Adaptive and Dependable Server I/O Networking Support in a High Performance Cluster Computing Environment

Hsing-Bung Chen, HPC-5

We present an adaptive and dependable server I/O fault-management mechanism used in equal cost multi-path routing to enable LANL's parallel scalable back bone (PaScalBB) (Fig. 1) high performance computing (HPC) systems to run computational jobs around the clock without service interruption during unexpected physical I/O link failures and connection loss. This mechanism, named Dead Server I/O Gateway Detection and Recovery (DGD) (Fig. 2), can detect a data path connectivity problem within seconds when it happens. Then DGD removes the entry of a dead I/O gateway from a multi-path routing table, migrates a connecting I/O path to an available entrance in a multi-path routing table, and preserves and resumes the existing I/O data stream. DGD can tolerate multiple single points of failures, keep the streaming I/O data moving, and seamlessly continue and finish computation jobs. Figure 3 illustrates the self-explained pseudo-code for the proposed DGD mechanism. We have developed a proof-of-concept implementation of this proposed DGD mechanism on a Linux cluster as a blueprint for a production-type reliability-availability-serviceability (RAS) solution. Figures 4 and 5 show the testing cases of using this DGD process on BlueSteel (256-node InfiniBand cluster). Eventually we plan to apply this process to all LANL's PaScalBB-based HPC production clusters.

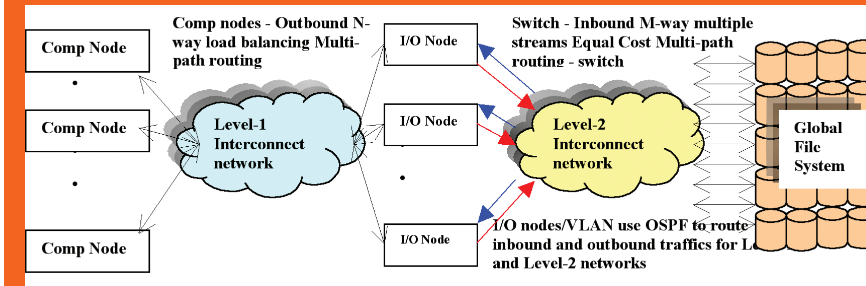

Fig. 1. Components used in LANL's PaScalBB backbone infrastructure.

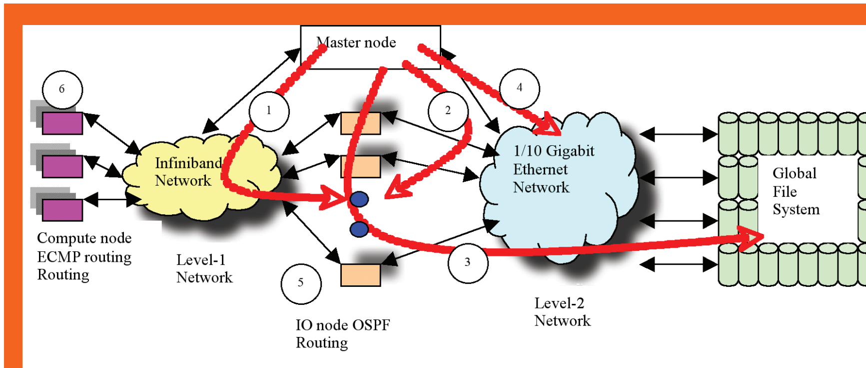

Fig. 2. DGD and ECMP route recovery system diagram and processing steps.

For further information contact Hsing-Bung Chen at hbchen@lanl.gov.

Funding Acknowledgments
- Department of Energy, National Nuclear Security Administration

```
initDGD( );
while (1) {
   1.   Check Level-1 connectivity:
            1.1   Master node → Level 1 Network → I/O nodes (Level 1 interface)
            1.2   Master node → Level 1 Network → I/O nodes (Level 2 interface)
   2.   Check Level-2 connectivity:
            2.1   Master node → Level 2 Network → I/O nodes (Level 2 interface)
            2.2   Master node → Level 2 Network → I/O nodes (Level 1 interface)
   3.   Check Global File System connectivity:
            Master node → Level 1 Network interface → Global File System
   4.   Check Status of the Level-2 Interconnect: Master node → Level 2 Storage network
   5.   Check OSPF status on both IO-gateway and Level-2 interconnect:
            OSPF routing status : Active vs. InActive
   Summarize of step1-step5 checking and set current I/O node status
   Update I/O nodes state machine and detect ECMP route fault-events
   if (PreviousStatus(I/O node)  equal CurretnStatus(I/O node)) {
            Active    → Active: No ECMP route change
            InActive → InActive: No ECMP route change
         ECMP route changed → NO
   }
   if (PreviousStatus(I/O node)  not equal CurretnStatus(I/O node)) {
            Active    → InActive: Remove dead/inactive I/O node from ECMP route
            InActive → Active: Add new Active I/O node back to ECMP route
         ECMP route changed → Yes
   }
   if (ECMP route changed) {
      6.    Adaptive Event-handling
         + Generating syslog information
         + Email DGD status updated notification
         + Record DGD history status
         + Adaptive route change on Equal Cost Multipath (ECMP) Route
            → Add or Remove I/O node from ECMP route
   }
   sleep N seconds; // kick-off DGD process every N seconds
}
```

Fig. 3. DGD processing pseudo code.

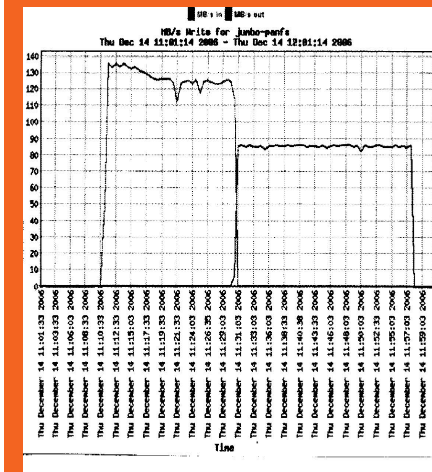

Fig. 4. Normal Write then Read with no dead I/O gateway.

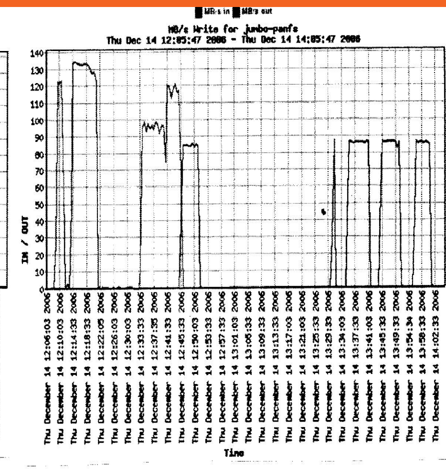

Fig. 5. Write then Read with multiple dead I/O gateways involved during testing.