System Level Testing Evolution and Resulting Test Methodology Used to Verify CHARM Functionality on POWER Systems

Authors:
Mike Duron
Paul Wojciak

July 27, 2011

Table of Contents

Background

Enterprise computing environments require constant availability. Time lost to unavailable compute resources can cost financial institutions millions of dollars [1,2]. Lost time can come from unscheduled outages. Such outages originate from influences outside the datacenter and from failing hardware components or software problems within the datacenter. Scheduled outages represent the larger contributor to enterprise system unavailability and lost time. Routine maintenance for system upgrades, code updates, or repairs are the biggest contributor to scheduled outages [3,4]. Generally when an unscheduled outage occurs with a computer system there will need to be one or more scheduled outages to repair the system. Further, upgrading system hardware capacity can require scheduled outages as well. Systems such as the IBM Power systems [8] family of servers have robust availability features. These features are designed to address both unplanned and planned outages within customer's enterprise computing environments. One of the key maintenance related feature sets of IBM Power systems servers is known as CEC Hot Add Repair and Maintenance (CHARM)[9].

Given that enterprise servers rarely experience significant lulls in their utilization, adding more capacity to those systems while they are running is very valuable to the customer. However, adding more physical processor, memory or IO hardware while the machine is running, presents a challenge. IBM's POWER enterprise class servers meet this challenge with CHARM. Additional nodes of compute resources can be added to the machine and those resources dynamically configured for use. Thus, hot node add, hot memory upgrade, and hot IO drawer add enable power systems to avoid scheduled outages for capacity upgrades. The next area of server unavailability addressed by CHARM relates to repairing hardware in the rare instance of a hardware failure. When hardware in an enterprise computer system experiences a failure, the system automatically restarts with the failing components logically isolated from the rest of the system. This allows customers to continue operations and defer the maintenance until a more convenient time. Using the capabilities of Hot Repair, IBM service personnel can replace the failing hardware while the server is running. The repaired hardware can then be dynamically returned to use by the customer applications. Hot repair allows for the repair of critical components within the power systems server in a manner most considerate of customer compute availability.

POWER6 CHARM Experience

The roll out of the server maintenance functions that are integral to CHARM occurred starting in POWER6 [10]. The main functions involved are:

- Hot node add
- Hot node memory upgrade
- Hot node repair
- Cold node repair

- Hot IO hub add (expanding on an existing capability from POWER5 of Hot IO drawer add)
- Hot IO hub repair
- Cold IO hub repair

These functions were released on the 9117-MMA and 9119-FHA machines after a lengthy set of tests. The tests for these original releases on POWER6 were modeled after the successful test efforts undertaken by the IBM system z team for that family of servers. The testing of Power systems CHARM functions, in fact some aspects of the design, were modeled after existing capabilities on system z servers. The system z team took three machine generations to mature their concurrent maintenance function set. The biggest challenge for POWER systems was the need to get all the CHARM functions tested and delivered in one machine generation.

Expectations were high regarding the launch of CHARM capability on POWER6. The functions appeared to perform well within the lab and objectives for release were met. One of the earliest POWER6 problems coming back from the field was related to a concurrent firmware driver load of the machine. It was discovered that after upgrading the server from one level of code to the next that the CHARM functions did not work. This exposed a testing gap to verify CHARM worked after a concurrent upgrade to new firmware driver. Another area of weakness that pointed back to test was some CHARM operations would fail if several of them were performed in sequence. The tests done in the lab were rarely executed back to back to back. Even at this early juncture, the field results began to indicate a susceptibility to secondary problems occurring during CHARM. This all served to motivate the team to scrutinize the test matrix. The results of this early review were to expand the matrix of CHARM tests executed in sequence with concurrent code updates included in the steps. In addition, new variations of secondary errors during CHARM were executed. The resulting test effort found a number of important defects that were fixed and released. During this period, user interface and procedure flow defects were frequently treated as required. In prior tests, this class of defect received much less consideration. The CHARM functions were enabled anew on POWER6 along with that level of firmware.

Again, the results from the field showed a different story regarding the use of CHARM functions on customer servers. Several trends were identified.

- Latent (secondary) problems in the hardware, whether from new defective parts or other parts failing in the machine during CHARM operations, were affecting concurrent operation success.
- Complex and lengthy procedures for executing maintenance actions were contributing to field personnel mistakes.
- Inexperience of field personnel performing maintenance operations on live machines was leading to unscheduled outages during CHARM.
- Unique machine configurations contributed to CHARM failures.

Improving CHARM Field Quality

All of these issues and more came to light in a new task force on CHARM field quality. There was no denying that CHARM stressed the different functions within POWER systems hardware and firmware in a way that had not been seen before. The sources of defects were found to reside outside CHARM firmware as often as they were found within the CHARM firmware. As a result of all this, many different improvements were set in motion. The focus of these improvements was to assure POWER7 CHARM launched successfully. The stabilization of POWER6 CHARM field quality was considered important as well. An end to end team was formed to address the challenges.

