13th ANNUAL WORKSHOP 2017

HOST BASED INFINIBAND NETWORK FABRIC MONITORING

Michael Aguilar

Jim Brandt Staff R&D, Dr. Benjamin Allan, and Dr. Douglas Pase

Sandia National Laboratories

[March, 2017]

Exceptional service in the national interest

Sandia National Laboratories

U.S. DEPARTMENT OF ENERGY

NNSA National Nuclear Security Administration

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000. SAND NO. 2013-6521C

OUTLINE

- **Motivation**
  - Create monitoring tool to see congestion and contention on Infiniband fabrics
- **Goals**
  - To better understand network data flow as it is routed through IB switches
  - Less congestion=more computational efficiency
- **Related Work**
  - Previous Infiniband fabric monitoring
- **Approach**
  - “Synchronously” retrieve metrics from all of the switch ports within the cluster Infiniband fabric
- **Initial Results**
  - How long does it take to get metrics from switch ports and what is the load on the system?
- **Future Work**
- **Conclusions**

MOTIVATION

MOTIVATION

- **Infiniband network fabric problems not well understood**
  - Congestion
    - Magnitude, extent, longevity
  - Attribution to sources
- **Current Infiniband fabric monitoring tools shortfalls**
  - Too low fidelity and lack sufficient synchronization to be of practical use in understanding congestion phenomena
    - Collection periods of minutes make attribution difficult
    - The unsynchronized nature of current tools makes it difficult to view system wide “snapshots”

GOALS

GOALS

- **Understand network data flow as it is routed through Infiniband Switches**
  - Internal RDMA data transfers, using Infiniband, often are made through several layers of Infiniband switches on large HPC systems
  - Allow a more complete inspection of dynamic data-flows during application runs
- **Develop Tools to identify network congestion within an IB fabric**
  - Counter monitoring at the injection and ejection points doesn't provide contention location information
    - Better understanding of factors that might cause congestion
- **Develop Tools and techniques for early identification and mitigation of network contention related problems**
  - Mitigate problems through feedback
    - Scheduler, resource manager, application

