrors: 0, SarTimeOuts: 0, OversizedSDUs: 0
*****snipped OAM*****
Status: UP

ATM: Static Map Classes

- Static map lists are a Cisco IOS® software feature that offers an alternative to using the ATMARP and InATMARP mechanisms; using them, you can associate a protocol address with an ATM address on an SVC, or with a VPI/VCI on a PVC

```
interface ATM1/1/0.200 multipoint
   ip address 2.2.2.1 255.255.255.0
   no ip directed-broadcast
   pvc 2/200
   protocol ip 2.2.2.2 broadcast
```
ATM: Static Map Classes (Cont.)

- You can check the mapping using the show atm map command; as you can see, the mapping of layer 3 to layer 2 addresses is permanent, rather than dynamic, as it was when we used inverse ARP

7500#show atm map
Map list ATM1/1/0.100_ATM_INARP : DYNAMIC
ip 1.1.1.2 maps to VC 19, VPI 2, VCI 100, ATM1/1/0.100

Map list ATM1/1/0.200pvc20 : PERMANENT
ip 2.2.2.2 maps to VC 20, VPI 2, VCI 200, ATM1/1/0.200 , broadcast

ATM: debug atm packet

Router#debug atm packet

Dec  7 10:21:16 : ATM2/IMA0.294(O):
VCD:0x5 VPI:0x7 VCI:0x0 DM:0x100 SAP:AAAA CTL:03 OUI:000000 TYPE:0800
Length:0x70

Dec  7 10:21:16 : 4500 0064 0032 0000 FF01 7643 0A90 9801 0A9 0 9802  0800
BAA2 0031 0ED1 0000

Dec  7 10:21:16 : 0000 5A75 5A50 ABCD ABCD ABCD ABCD ABCD ABCD ABCD

Dec  7 10:21:16 : ABCD ABCD ABCD ABCD ABCD ABCD ABCD

Dec  7 10:21:16 : ..
ATM: debug atm packet (Breakdown)

- Let's look at what this output means:
  
  Dec 7 10:21:16 : ATM2/IMA0.294(O):
  
  VCD:0x5 VPI:0x7 VCI:0xC0 DM:0x100 SAP:AAAA CTL:03
  OUI:000000 TYPE:0800 Length:0x70

  - **ATM2/IMA0.294(O)**—the packet is an output packet

  - **VPI:0x7 VCI:0xC0**—the packet is being transmitted on VPI 7 and VCI 192 (0xC0)

  - These values are provided in hexadecimal format; convert them to decimal to ensure that the router is using the correct PVC values in the ATM five-byte header

ATM: debug atm packet (Breakdown)

Dec 7 10:21:16 : ATM2/IMA0.294(O):

VCD:0x5 VPI:0x7 VCI:0xC0 DM:0x100 SAP:AAAA CTL:03 OUI:000000 TYPE:0800 Length:0x70

Dec 7 10:21:16 : ABCD ABCD ABCD ABCD ABCD

- **SAP:AAAA**—a SNAP header follows

- **OUI:000000**—the following PID is an EtherType

- **TYPE: 0800**—this is a "well-known" EtherType value for the IP

- **ABCD ABCD ABCD**—default payload pattern of a ping packet
debug atm errors

- Ensure that the router knows which VC to use to reach the remote destination; use the debug atm errors command on the interface; this debug command is non-intrusive and it only produces output if there are a lot of ATM errors.

  Do you see a line similar to the following in the output?

  Jul 12 05:01:26: ATM(ATM6/0): Encapsulation error1, link=7, host=B010117

- The problem may be that you have miss-configured the static mappings.

  We can see from this output that the ping fails and the encapsulation error message is recorded; if we look at the hexadecimal value B010117, here is what we can see:

  B -> 11
  01 -> 1
  01 -> 1
  17 -> 23

  The hexadecimal value displayed is 11.1.1.23, the address we are trying to ping.
ATM: What’s an ATM CRC Error?