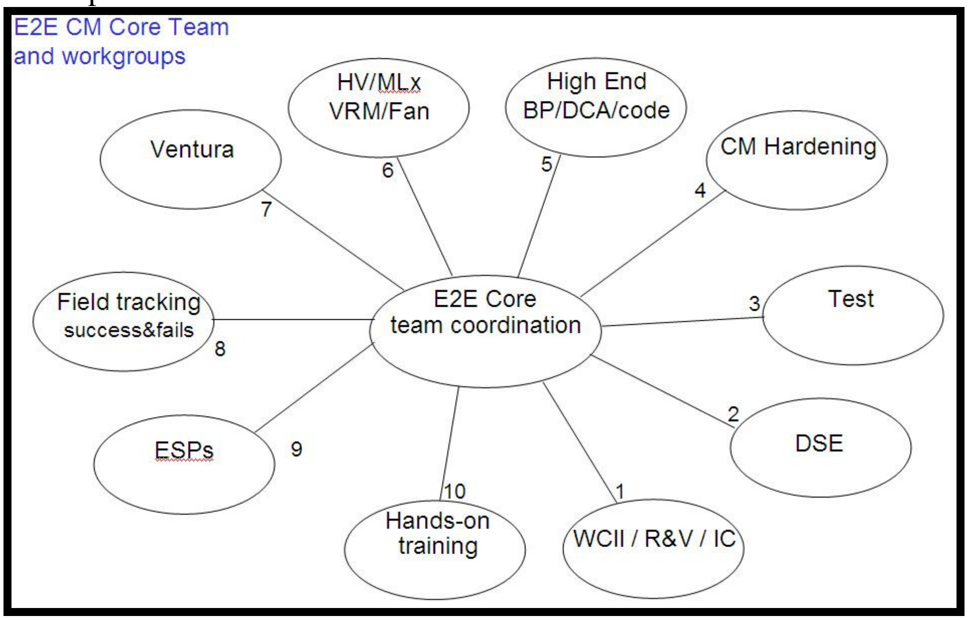

Figure 1 POWER Systems End to End Core Team Structure

Each of the workgroups in the core team had a specific mission.

- There were multiple teams created to address the issue of secondary hardware failures during CHARM. These were the Ventura, HV/MLx VRM/Fan, High End BP/DCA/code and cable teams. Together they encompassed nearly all of the secondary failure cases due to hardware experienced on POWER6 during CHARM. These teams were tasked with recommending improvements in hardware and code to eliminate the occurrence of secondary hardware errors.

- The Concurrent Maintenance (CM) hardening team was to improve the ability to handle secondary errors during CHARM without crashing the machine or preventing the procedure from being restarted and completed successfully.

- The Display Service Effects (DSE) and Worldwide Customized Installation Instructions (WCII), Repair & Verify (R&V), and InfoCenter (IC) teams were tasked with improving and simplifying the complex and error prone service procedures and instructions relied upon by field personnel. The existing procedures had contributed to several outages and missed service windows so fixing usability was critical to CHARM success.

- The hands-on training team was responsible for improving the education and training of the field personnel with respect to CHARM. There were several cases of improper actions taken by IBM servicers that directly led to machine outages. The complex procedures were a large contributor to this. However, the team felt that servicer training received in some geographies across the globe could be improved.

- The Early Support Program (ESP) team was engaged to help line up accounts to test CHARM functions prior to general availability. This was deemed important to get an earlier sense of CHARM quality through our internal IBM accounts.

- The task force had observed that there was incomplete information regarding the success of every CHARM operation in the field on POWER6. Details of failures surfaced quickly but when the operations succeeded there was generally no easy way for the development lab to know. This tended to skew the success rate results but by how much no one knew for sure. Thus, the field tracking success & fails workgroup was formed to offer a proposal on data collection. This team was responsible for setting up a system for CHARM success metrics to automatically be returned to IBM, tallied and reported on. This information would be used to judge the effectiveness of all the actions taken to improve CHARM.

- The test team was formed to devise a plan for testing CHARM that could dramatically improve the field success and drive outages due to CHARM failures in the field to zero.

- Since such a large collection of disciplines were involved in this effort, leaders were appointed and a core team schedule put in place to coordinate the efforts.

Of the several initiatives undertaken for the launch of CHARM on the POWER7 Mid-Range machine, the system test related changes were the most sweeping. Many significant improvements were made. These improvements drew upon everything from research techniques to aid test trial selection to vastly expanding the representative set of hardware configurations used by test. The next section of this paper will detail the test methodology necessary to verify the POWER7 CHARM[12] function.

A New Test Methodology

To deliver better field performance the Technical Leaders from the Test Team devised a new test methodology. Mike Duron and Paul Wojciak are the two Technical Leaders that designed and implemented this novel test methodology. This methodology was designed to inject secondary errors to improve the system's ability to tolerate such errors. In addition, this innovative methodology forced every test case to see at least two unique hardware configurations. This section details the new test methodology from the acronym we use, the test mind set, test case generation and other testing aspects used for this effort.

The Reliability, Availability, and Serviceability (RAS)[11] function where a service person will operate on a portion of the system while the rest of the system continues to run is called Concurrent Maintenance (CM). The hardware concurrent maintenance function came on line subsystem by subsystem. First, IO concurrent maintenance became available then concurrent maintenance of the power components, and lastly, concurrent maintenance of the central electronics complex. The first acronym for this function was CCM or CEC Concurrent Maintenance. The final acronym is now
CHARM – CEC Hot Add Repair and Maintenance.