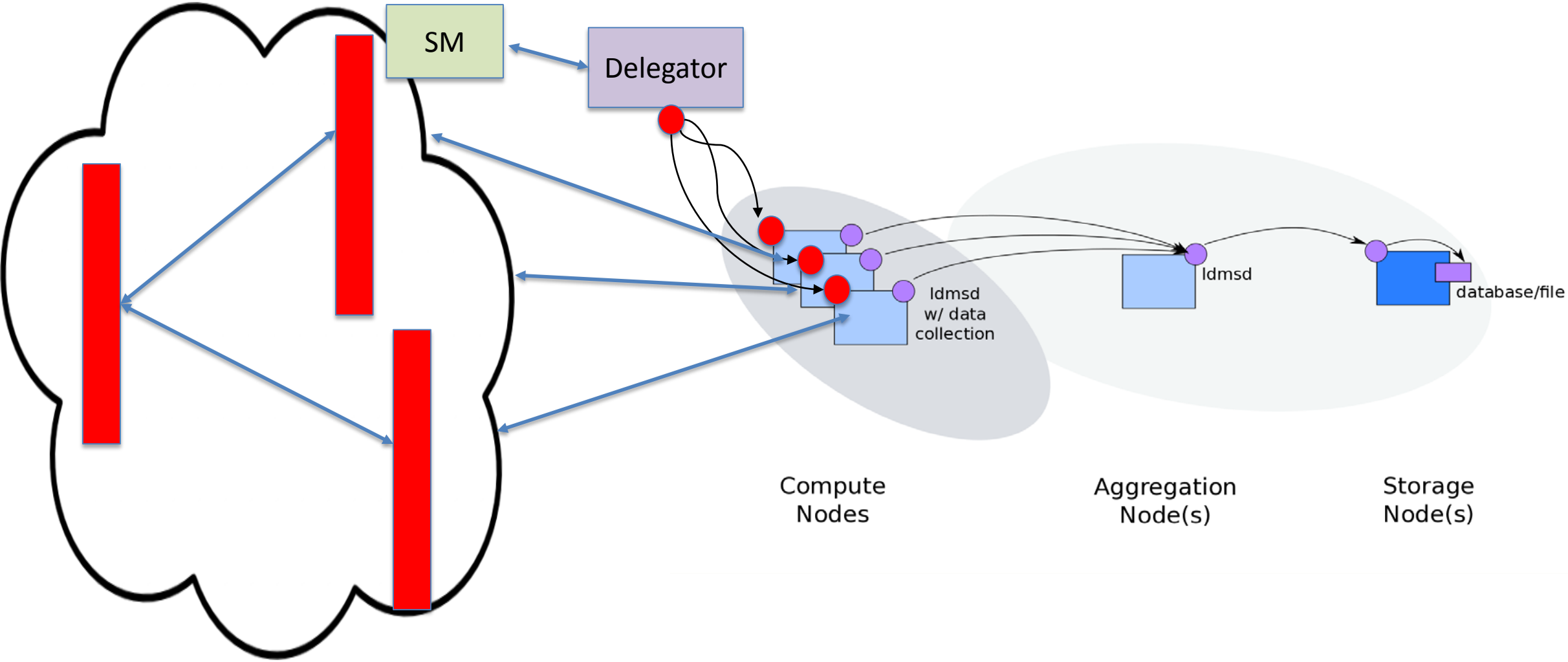
RELATED WORK

RELATED WORK

- **INAM and INAM^2 allow congestion monitoring of the entire subnet, including switches**
  - Because MAD port counter metrics are retrieved from Open Subnet Manager sweeps, no mention is made about time synchronization
- **Ganglia doesn’t store metric data for analysis, doesn’t store congestion metrics, and can only monitor endpoint HCAs**
- **Subnet Manager can query congestion metrics, during a sweep, over a very wide window sampling of time making dynamic changes in traffic flow during application runs difficult to analyze**
  - Congestion metrics are available at a large delay in time
- **Nagios can be programmed to retrieve congestion metrics**
  - It has a very wide window of sampling time plus no time-stamp to help analyze the metric data
- **The LDMS sampler Sysclassib provides timely data-gathering**
  - At each compute node, the Sysclassib sampler reads metrics from the local HCA LID and port
    - Reads congestion metrics at the endpoints

APPROACH

APPROACH


SM
Delegator
ldmsd
w/ data
collection
ldmsd
database/file
Compute
Nodes
Aggregation
Node(s)
Storage
Node(s)

APPROACH

- **Create a distributed LDMS sampler to ‘synchronously’ retrieve metrics from all of the switch ports within the cluster Infiniband fabric**
  - We were able to gather metrics from both compute node end-points, but we wanted to gather metrics from the switch ports between the compute nodes, as well
    - By sampling data from switches within an HPC system and all active HCAs at the end-points, detailed information on data flows can be analyzed
      - The user can specify a subset of available switch port metrics to collect. If a set of metrics isn’t chosen by a user, the default setting is to display all of the metrics
      - To allow easy parsing of resulting data, all of the metrics should display the LID and port identifiers
      - Allow the user to pick network neighborhood switch ports to check from available sampler HCAs
  - LDMS samplers maintain a time-synchronized metric gathering and report when there are time synchronization issues
  - The sampler provides network connection information, as well

INFINIBAND AND MAD QUERIES CAN BE PERFORMED FROM ANY HCA WITHIN THE CONNECTED FABRIC

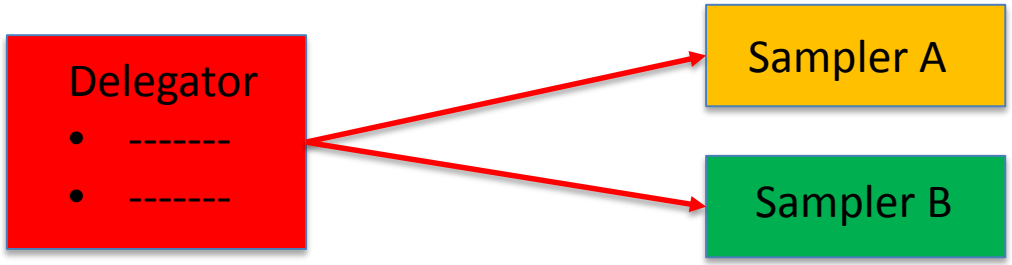
- **Metrics can be read using MAD queries through VL15 on any LID and port within the fabric**
  - A sampler can be created to read RDMA network transmission metrics within the fabric
    - By gathering full data metrics from switches within the fabric, a more complete picture can be gathered of how much data is traversing each switch port
- **The load of monitoring the HCAs and connections in the different layers of switches can be spread across all nodes**
  - The connectivity is itself important metric information
  - OFED libnetdisc can be use to find connectivity and determine how to divide up the query tasks

