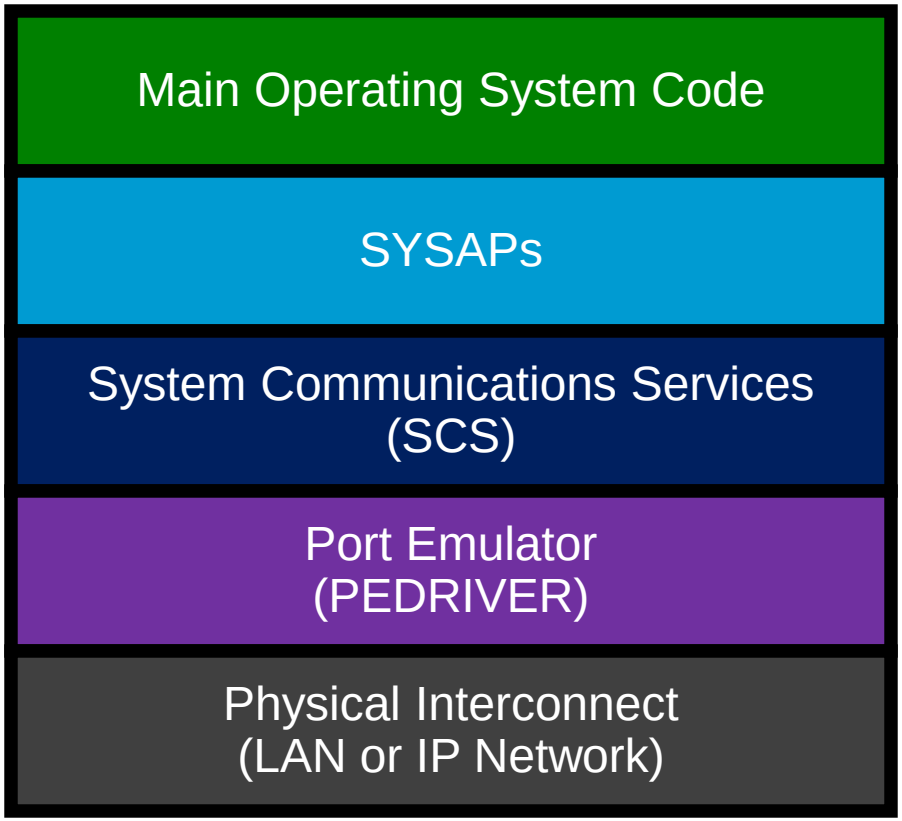
Hewlett Packard
Enterprise

Assessing Your OpenVMS Cluster Interconnect

Part 2: Performance

Keith Parris
Engineer, Hewlett Packard Enterprise

Cluster Communications Layers



SCS Credits

- OpenVMS Cluster System Communications Services (SCS) uses credit-based flow control
  - Each SYSAP-to-SYSAP connections has a set number of available credits
  - Transmitting a sequenced message requires a credit
  - Receiving an ACKnowledgement for a message frees a credit
- Credit Waits are a possible indication of some combination of:
  - Insufficient credits,
  - Excessive inter-node latency, or
  - Excessive service time

  and in any case, a potential adverse impact to performance
- Detecting SCS Credit Waits:
  - $ SHOW CLUSTER /CONTINUOUS with ADD CONNECTIONS,REM_PROC_NAME,CR_WAITS and SET CR_WAITS /WIDTH=10
  - T4 data T4_*_SCS.CSV data file, viewed with Tlviz or checked/graphed with CSVPNG
  - SDA> SHOW CONNECTIONS