Test Team Mindset

I will attempt to describe the mind set used by System test then compare it to the mind set used by CHARM test. First let me say that the mind set used by CHARM does not always necessarily fit the goals of traditional test methods. It's been my observation that System test operates in such a manner as to deliver a completed test plan to the product. In doing so, System test is quick to hit a fail, write a defect, give the development team a couple hours to pull their data then move on to the next test case. Near GA System test can deliver a completed test plan with many unrecreatable defects. In addition, the mind set devised for CHARM testing revolves around the fact that we can't simply deliver a completed test plan at GA time but we have to deliver a completely tested, fails exposed, fails debugged to root cause, and fixes tested firmware, software, and hardware package. To accomplish this goal, the CHARM team would freeze upon discovering a fail then allow all the time necessary to accomplish the debug, generate a fix, then test the fix on the system that exposed the fail. The CHARM test mind set is defect driven as opposed to schedule driven. With the dismal POWER6 field results we forced ourselves to go beyond the traditional test role to get to the end point of delivering a product that works. *In our opinion the quickest way to a working function is to root cause everything you find when you find it because you may not be able to recreate the same fail again.*

Test Case Selection With Pairwise Combinations

Test case selection and generation is always a challenge. Since the number of permutations dealing with system configurations, maintainable Field Replaceable Units or FRUs, energy scale modes and performance modes is near infinite, we needed a method to intelligently select a subset of the combinations of these attributes which could then become a manageable, testable set of test cases. We used a pairwise combinatorial testing approach [5,6]. The FoCuS tool from the Haifa Research team was used for intelligent test case generation. FoCuS will take system attributes (number of nodes, node positions, list of FRUs, energy scale and performance modes) and give you two pieces of information.

1. The total number of permutations possible
2. An intelligent subset of permutations that scientifically gives you a high degree of coverage. FoCuS provides at least 90% coverage for pairwise combinations of our CHARM test attributes.

In the case of the POWER7 MR testing FoCuS said we had over 200,000 possible permutations and suggested a list of about 120 test cases with the attributes set as prescribed to get the 90% coverage.

Secondary Error Handling Tests

We also required testing the system's ability to tolerate secondary errors. This was accomplished with error injection at critical times during the CHARM operations. This included but was not limited to injecting a reset reload on the Service Processor, taking cores offline by producing core checkstops, and disconnecting cables in the power subsystem. We called this bad path testing. We also did good path testing where we did not inject any errors and allowed the CHARM operation to go down the good code path. It became very obvious that the bad path testing was flushing out a significant amount of bugs compared to the good path testing so we required the ratio of bad to good path testing be 2 to 1. To further manage this testing we split the entire test plan into several waves with each wave representing a cross section of all possible CHARM operations and with each wave comprised of a bad to good path ratio of 2:1 or higher.

Pass On Two Machines

More invention was needed as we discussed past fails where the exact CHARM operation done in test was attempted in the field and failed. We decided that before we would classify a test case as a pass it had to be run on two unique configurations. Every CHARM test case was run on two different systems. On our mid range system we found that out of 39 total test cases, six test cases exposed bugs only by running them on a 2^{nd} system.

Configuration Dependencies

Another phenomenon that I consider part of using two unique systems was evident when we used different test personnel. With different test personnel we get different partition configurations. We unearthed a code bug because one tester decided to generate a partition with fractional (on a boundary of less than 1G) memory. Another tester was able to expose a bug and make it more repeatable by not running any workload. We have since added the workload parameter into the test generation process.

Physical Part Movements During CHARM

Physical hardware movement is also required. To test this as it would be used in the field on a customer system required us to physically move hardware. We drove to move every single FRU on the POWER7 MR testing. This was particularly challenging when most of our testing was done on borrowed systems. These borrowed systems are joint engineering and manufacturing test (JEMT) systems that are delivered to test and returned after a period of time. What was done for POWER7 MR is that non-JEMT nodes were purchased and each system was reconfigured with say 3 JEMT nodes and 1 non-JEMT node and the CHARM operation was done on the non-JEMT node. While this idea got the job done it's not without many logistical challenges since every JEMT system would have to be returned in its original configuration and moving those heavy nodes is a two person job. This physical hardware movement proved its worth at the very end of the last wave of test for the 721 release where a fatal design error was found that would take the system down when doing a Service Processor repair. You may ask why this fail took so long to find? The answer is that this function of hot repairing the Service Processor was the last function to come on line. In addition, there were many issues found along the way with respect to network connections, syncing each service processor with the other and the fact that the code sync function could not be tested with any patches on the system. Once all these were cleared out of the way and we did not have any patches to deal with is when the base lack of stability became evident and obvious. This bug ended up delaying the availability of several CHARM operations including Service Processor and some Node A & B repairs and node B adds.