- On an ATM interface, CRC errors occur when the ATM network provider drops some cells of a total packet in the switch “cloud”; this can be because you are transmitting packets faster than the carrier has configured your Sustainable Cell Rate (SCR).
- The ATM interface detects these lost cells on the receiving end when the segmentation and reassembly (SAR) function reassembles the cells to create a complete packet again; thus, CRC errors on ATM interfaces may point to a mismatch in traffic shaping and traffic policing parameters.

ATM: Which CRC Are We Checking?

- Request for Comments (RFC) 1483, Multiprotocol Encapsulation over ATM Adaptation Layer 5, defines aal5snap encapsulation, as well as defining how it should use the AAL5 trailer.
- ATM supports five ATM adaptation layers (AALs); AAL5 appends an eight-byte trailer to the common part convergence sublayer protocol data unit (CPCS-PDU).
- This CRC fills the last four bytes of the trailer and protects most of the CPCS-PDU, except for the actual CRC field itself.
### ATM: Which CRC Are We Checking?

<table>
<thead>
<tr>
<th>Layer Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPCS-PDU Payload (up to 2 2/16-1 octets)</td>
</tr>
<tr>
<td>PAD (0-47 octet)</td>
</tr>
<tr>
<td>CPCS (1 octet)</td>
</tr>
<tr>
<td>CPI (1 octet)</td>
</tr>
<tr>
<td>Length (2 octets)</td>
</tr>
<tr>
<td>CRC (4 octets)</td>
</tr>
<tr>
<td>CPCS-PDU Trailer</td>
</tr>
</tbody>
</table>

### ATM: Finding CRC Errors

- Use the command **“show interfaces atm slot/port”** to identify the presence of CRC errors on an ATM interface
**ATM: CRC Example**

Router#show interfaces atm 4/0
ATM4/0 is up, line protocol is up
Hardware is cxBus ATM
Internet address is 131.108.97.165, subnet mask is 255.255.255.0
MTU 4470 bytes, BW 100000 Kbit, DLY 100 usec, rely 255/255, load 1/255
ATM E164 Auto Conversion Interface
Encapsulation ATM, loopback not set, keepalive set (10 sec)
Encapsulation(s): AAL5, PVC mode
256 TX buffers, 256 RX buffers, 1024 Maximum VCs, 1 Current VCs
Signalling vc = 1, vpi = 0, vci = 5
ATM NSAP address: BC.CDEF.01.234567.890A.BCDE.F012.3456.7890.1234.13
Last input 0:00:05, output 0:00:05, output hang never
Last clearing of "show interface" counters never
Output queue 0/40, 0 drops; input queue 0/75, 0 drops
Five minute input rate 0 bits/sec, 0 packets/sec
Five minute output rate 0 bits/sec, 0 packets/sec
144 packets input, 31480 bytes, 0 no buffer
13 input errors, 12 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
154 packets output, 4228 bytes, 0 underruns
0 output errors, 0 collisions, 1 interface resets, 0 restarts

**ATM: Reasons for CRC Errors**

- Potential reasons for ATM CRC errors:
  - Dropped cells due to traffic policing in the ATM cloud on one or more VCs
  - Noise, gain hits, or other transmission problems on the data-link equipment
  - A faulty or failing ATM interface
ATM: When Do CRC Errors Occur?

- Do they occur during certain times of the day or during periods of high traffic?
  - If so, you may be exceeding the traffic shaping parameters agreed with your ATM service provider
  - Look into the switch cloud and determine whether there is congestion.; this might involve asking the provider
- Confirm your traffic shaping parameters with your provider
  - Does your provider see any cells with the cell loss priority (CLP) bit in the ATM header set to one (1)?
  - Has the provider recorded dropped cells on its switch interfaces?
- Test the line using pings with various IP packet sizes using extended ping

ATM: Are Cells Dropping Now?