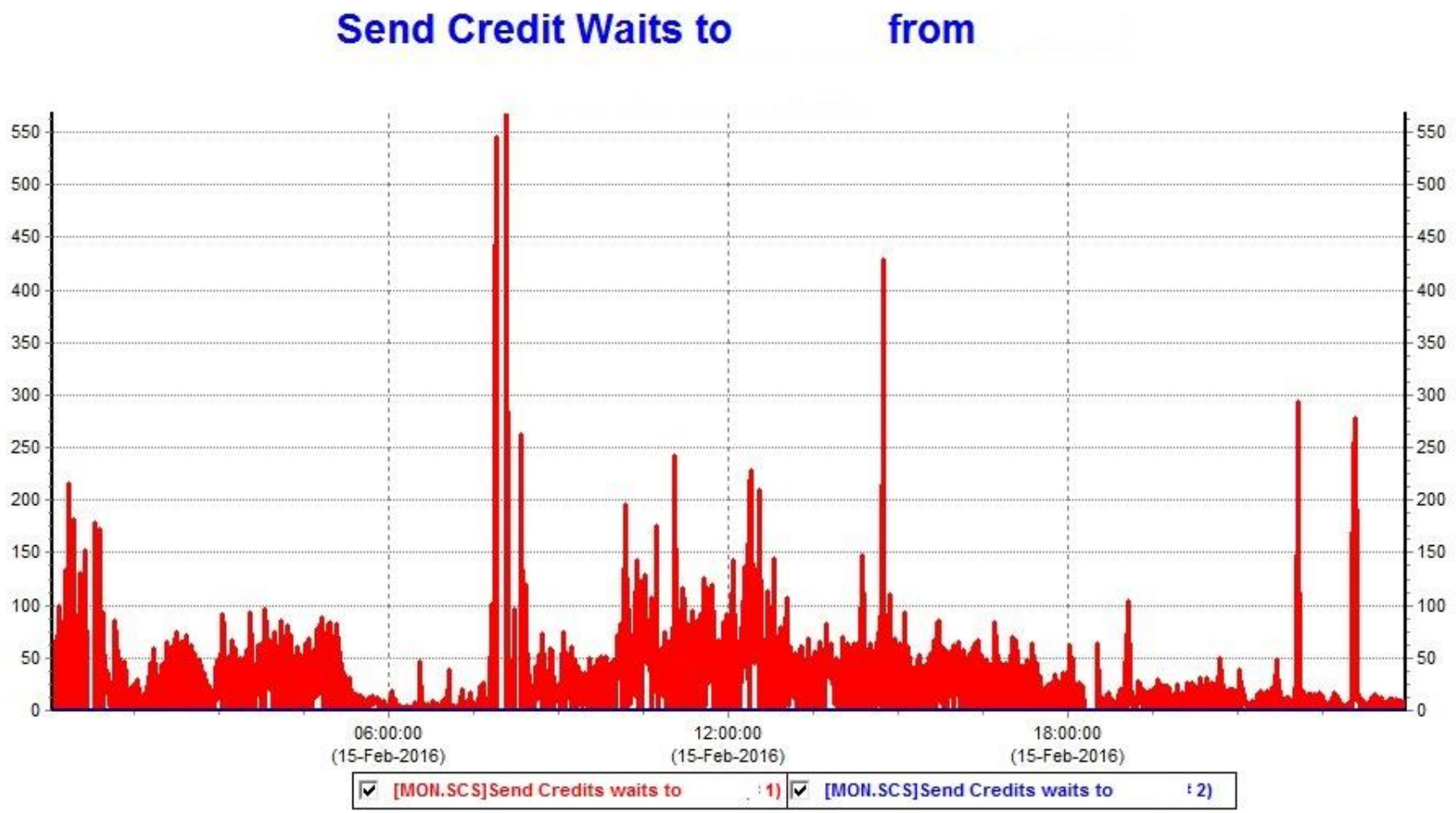
SCS Credits

– Detecting with $ SHOW CLUSTER /CONTINUOUS

```
View of Cluster from system ID 1133  node: NODE9          29-NOV-2012 09:24:55
+--------+--------------+---------+------------------+---------+----------+
|        SYSTEMS         | MEMBERS |        CONNECTIONS         | COUNTERS |
+--------+--------------+---------+------------------+---------+----------+
|  NODE  |   SOFTWARE   |  STATUS |   LOC_PROC_NAME  | CON_STA | CR_WAITS |
+--------+--------------+---------+------------------+---------+----------+
|        |              |         | VMS$DISK_CL_DRVR | OPEN    |        0 |
| NODE7  | VMS V8.4     | MEMBER  | SCA$TRANSPORT    | OPEN    |      139 |
|        |              |         | VMS$DISK_CL_DRVR | OPEN    |        0 |
|        |              |         | VMS$VAXcluster   | OPEN    |    36760 |
|        |              |         | MSCP$DISK        | OPEN    |        0 |
| NOD12  | VMS V8.4     | MEMBER  | VMS$DISK_CL_DRVR | OPEN    |        0 |
|        |              |         | MSCP$DISK        | OPEN    |        0 |
|        |              |         | VMS$VAXcluster   | OPEN    |    61575 |
| NODE8  | VMS V8.4     | MEMBER  | VMS$DISK_CL_DRVR | OPEN    |        0 |
|        |              |         | MSCP$DISK        | OPEN    |        0 |
|        |              |         | VMS$VAXcluster   | OPEN    |    36345 |
| NODE6  | VMS V8.4     | MEMBER  | VMS$DISK_CL_DRVR | OPEN    |        0 |
|        |              |         | MSCP$DISK        | OPEN    |        0 |
|        |              |         | VMS$VAXcluster   | OPEN    |    32668 |
└────────┴──────────────┴─────────┴──────────────────┴─────────┴──────────┘
```

SCS Credits

– Detecting with T4 SCS data collector and TLviz



SCS Credits

– Detecting with SDA

VMS$VAXcluster is adjustable in both directions, by raising CLUSTER_CREDITS on both nodes:

```
SDA> SHOW CONNECTIONS
...
VMScluster data structures
--------------------------
          --- Connection Descriptor Table (CDT) 885D4D40 ---
State:          0002 open             Local Process:        VMS$VAXcluster
Blocked State:  0000                  Remote Node::Process: VC1::VMS$VAXcluster

Local Con. ID   2A910006    Datagrams sent         0    Message queue   885D4D7C
Remote Con. ID  81650007    Datagrams rcvd         0    Send Credit Q.  885D4D84
Receive Credit       127    Datagram discard       0    PB address      889D1280
Send Credit          128    Message Sends    5901883    PDT address     882AE7B8
