Using GPFS 2.2 To Enable a Cross Platform Accessibility of Single Storage

W. Baird

National Energy Research Scientific Computing Division
Ernest Orlando Lawrence Berkeley National Laboratory
University of California
Berkeley, California 94720

With IBM's aid I have conducted a cross compatibility test of GPFS 2.2 between an IBM F50 Power2 running AIX 5.2 ML/3 and 8 Dual Pentium 4/2.2 GHz running Redhat 9.0. The objective was to demonstrate a single shared instance of the file system and storage between the disparate operating systems and hardware systems. The cross compatibility test was successful. The chronology of events that led to this successful test are documented below.

1.0 Introduction

Increasingly, scientific computing has been seeking ways to share data among different computing resources. What started with NFS with its known capabilities to share data among multiple computing systems without requiring the replication of that data has grown into a very large collection of cluster file systems. Multiple vendors have proposed different solutions. IBM has put forward its General Purpose File System (GPFS) as one solution.

The National Energy Research Scientific Computing Center (NERSC) has been investigating standardizing on a center wide shared file system for all its computing resources through the Global Unified Parallel File System (GUPFS) project. Standardizing on a unified center wide file system would allow researchers to maximize their time and allocated resources to computation and analysis rather than management of data. One of the driving requirements for this capability is to be able to use the same file system across multiple platforms. Additionally, it is required that the exact same instance of the file system be shared among very dissimilar computing platforms.

IBM has stated that their GPFS software would allow for Linux and AIX systems to simultaneously access the same data and storage. An attempt was made to configure and run a GPFS 2.2 cross platform compatibility test using a small cluster of eight Dual Pentium 4/2.2 GHz running Redhat 9.0 and a F50 Power2 running AIX 5.2. The objective was to demonstrate that a single GPFS instance was able to run on a Linux cluster and AIX node.

An Overview of GPFS

2.1 Definitions

GPFS is one of IBM's cluster file systems. GPFS' acronym literally stands for General Purpose File System. This is the file system being evaluated here for the process of building an operating system and hardware cross compatible shared storage.

RSCT is another IBM software package. RSCT's acronym is Reliable Scalable Cluster Technology. This provides group services and topology services for IBM software under Linux and AIX.

A GPFS instance is called a "cluster". This is not to be confused with the actual hardware of the system or systems. It is entirely possibly to have two different hardware clusters. For example, an IBM SP and an Intel Pentium cluster to participate in a single GPFS "cluster". This also should not be confused with a single mounted file system of GPFS. The "cluster" is only the nodes and configuration information that the instance needs to know to operate.

2.2 Creating a GPFS "Cluster"

When creating a GPFS "cluster", there are several steps that are done in two phases. The first phase is to configure the Reliable Scalable Cluster Technology (RSCT) software. The second phase is to configure the GPFS software specifically.

2.2.1 Configuring RSCT

The first phase is to configure the Reliable Scalable Cluster Technology (RSCT) software. GPFS version 2.2 is dependent upon the RSCT group services provided by a RSCT domain. The steps involved are relatively simple. There are only three steps for this first phase.

Phase One Commands:

Step:	Command:
1.1) Prepare the Nodes to establish the domain of trust:	*preprpnode*.
1.2) Make the domain:	*mkrpdomain*.
1.3) Start the RSCT domain:	*startrpdomain*.

preprpnode is the first command run. It must be run on all the different systems participating in an RSCT domain. It prepares the security features of a peer domain such that it is capable of distinguishing between trusted and untrusted nodes. Public keys are exchanged in the process and access control list permissions are granted between the trusted nodes.

mkrpdomain is the creation command for the RSCT peer domain. It provides and propagates the configuration information for that peer domain. It does not actually start the domain services. A single node may participate in multiple domains, but only one may be active at any given time. This command does not need to be run on each node. One node that will be a participant in the peer domain is sufficient.

startrpdomain is the command that starts the RSCT peer domain services. It offers an invitation to each of the nodes within configuration. If the nodes are already online no action is taken, but if the nodes are offline they are brought up within the inviting domain. This command needs to be run once on a single node within the desired configuration.

2.2.2 Configuring GPFS

After the RSCT domain has been configured and established, the second phase of setting up GPFS is undertaken. The GPFS software is then configured. There are several steps in the second phase of setting up a GPFS "cluster" involved in constitute the procedure used to create a GPFS cluster and file system.