Service Processor Repair Instability Root Cause

For those interested this is the technical explanation of this bug. First let me add that we had always felt that the Service Processor repair operation was just not stable. When we finally had fixes for all known issues we still hit the instability that exposed this bug. On with the explanation….the Service Processor repair requires the workload on the node to be evacuated to other processors and memory in the system[13]. The next step involves disconnecting cables that connect each node[14] to each other node (nodes are also referred to as books or drawers) [15] then removing the Service Processor itself. The bug was exposed when reconnecting the cabling after inserting a replacement Service Processor. The root cause was the fact that negative active cfam reset lines run through those cables and inserting a negative active cfam reset signal into a powered off node, which acts like a big capacitor will glitch that signal to a logical 0 to the point that it caused a system wide checkstop. The fix was to add a series filter resistor on that signal between the hot signal and the powered off node. This resistor was added to the Service Processor FRU so that the field population of systems would not require replacement Service Processors for CHARM as long as replacing a failed Service Processor replaced with a new Service Processor that has the filter resistor.

Real Time Test Plan Modifications

Unlike traditional testing we required the test plan change on the fly when necessary. With a nod to defect based testing [7], also referred to as yield based testing, experience from failures seen during the testing was used to guide subsequent test trial content. Continued experience from the field on POWER6 also helped inform test trial changes during the test waves. On the 3rd wave of POWER7 MR CHARM testing we identified bug injections and weak functional areas that we used to modify the remaining tests to target these methods and weak areas. We found that the loss of connection to the management console found bugs along with inserting Service Processor reset reloads. Studying the test results of one complete wave showed the weak functional areas being Service Processor repairs and anything having to do with system clocking.

CHARM testing turned out to be a rigorous test of the system firmware and the base hardware. The development CHARM function owner claimed that CHARM testing does for the firmware stack what HTX, a POWER architecture stress suite, does for hardware test.

Usability Matters … A Lot

There are two facets related to CHARM testing. One aspect is that of finding functional fails. These are tangible fails that prevent a successful CHARM operation. The other facet is that of usability. Traditional system test practices are that functional issues can stop a GA but not usability issues. CHARM testing included the use of service personnel from around the globe primarily to travel to the Austin test lab to get true, intense hands on experience with these CHARM operations and secondarily to help in the test effort. These service reps pointed out many usability issues and were also used in a usability study. The usability study scripted a set of CHARM operations run by inexperienced servicers and evaluated the resulting test data through Failure Mode Effects Analysis. The results of this test demonstrated that product usability will suffer when there is only a functionality focused test effort. This was especially true for a complex function set like CHARM where a great number of manual procedure steps are combined with physical cable and parts movement in live customer machines.

Appling These Test Methodologies Elsewhere

How applicable are these new test methodologies to other aspects of test? If you're testing is all good path testing then there is not much applicability. The first natural extension of these methodologies applies to the base RAS test area. The RAS Test Lead was the first to pick up the requirement to run test cases on multiple systems. Another important aspect is to freeze a system that is in the error state until the root cause has been discovered, then have development generate a fix to be tested on that very system. This applies to all good and bad path testing in all of the test areas. Using pairwise combinatorial test selection to generate an intelligent list of test cases out of an almost infinite number of permutations works well on CHARM testing but would also work well with many other areas of testing such as IO adapter compatibility testing or software application, middleware and operating system combinations. Bad path testing with error injects makes for a more robust system and this aspect is applicable to all test areas. Yield based testing, which requires changes on the fly to the test plan, also has applicability to all test areas. What is currently failing is indicative of weak areas that need to be targeted more. I just can't see doing it any other way.

Once we presented this test plan to the End to End team it was suggested, and let me paraphrase: "Now that we have a good test plan to tackle the CEC CHARM ops what are we going to do for the CHARM ops having to do with the power subsystem components?" The power subsystem is a traditionally troublesome area. It was clear at this point that we had come up with a gold standard way of testing that was desired to be applied to other areas of test. This kind of testing comes at a price. For CHARM you need to physically move hardware requiring the test teams to buy capital systems since only part of the testing can be accomplished on JEMT hardware. You need many systems and many test personnel. Later in this document we will document the field results to date.

From a test perspective we had to ask ourselves: Where is the quality going to come from? Were we just testing quality into the product? This new innovative test plan was widely publicized but nobody knew what the development team was doing to improve quality. We asked the question and I want to document it here. When we started testing the High End system we started with a wave 0 of five good path test cases. All five failed right away. The fails resulted from a refactoring effort that the firmware team was engaged in for several years prior. In this specific example the refactoring was done to prevent any and every code module from keeping a private copy of the hardware representation of the system. This refactoring was very necessary to prevent future bugs where one code module would keep a private copy of the hardware and keep its copy updated but another code module's copy was not updated. If you think about it you can see how this could lead to many problems. The firmware team restructured the code to keep a single copy of the hardware in a database that all code modules could reference. The bug was exposed when CHARM test started hitting time outs where the code was accessing this database. The only reason CHARM test hit this problem but unit and function test did not is that System Test has the richly configured systems. Again the traditional process of starving development test with minimal configs cost us the ability to shift left these set of problems and ironically the code restructuring done to prevent future problems went out and generated more problems.