Min. Rec. Credit       0    Message Recvs   11503234    Error Notify    D03782C8
Pend Rec. Credit       1    Mess Sends NoFP  5901883    Receive Buffer  885D6F60
Initial Rec. Credit  128    Mess Recvs NoFP 11503234    Connect Data    885ED470
Rem. Sta.   0000000000F9    Send Data Init.       23    Aux. Structure  885ED440
Rej/Disconn Reason     0    Req Data Init.        26    Fast Recvmsg Rq 00000000
Queued for BDLT        0    Bytes Sent         43344    Fast Recvmsg PM 00000000
Queued Send Credit     0    Bytes rcvd         44708    Change Affinity 00000000
                            Tot bytes map     161088
...
SYSGEN> show cluster_credits
Parameter Name            Current    Default     Min.       Max.   Unit  Dynamic
--------------            -------    -------   -------    -------  ----  -------
CLUSTER_CREDITS               128         32        10        128 Credits
```

SCS Credits

– Detecting with SDA

MSCP is adjustable in only one direction (from client to server, by raising MSCP_CREDITS on MSCP-serving node):

```
SDA> SHOW CONNECTIONS
...
VMScluster data structures
--------------------------
          --- Connection Descriptor Table (CDT) 885D6680 ---
State:          0002 open              Local Process:        VMS$DISK_CL_DRVR
Blocked State:  0000                   Remote Node::Process: VC1::MSCP$DISK

Local Con. ID   2AE20007    Datagrams sent          0    Message queue   885D66BC
Remote Con. ID  81680006    Datagrams rcvd          0    Send Credit Q.  885D66C4
Receive Credit        10    Datagram discard        0    PB address      889D1280
Send Credit           31    Message Sends      117031    PDT address     882AE7B8
Min. Rec. Credit       1    Message Recvs      117031    Error Notify    D04ECC18
Pend Rec. Credit       0    Mess Sends NoFP    117031    Receive Buffer  889DA0E0
Initial Rec. Credit   10    Mess Recvs NoFP    117031    Connect Data    D04E2A3C
Rem. Sta.   0000000000F9    Send Data Init.         0    Aux. Structure  886CFB40
Rej/Disconn Reason     0    Req Data Init.          0    Fast Recvmsg Rq D04ECBD8
Queued for BDLT        0    Bytes Sent              0    Fast Recvmsg PM D04ECBB8
Queued Send Credit     0    Bytes rcvd              0    Change Affinity D04ECB98
                            Tot bytes map           0
...
SYSGEN> SHOW MSCP_CREDITS
Parameter Name            Current    Default     Min.       Max.   Unit  Dynamic
--------------            -------    -------   -------    -------  ----  -------
MSCP_CREDITS                   32         32         2       1024 Coded-valu
```