Phase Two Commands:	
Step:	Command:
2.1) Creating the "cluster":	*mmcrcluster*.
2.2) Setting up the configuration:	*mmconfig*.
2.3) Setting up the license:	*mmchconfig license="..."*.
2.4) Starting GPFS:	*mmstartup*.
2.5) Creating the Network Storage Devices:	*mmcrnsd*.
2.6) Making the file system:	*mmcrfs*.

mmcrcluster is the command run to create the GPFS "cluster". With this command, the primary and secondary configuration servers, all of the nodes, and protocols used for communication (in our case, ssh and scp) are defined. The "*-t*" option for *mmcrcluster* is to define the type of cluster being run. The only valid value is "lc" (Linux Cluster) for linux or mixed OS nodes (Linux and AIX).

mmconfig defines the roles the different nodes are to be using in the GPFS "cluster" (either client or manager as the case may be).

mmchconfig license="…" is where the GPFS license is set up.

mmstartup actually starts the GPFS daemons. In this case, it is preferred to use the "*–a*" flag initiating the GPFS daemons on all the nodes defined in the "cluster" rather than running *mmstartup* on every single node individually.

mmcrnsd is the command where the network storage devices (NSDs) are created. Which node will serve the disk, which fail over group it is in and defining the backup server are all part of what this command specifies. It is recommended that *mmcrnsd* be run with a file defining the NSDs and their parameters. This saves repetition of the command and allows for much smaller command lines without losing all of the options that might be otherwise.

mmcrfs creates the file system. Block sizes, mount point, which "cluster" is to be used, what device is to be created, etc. are specified and configured here. It should be noted that a single "cluster" might have more than one file system created by *mmcrfs* on it at any one given time.

Completed in the correct order, and with the proper parameters, these actions will properly set up a RSCT domain, initialize a GPFS "cluster", and create a GPFS file system.

3.0 The Experiences and Problems Encountered

The GPFS test setup was using a portion of the GUPFS testbed. This is a collection of Linux nodes running Redhat Linux. Also on loan from the NERSC Computational Systems Group to the GUPFS project was an IBM F50 Power2 running AIX 5.2. The diagram below demonstrates the network, SAN, and storage configuration. With this equipment, the attempt was undertaken to configure and run GPFS 2.2 across both platforms. Described here are the issues in the order they were encountered.

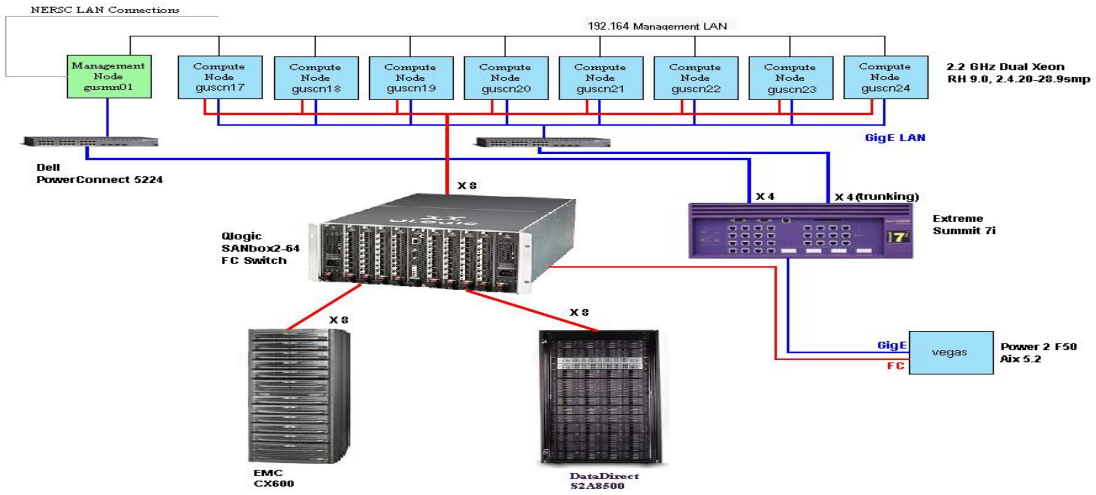

Figure 1.0: Testbed Hardware Configuration

3.1 GPFS configured with "nonexistent" options for *mmcrcluster*

Normally, the first step to configuring GPFS, not mentioned above, is to configure the Reliable Scalable Cluster Technology (RSCT) software services. According to the IBM GPFS documentation and observed behavior, if "*mmcrcluster -t lc <...>*" is run without configuring RSCT ahead of time, it will complain that it must join an already existing RSCT domain, error out, and fail to create a GPFS "cluster". There is an undocumented method around this.