Predicting Field Success Rate from System Test Experience

One of the challenges associated with CHARM testing has been forecasting the success of the functions in the field on real customer machines. The success of any given product in the field should correlate with the experience during test and align with the final success metric for that test effort. The CHARM functions have defied those indicators on two separate occasions. The lack of success in the field related very directly to test plan gaps. The test cases in place originally were found to lack coverage for many of the bad path tests. There were many more secondary error categories found to occur during

CHARM when those functions were used on customer machines. These secondary errors exposed firmware to error cases for which no handling existed.

The other factor that contributed to a lack of correlation between the test experience and the field experience for CHARM relates to user errors. There were many more user errors experienced in the field when minimally trained field personnel were asked to perform the complex steps and mechanical procedures associated with CHARM. The experience of the testers inhibited usability getting the light that it needed prior to GA. CHARM is very complex to perform. The user is operating on a live system where one mistake can take down the whole compute complex. Many service personnel asked to perform functions Hot through CHARM had never done that before. Their first experience with CHARM was on a live production customer machine. This inevitably led to some customer outages.

Having understood the lack of correlation between test results and field success for CHARM, the approaches outlined in this document were undertaken. The metrics arrived at were called the Initial Trial Success and Final Trial Success.

Initial Trial Success

This denotes whether a test trial succeeds the first time it is attempted. In the context of two machines per test trial, each trial must succeed on first attempt on two different machines to be considered an initial success. This aspect of trial success was felt to portray delivered quality into test and also to have been an identifiable weakness in the POWER6 CHARM testing. The initial success of POWER6 CHARM was very low, under 50%. Could this have been an early indicator that necessary quality was not being achieved? The test team believed initial success was important so began tracking that for POWER7.

Final Trial Success

This denotes whether a test trial can ultimately succeed when all available fixes are applied. Reliance on final trial success is common practice throughout test groups for any product regardless where you are. The final trial successes had been good in the past test efforts for CHARM. The emphasis was thus on additional metrics that would be better indicators of success in the field.

Usability and early internal customer experience

As previously mentioned, a separate usability study was conducted for CHARM on POWER7. This provided a metric of success using FMEA that relied on servicers with little to no prior CHARM experience much like would really occur in the field. This data supplemented the initial and final test success metrics. Prior to full availability of the new function, select internal accounts are involved in testing out the new code. This provides a production customer environment with servicer skills that have much more familiarity with CHARM but not nearly as much as the testers. The internal account servicers provide reports of their experience to the lab. Tests conducted in this manner provide a realistic gage of the CHARM function's readiness for general availability.

Thus, through use of initial trial success, final trial success, usability study results, and internal account CHARM experience a more cohesive picture is developed regarding function readiness. The following table depicts the metric values used for POWER7 MR CHARM testing.

Table 1 Sample CHARM Success Metrics

Metric	Target	Comments
Initial Trial Success	60% or better	This number was based on doubling the prior numbers from the POWER6 products.
Final Trial Success	100%	Deviations to the 100% need to have fixes destined for the first product service pack.
Usability Test	FMEA: RPN > 335	Scale of 1-10 for each. customer impact x frequency of occurrence x likelihood to detect
Internal account experience	All Trials succeed	Expect success for all trails attempted at internal accounts

Preliminary CHARM Field Results for POWER7 MMB/MHB

The real measure of success regarding CHARM test is the field results. Once a sufficient number of CHARM actions have been completed on customer machines, the success metrics will bear more weight. Preliminary results allow a view into the trend we may be seeing. The target for success of CHARM is 90% or better with no machine crashes. Thus, a CHARM action on a machine can succeed, abort, or crash. Both aborts and crashes are counted as failures. Testing for CHARM focused extensively on bad path trials not taking machines down and the operation being retryable after service is aborted. This should be evident in the data. The chart below portrays POWER7 MMB/MHB CHARM field results.

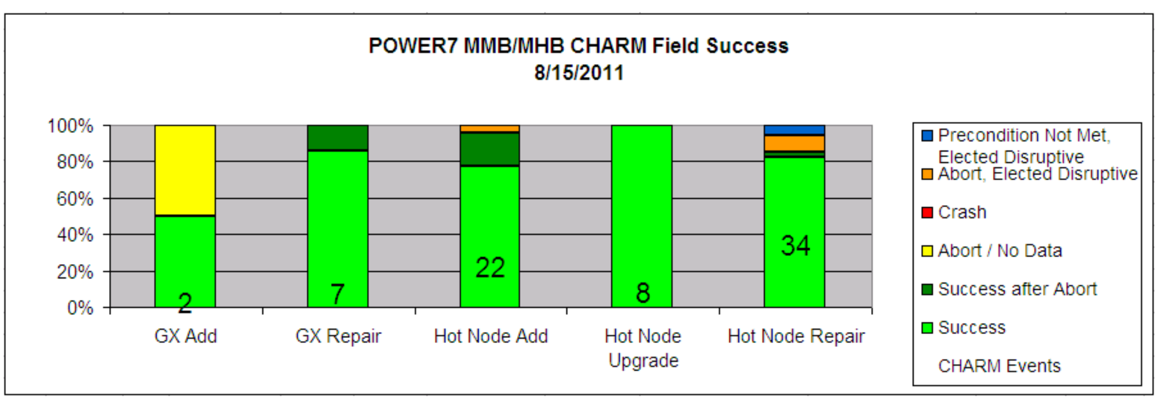

Figure 2 Preliminary POWER7 MMB/MHB CHARM Field Success

There have been quite a few CHARM actions in the field. The intent is to consider measurement of a given type of CHARM function statistically significant when at least 30 samples have been collected. There have been 34 measured CHARM events for Hot Node repair. So this category of CHARM data has reached our criteria for statistical significance.

