# IBM POWER6 Processor-based Systems: Designed for Availability

IBM System p Platform

Reliability, Availability and Serviceability (RAS)

Daniel Henderson, Brian Warner and Jim Mitchell

June 11, 2007

# 1 Overview: Designing for Availability

The IBM POWER6™ Availability strategy is deeply rooted in an extensive history spanning multiple decades of mainframe development. By leveraging IBM's extensive background in Predictive Failure Analysis™ and dynamic system adaptation, the Availability team has helped to create a unique processor that unleashes significant value to the client.

A cornerstone of IBM's POWER6 Availability strategy is the ability to perform effective Predictive Failure Analysis, or PFA. IBM's extensive First Failure Data Capture (FFDC) design, long an integral component of the POWER™ server line, enables an unprecedented level of self-awareness in POWER6 processor-based systems. The FFDC method allows the system to proactively analyze situations that indicate an impending failure, in many cases removing the questionable component from use before it can cause a threat to system stability.

IBM POWER6 processor-based systems have a number of new features which enable them to dynamically adjust when issues arise that threaten availability. Most notably, POWER6 processor-based systems introduce the POWER6 Processor Instruction Retry suite of tools, which includes Processor Instruction Retry, Alternate Processor Recovery, Partition Availability Prioritization, and Single Processor Checkstop. Taken together, in many failure scenarios these features allow a POWER6 processor-based system to recover transparently without an impact on a partition using the core.

# 2 Detecting and Deallocating Failing Components

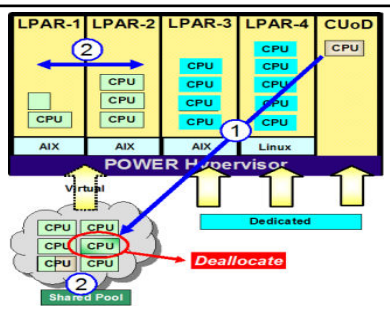
Run-time correctable/recoverable errors are monitored to determine if there is a pattern of errors or a "trend towards uncorrectability." Should these components reach a predefined error limit, the Service Processor will initiate an action to deconfigure the "faulty" hardware, helping avoid a potential system outage and enhancing system availability. Error limits are preset by IBM engineers based upon historic patterns of component behavior in a variety of operating environments. Error thresholds are typically supported by algorithms that include a time-based count of recoverable errors; that is, the Service Processor responds to a condition of too many errors in a defined time span.

## *2.1 Persistent Deallocation*

To enhance system availability, a component that is identified for deallocation or deconfiguration on a POWER6 server will be flagged for persistent deallocation. Component removal can occur either dynamically (while the system is running) or at boot-time (IPL), depending both on the type of fault and when the fault is detected.

In addition, run-time unrecoverable hardware faults can be deconfigured from the system after the first occurrence. The system can be rebooted immediately after failure and resume operation on the remaining good hardware. This prevents the same "faulty" hardware from affecting the system operation again while the repair action is deferred to a more convenient, less critical time for the user operation.

## 2.2 Dynamic Processor Deallocation




Dynamic Processor Deallocation from the shared pool uses a similar strategy (but may affect up to ten partitions). First, look for available Capacity on Demand (CoD) processor(s). If not available, determine if there is one CPU's worth of performance available in the pool. If so, rebalance the pool to allocate the unused resource.

If the shared pool doesn't have enough available resource, query the partitions and attempt to reduce entitled capacities to obtain the needed performance.

First introduced with the IBM RS/6000® S80 server, Dynamic Processor Deallocation enables automatic deconfiguration of an error-prone processor core before it causes an unrecoverable system error (unscheduled server outage). Dynamic Processor Deallocation relies upon the Service Processor's ability to use FFDC-generated recoverable error information and to notify the POWER Hypervisor when the processor core reaches its predefined error limit. The POWER Hypervisor, in conjunction with the operating system (OS), will then "drain" the run-queue for the offending core, redistribute the work to the remaining CPUs, deallocate the offending CPU, and continue normal operation, although potentially at a lower level of system performance.[1]

IBM's logical partitioning (LPAR) strategy enables a number of additional system availability improvements. IBM's logical partitioning allows any processor to be shared with any partition on the system. This enables the following sequential scenarios for deallocation:

1. **An unlicensed Capacity on Demand (CoD) CPU** will by default be automatically used for Dynamic Processor Sparing. In this case, the unlicensed CPU can be configured to automatically "back-fill" for the deallocated bad processor core. In most cases, this operation is transparent to the system administrator and to end users, and the server continues normal operation with full functionality and performance.
2. **If no CoD processor core is available,** the POWER Hypervisor attempts to locate an un-allocated core somewhere in the system, in which case it will be substituted in and the failing CPU deallocated.
3. **If no spare processor core is available,** the POWER Hypervisor attempts to locate a sum total of 1.00 spare processor capacity from a shared processor pool (when enabled) – a capability enabled by logical partitioning, because resources can be shared anywhere – and redistributes the workload.
4. **If the requisite spare capacity is not available,** the POWER Hypervisor will determine how many processor core capacity units each partition will need to relinquish to create at least 1.00 processor capacity units from a shared pool. The POWER Hypervisor uses an algorithm based on partition utilization and the defined partition minimum and maximums for CPU equivalents to calculate capacity units to be requested from each partition. The POWER Hypervisor will then notify the operating system (via an error entry) that processor units and/or

[1] While AIX® V4.3.3 precluded the ability for a SMP server to revert to a uniprocessor (i.e., a 2-core to a 1-core configuration), this limitation was lifted with the release of AIX Version 5.2.

virtual processors need to be varied off-line. Once a full core equivalent is attained, the CPU deallocation event occurs.

5. **The deallocation event will not be successful** if the POWER Hypervisor and OS cannot create a full core equivalent. This will result in an error message and the requirement for a system administrator to take corrective action. In all cases, a log entry will be made for each partition that could use the physical core in question.

## *2.3 POWER6 Processor Instruction Retry*

POWER6 processor-based systems introduce the Processor Instruction Retry suite of mainframe-class recovery features, which significantly reduce scenarios which would result in a checkstop. This powerful new suite of tools includes:

- **Processor Instruction Retry** – Automatically retry a failed instruction and continue execution.
- **Alternate Processor Recovery** – Interrupt a repeatedly-failing instruction and move to a new processor and continue execution.
- **Partition Availability Priority** – In the event spare capacity is not found, a properly-configured system will utilize proactively defined priorities that specify (for example) that capacity should be first obtained from your test environment instead of your financial accounting system.
- **Processor Contained Checkstop** – When all else fails, in almost all cases (excepting the POWER Hypervisor) a termination will be contained to the single partition using the faulty core.

### 2.3.1 Processor Instruction Retry

By combining enhanced error identification information with an integrated Recovery Unit, a POWER6 microprocessor can use **Processor Instruction Retry** to transparently recovery from a wider variety of fault conditions than could be handled in earlier POWER processor cores. Examples of situations handled by Processor Instruction Retry include “non-predicted” fault conditions undiscovered through predictive failure techniques, and transient faults such as bit flips from cosmic rays. This mechanism allows the processor to recover completely and transparently from what would otherwise have caused an application, partition, or system outage.

### 2.3.2 Alternate Processor Recovery

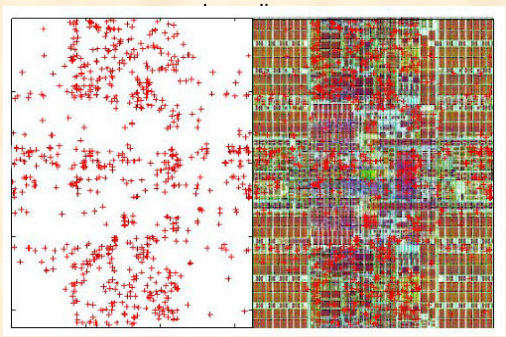
For solid core faults, retrying the operation on the same processor core will not be effective because the error will reappear each time. For many such cases, the **Alternate Processor Recovery** feature will deallocate and deconfigure a failing processor, moving the instruction stream over to and restarting it on a spare processor. These operations can be accomplished by the POWER Hypervisor and POWER6 hardware without application interruption, allowing processing to continue unimpeded.

