MPT Fusion Linux OS Driver Release Notes

Driver Version: mptlinux-4.00.21.00-1

11/02/2007

Compatibility:
- 53C1020, 53C1030, 53C1035 (Scsi Mode)
- FC909, FC919, FC929
- FC919X, FC929X, FC929XL, FC939X, FC949X, FC949ES
- SAS 1064, 1068
- SAS 1064E, 1068E, 1078E

Special Notes For This Build:
- No special notes available at this time.

Current Errata

- These drivers do not include disk dump support that is sometimes added by certain distro's. This appears to be replaced by kdump. Here is help for using kdump: http://lse.sourceforge.net/kdump/
- Kdump doesn't work for flashless controllers.
- Fusion.mptctl will not be provided on RHEL5 DUDs, nor KMP RPMS. This is the service script that autoloads the mptctl driver. You will find this only on SLES10 DUDS, and generic RPMS(found in rpm-1 folder).

Driver Release Package Contents

• *Source tarball*

      Source tarball is named as: mptlinux-<version>-src.tar.gz

      where:
      <version>   =   version tag for this rpm

• *RPM Binaries:*

RPM images are named as: mptlinux-<version>-<release>-<os>.<arch>.rpm

      where:
      <version>   =   version tag for this rpm
      <release>   =   release tag for this rpm
      <os>   =   {rhel5, sles10}
      <arch>   =   ia64 - Itanium Processor
      x86_64 - Opteron Processor, w/ x86_64 install
      i686 - i686 or later processor (Red Hat)
      i586 - x86 installations (SuSE)
      ppc – Power PC (64 bit)

• *Driver Update Disks*

dud images are named : mptlinux-<version>-<release>-<os>.<arch>.dd.gz

      where:
      <version>   =   version tag for this rpm
      <release>   =   release tag for this rpm
      <os>   =   {rhel5, sles10}
      <arch>   =   ia64 - Itanium Processor
      x86_64 - Opteron Processor, w/ x86_64 install
      i686 - i686 or later processor (Red Hat)
      i586 - x86 installations (SuSE)
      ppc – Power PC (64 bit)

• *DKMS*

DKMS tarball is named : mptlinux-<version>-<release>.dkms.tar.gz

       where:
       <version>   =   version tag for this rpm
       <release>   =   release tag for this rpm

This is an rpm that provide prebuilt binaries for major releases, and will compile drivers on the fly for the other kernels. Here is help on dkms: http://linux.dell.com/dkms/dkms.html

• *SRPMS*

SRPM images are named mptlinux-<version>-<release>.src.rpm

       where:
       <version>   =   version tag for this rpm
       <release>   =   release tag for this rpm

SuSE:

i686
       SLES 10
           Gold (2.6.16.21-0.8)   (bigsmp, debug, default, kdump, smp, xen)
       SLES 10 SP1
           Gold (2.6.16.46-0.12)   (bigsmp, debug, default, kdump, smp, xen)
x86_64

       SLES10
           Gold (2.6.16.21-0.8)   (debug, default, kdump, smp, xen)
       SLES10 SP1
           Gold (2.6.16.46-0.12)   (debug, default, kdump, smp, xen)
ia64
       SLES10
           Gold (2.6.16.21-0.8)   (debug, default)
       SLES10 SP1
           Gold (2.6.16.46-0.12)   (debug, default)
ppc

       SLES10
           Gold (2.6.16.21-0.8)   (debug, default, iseries64, ppc64)
       SLES10 SP1
           Gold (2.6.16.46-0.12)   (debug, default, iseries64, ppc64)

Red Hat:

i686
RHEL5
Gold (2.6.18-8)   (el5, el5PAE, el5xen)
Update 1 (2.6.18-53)   (el5, el5PAE, el5xen, el5debug)

x86_64

RHEL5
Gold (2.6.18-8)   (el5, el5xen)
Update 1 (2.6.18-53)   (el5, el5xen, el5debug)

ia64
RHEL5
Gold (2.6.18-8)   (el5, el5xen)
Update 1 (2.6.18-53)   (el5, el5xen, el5debug)

ppc
RHEL5
Gold (2.6.18-8)   (el5, el5kdump)
Update 1 (2.6.18-53)   (el5, el5kdump, el5debug)

Major Changes For Version 04.00.21.00-1
Release Date: 11/02/2007

General Changes

- Adding rpm/dud binaries for RHEL5.1 GA.
- Remove "LSI Logic" strings from the driver update disks disk-info and dud.config files.   This was a customer request.