From looking at the data, there have been no crashes in the field using CHARM and most instances of aborts have been able to be retried successfully. As the paper has described, so many things can affect a CHARM operation that some number of aborts are expected. In some cases, the CHARM operation cannot be done without shutting down one or more partitions on the customer's machine. Many of these situations are not recognized until the servicer is attempting to proceed with the hot add or repair operation. The panels guide the servicer with respect to preparedness. When faced with shutting down production partitions, some customers will choose to schedule a power off at a later time rather than proceed with the hot add or repair operation.

The events leading to aborts in this field data were analyzed. In most cases, the CHARM operation was tried again and completed successfully after the system preparedness was resolved. In a few cases the customer directed the servicer to reschedule and perform the repair or upgrade with power off. There were multiple instances of errors encountered during the CHARM operation that led to a delay and abort. Those errors were mostly related to SMP cable plugging. There was also a case where the servicer did not correctly identify a misplugged FSI pass-thru card before trying and aborting the CHARM operation several times. By and large, the CHARM field performance on POWER7 MMB/MHB has been very good. While more experience is required before drawing conclusions, the early data suggest a more desirable trend.

Conclusions

The ability to support continuous availability of power systems matters greatly for enterprise customers. RAS functions like CEC Hot Add Repair Maintenance (CHARM) are a key aspect in meeting that need. Delivering on those promises for POWER enterprise systems has not been without challenge. The experience from POWER6 launch of CHARM functions drove substantial innovation in the test of those functions on POWER7.

The test approach for CHARM on POWER7 presumed that the characteristics of previous tests were insufficient to achieve a 90+% success rate of those functions in the field. The problem was attacked from several directions resulting in a long list of innovative changes to the test effort.

1. New test mind set – defect driven
2. Agile approach to test cycle using test Waves
3. 2:1 bad path trial to good path trial ratio
4. Pairwise test trial selection using FoCuS

5. Trial pass criteria requiring success on two unique configurations
6. Mandatory hardware movement for subset of add, upgrade, and maintenance trials
7. JEMT/CAP system hardware
8. Change test plan on the fly to accomplish yield based testing

While implementing this test effort on POWER7, there were a number of significant technical challenges faced and overcome. There are two areas that challenged us when it came to implementing the test plan. These areas are technical issues dealing with error injections and the other being logistical. Generally, error injection procedures were problematic. For example, the procedure to inject a reset reload on the service processor is accomplished simply by entering a command but when it came time to recreate a fail seen from such an injection the test team could not always guarantee the inject would be performed at exactly the same time in the CHARM operation. Also, the service processor would not always act on the command entered exactly the same way in subsequent recreate attempts. In some cases we found multiple problems and in other cases we could not recreate issues that were found. On the logistical front the test status reporting was so novel that management had to be briefed on the terminology. Since we required that a test case only pass upon being exposed to two unique system configurations we came up with several terms to explain the progress being made.

- Total Test Cases – for example 10
- Total Attempts – 10 test cases across 2 different systems = 20 total
- 1st Machine Attempt
- 1st Machine Attempt Pass
- 1st Machine Attempt Fail
- 2nd Machine Attempt
- 2nd Machine Attempt Pass
- 2nd Machine Attempt Fail

A test case would be classified a pass only after experiencing a 1st Machine Attempt Pass and a 2nd Machine Attempt Pass. In the case where both of these attempts passed the first time through we labeled it an Initial Success. This reporting is much more involved compared to traditional test status reporting where you have test case totals and total attempts where a single test case is attempted and passed or failed on a single test system.

The early field results from the POWER7 MMB/MHB machines are encouraging. There remains insufficient CHARM events for full measurements, but the events that we do have indicate no crashes and that operations succeed upon retry as was central to the design improvements brought in with POWER7. The complexity of the full set of add, upgrade, and maintenance operations still seems to be affecting the results. This is evidenced by a couple cases where the servicer made mistakes leading to aborts. These situations are being addressed for improvement in future releases. The good news is that the system no longer crashes from these types of servicer errors.

In many ways the POWER7 system test for CHARM can be considered the gold standard for hardware testing. One must understand that not all aspects of this test approach need be applied to accrue the benefits. In fact, not all aspects of the testing easily apply to all types of hardware testing. For instance, use of pairwise test selection lends itself more to test spaces consisting of many different attributes where there is less distinction between members of each attribute group. However, test aspects like the ratio of bad path trials to good path trials or requiring each trial to be performed on two unique machines are very applicable to nearly every type of hardware test. Testers are encouraged to evaluate application of this methodology in their domains of ownership. This report clearly demonstrates there is experience to be gained by other teams facing similar opportunities.