The technique to bypass the creation of an RSCT domain or configure RSCT at all is to use the "*-t lc/lc*" option. High Availability Topology Services (HATS) and High Availability Group Services (HAGS) will still run and seem to be needed by GPFS, but they do not seem to have the drawbacks using an RSCT domain. Namely this means that an arbitrary number of nodes may be taken down or brought up without ever worrying about quorum. Nor are the nodes configured at all by human intervention. This approach has some attractive benefits.

Based on the experience within the GUPFS project, the "*-t lc/lc*" option has worked as far back as GPFS version 1.3. However, when the GUPFS project set out to get GPFS working cross platform with AIX, GPFS with the "*-t lc/lc*" option was quickly discovered not to work under AIX. When trying to add the AIX node to an existing GPFS "cluster",

mmaddcluster and *mmaddnode* reported that the AIX node was unable to contact the RSCT domain. The next step attempted to create a GPFS cluster from scratch, using one of the Linux nodes to issue the commands, and the same error occurred.

At this point, it was decided to try configuring the GPFS "cluster" from the AIX node. It was suspected that perhaps there might be a peculiarity in the software requiring that the AIX node be the computer where the commands are issued. When *mmcrcluster* is run on the AIX node with the "*-t lc/lc*" a new error occurred. *mmcrcluster* complained that the requisite software was not installed. Speaking with IBM's GPFS architects, "*-t lc/lc*" is not a valid option and RSCT must be configured first. Yet, "*-t lc/lc*" works under Linux, and under AIX, it complains of missing software sets, not invalid command flags. In speaking with the developers, it seems that this is a 'feature' that is intended to do exactly what it appears to: avoid having to configure RSCT. It should again be noted that this feature has not been ported to AIX as of the writing of this paper.

3.2 Documentation Errors

Initially, the documentation, printed and man pages, was less than perfect. The print documentation, even when the documentation was sent in an updated form by IBM support, was erroneous (numerous references to GPFS 1.3 in the RSCT documentation, for example, when the RSCT software has been updated since GPFS 2.2 was released). The man pages were also incorrect. The latter has been addressed and corrected in later patches. This caused some missteps with *preprpnode* that IBM support was able to replicate and correct after a day's worth of work.

3.3 AIX RMC Daemon Problems

After being directed to the correct methodology for setting up the RSCT domain and GPFS "cluster", and after getting corrected RSCT documentation and GPFS procedures, there were further software bug issues. *preprpnode* and *mkrpdomain* worked as per instructions and without fault. Using *lsrpdomain,* all the nodes, including the AIX box, listed the inactive domains showing the one created using *mkrpdomain*. Starting the domain was next. This failed.

The failure was on the AIX box. The "rmc" (Resource Management and Control) daemon had died. When running an *lsrpnode* on the Linux nodes, the AIX node was listed as 'Online Pending'. When *lsrpnode* was run on the AIX box, it was unable to contact the rmc daemon. A quick check found that the daemon had died. After a timeout period, generally on the order of five minutes, the *lsrpnode* when run on the Linux box finally lists the AIX box as offline.

Restarting the daemon on the AIX box and then running *lsrpnode* on a Linux box would show that the AIX box would cycle through "Online", "Online Pending", and "Offline" over the course of a few minutes. The transition from "Online" to "Online Pending" was consistent with the observed four seconds. This was carefully repeated, described, and captured using *ctsnap* sent to IBM Support for analysis.

After removing the configuration for RSCT on the AIX box, the daemon would run without crashing when restarted. Additionally, this would also be the case when the daemon was configured with the AIX node was in a RSCT domain. The rmc daemon would only crash once communication with the Linux nodes were started.

At this point, IBM Support stated that the RSCT file sets needed to be updated. This was done. The RSCT software level was 2.3.2.0 after the update. The problem persisted. However, IBM Support was unable to replicate the problem with the rmc daemon dying on the start of communication between the Linux nodes and AIX node. They were able to get SP nodes and Linux boxes to talk to one another using RSCT and GPFS without issue. Later, they would get a F50, like the node on which we were testing, and configured it identically. Still they were unable to reproduce the problem.

At this point, a series of debug rmc daemons were sent from the IBM developers. The hope was that the daemons would find where exactly the failure was taking place. Based on discussions with the RSCT developers and IBM support, one of the daemons seemed to indicate the failure was a known problem that had been addressed in one of the updates. This turned out to be one of several misdiagnoses. The next iteration of diagnoses suggested there was a problem with the AIX 5.2 kernel. After another iteration, there appeared to be two problems with the kernel. After that, it was declared to be a broken pipe. At this point, IBM asked if we could use the kernel debugger to crash the node while running a special rmc daemon. This daemon had an extraordinary amount of sleeps and debug information included in order to capture the specific event by slowing down the barrage of actions that the rmc daemon was performing, thereby capturing the specific event, with one specifically causing the problem at the exact time of the daemon's death.