Defect fixes

- **Issue:** Controller fails to resume coming out of hibernation.
- **Reported by:** Customer
- **To Reproduce:** From the console, do 'echo standby > /sys/power/state', then press power button to bring system out of standby.
- **Description of Change:** The solution is to issue a diagnostic reset from the mptbase_resume entry point. This is only required for 1064E, 1068E. The contents of the errata is as follows: The Doorbell register contents do not get cleared when the controller is put into the D3 PCIe power management state. Upon returning to the D0 PM state, the contents of the register will be stale data. This is problematic when the controller firmware is in IOC State Ready when it is moved to the D3 power state. After being brought back to the D0 power state, the doorbell will incorrectly signal the host driver that firmware is ready to begin processing MPT commands.

Defect fixes

- **Issue:** Panic loading mptctl and mptsas(or mptfc) at the same time.
- **Reported by:** Customer
- **To Reproduce:** From the console, do 'insmod mptsas.ko &; insmod mptctl.ko'
- **Description of Change:** Adding sanity check in mpt_device_driver_register, to insure ioc object has been created prior to calling the probe entry point. Also delete the calling of mptctl_remove from the mpt_probe entry point.

Major Changes For Version 04.00.20.00-1
Release Date: 10/10/2007

General Changes

- Adding rpm/dud binaries for RHEL5.1 pre-release.
- Upstream linux kernel merges.
- Dual Port Feature testing fixes: DUAL_PORT events respond to device added/deleted, bug fixs in mpt_raid_phys_disk_pg1, add new function mpt_raid_phys_disk_get_num_paths, and the rewrite of mptsas_find_phyinfo_by_phys_disk_num
- Continuation Reply support in mptctl: add msg_context member in MPT_MGMT struct keeping track of current request, from mpt_reply added sanity check on new mst_context member ignoring processing reply if that reply was not expected, from mpt_reply check CONTINUATION_REPLY bit by not reposting the the message frame back to FIFO, from mptctl_release_diag_buffer change expected ioc-status from DIAGNOSTIC_RELEASE to SUCCESS, and add macro SET_MGMT_MSG_CONTEXT.
- The device queue_depth was being set to 1 from mptscsih_slave_configure entry point when the target_id is larger than sh->max_id. For SAS, the transport layer assigns the target_id which is a 32 bit integer value, so the checks on sh->max_id need to be ignored. This sanity check was intended for SPI bus which is limited to 16 devices per bus. The fix in the release is to not allow SAS bus type to enter this sanity check.
- Some debug logging modifications for SPI protocol in mptspi.c and mptscsih.c, for instance functions mptspi_print_write_nego and mptspi_print_read_nego were modified to use starget_printk.
- Fix for SAS, handle discrepancy between phy_page_0 and sas_iounit_page_0 when using Y-cable(4 phy wide port split into two sets of 2 phy). The handle in phy_page_0 was the same for both sets of 2 phy which is incorrect.   The sas_iounit_page_0 was correct in providing unique sas addresses for both ports of the Y-cable. So to fix this, the function mptsas_probe_hba_phys will pick up the correct handle sas_iounit_page_0 in setting the identify.handle.

Defect fixes

- **Issue**: Driver hang when loading CSMI agents in RHEL5
- **Reported by**: customer
- **To Reproduce**: csmitest using SMP_PASSTHRU
- **Description of Change**: Fix stack overflow due to over 2K byte buffer using function stack memory. Change involves allocation of memory on the fly which uses free slab memory.

- **Issue**: CSMI agents are not returning any data for SMP_PASSTHRU
- **Reported by**: customer
- **To Reproduce**: csmitest using SMP_PASSTHRU
- **Description of Change**: The macros CLEAR_MGMT_STATUS added in previous release was clearing out a bit in the status flags that indicating there was data available from firmware. This macro was removed. In addition, the MsgContext was being cleared prior to sending the request, and when the reply arrived in mptctl_reply, it failed the sanity check. Fix this by not clearing out the MsgContext.