- Determine if the CRC counter is incrementing or whether it is a historical value from a problem that has now been corrected
  - Execute the show interfaces atm command multiple times over a few hours or days
  - Clear the counters if appropriate for easier troubleshooting
  - Is the circuit new? Has it ever worked without CRC errors?
ATM: Isolate the Problem

• Perform loopback tests on the line to determine whether the CRC errors point to noise or other transmission problems
  
  Create a test PVC on the two ATM interfaces and assign IP addresses
  
  Use the loopback line command on the remote ATM router interface

ATM: Is It a Hardware Problem?

• Place a physical loopback on the interface and ping your own IP address

• Create a soft loopback with the loopback diagnostic and atm clock internal commands on the main ATM interface

• If loopbacks fail, try other ports on the same card

• If card looks suspect, move it to another slot in the chassis and retest
ATM: Using OAM for Troubleshooting

- If a communication problem occurs on a PVC (no traffic going one way or the other), the PVC remains UP on the end-devices; therefore, routing entries that were pointing to that PVC remain in the routing table for a certain time and as a result, packets will be lost; the solution to this problem is to use Operation and Maintenance (OAM) to detect such failures and allow the PVC to disconnect if it is disrupted along its path.

ATM: Sample Network Diagram
ATM: Detecting Failures

- In order to detect a failure along the PVC path on an end-device, OAM uses specific cells:
  - Loopback cells
  - Continuity Check (CC) cells
  - Alarm Indication Signal (AIS) cells
  - RDI cells (Remote Detection Indication)

ATM: Detecting Failures (Cont.)

- There are three conditions to declare a PVC UP:
  - The router receives a configured number of successive end-to-end F5 OAM loopback cell replies
  - The router does not receive F5-AIS cells for 3 seconds
  - The router does not receive F5-RDI cells for 3 seconds
Tries:0
Tries:0

Tries:0

At This Point, the Sub-Interface Corresponding to PVC 1/116 on Guilder Is Shut Down
ATM: debug atm oam (Cell loss) (Cont.)

Tries:0

No Reply To The Loopback Cell Just Sent

Mar 30 14:48:45.676: ATM OAM LOOP(ATM2/0/0.116) O: VCD#4 VC 1/116 CTag:438A
Tries:5 <- the router makes 5 retries before declaring the PVC down
Mar 30 14:48:46.676: %LINEPROTO-5-UPDOWN: Line protocol on Interface ATM2/0/0.116,changed state to down

***snipped – 4 retries***
ATM: sh atm pvc (OAM)

Bernard# sh atm pvc 1/116

"snip"

AAL5-LLC/SNAP, etype:0x0, Flags: 0xC20, VCmode: 0x0
OAM frequency: 10 second(s), OAM retry frequency: 1 second(s)
OAM up retry count: 3, OAM down retry count: 5
OAM Loopback status: OAM Sent
"snip"

OAM cells received: 9
F5 InEndloop: 9, F5 InSegloop: 0, F5 InAIS: 0, F5 InRDI: 0
F4 InEndloop: 0, F4 InSegloop: 0, F4 InAIS: 0, F4 InRDI: 0
OAM cells sent: 18
F5 OutEndloop: 18, F5 OutSegloop: 0, F5 OutRDI: 0
F4 OutEndloop: 0, F4 OutSegloop: 0, F4 OutRDI: 0
OAM cell drops: 0
Status: DOWN, State: NOT_VERIFIED

ATM: AIS/RDI Cells (Diagram)

• Upon detection of a failure, a device configured for OAM sends AIS frames downstream and sends RDI frames upstream
Session Goal

• Troubleshooting process
  Check your physical layer first
  Use your debugs
  Use your show commands

Troubleshooting WAN Protocols
Session WMS-301
Please Complete Your Evaluation Form