DELEGATOR DAEMON

- **A delegator daemon gathers information on network connections and available LIDs and ports**
  - With the LIDs and ports gathered, each sampler is then provided with a subset of the network to gather metrics from
    - Division of labor within the subnet
    - Each sampler was provided a list of LIDs and ports, the brand and type of network port, and the connecting LID and port
  - Samplers were given their identities and work to be performed by their component identifier numbers
    - An input file was used to list the metrics that the LDMS sampler was to gather from each port
    - An input parameter would provide the delegator with information on the brand and type of Infiniband hardware to expect
    - Finally, an input file containing both the quantity of samplers used in metric gathering and the some user-defined LIDS and ports to sample from
      - Allowed the LDMS IBFabric sampler to gather metrics from locally connected switches

DELEGATOR DAEMON

- **Read the entire Infiniband fabric, create a list of HCAs, distribute the HCAs to fabric samplers**
- **Preconditions:**
  - Multi-threading support active on the host used for the port delegator
  - HCA network connection for the Infiniband network to sample with
  - OFED libraries libmad, libibmad, and libnetdisc used by the port delegator
- **Delegation Steps:**
  - Check for a list of user-defined metrics to check, if none is available, then provide a default list of every metric
  - Check for a list of sampler port assignments. If no port assignments exist then evenly spread out the port assignments among the samplers



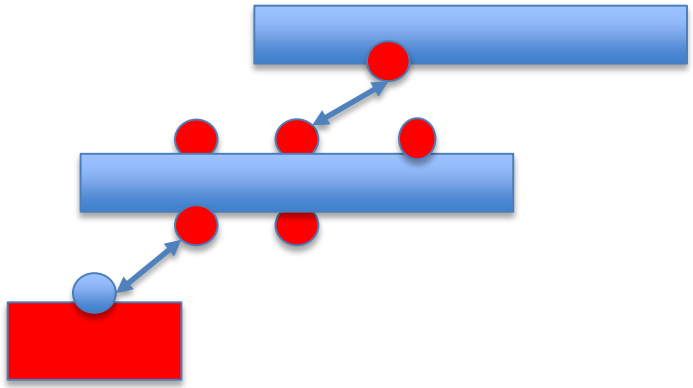
  - Acquire a map of the Infiniband fabric using libnetdisc
  - Create a list of recovered HCAs using an array of metrics. Each element in the list should contain a LID identifier, a port identifier, a remote LID identifier, a remote port identifier, and if the HCA is 32-bit or 64-bit
  - Upon receiving a component identifier from a sampler, provide a list of switch ports (LID, port) to sample, and a list of metrics

DELEGATOR DAEMON

- **Sample Metric Selection File**
  - Configuration file allows a choice of target metrics for the user
    - Traffic volume, congestion, link errors, remote connection port, etc.
      - Transmit waits report back pressure, VL15 drops will indicate that metric data coming from a switch port should be suspect
  - The input file metrics mirror the MAD query names
    - Currently, the input metrics that we collect are a ‘core group’ of common metrics that all types and brands of switch ports would use
- **The user specifies the brand and type of switch hardware that is installed within the fabric that they would like to sample**
  - Currently supports only homegeneous fabrics
  - We will extend the delegator to support heterogeneous network environments