Major Changes For Version 04.00.19.00-1
Release Date: 09/28/2007

Defect fixes

- **Issue**: Driver panic
- **Reported by**: customer
- **To Reproduce**: Load mptspi driver when the controller has IR firmware.
- **Description of Change**: When there is IR firmware, we create a virtual channel for the hidden raid component. In mptspi_slave_alloc, we return -ENXIO when there are no devices in the mptscsih_is_phys_disk_sanity check. Due to the early exit, sdev->hostdata has not been set. The mptspi_slave_destroy is still called, even though the slave_alloc return failure. From mptscsih_slave_destroy, we expect sdev->hostdata to be a valid pointer. There is a sanity check in the built-in driver sources for non-zero sdev->hostdata, however it was removed some time ago in the internal LSI driver sources.

Major Changes For Version 04.00.18.00-1
Release Date: 09/25/2007

General Changes

- Enhancements in the error handling of SAS hotplug device_add events. From from mptsas_test_unit_ready, added support for return codes : MPT_SCANDV_DID_RESET, MPT_SCANDV_SELECTION_TIMEOUT, and MPT_SCANDV_SOME_ERROR. The new support would recognize these code and take appropriate action. Also added were some printks that are useful in determining the failures that could occur during this function.
- Fix compilation error in newer kernels over scsilun_to_int. The fix converted using a different kernel API called int_to_scsilun, which

performed the same functionality in obtaining the lun number. This fix is from the function mptscsih_search_running_cmds.

Major Changes For Version 04.00.17.00-1
Release Date: 09/24/2007

General Changes

- Merged code from mainstream drivers
- Add dual port support for Raid Volumes having two paths to a single device. Support added include: (1) new asyn firmware events { MPI_EVENT_IR2_RC_DUAL_PORT_REMOVED and MPI_EVENT_IR2_RC_DUAL_PORT_ADDED}, (2) mptscsih_is_phys_disk and mptscsih_raid_id_to_num functions added support for recognizing dual ports, (3) new API function mpt_raid_phys_disk_pg1 to read this config page, and (4) csmisas_get_raid_config ioctls to return a Drive Type of CSMI_SAS_DRIVE_TYPE_DUAL_PORT_SAS flag.

Major Changes For Version 04.00.16.00-1
Release Date: 09/04/2007

General Changes

- For peripheral devices requiring media, such as tapes and cdroms, the devices were not being reported to upper layers when no medium was present.   This issue also occurs for devices that report "LU self configuring" when devices are hot added after driver load time. The fix was to add the appropriate sense code handling for both cases in function mptsas_test_unit_ready.

Major Changes For Version 04.00.15.00-1
Release Date: 08/31/2007

General Changes

- Created three versions of mpt_add_sge {32 bit, 64 bit, and 1078 64bit errata fix}, and setting the appropriate handler in the ioc->add_sge function. This should help with performance, by reducing extra if/then/else checks based on the three different paths.
- Broadcast AEN fix's: incorrect LUN was passed for task_management request, clear_task_set was being sent, when it should have been abort_task_set, added ioc->broadcast_aen_busy flag so we disregard any

aen events that occur when a current aen event is being processed, and requeue aen event if there is already task mangement request in progress.
- Beautify task management debug prints.
- mptscsih_change_queue_depth – set the queue depth to 1 when sdev->tagged_supported flag is cleared.   This insures that devices queues are set to 1 when the device reports they don't support tags.
- Fix issue where extra device nodes were created mapped to the same device. This problem occurs when the HBA has target_mode enabled.
- Fix compile errors when compiling in driver in Linux kernels greater than 2.6.20 (tip is 2.6.23).
- Put controller in READY_STATE when driver unloads.

Major Changes For Version 04.00.14.00-1
Release Date: 08/17/2007

General Changes

- 1078 DMA HW Issue fix –

1)   Ensure that the message frames are NOT located in the trouble region. There is no remapping available for message frames, they must be allocated outside the problem region.
2)   Ensure that Sense buffers are NOT in the trouble region. There is no remapping available
3)   Walk through the SGE entries and if any are inside the trouble region then they need to be remapped to use POM2 addressing mode as discussed below.