Session WMS-301
Reference Slides

Cabling and Lead States
T1: DB-15 to RJ-45 Cable Pin Out

<table>
<thead>
<tr>
<th>DB15 Pin</th>
<th>RJ45 Pin</th>
<th>Signal Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5</td>
<td>TX Tip (T1)</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>RX Tip (T)</td>
</tr>
<tr>
<td>9</td>
<td>4</td>
<td>TX Ring (R1)</td>
</tr>
<tr>
<td>11</td>
<td>1</td>
<td>RX Ring (R)</td>
</tr>
</tbody>
</table>

- The above pin out is for a “straight through” cable
- This cable is typically used to connect a CSU to the telco demarcation point

T1: Cabling and Power Issues

- Shielded cables should connect CSU/DSU to Telco demarcation point
- Unshielded cables expose TX and RX pairs to potential cross talk which causes BPVs and CRC-6 errors
- A T1 voltage pulse must have a base-to-peak amplitude between 2.4V and 3.6V
Serial Control Leads

- Transmit Data (TD)
- Receive Data (RD)
- Request to Send (RTS)
- Clear to Send (CTS)
- Data Terminal Ready (DTR)
- Data Set Ready (DSR)
- Carrier Detect (CD or DCD)
- Receive Clock (RxC)
- Transmit Clock (TxC)
- Transmit Clock External (TxCE or SCTE)

Special Functionality

V.35 Signal Mapping

<table>
<thead>
<tr>
<th>DCE 34 Pin</th>
<th>Signal Name</th>
<th>DTE 34 Pin</th>
<th>DCE 34 Pin</th>
<th>Signal Name</th>
<th>DTE 34 Pin</th>
</tr>
</thead>
<tbody>
<tr>
<td>R</td>
<td>T</td>
<td>P</td>
<td>E</td>
<td>F</td>
<td></td>
</tr>
<tr>
<td>T</td>
<td>TD+</td>
<td></td>
<td></td>
<td>DCD</td>
<td></td>
</tr>
<tr>
<td>P</td>
<td>TD-</td>
<td>R</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>S</td>
<td>RD+</td>
<td>T</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>RF-</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>RTS</td>
<td>C</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>CTS</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>H</td>
<td>DTR</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>V</td>
<td>RXC+</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X</td>
<td>RXC-</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Y</td>
<td>TXC+</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AA</td>
<td>TXC-</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>U</td>
<td>SCTE+</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>W</td>
<td>SCTE-</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Copyright © 1998, Cisco Systems, Inc. All rights reserved. Printed in USA.
Presentation_ID.scr
34 Pin “Winchester” Connector Pinout

Frame Formats
Cisco’s HDLC: Frame Format

flag = start/end of frame = 0x7E
(Other special characters: Idle = 0xFF, Abort = 0x7F)
address = this is really a frame type field
0x0F = Unicast Frame
0x80 = Broadcast Frame (used if upper layer packet is a broadcast)
0x40 = Padded Frame
0x20 = Compressed Frame
Protocol = the Ethernet type of the encapsulated data:
TYPE_IP10MB 0x0800 IP
TYPE_IEEE_SPANNING 0x4242 DSAP/SSAP for IEEE bridge spanning protocol
TYPE_DECNET 0x6003 DECnet phase IV
TYPE_BRIDGE 0x6558 Bridged Ethernet/802.3 packet
TYPE.Reverse_ARP 0x8035 cisco SLARP
TYPE_DEC_SPANNING 0x8038 DEC bridge spanning tree protocol
TYPE_EtherTalk 0x809b Apple EtherTalk
TYPE_AARP 0x80f3 Appletalk ARP
TYPE_NOVELL1 0x8137 Novell IPX
TYPE_CLNS 0xFEFE ISO CLNP/ISO ES-IS DSAP/SSAP

PPP: Frame Format

Modeled after HDLC frame w/addition of protocol field
PPP frame size ranges from 10 bytes to 4 bytes (when PFC and ACFC compression are enabled, and w/o ending flag)
Ending flag only needed on single frame or final frame of a sequence

*Omitted When Address and Control Field Compression (ACFC) Is Used
**Only 1 Byte When Protocol Field Compression (PFC) Is Used
### Frame Relay Packet

<table>
<thead>
<tr>
<th>Bits</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q.922 Header</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>O.922</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Protocol</td>
<td>NLPID</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0x08</td>
<td>0x80</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Q.933</td>
<td>SNAP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0x81</td>
<td>ISO CLNP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0xCC</td>
<td>Internet IP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0xB1</td>
<td>FRF.12</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**NOTE:** Cisco Encap uses 2 byte Ethertype instead of the Control and NLPID bytes.