DELEGATOR DAEMON

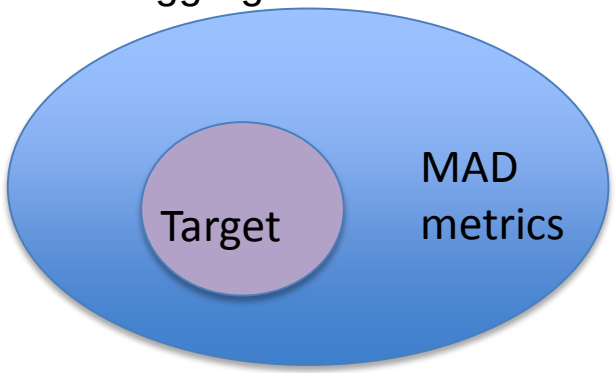
- **A second input file for users to tell the delegator how many samplers are being used to gather metrics in the fabric and also to provide a list of LIDs and ports that the user might want to match to individual samplers**
  - The user can target a list of Lids and ports to a specific sampler
    - match a sampler to a set of switch ports that might be a better choice for gathering metrics from



- **Any switch ports that are in the network that aren't user-chosen are divided up evenly across samplers**

SAMPLERS

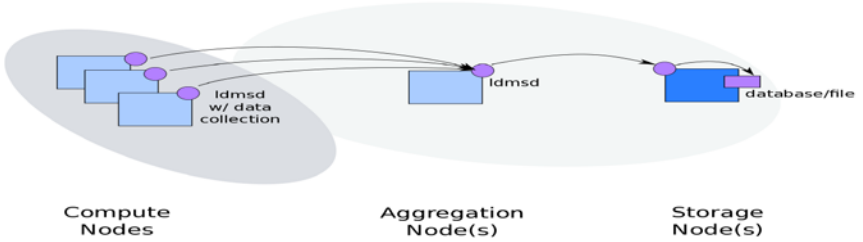
- During configuration, the new IBFabric sampler communicates its component identifier to the delegator then gets a group of LIDs and ports to check
- MAD queries are performed using each switch LID and individual port identifier. With the user target metrics, the data is gathered into a complete LDMS sampling set
  - When a MAD query is made, a complete set of metrics from the port is received. The sampler decodes and assembles a subset of target metrics for the the LDMS aggregator



  - Each LDMS set size is designed to be dynamically allocated by the number of LIDs and ports to sample and the user target metrics

SAMPLERS

- **Using a list of HCAs to check and a list of requested metrics, sample and report the metrics to an LDMS aggregator service**
- **Preconditions:**
  - Infiniband connection to cluster network fabric
  - A TCP/IP network connection to the port delegator service
  - Libraries ibmad, ibumad
- **Sampler operational steps:**
  - Start up an LDMS sampler process
  - Using LDMS control, load the ibfabric sampler
  - Configure the sampler including the component identifier
  - Request a list of HCAs and user defined metrics to check using the component identifier
  - Vary the length of reported data by the number of HCAs and the number of metrics to check
  - Begin to report back metrics on an LDMS defined sampling interval



SAMPLERS

- **Sample LDMS metric output from IBFabric:**

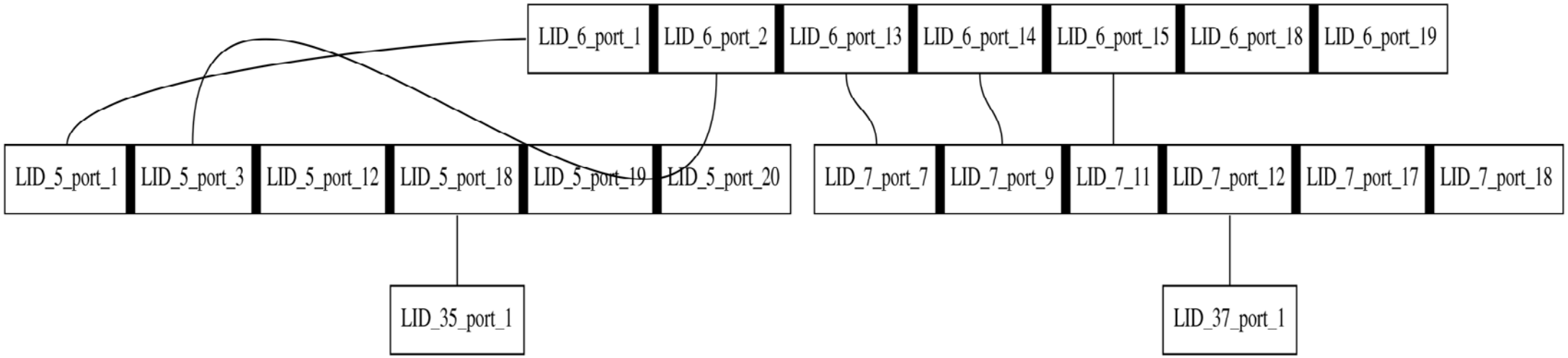
```
[root@shaun4 brandt]# ldms_ls -h localhost -x sock -p 60020 -v
shaun4/sysclassib: consistent, last update: Thu Mar 30 06:06:21 2017 [2500us]
  METADATA --------
    Producer Name : shaun4
    Instance Name : shaun4/sysclassib
      Schema Name : sysclassib_4
             Size : 2760
     Metric Count : 47
               GN : 3
  DATA ------------
        Timestamp : Thu Mar 30 06:06:21 2017 [2500us]
         Duration : [0.000350s]
       Consistent : TRUE
             Size : 408
               GN : 146917
  -----------------
```