SCS Credits

– Detecting with SDA

Some are not adjustable at all (e.g. SCA$TRANSPORT, used for queue manager and DECdtm):

```
SDA> SHOW CONNECTIONS
...
VMScluster data structures
--------------------------
          --- Connection Descriptor Table (CDT) 886CC480 ---
State:          0002 open             Local Process:        SCA$TRANSPORT
Blocked State:  0000                  Remote Node::Process: VC5::SCA$TRANSPORT

Local Con. ID   2A910014    Datagrams sent         0    Message queue   886CC4BC
Remote Con. ID  81680012    Datagrams rcvd         0    Send Credit Q.  886CC4C4
Receive Credit         6    Datagram discard       0    PB address      885EF1C0
Send Credit            5    Message Sends          4    PDT address     882AE7B8
Min. Rec. Credit       0    Message Recvs          4    Error Notify    D038C070
Pend Rec. Credit       0    Mess Sends NoFP        4    Receive Buffer  887277E0
Initial Rec. Credit    6    Mess Recvs NoFP        4    Connect Data    8872AAB0
Rem. Sta.   0000000000F8    Send Data Init.        0    Aux. Structure  8872AA00
Rej/Disconn Reason     0    Req Data Init.         0    Fast Recvmsg Rq 00000000
Queued for BDLT        0    Bytes Sent             0    Fast Recvmsg PM 00000000
Queued Send Credit     0    Bytes rcvd             0    Change Affinity 00000000
                            Tot bytes map          0
```

SCS Credits

– Mitigating Credit Waits by raising available SCS credit counts

– SCS Credits for different SYSAPs are controlled in different ways:
  – VMS$VAXcluster – to – VMS$VAXcluster SYSAP connection:
    – SYSGEN parameter CLUSTER_CREDITS
  – VMS$DISK_CL_DRVR – to – MSCP$DISK SYSAP connection:
    – SYSGEN parameter MSCP$CREDITS on client node
  – Other SYSAPs: Hard-coded values (talk to HPE Technology Services Support)

PEDRIVER performance considerations

– Parallel transmission with multiple channels (paths) in Equivalent Channel Set (ECS)

– PEDRIVER Transmit Window Size

– Avoid saturation in interrupt state of the CPU handling PEDRIVER

PEDRIVER performance considerations

– Equivalent Channel Set (ECS)

– Parallel transmission with multiple channels (paths) in Equivalent Channel Set (ECS)
  – ECS is the set of channels on which we transmit data to a given remote node, in a round-robin fashion
    – We can always receive on a channel whether or not it is presently in the ECS
  – Channels are classified by:
    – Are they presently in the ECS or not? (Yes or No)
    – Packet loss rate (Tight or Lossy)
    – Packet maximum payload size (Inferior, Peer, or Superior)
    – Latency (Fast or Slow)
    – Examples:
    – Y(T,P,F) – Good path
    – N(T,I,F) – Inferior payload size, perhaps IPCI path in presence of LANCI path
    – N(T,P,S) – Slow latency path
    – N(L,P,F) – High packet loss rate