### *2.3.2.1 Locating Spare Capacity*

Using an algorithm similar to that employed by Dynamic Processor Deallocation the POWER Hypervisor manages the process of acquiring a spare processor core.

1. **If there is a fault, is there a spare, unlicensed CoD core?** If a spare exists, it will be used without charge.
2. **If there is no spare, is there an unused processor?** If there are processors not assigned to any partition, the one with the closest affinity to the faulty processor is used as a spare.
3. **If there is no unused processor, can we make room?** If no spare is available, then the POWER Hypervisor will attempt to "make room" for the instruction thread by over-committing hardware resources or, if necessary, terminating lower priority partitions. Clients manage this process by using a new metric, set using the HMC: the Partition Availability Priority.

POWER6™ Test System mounted in



As part of the process to verify the coverage model, the latch flip distribution (left) was overlaid on a POWER6™ die photo (right).

**A Test to Verify Automatic Error Recovery**

To validate the effectiveness of the RAS techniques in the POWER6 processor, an IBM engineering team created a test scenario to "inject" random errors in the cores.

Using a proton beam generator, engineers irradiated a POWER6 chip with a proton beam, injecting over $10^{12}$ high-energy protons into the chip, at more than six orders of magnitude higher flux than would normally be seen by a system in a typical application. The team employed a methodical procedure to correlate an error coverage model with measured system response under test.

The test team concluded that the POWER6 microprocessor demonstrated dramatic improvements in soft-error recovery over previously published results. They reasoned that their success was likely due to key design decisions:

1. Error detection and recovery on data flow logic provides the ability to recover most errors. ECC, parity, and residue checking are used to protect data paths.
2. Control checking provides fault detection and stops execution prior to modification of critical data. IBM employs both direct and indirect checking on control logic and state machines.
3. Extensive clock gating prohibits faults injected in non-essential logic blocks from propagating to architected state.
4. Special Uncorrectable Error handling avoids errors on speculative paths.

Results showed that the POWER6 microprocessor has industry leading robustness with respect to soft errors in the open systems space.

### *2.3.2.2 Partition Availability Priority*

Starting with POWER6 technology, administrators are able to rank order partitions by numeric priority. Partitions receive an integer rating with the lowest priority partition rated at "0" and the highest priority partition valued at "255." The default value is set at "127" for standard partitions and "192" for Virtual I/O Server (VIOS) partitions. Partition Availability Priorities are set for both dedicated and shared partitions.

To initiate Alternate Processor Recovery when a spare processor is not available, the

POWER Hypervisor uses the Partition Availability Priority to determine the best way to maintain unimpeded operation of high priority partitions. This mechanism enables a customer to ensure that when a split-second decision must be made over how best to use remaining resources, a critical database or virtual I/O server is defined to be far more important than a test partition.

### 2.3.3 Processor-Contained Checkstop

If a specific processor detected fault cannot be recovered by Processor Instruction Retry and Alternate Processor Recovery is not an option, then the POWER Hypervisor will terminate (checkstop) the partition that was using the processor core when the fault was identified. In general, this limits the outage to a single partition. However, if the failed CPU was executing a POWER Hypervisor instruction and the saved state is determined to be invalid, the server will be rebooted.

## *2.4 Memory Protection*

Memory and cache arrays are comprised of data "bit lines" that feed into a memory word. A memory word is addressed by the system as a single element. Depending on the size and addressability of the memory element, each data bit line may include thousands of individual bits (memory cells). For example:

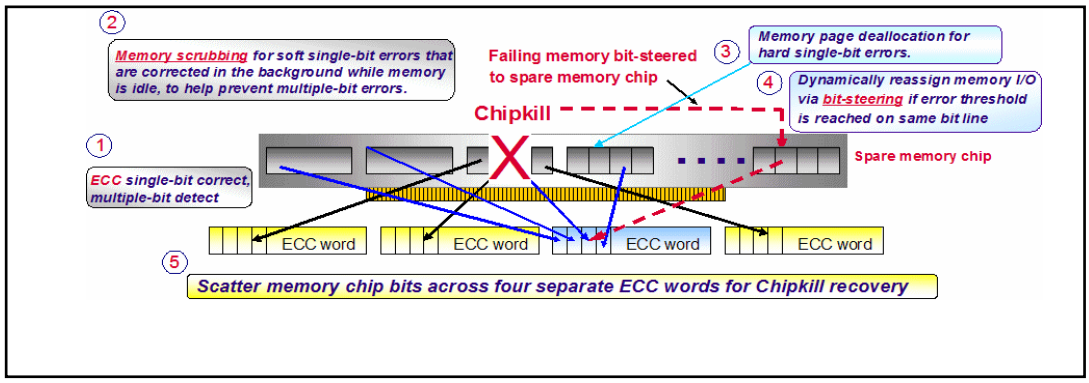
- A single memory module on a memory DIMM (Dual Inline Memory Module) may have a capacity of 1 Gbits, and supply eight "bit lines" of data for an ECC word. In this case, each bit line in the ECC word holds 128 Mbits behind it, corresponding to more than 128 million memory cell addresses.
- A 32 KB L1 cache with a 16-byte memory word, on the other hand, would only have 2 Kbits behind each memory bit line.

A memory protection architecture that provides good error resilience for a relatively small L1 cache may be very inadequate for protecting the much larger system main store. Therefore, a variety of different protection schemes are used in IBM POWER6 processor-based systems to avoid uncorrectable errors in memory. Memory protection plans must take into account many factors including size, desired performance, and memory array manufacturing characteristics.

POWER6™ processor-based systems have a number of protection schemes designed to prevent, protect, or limit the effect of errors in main memory. These capabilities include:

- **Hardware Scrubbing**, which is IBM's method of dealing with transient, or soft, errors. POWER6 processor-based systems periodically addresses all memory locations and rewrites any with an ECC error are re-written with the correct data.
- **ECC**, or Error Correcting Code, allows a system to detect up to two errors in a memory word and correct one. However, if more than one bit is corrupted without additional correction techniques, a system will fail. For example, a burst error (sequential bad bits) or DRAM failure will not be tolerated by a system that exclusively uses ECC. For this reason, IBM developed Chipkill™ memory.

- **Chipkill** is IBM's proprietary enhancement of ECC that enables a system to sustain the failure of an entire DRAM. Chipkill works by spreading the bit lines from a DRAM over multiple ECC words, so that a catastrophic DRAM failure would affect at most one bit in each word. Barring a future additional single bit error, the system can continue indefinitely in this state with no performance degradation until the failed DIMM can be replaced. To avoid this scenario, POWER6 processor-based systems utilize a technology called Redundant Bit Steering.
- **Redundant Bit Steering** is IBM's means of avoiding situations where multiple single-bit errors align to create a multi-bit error. In the event that a POWER6 processor-based system detects an abnormal number of errors on a bit line, it can dynamically "steer" the data stored at this bit line into one of a number of spare lines. This both reduces exposure to multi-bit errors as well as helps to defer maintenance until all redundant bits have been used.




### 2.4.1 Memory Page Deallocation

While coincident single cell errors in separate memory chips is a statistic rarity, POWER6 processor-based systems can contain these errors using a memory page deallocation scheme for partitions running AIX and for memory pages owned by the POWER Hypervisor. If a memory address experiences an uncorrectable or repeated correctable single cell error, the Service Processor sends the memory page address[2] to the POWER Hypervisor to be marked for deallocation.

1. Pages used by the POWER Hypervisor are deallocated as soon as the page is released.
2. In other cases, the POWER Hypervisor notifies the owning partition that the page should be deallocated. Where possible, the operating system moves any data currently contained in that memory area to another memory area and removes the page(s) associated with this error from its memory map, no longer addressing these pages. The operating system performs memory page deallocation without any user intervention and is transparent to end users and applications.

[2] Support for 4K and 16K pages only.