There was a miscommunication about where to run the kernel debugger (KDB): /dev/console was actually /dev/lft0. KDB does not run there. It requires that it be run through a serial connection. This detail was missed by IBM Support personnel, but was not missed by the KDB developer during a very short conference call. At that point, after finding the serial cable with the proper number of pins, it was possible to crash the AIX node as per instructions. At this point, we were able to correctly collect the requested debugging information by using the serial connection.

During a separate investigation, it was determined that the kernel that was running on the AIX 5.2 node in the GUPFS testbed, was in fact 32-bit, not 64-bit as was originally anticipated. Since IBM already had a significant amount of data, it was decided to boot into the 64-bit kernel. This was a mistake. During booting, libc was destroyed. This caused the node to hang while booting. Quick fixes and restores were insufficient for recovery. A full reinstall was the next step.

3.4 The Reinstall

The reinstall took quite a while. This was primarily due to conflicting priorities. However, in addition, the low performance of the AIX node compounded the reinstall time. This made the actual reinstalling time rather slow. Configurations were adjusted to allow the node to participate in the linux cluster, and the testing was able to continue.

3.5 GPFS with IBM Procedures a Success

After the reinstall of the AIX node (vegas), an *lslpp* listed that the RSCT software had been updated to 2.3.3.0. Following the same procedure as before (*preprpnode*, *mkrpdomain*, *startrpdomain*), the procedure went smoothly. No problems were encountered. RSCT's rmc daemon stayed up and running under AIX. To verify this would remain the case, the RSCT domain was left up and running until the next day. The AIX node was then rebooted. At that time, the AIX node returned to the RSCT domain without incident. The procedure developed for building the GPFS "cluster" was then started. *mmcrcluster* was run with the recommended and outlined procedure where *"-t lc"* rather than "*-t lc/lc"* was used.

At the point where the file system was to be created there were problems. The error message back stated that the disks reported a different size than what had been specified when discovered by previous GPFS configuration commands. This seems to be related to the fact that the underlying storage was the Data Direct 8500 and the way it presents LUNS to the Linux and AIX nodes. Since the point of the exercise was not to troubleshoot the Data Direct and its interaction with GPFS, the configuration was changed to use the EMC storage instead. Creating the NSDs and the file system went smoothly and without incident. All the nodes were requested at that point to mount the file system and did so. A 1.2 terabyte file system was successfully created on Linux nodes and a single AIX node.

4.0 Conclusions

Despite the long time line, the test of GPFS 2.2's cross compatibility was basically a success. The progress made by GPFS in moving towards a platform agnostic file system is very encouraging from NERSC's perspective. The expansion of capability makes for some capabilities that can be used by NERSC and probably other HPC centers. However, there are issues to be sorted out by further investigations at NERSC. Additionally, IBM must address some of the issues raised by this first iteration of testing.

4.1 Remaining Issues to be Resolved

Even though the test was a success, there are still some issues that need to be followed up. These are some very serious in some cases, such as security and architecture questions. On the other hand, some are merely interesting for the explanation as to why they exist, such as the "*-t lc/lc*" option for *mmcrcluster*. Some are questions about how IBM plans to support cross platform software under its current model. All of them ought to be addressed by IBM.

4.1.1 RMC Issues

There are still issues to be explored with RSCT. Based on our findings on AIX and Linux nodes, for proper communication to take place between Linux and AIX nodes, the RSCT software level must be 2.3.3.0. IBM maintains that this is not the case that RSCT on AIX should work under RSCT 2.3.2.0. Answers as to what it is going on are required.

IBM was unable to replicate the problems experienced on the Linux-AIX compatibility test bed. IBM has also stated that the debug information gathered indicated that the rmc daemon was dying specifically on writes to particular file handles under AIX. This only seems to take place when communicating with the Linux cluster. Why this is happening ought to be explored and an explanation provided. The RSCT 2.3.3.0 release may have fixed the problem. However, what has been fixed has not been determined by NERSC.

4.1.2 The Nonexistent *mmcrcluster* Option

The *"-t lc/lc*" option for *mmcrcluster* ought to be explained. What exactly are its limitations? Is this a part of the future path of GPFS? Is RSCT becoming obsolete? If so, when? Why is the "*-t lc/lc*" option undocumented and considered an invalid option by the GPFS architects, yet has existed since GPFS version 1.3? This option works quite well. It appears to give the GPFS cluster quorumless operation even when running in NSD server mode. The positive side effect of quorumless operation is that should the "cluster" require more than fifty percent of these nodes to undergo maintanence, GPFS can remain up and available for users during that time frame.