PEDRIVER performance considerations

– PEDRIVER Transmit Window

– PEDRIVER Transmit Window Size

- Maximum number of outstanding sequenced packets PEDRIVER will allow
  - Acknowledgement of a sequenced packet allows another packet to transmit
- PEDRIVER calculates default PEDRIVER Maximum Transmit Window to a node based on:
  - Speed of interconnect:
    - 10 megabit Ethernet: 8
    - 100 megabit “Fast” Ethernet: 16
    - FDDI: 31
    - 1 gigabit Ethernet: 32
    - 10 gigabit Ethernet: 64
  - Number of parallel channels (paths) through the LAN or IP network
  - Example:
    - 2 parallel channels at 1-gigabit bandwidth = 2 x 32 = 64
- Starting with OpenVMS version 8.3, Maximum PEDRIVER Transmit Window may be modified using:
  - SCACP> SET VC *nodename* /WINDOW={RECEIVE | TRANSMIT} = *n*
    - When increasing, raise receive side first; when reducing, cut transmit side first

PEDRIVER performance considerations

– PEDRIVER Transmit Window

– PEDRIVER Transmit Window Size

- Starting with OpenVMS version 8.3, Maximum PEDRIVER Transmit Window may be modified using:
  - SCACP> SET VC ***nodename*** /WINDOW={RECEIVE or TRANSMIT} = ***n***
    - When increasing, raise receive side first; when reducing, cut transmit side first
- An appropriate PEDRIVER Transmit Window size may be calculated using:
  - SCACP> CALCULATE WINDOW_SIZE /SPEED=***n*** –
    /DISTANCE=[KILOMETERS or MILES]=***d*** –
    /OPTIMIZE=[IO or LOCKING]

where ***n*** where is the inter-site link speed in megabits per second, while ***d*** is the distance between sites.