#### SNAP Format - 3 Byte OUI and 2 Byte PID

- **OUI:**
  - 0x00 00 00 routed
  - 0x00 80 C2 bridged

- **PID**
  - EtherType
  - With Preserved FCS
  - 0x00 01
  - 0x00 02
  - 0x00 03
  - 0x00 04

- **W/O FCS**
  - 0x00 07
  - 0x00 08
  - 0x00 09
  - 0x00 0A
  - 0x00 0D
  - 0x00 0E
  - 0x00 0F

#### Media

- 802.3/Ethernet
- 802.4
- 802.5
- FDDI
- Fragments
- BPDU’s
- Source-routing
- BPDU’s

### NNI Cell Header

- **GFC**—4 bits of generic flow control that are used to provide local functions, such as identifying multiple stations that share a single ATM interface; the GFC field is typically not used and is set to a default value.
- **VPI**—8 bits of virtual path identifier that is used, in conjunction with the VCI, to identify the next destination of a cell as it passes through a series of ATM switch routers on its way to its destination.
- **VCI**—16 bits of virtual channel identifier that is used, in conjunction with the VPI, to identify the next destination of a cell as it passes through a series of ATM switch routers on its way to its destination.
- **PT**—3 bits of payload type; the first bit indicates whether the cell contains user data or control data; if the cell contains user data, the second bit indicates congestion, and the third bit indicates whether the cell is the last in a series of cells that represent a single AAL5 frame.
- **CLP**—1 bit of congestion loss priority that indicates whether the cell should be discarded if it encounters extreme congestion as it moves through the network.
- **HEC**—8 bits of header error control that are a checksum calculated only on the header itself.
### NNI Cell Header

- **VPI**—8 bits of virtual path identifier that is used, in conjunction with the VCI, to identify the next destination of a cell as it passes through a series of ATM switch routers on its way to its destination.
- **VCI**—16 bits of virtual channel identifier that is used, in conjunction with the VPI, to identify the next destination of a cell as it passes through a series of ATM switch routers on its way to its destination.
- **PT**—3 bits of payload type; the first bit indicates whether the cell contains user data or control data; if the cell contains user data, the second bit indicates congestion, and the third bit indicates whether the cell is the last in a series of cells that represent a single AAL5 frame.
- **CLP**—1 bit of congestion loss priority that indicates whether the cell should be discarded if it encounters extreme congestion as it moves through the network.
- **HEC**—8 bits of header error control that are a checksum calculated only on the header itself.
- The GFC field is not present in the format of the NNI header; instead, the VPI field occupies the first 12 bits, which allows ATM switch routers to assign larger VPI values; with that exception, the format of the NNI header is identical to the format of the UNI header.

### OAM Cell Format

- **OAM cell type** is coded as 0001.
- **OAM function type** is coded as 0010.
- 350 bits that are specific to the OAM type are divided into the following elements:
  - **Loopback indicator**—a bit that is set to 1 before the cell is looped back. The loopback node then sets the bit to 0, indicating it has been looped back.
  - **Correlation tag**—identifies (correlates) related OAM cells within the same connection.
  - **Loopback location ID**—an optional field that identifies the site that is to loopback the cell.
  - **Source ID**—An optional field that identifies the site generating the cell.

---

Copyright © 1998, Cisco Systems, Inc. All rights reserved. Printed in USA.

Presentation_ID.scr