4.1.3 IBM Support Issues

The final issue is with IBM support. While some of the individuals did heroic efforts, it definitely seemed as though the interaction between the different groups - RSCT on Linux, RSCT on AIX, KDB, and support - were disjointed. There was a definite feel that the interaction between AIX and Linux on this level was very new territory and really belonged in the beta, rather than supported category. The amount of debug information (dumps, snaps, etc.) provided was far more than what is normally provided when debugging a problem of similar magnitude to an IBM SP. Yet despite that, the amount of useful information produced and the time taken to resolve the issue was over a month from the start of the cross platform test. Streamlining the IBM cross platform process and cross training is highly recommended for the AIX and Linux personnel. It might be suggested that there be personnel that either work both sides on the development or there ought to be a special section of support set aside for cross platform issues. Based on IBM's announced operating system support strategy, cross platform issues are going to become more and more common.

4.2 Future Investigations:

Further testing is needed for verification of success. Only mounting the file system and doing simple file creation and rudimentary testing is not enough to claim complete success with cross platform compatibility. The successful sharing of a GPFS file system across Linux and AIX nodes is a milestone for GPFS, and a significant one to be sure, but more needs to be done.

4.2.1 32-bit vs. 64-bit Platform Compatibility

The necessary testing of compatibility of GPFS needs to be done between the 32-bit Linux 2.4 kernel and the 64-bit AIX 5.2 kernel. This has not been done. This is due to the issues encountered when the attempt was made to move AIX node to the 64-bit kernel because of hardware incompatibilities.

4.2.2 Stress testing

Additionally, stress testing needs to be performed. GPFS is up and running, but at what point will it fall over and die while running cross platform has not been ascertained. It is not known whether or not the GPFS 2.2 heterogeneous instance will be as forgiving and stable as on a homogeneous hardware setup. For the stress and performance testing to be done successfully, a faster set of AIX nodes will be needed than the F50 used.

4.2.3 Security Explorations

There are major concerns with taking the file systems out into a multiple disparate operating system scenario. Not the least of which is security. This is something that needs to be thoroughly investigated with GPFS. As it stands right now, there are serious questions about if one machine is compromised; will the others in the "cluster" do as well? If root access is gained on one node, will the others be automatically gained too? How well is the user data protected under GPFS 2.2? All of these concerns need to be investigated and addressed in future analyses.

4.2.4 Data Integrity

Finally, data integrity needs to be verified. Is there data corruption problems related to the differing platforms for GPFS? If so, what are they? Rigorous testing is required and has yet to be done.

4.3 SUMMARY:

There are issues with the documentation. There are definite support issues. There are pending questions about reliability, performance, and various specifics with the configuration options. However, as far as basic functionality is concerned, GPFS 2.2 with the latest patches from IBM, does work. Name space is successfully shared between the AIX and Linux clients. Files can be successfully accessed from nodes running either OS successfully. For a first iteration for the verification of the claimed functionality, the testing and software provided a success.

5.0 Acknowledgements

The author would like to gratefully thank Nicholas Cardo and Tavia Stone of NERSC's Computational Systems Group. Without their help on the AIX side of this test, it would have been more time consuming and a greater learning curve. I would also like to thank the GUPFS project members for their tutoring in writing this paper.

This work was supported by the Director, Office of Science, Office of Advanced Scientific Computing Research, Mathematical, Information, and Computational Sciences Division, of the U.S. Department of Energy under Contract No. DE-AC03-76SF00098. This paper has been submitted as an LBNL Technical Report LBNL/PUB-920.

6.0 Appendix A: Actual Commands Used

6.1 RSCT Commands

preprpnode -TV -f rsct_nodes2 # run on each node # contained in rsct_nodes2

mkrpdomain -f rsct_nodes2 -TV # run on guscn17

startrpdomain aixlinuxtest # run on guscn17

6.2 GPFS Commands

mmcrcluster -t lc -n /usr/local/gpfs/config/040614/clstrtwo_nodes -p guscn17-ge0.gus.nersc.gov -s guscn18-ge0.gus.nersc.gov -r /usr/bin/ssh -R /usr/bin/scp

mmconfig -n /usr/local/gpfs/config/040614/clstrtwo_nodes.conf -C gus2

mmchconfig license="<munched>"

mmstartup -a

mmcrnsd -F /usr/local/gpfs/config/040614/nsd.descfile.nsd2 -v no

mmcrfs /mnt/NSD2gpfs nsd2 -F /usr/local/gpfs/config/040614/nsd.descfile.nsd2 -C gus2 -E yes -v no