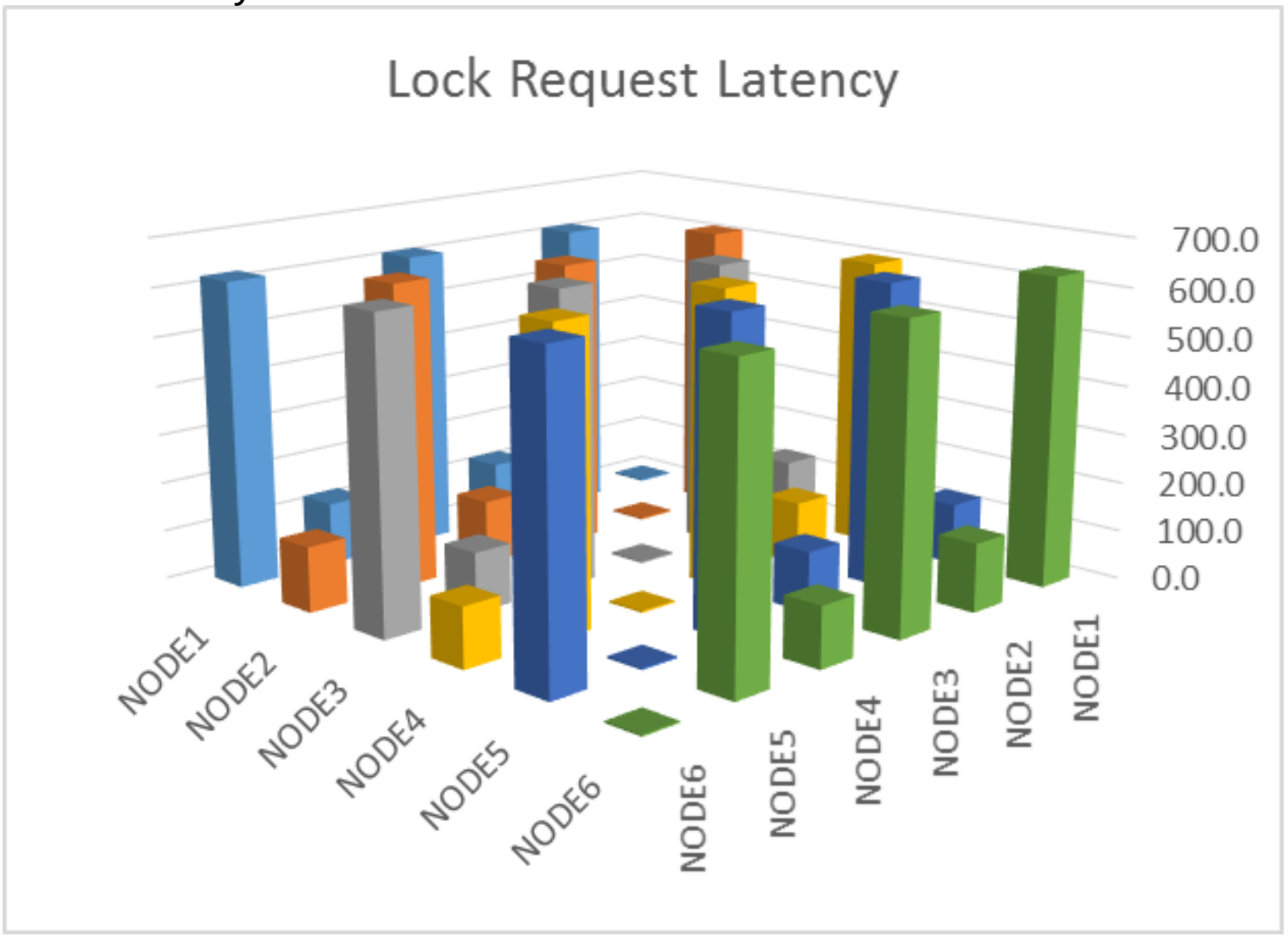
PEDRIVER performance considerations

– PEDRIVER Transmit Window

– PEDRIVER Transmit Window Size
  – Current Transmit Window Size can grow and shrink over time:
    – Starts out at 1
    – Grows with successful acknowledgements of sequenced packets
      – Tends to grow fairly slowly over time
      – The more the traffic, the faster it can grow
    – Cut back if packets need to be retransmitted, either because
      – Sequenced packet itself is lost, or
      – Acknowledgement for the sequenced packet is lost on the way back
    – If 1 retransmission occurs, Current transmit window is cut back to ½ of Maximum
    – If multiple retransmissions occur in a short timeframe, Current transmit window is cut back down to 1

Lock Request Latency

– As measured by LOCKTIME.COM tool



Case Study

– Dropped packets and high latency in the network

– 2-site OpenVMS Cluster, Volume Shadowing, MSCP-Served remote disk access

– During high workload periods, eventually application performance was affected so badly the application became useless

– Checked T4 data for saturation of any CPU in Interrupt State – no problem

– Enabled Jumbo Packets -- no help

Case Study

- Dropped packets and high latency in the network

- Identified that poor performance was due to problems in inter-site cluster interconnect:
  - To emulate LAN bridging, vendor uses MPLS encapsulation, 8 MPLS hops
    - High inter-site latency (almost 5 milliseconds according to LOCKTIME.COM, despite short inter-site distance)
  - Small packets being dropped (based on RFC 2544 test results)
    - Inter-site link via 2 different layered vendors; packet loss in lowest-layer provider
- Lost packets caused PEDRIVER Current transmit window size to be cut back, adversely affecting remote disk writes for Volume Shadowing to the remote site. Intersystems Caché is very tolerant to write latencies, but eventually things backed up to the application level.
- Tried using “new” (since 2009) T4 PEDRIVER VC Data Collector in version T4.4 to track PEDRIVER Current transmit window value
  - Conclusion: This data collector was not ready for prime time

Case Study

- Dropped packets and high latency in the network

- Solutions suggested:
  - Investigate lower-latency alternative to MPLS encapsulation (particularly with 8 MPLS hops) for emulating LAN bridging
    - Market demand for IP segments across sites to support live VM migration for DR has created new solutions in recent years
  - IP as a Cluster Interconnect over native IP network, to replace MPLS encapsulation with straightforward IP network
  - SAN Extension instead of MSCP Serving, coupled with using Cisco I/O Accelerator or Brocade Fast Write for remote writes in 1 round trip instead of 2 round trips
- Solutions tried:
  - Upper-layer network provider added router at each end to track loss of packets
    - Added 1 more millisecond of latency ☹
- Solution: Customer had a non-zero Recovery Point Objective value, which allowed them to move from OpenVMS Multi-Site Clustering and Host-Based Volume Shadowing to Intersystems Caché asynchronous replication