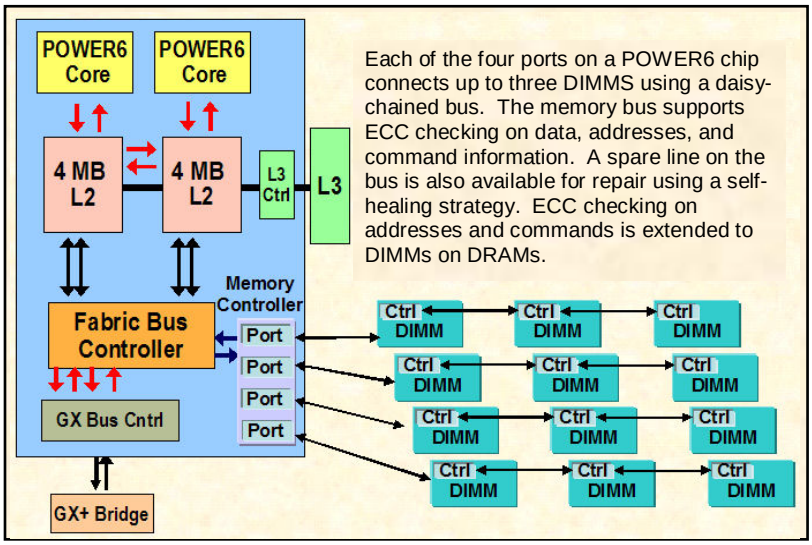
3. The POWER Hypervisor maintains a list of pages marked for deallocation during the current platform IPL. During a partition IPL, the partition receives a list of all the bad pages in its address space. In addition, if memory is dynamically added to a partition (through a Dynamic LPAR operation), the POWER Hypervisor warns the operating system if memory pages are included which need to be deallocated.

While memory page deallocation will not provide additional availability for the unlikely alignment of two simultaneous single memory cell errors, it will address the subset of errors that can occur when a solid single cell failure precedes a more catastrophic bit line failure or even the rare alignment with a future single memory cell error. Memory page deallocation handles single cell failures, but, because of the sheer size of data in a data bit line, it may be inadequate for dealing with more catastrophic failures. Redundant bit steering will continue to be the preferred method for dealing with these types of problems.

Finally, should an uncorrectable error occur, the system can deallocate the memory group associated with the error on all subsequent system reboots until the memory is repaired. This is intended to guard against future uncorrectable errors while waiting for parts replacement.

### 2.4.2 Memory Control Hierarchy

A memory controller on a POWER6 processor-based system is designed with four ports. Each port connects up to three DIMMS using a daisy-chained bus. The memory bus supports ECC checking on data, addresses, and command information. A spare line on the bus is also available for repair using a self-healing strategy. In addition, ECC checking on addresses and commands is extended to DIMMs on DRAMs.




Because it uses a "daisy-chained" memory access topology, a System p™ 570 server can deconfigure a DIMM that encounters a DRAM fault, without deconfiguring the bus controller -- even if the bus controller is contained on the DIMM.

### 2.4.3 Memory Deconfiguration

Defective memory discovered at boot time will be automatically switched off, unless it is already the minimum amount required to boot. If a memory fault is detected by the Service Processor at boot time, the affected memory will be marked as bad and will not be used on subsequent reboots. (Memory Persistent Deallocation).

If the Service Processor identifies faulty memory in a server that includes CoD memory, the POWER Hypervisor attempts to replace the faulty memory with available CoD memory.

As faulty resources are automatically "demoted" to the system's unlicensed resource pool, working resources are included in the active memory space. Since these activities reduce the amount of CoD memory available for future use, repair of the faulty memory should be scheduled as soon as is convenient.

Upon reboot, if not enough memory is available to meet minimum partition requirements the POWER Hypervisor will reduce the capacity of one or more partitions. The HMC receives notification of the failed component, triggering a service call.

# 3 Special Uncorrectable Error Handling

While it's a rare occurrence, despite all precautions built into the server an uncorrectable data error can occur in memory or a cache. POWER6 processor-based systems attempt to limit, to the least possible disruption, the impact of an uncorrectable error using a well-defined strategy that first considers the data source. Sometimes, an uncorrectable error is transient in nature and occurs in data that can be recovered from another repository. For example:

- Data in the processor's instruction cache is never modified within the cache itself. Therefore, an uncorrectable error discovered in the cache is treated like an ordinary cache miss, and correct data is loaded from the L2 cache.
- The POWER6 processor-based system's L3 cache can hold an unmodified copy of data in a portion of main memory. In this case, an uncorrectable error in the L3 cache would simply trigger a "reload" of a cache line from main memory.