- **Sample LDMS metric output from IBFabric:**

```
D u64       22.6#VL15_dropped                0
D f32       22.6#VL15_dropped.rate           0.000000
D u64       22.6#port_xmit_data              436309560
D f32       22.6#port_xmit_data.rate         144.081116
D u64       22.6#port_rcv_data               436372992
D f32       22.6#port_rcv_data.rate          144.081116
D u64       22.6#port_xmit_packets           6059855
D f32       22.6#port_xmit_packets.rate      2.001127
D u64       22.6#port_rcv_packets            6060736
D f32       22.6#port_rcv_packets.rate       2.001127
D u64       22.6#port_xmit_wait              0
D f32       22.6#port_xmit_wait.rate         0.000000
```

SAMPLERS

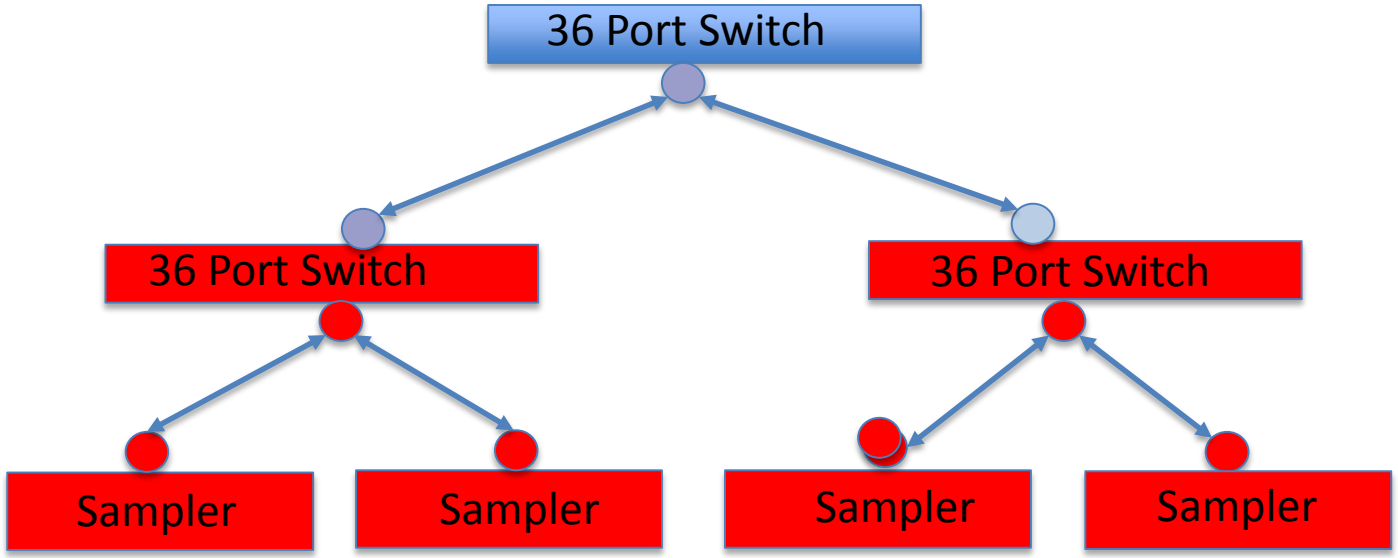
- Our sampler can pass on LID/port to connecting LID/port connections to aid a user in correlating metric data, as well.