Trouble Region:
–   0x0000_0009_0000_0000 through 0x0000_0009_1000_FFFF
–   0x0000_0009_1800_0000 through 0x0000_0009_1800_0FFF
–   0x0000_0009_1C00_0000 through 0x0000_0009_1C00_0FFF

If an SGE entry points to something in the troubled region then the following needs to be changed for the SGE entry.

1) Set the Local Address bit in the SGE Flags field. MPI_SGE_FLAGS_LOCAL_ADDRESS
2) Ensure are using 64-bit SGEs
3) Set MSb (Bit 63) of the 64-bit address, this will indicate buffer location is Host Memory.

- Broadcast AEN support for multi-initiator devices.

- DIAG BUFFER – IOCTL support
- Sparse fixes
- Phase 12 mpi headers v1.5.17
- Command line option mpt_sdev_queue_depth, and tunable /sys/class/mptsas/parameters/mpt_sdev_queue_depth, for globally adjusting the device queue depth
- cb_idx fixes found during code review
- domain validation fixes – set default period to 0xF, instead of 0xA

Major Changes For Version 04.00.13.00-1
Release Date: 07/20/2007

General Changes

- Merged CSMI IOCTL code; all three driver streams using same common code.

Major Changes For Version 04.00.12.00-1
Release Date: 07/18/2007

General Changes

- ATTO 4LD support: rebranded 53C1030, required a different PCI vendor id, and spi_port_page_2 is in a different format than specified in the mpi spec, so we add code to reformat the bits into our format.
- Merged in the sysfs patch changes requested by the linux community: driver_version removed, removed usage of buf_out, and hba_properites obtains driver_version from /sys/module/mptbase/version.
- Removed setting the NB bits for task management request that is sent via high-priority-request-q.
- cb_idx changes: mpt_get_cb_idx function changed to scans the ids in descending order, and MPT_MAX_PROTOCAL_DRIVERS changed back to 0xF.
- Brocade support for loading mptctl at boot time in the RPMs. This involved updating the init script fusion.mptctl.

Major Changes For Version 04.00.11.00-1
Release Date: 06/26/2007

General Changes

- SLES10 Duds were not working. The problem was the directory path to the drivers on the duds was incorrect.
- The CSMI IOCTL routine csmisas_get_raid_config has been changed to report the correct drive count when either DRIVES or DEVICE_ID is requested.
- The CSMI IOCTL routine csmisas_get_raid_config will return a place holder for missing drives (actually implemented in the 4.00.08.00 version)
- "LSI Logic Corporation" copyright changed to "LSI Corporation" through out the sources. And all other references to Logic have been removed.
- Mpt_linux_developer@lsi.com mailto changed to DL_MPTFusionLinux@lsi.com
- SendEventNotification changed to send request via doorbell using handshaking protocol. Interrupts are enabled after this call completes.
- The watchdog FAULT handler was changed to run in user context.
- The MPT_IOC_POST_RESET callbacks after soft and hard reset are invoked even though the controller didn't recover. If the controller didn't recover, the ioc->active flag is set to zero. When ioc->active is zero, the callback handlers will prevent sending commands that require interrupts.
- The callback index called cb_idx found through out the code was changed from integer to u8. mpt_stm_index and mpt_lan_index were deleted, and cb_idx is obtained thru new function mpt_get_cb_idx. All the associated callback index's and functions that return those values are all changed to u8.
- Cleanup mptctl prints and debug messages so they are using the MYIOC_s_XXX_FMT format.
- Fix bug for MPT_COMMAND ioctls that sends mpi_functions equal to "task management" request. For SAS the callback index was not set properly, thus having its completion routine routed to the wrong function. From mpt_put_msg_frame_hi_pri, the input parameter was changed from ioc->TaskCtx was to mptctl_id.
- Big Endian fixes in mptsas_send_discovery_event and mptsas_send_raid_event. Under power pc, when a volume was activated via lsiutil (for an inactive volume), the device driver was not reporting the volume to upper layers, hence the volume wasn't seen.
- mpt_fc_log_info – rewrite, so it includes lsi/mpi_log_fc.h
- Cleanup error handling entry points print and debug info.
- Delete lsi/mpi_inb.h – unused.
- mpt_diag_reset – code has been added to break early from the "wait-for-ready" loop after diag_reset. The early break occurs when when the card is found to be in FAULT state after one iteration.
- Mpt_do_ioc_recovery – force reset of alternate controller when the reason is MPT_HOSTEVENT_IOC_RECOVER.