In cases where the data cannot be recovered from another source, a technique called Special Uncorrectable Error (SUE) handling is used to determine whether the corruption is truly a threat to the system. If, as may sometimes be the case, the data is never actually used but is simply over-written, then the error condition can safely be voided and the system will continue to operate normally.

When an uncorrectable error is detected, the system modifies the associated ECC word, thereby signaling to the rest of the system that the "standard" ECC is no longer valid. The Service Processor is then notified, and takes appropriate actions. When running AIX V5.2 or greater or Linux®[3] and a process attempts to use the data, the OS is informed of the error and terminates only the specific user program. It is only in the case where the corrupt data is used by the POWER Hypervisor that the entire system must be rebooted, thereby preserving overall system integrity.

Depending upon system configuration and source of the data, errors encountered during I/O operations many not result in a machine check. Instead, the incorrect data may be handled by the processor host bridge (PHB) chip. When the PHB chip detects a problem it rejects the data, preventing data being written to the I/O device. The PHB then enters a "freeze" mode halting normal operations. Depending on the model and type of I/O being used, the freeze may include the entire PHB chip, or simply a single bridge. This results in the loss of all I/O operations that use the frozen hardware until a power-on reset of the PHB. The

[3] SLES 10 SP1 or later, and in RHEL 4.5 or later (including RHEL 5.1).

impact to partition(s) depends on how the I/O is configured for redundancy. In a server configured for “fail-over” availability, redundant adapters spanning multiple PHB chips could enable the system to recover transparently, without partition loss.

# 4 Cache Protection Mechanisms

POWER6 processor-based systems are designed with cache protection mechanisms, including cache line delete in both L2 and L3 arrays, Processor Instruction Retry and Alternate Processor Recovery protection on L1-I and L1-D, and redundant “Repair” bits in L1-I, L1-D, and L2 caches, as well as L2 and L3 directories.

## *4.1 L1 Instruction and Data Array Protection*

The POWER6™ processor’s instruction and data caches are protected against transient errors using the POWER6 Processor Instruction Retry feature and against solid failures by Alternate Processor Recovery, both mentioned earlier. In addition, faults in the SLB array are recoverable by the POWER Hypervisor.

## *4.2 L2 Array Protection*

On a POWER6™ processor-based system, the L2 cache is protected by ECC, which provides single-bit error correction and double-bit error detection. Single-bit errors are corrected before forwarding to the processor, and subsequently written back to L2.

Like the other data caches and main memory, uncorrectable errors are handled during run-time by the Special Uncorrectable Error handling mechanism. Correctable cache errors are logged and if the error reaches a threshold, a Dynamic Processor Deallocation event is initiated.

Starting with POWER6 processor-based systems, the L2 cache is further protected by incorporating a dynamic cache line delete algorithm similar to the feature used in the L3 cache. Up to six L2 cache lines may be automatically deleted. It is not likely that deletion of a few cache lines will adversely affect server performance. When six cache lines have been repaired, the L2 is marked for persistent deconfiguration on subsequent system reboots until it can be replaced.

## *4.3 L3 Array Protection*

In addition to protection through ECC and Special Uncorrectable Error handling, the L3 cache also incorporates technology to handle memory cell errors via a special cache line delete algorithm. During system run-time, a correctable error is reported as a recoverable error to the Service Processor. If an individual cache line reaches its predictive error threshold, it will be dynamically deleted.

The state of L3 cache line delete will be maintained in a “deallocation record,” and will persist through future reboots. This ensures that cache lines “varied offline” by the server will remain offline should the server be rebooted, and don’t need to be rediscovered each time.

These “error prone” lines cannot then cause system operational problems.