References and Acknowledgements

1. Arnold A, Chief Technology Officer, Vision Solutions. Assessing the Financial Impact of Downtime (26th April 2010). IT-Director.com website. Available at: http://www.it-director.com/business/costs/content.php?cid=12043. Accessed June 27, 2011.
2. Martinez H. How Much Does Downtime Really Cost? (August 6, 2009). InfoManagement Direct website. Available at: http://www.information-management.com/infodirect/2009_133/downtime_cost-10015855-1.html. Accessed June 27, 2011.
3. Weygant PA. Clusters for High Availability. 2nd Edition. Prentice Hall; 2005: Chapter 1.
4. Brodkin J. Amazon: Bad execution during planned upgrade caused outage (April 29, 2011). Network World website. Available at: http://www.networkworld.com/news/2011/042911-amazon-explanation.html. Accessed June 27, 2011.
5. Kuhn R, Kacker R. Advanced Combinatorial Testing System (2009). NIST – Computer Security Resource Center. Available at: http://csrc.nist.gov/groups/SNS/acts/index.html. Accessed March 13, 2011.
6. Czerwonka J. Pairwise Testing – Combinatorial Test Case Generation (May 2011). Pairwise.org. Available at: http://pairwise.org. Accessed April 4, 2011.
7. van Veenendaal E. Standard glossary of terms used in Software Testing. Version 2.1. 'Glossary Working Party', International Software Testing Qualifications Board; 2010. Available at: http://istqb.org/download/attachments/2326555/ISTQB+Glossary+of+Testing+Terms+2+1.pdf. Accessed June 27, 2011.
8. Arroyo, R. X. Harrington, R. J. Hartman, S. P. Nguyen, T. IBM POWER7 systems. IBM Journal of Research and Development. Volume 55 Issue 3; 2011: Page(s): 2:1 - 2:13
9. Eide C, Kitamorn A, Kumar A, Larson D, Lo E, Mehta C, Nayar N, Patel J, Stallman A. IBM Power 770/780 and 795 Servers CEC Hot Add & Repair Maintenance Technical Overview (May 2011). Available at: http://www-03.ibm.com/systems/power/hardware/whitepapers/770_780_cec.html. Accessed June 27, 2011.
10. Eide C, Kitamorn A, Larson D, Lo E, Nayar N, Patel J, Stallman A. IBM Power 595 and 570 Servers CEC Concurrent Maintenance Technical Overview (April

2009). Available at: http://www-03.ibm.com/systems/power/hardware/whitepapers/595_570_cec.html. Accessed June 27, 2011.

11. Henderson D, Mitchell J, Ahrens G. POWER7® System RAS Key Aspects of Power Systems™ Reliability, Availability, and Serviceability (November 2010). Available at http://www-03.ibm.com/systems/power/hardware/whitepapers/ras7.html. Accessed June 27, 2011.
12. Eide C, Kitamorn A, Kumar A, Larson D, Lo E, Mehta C, Nayar N, Patel J, Stallman A. IBM Power 770/780 and 795 Servers CEC Hot Add & Repair Maintenance Technical Overview (May 2011). Available at: http://www-03.ibm.com/systems/power/hardware/whitepapers/770_780_cec.html. Accessed June 27, 2011.
13. Eide C, Kitamorn A, Kumar A, Larson D, Lo E, Mehta C, Nayar N, Patel J, Stallman A. IBM Power 770/780 and 795 Servers CEC Hot Add & Repair Maintenance Technical Overview (May 2011). Available at: http://www-03.ibm.com/systems/power/hardware/whitepapers/770_780_cec.html. Page 45.
14. Arroyo, R. X. Harrington, R. J. Hartman, S. P. Nguyen, T. IBM POWER7 systems. IBM Journal of Research and Development. Volume 55 Issue 3; 2011: Page: 2:1
15. Arroyo, R. X. Harrington, R. J. Hartman, S. P. Nguyen, T. IBM POWER7 systems. IBM Journal of Research and Development. Volume 55 Issue 3; 2011: Page(s): 2:9 Figure 6 and 2:11 Figure 8
16. Arroyo, R. X. Harrington, R. J. Hartman, S. P. Nguyen, T. IBM POWER7 systems. IBM Journal of Research and Development. Volume 55 Issue 3; 2011: Page(s): 2:10 Figure 7 and 2:11 Figure 8
17. Sinharoy, B. Kalla, R. Starke, W. J. Le, H. Q. Cargnoni, R. Van Norstrand, J. A. Ronchetti, B. J. Stuecheli, J. Leenstra, J. Guthrie, G. L. Nguyen, D. Q. Blaner, B. Marino, C. F. Retter, E. Williams, P. IBM POWER7 multicore server processor. IBM Journal of Research and Development. Volume 55 Issue 3; 2011: Page(s): 1:4
18. McCreary, H. Y. Broyles, M. A. Floyd, M. S. Geissler, A. J. Hartman, S. P. Rawson, F. L. Rosedahl, T. J. Rubio, J. C. Ware, M. S. EnergyScale for IBM POWER6 microprocessor-based systems. IBM Journal of Research and Development. Volume 51 Issue 6; 2011: Page(s): 776
19. Chen, A. D. Cruickshank, J. Costantini, C. Haug, V. Maciel, C. D. Schmidt, J. T. IBM POWER 795: Technical Overview and Introduction. Available at: http://www.redbooks.ibm.com/abstracts/redp4640.html?Open&pdfbookmark, Page(s): 60
20. Floyd, M. Allen-Ware, M. Rajamani, K. Brock, B. Lefurgy, C. Drake, A. J. Pesantez, L. Gloekler, T. Tierno, J. A. Bose, P. Buyuktosunoglu, A. Introducing the Adaptive Energy Management Features of the POWER7 Chip. Micro Hot Chips 22. Volume 31 Issue 2; 2011; Page(s) 69

