IB Monitoring Through the Console

Jesse Martinez
Los Alamos National Laboratory
LA-UR-14-21958

April 3rd, 2013

UNCLASSIFIED

Operated by Los Alamos National Security, LLC for the U.S. Department of Energy's NNSA

Outline

- Monitoring Methods
  - Errors
  - Performance
- Use of Console
- Analysis and Reporting
- Future Implementations

Monitoring at LANL

- Monitoring is done per each cluster's fabric
  - Range from 8 node to 1600 node clusters
    - DDR, QDR, FDR systems
  - OpenSM 3.3.6 to 3.3.16-1
- Monitoring at near real time:
  - Fabric Errors
  - Non Optimal Links
  - Performance Issues
  - Bandwidth and Latency (Susan Coulter)
  - Throughput

IBMon2

- Developed by Susan Coulter
- Suite of scripts designed to look for InfiniBand hardware errors as well as performance metrics
- Runs off master nodes for each cluster
  - Where subnet manager is located
- Forwards messages to both Zenoss and Splunk
- Thresholds are set to trigger fabric errors and performance issues to send to operators and system administrators

Error Monitoring Methods

- Subnet Manager gathers counters from IB fabric continuously
- Scripts written to gather this data and convert it to readable format
  - Local Device: [Error == Counter] - (Remote Device)
- Error counters reset every half hour
  - Allows to monitor errors at near real time
  - Automatically disabled during Dedicated Service Time (DST)
- Errors messages recorded in syslog for each fabric

Performance Monitoring Methods

- Scripts written to gather transmit and receive data from ports throughout fabric
  - Recalculates actual data across 4 links and converts to MB
- Performance counters reset every half hour
- Throughput calculated based on transmit and receive data
  - Converts performance counters to Average MB/s
  - MB/30 minutes → ~MB/s
- Can look at overall cluster or port usage every half hour

Counters through Console

- Before: ibqueryerrors calls
  - Used before to gather errors and congestion counters on the fabric and modified by scripts
- OpenSM console used now to dump fabric counters via PerfMgr every half hour
  - Allows counters to be gathered continuously over fabric without additional calls from our scripts
  - Scripts parse dump file for information to gather error and performance counters
  - Calculations done on master nodes

Console Output

```
OpenSM $ help
Supported commands and syntax:
help [<command>]
quit (not valid in local mode; use ctl-c)
loglevel [<log-level>]
permodlog
priority [<sm-priority>]
resweep [heavy|light]
reroute
sweep [on|off]
status [loop]
logflush -- flush the opensm.log file
querylid lid -- print internal information about the lid specified
portstatus [ca|switch|router]
switchbalance [verbose] [guid]
lidbalance [switchguid]
dump_conf
update_desc
version -- print the OSM version
perfmgr(pm) [enable|disable
        |clear_counters|dump_counters|print_counters(pc)|print_errors(pe)
        |set_rm_nodes|clear_rm_nodes|clear_inactive
        |dump_redir|clear_redir
        |sweep|sweep_time[seconds]]
dump_portguid [file filename] regexp1 [regexp2 [regexp3 ...]] -- Dump port GUID matching a regexp
OpenSM $
```

Monitoring through Console

- Scripts search over all ports on hardware through dump file (Spine/Line cards, HCAs)
  - Locate at /var/log/opensm_port_counters.log
- Grep for non zero counters for errors
  - SymbolErrors, PortRcv, LinkedDowned, etc.
- Use source device/port to find remote device/port
  - Through ibnetdiscover parse
- Gathers performance metrics per port
- Sends error events to syslog and Zenoss
- Stores performance numbers in file (read by Splunk)

PerfMgr Dump File

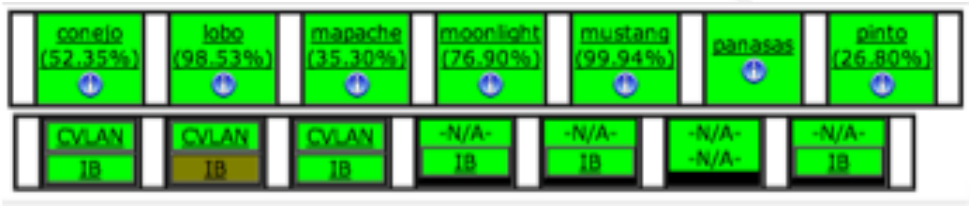
"mu1456" 0x2c9000100d050 active TRUE port 1