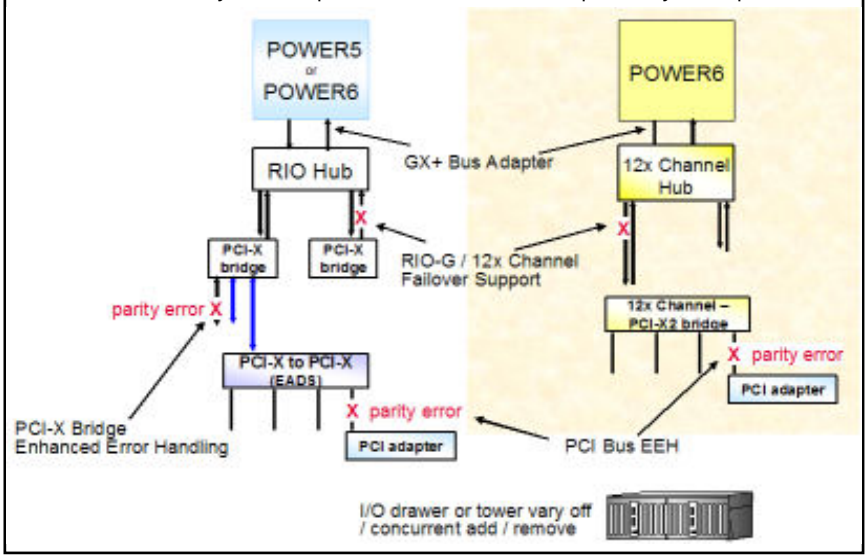
A POWER6 processor-based system can dynamically delete up to 14 L3 cache lines. Again, it is not likely that deletion of a few cache lines will adversely affect server performance. If this total is reached, the L3 is marked for persistent deconfiguration on subsequent system reboots until repair.

While hardware scrubbing has been a feature in POWER main memory for many years, POWER6 processor-based systems introduce a hardware-assisted L3 cache memory scrubbing feature. All L3 cache memory is periodically addressed, and any address with an ECC error is rewritten with the faulty data corrected. In this way, soft errors are automatically removed from L3 cache memory, decreasing the chances of encountering multi-bit memory errors.

# 5 PCI Error Recovery

IBM estimates that PCI adapters can account for a significant portion – up to 25% – of the hardware based error opportunity on a large server. While servers that rely on “boot time” diagnostics can identify failing components to be replaced by “hot-swap” and reconfiguration, run time errors pose a more significant problem.

IBM has long built servers with redundant physical I/O paths using CRC checking and failover support to protect RIO server connections from the CEC to the I/O drawers or towers. IBM extended this data protection by introducing first-in-the-industry Extended Error Handling to allow recovery from PCI-bus error conditions. POWER5™ and POWER6 processor-based systems add recovery features to handle potential errors in the Processor Host Bridge (PCI bridge), and GX+ bus adapter. These features provide improved diagnosis, isolation, and management of errors in the server I/O path and new opportunities for concurrent maintenance — to allow faster recovery from I/O path errors, often without impact to system operation.




PCI adapters are generally complex designs involving extensive “on-board” instruction processing, often on embedded microcontrollers. Because these are usually cost sensitive designs, they tend to use industry standard grade components, avoiding the more expensive (and higher quality) parts used in other parts of the server. As a result, they may be more likely to encounter internal microcode errors, and/or many of the hardware errors described for the rest of the server.

The traditional means of handling these problems is through adapter internal error reporting and recovery techniques in combination with operating system device driver management and diagnostics. In some cases, an error in the adapter may cause transmission of bad data on the PCI bus itself, resulting in a hardware detected parity error and causing a global machine check interrupt, eventually requiring a system reboot to

continue.

In 2001, IBM introduced a methodology that uses a combination of system firmware and “Extended Error Handling” (EEH) device drivers that allows recovery from intermittent PCI bus errors. This approach works by recovering and resetting the adapter, thereby initiating system recovery for a permanent PCI bus error. Rather than failing immediately, the faulty device is “frozen” and restarted, preventing a machine check. POWER6 technology extends this capability to PCIe bus errors, and includes expanded Linux support for EEH as well.

# 6 IBM POWER6 processor-based systems – Designed for Availability

IBM POWER6 processor-based systems continue to expand the POWER legacy for availability, leveraging IBM’s mainframe legacy to deliver best of breed features such as processor instruction retry and alternate processor recovery coupled with new innovations such as partition availability priorities, while expanding upon existing features such as EEH and memory resilience technologies. The Availability teams at IBM are continuing to execute upon the POWER6 RAS roadmap, delivering evolutions in value and availability never before seen in a POWER product.


IBM Corporation
Systems and Technology Group
Route 100
Somers, New York 10589

Produced in the United States of America
June 2007


PSW03020-USEN-01