INITIAL RESULTS

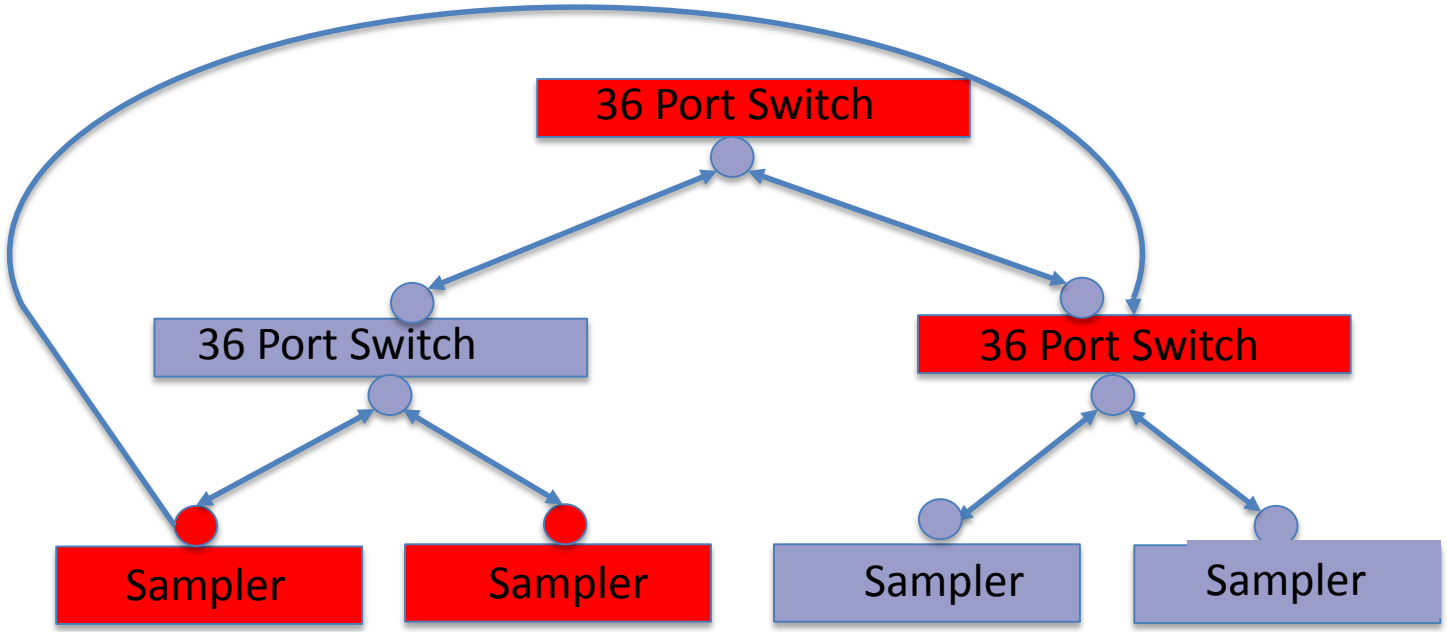
INITIAL RESULTS 3-SWITCH TRAVERSE

- **How long does it take to gather metrics that are locally connected to the sampler?**
  - Test Set Up