We would like to thank the following individuals for their contributions to material referenced in this paper.

1. Chris Aguinaldo
2. Bob Landa
3. Erlander Lo
4. Jayesh Patel

Appendix: Additional Details

I will now discuss an important example of using the FoCuS research tool to generate an intelligent meaningful subset of test cases along with changing the test plan on the fly based on the root cause of fails being discovered. Testing on the high end system model 795, a 32 socket system[16] during wave 3 exposed several stop ship defects all resulting in system wide checkstops. The root cause of these fails was a firmware bug where the frequency of a core was set on the high side while the voltage for that core was set on the low side. This resulted in data corruption which symptomatically resulted in L3 hangs, PHYP terminates, and illegal address errors. On POWER7 each processor chip has up to eight cores, each of which can independently run a low frequency or a high frequency as long as the core voltage is set to match[17]. High frequency requires high voltage and low frequency can be run with a lower voltage. Once this root cause was discovered, the Processor team, Firmware team, and the Test team studied the test results to understand why CHARM testing could not find this earlier. There was another effort to understand why the frequency voltage testing done as part of the Partition-level Power Management (PPM) testing did not find this fail earlier. PPM was a new function for POWER7 as compared to the system-level power management on POWER6[18]. The necessary ingredients needed to hit this fail where we caused the firmware do go down the path of setting high frequency with low voltage were

a. Running a CHARM operation so that you go down the defective code path.
b. Turbo Core mode[19] which is a performance mode requiring high frequency at high voltage.
c. DPS or dynamic power saving mode[20] specifically DPO_Not_max_perf mode which is a mode that wants to lower frequency.
d. Some unknown ingredient.

The discussion culminated with the realization that depending on the workload being run we could potentially hide this problem. Most CHARM testing was done with medium to heavy workload. The Processor team pointed out that to aggravate the problem would require us to run little to no workload. The missing ingredient was a deliberate attempt to run very light to zero workload. The actions taken at this point were to change the test plan to target light, medium and heavy workloads. Both the CHARM regression testing and the PPM regression testing were changed on the fly to test these attributes before GA. In addition we added a workload attribute to the FoCuS attributes so that we have an intelligent set of tests that now cover workload. This immediate feedback from a root cause discovery and study of the testing done with a great desire to shift left discovery of these types of problem is what we call Yield Based Testing (Defect based testing). This methodology is always applied in CHARM testing and applied when necessary in System test.

Terminology

CHARM: CEC Hot Add & Repair Maintenance
E2E: End to End team formed via task force to tackle getting a quality concurrent maintenance function to deliver to our customers
IO: Input/Output
VRM: Voltage Regulator Module
BP: Bulk Power
DCA: Distributed Converter Assembly – High End system power component
CM: Concurrent Maintenance
CEC : Central Electronics Complex
CCM: CEC Concurrent Maintenance
DSE: Display Service Effect – a panel that displays the results of checking that the node to be repaired is completely evacuated of any workload
WCII: Worldwide Customizable Installation Instructions used primarily for node add operations
IC: InfoCenter
ESP: Early Support Program
RAS: Reliability, Availability and Serviceability
FRU: Field Replaceable Unit
MR: MidRange system
HE: High End system
JEMT: Joint Engineering Manufacturing and Test systems
CAP: A test system that is purchased with CAPital expense dollars where depreciation has to be paid – as opposed to JEMT test systems that are borrowed from manufacturing
HTX: Hardware Test Executive – a system test exerciser suite
SMP: Symmetrical Multi-Processor
HV: High Volume
MLx: IBM Power 570 – 9117 - MMA
MMA: IBM Power 570 – 9117 – MMA – POWER6 Mid Range system
FHA: IBM Power 595 – 9119 - FHA - POWER6 High End system
MMB: IBM Power 770 - 9117-MMB – POWER7 Mid Range system
MHB: IBM Power 780 – 9117-MHB – POWER7 High End system
FSI: FRU Support Interface
FSP: Flexible Service Processor
FMEA: Failure Mode Effect Analysis
L3: Level 3 Cache
PHYP: P Series Hypervisor
PPM: Partition-level Power Management
DPS: Dynamic Power Save – a power savings mode of operation
GA: General Availability – of a new system or system firmware release

IBM Corporation
Systems and Technology Group
Route 100
Somers, New York 10589
Produced in the United States of America
July 2011