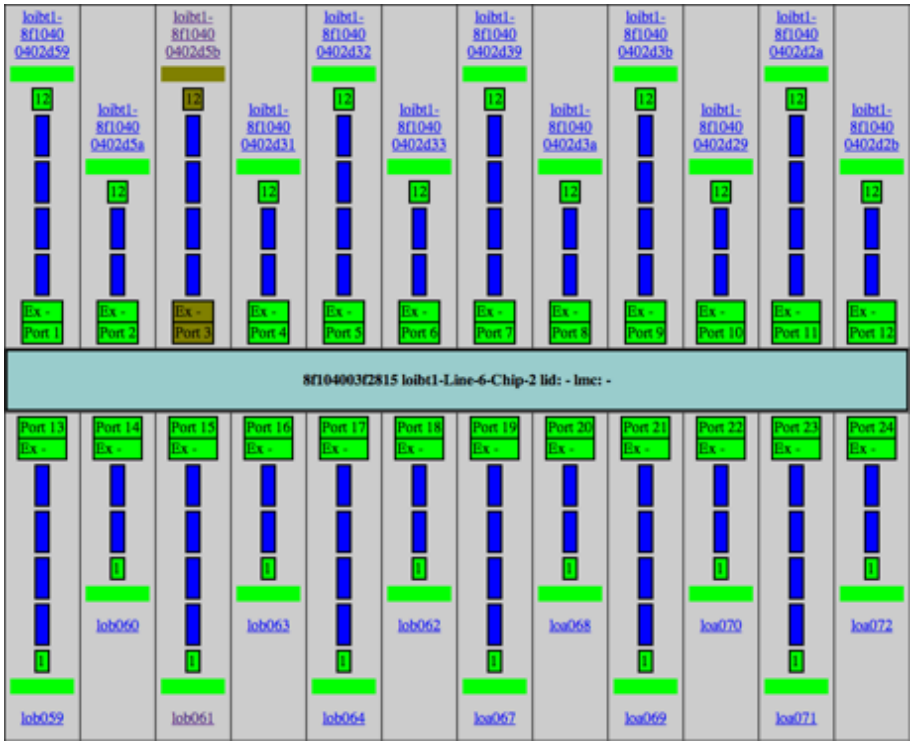
Last Reset	: Wed Mar 26 16:03:03 2014
Last Error Update	: Wed Mar 26 16:30:03 2014
symbol_err_cnt	: 0
link_err_recover	: 0
link_downed	: 0
rcv_err	: 0
rcv_rem_phys_err	: 0
rcv_switch_relay_err	: 0
xmit_discards	: 0
xmit_constraint_err	: 0
rcv_constraint_err	: 0
link_integrity_err	: 0
buf_overrun_err	: 0
vl15_dropped	: 0
Last Data Update	: Wed Mar 26 16:30:03 2014
xmit_data	: 141965786566 (528.864GB)
rcv_data	: 142302013218 (530.116GB)
xmit_pkts	: 706078664 (673.369M)
rcv_pkts	: 706229268 (673.513M)
unicast_xmit_pkts	: 0 (0.000)
unicast_rcv_pkts	: 0 (0.000)
multicast_xmit_pkts	: 0 (0.000)
multicast_rcv_pkts	: 0 (0.000)

Error Analysis and Reporting

- Two methods for monitoring errors
  - Zenoss
  - Splunk
- Why both?
  - Preference
  - Zenoss designed for real time virtualization of clusters to monitor errors
    - IB grid sent to Zenoss for virtualization
    - Automatically clear events
  - Splunk designed for analysis and benchmarking of performance and alerts