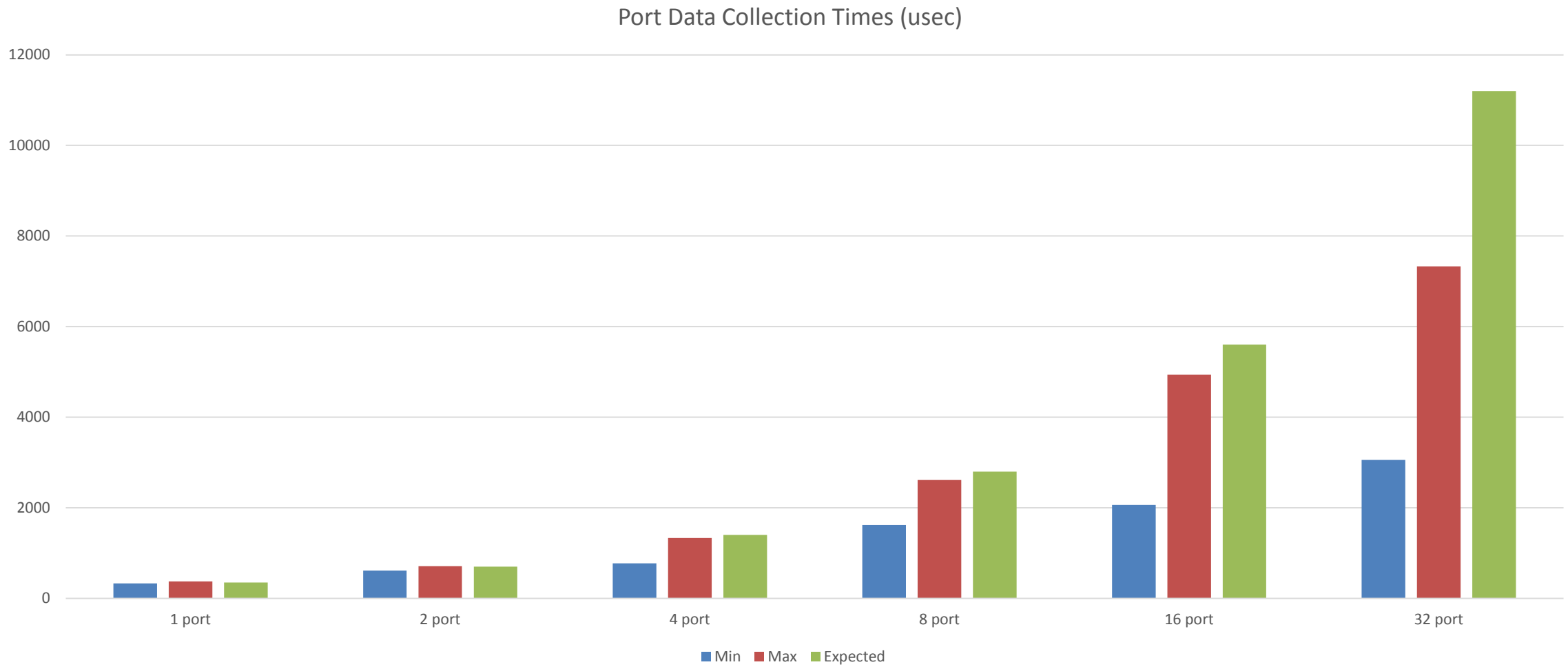


INITIAL RESULTS 3-SWITCH TRAVERSE

- **How long does it take to gather metrics that are remotely connected to the sampler?**
  - Test Set Up



INITIAL RESULTS 3-SWITCH TRAVERSE



SAMPLER LOAD ON SYSTEM

- **How much performance loss is there for running the LDMS sampler daemon and generating high levels of MPI traffic?**
  - An in-house Sandia MPI Performance tool (MPIPerf) was selected to run 500 randomized types of data traffic consisting of different data sizes with common MPI calls
  - No statistical difference in computational time for application runs was noticed on any of the test runs

FUTURE WORK

UPGRADES AND FUTURE WORK

- **Currently, if a sampler goes down, the metric gathering for a group of switch ports is lost**
  - We might like a way to notify the other samplers to be notified that a sampler is down and to begin sampling switch ports inn the absence of a lost sampler
  - Alternatively, we can collect data from each switch port using N sampling hosts to provide n-way redundancy and filter after to eliminate duplicates
- **Create a fabric sampler to work with OmniPath**

CONCLUSIONS

CONCLUSION

- **New measurement tool developed to help reveal congestion on physical Infiniband networks**
- **Demonstrated low overhead on test cluster hardware**
- **Reveals directly the previously inaccessible data switch behavior with time synchronization**

13th ANNUAL WORKSHOP 2017

THANK YOU

Michael Aguilar

Sandia National Laboratories

Exceptional service in the national interest

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000. SAND NO. 2013-6521C