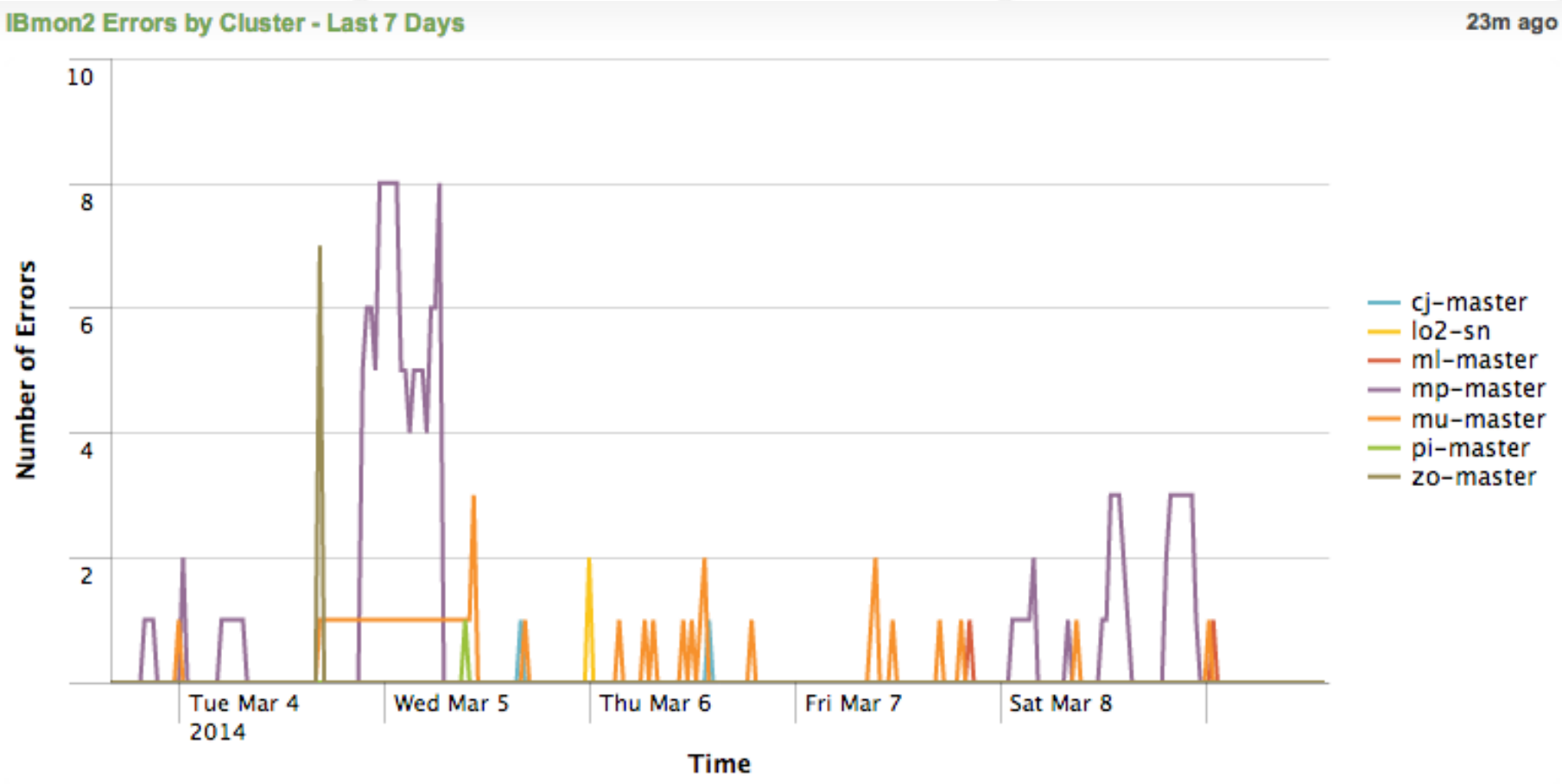
Zenoss Example



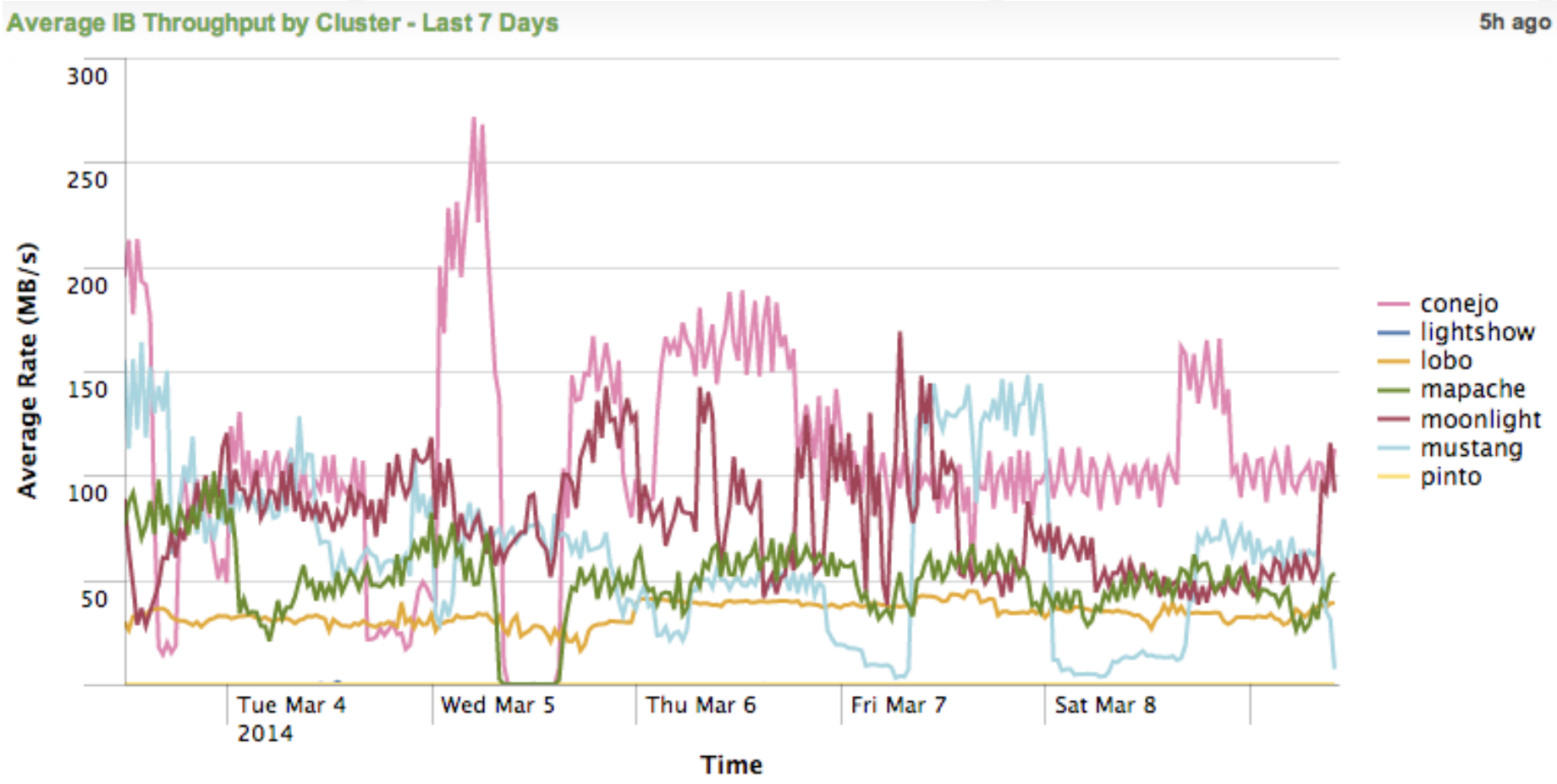


UNCLASSIFIED

Splunk Example



Splunk Example



Future Modifications

- Compatible IBmon2 for InfiniBand fabrics
  - Configuration Standards
    - Different fabric rates
    - Difference organizational implementations
- Pulling additional counters to look for trends in performance and error analysis
  - PortXmitWait
- Robust design to handle upgrades

Questions?