Major Changes For Version 04.00.10.00-1

Release Date: 06/13/2007

General Changes

- Driver was striping the first three characters of product name. For instance, "BRE040" is being displayed as "040".   The fix implemented in MptDisplayIocCapabilities, is to delete the code which strips of the first three characters of "ioc->prod_name".

Major Changes For Version 04.00.09.00-1
Release Date: 06/11/2007

General Changes

- Added support in mptfc.c for pci vendor id (0x1657) – FC949E, and displaying the string "BRE040" when the driver loads
- Add support for displaying revision strings for 1064E B3, 1068E B3, 1078 C2, and 1030T A3 when the driver loads.
- Removed the "ioc->active" check in mptctl_ioctl.

Major Changes For Version 04.00.08.00-1
Release Date: 06/1/2007

General Changes

- MPI v1.5.16 Headers
- SLES10 SP1 support
- Watchdog timer for monitoring controller ioc state for FAULTs
- Several Task management Updates, issue Message Unit Reset as first level recovery from timeouts, task_abort always returns SUCCESS when it determines the outstanding command was completed.
- 1030 errata fix - on a slower PCI bus (75 MHz or less) on a non-paced (below U320 speed) 1030 data transfer the DMA engine can be prematurely shut off, resulting in a data underrun condition (even though the target device transferred all of the data). This happens on a Read or Verify only (not on a Write).
- Increase SAS task mangement timeouts to 30 seconds
- Increase mpt_config timeout to 15 seconds
- Fix panic when creating IME volumes using cfggen
- Add command line option "mpt_disable_hotplug_remove: to disable hotplug remove in mptsas. Also sysfs attribute to toggle this flag while driver is running.
- Add hi-priority request queue support for SAS task management requests.

- Firmware download boot updates

Major Changes For Version 04.00.07.00-1
Release Date: 05/10/2007

General Changes
- Fix driver compilation errors with SLES10 SP1.   This was fixed by enabling the MPT_WIDE_PORT_API defines at the top of mptsas.c
- CSMI_GET_CNTRL_CONFIG – set the uHighPart variable when BaseMemoryAddress is found to be above the 32bit limit.
- The functions mptscsih_do_cmd and mpt_config, which send internal commands, were cleaned up. The handling of internal command timeouts using watchdog timers were deleted, replaced with the wait_for_completion_timeout API and entry/exit mutex's.
- The flag hd->resetPending was renamed to hd->quiesce_io. This flag used to queisce io during host reset. The use of this flag was expanded to queisce io for any task management request as well as SAS discovery. For hard reset, IO will be queisced for the alternate controller when the controller is for dual pci function, such as the 1030.   Use the spin lock diagLock, instead of FreeQlock when setting or clearing the quiesce_io flag.
- From mptbase.h, deleted several unused structure variables in MPT_ADAPTER, CONFIGPARAMS, and MPT_SCSI_HOST.   Deleted MPT_LOCAL_REPLY structures, and moved sas_mapping and sas_device_info structures over to mptsas.h.
- SAS hotplug enhancements – for hot add case, re-queue the event on 5 second intervals for 1 minute until the device is found in the topology. Added unique exit codes for mptsas_add_end_device to help with debugging hot add events.
- Add support to report a deleted RAID volume to upper layers when a firmware event is received saying the volume has been disabled.
- Added support in mptfc for displaying port link speed.
- Fix setting SMART event data in mptscsih_copy_sense_data, so the addition of channel, target, and lun are using "|" instead of "||".

Major Changes For Version 04.00.06.00-3
Release Date: 04/26/2007

General Changes
- Adding KMP support, and new subfolder kmp-${release_version}

Major Changes For Version 04.00.06.00-2
Release Date: 04/25/2007

General Changes

- RHEL5, adding support for installing RPM after a DUD install.

Major Changes For Version 04.00.06.00-1
Release Date: 04/10/2007

General Changes

- Documentation headers were added through out mptsas.c
- The function mpt_HardResetHandler input parameter is changed from NO_SLEEP to CAN_SLEEP, called from mpt_timer_expired, mptscsih_taskmgmt_complete, and mptscsih_timer_expired.
- Added MPI_EVENT_SAS_DEV_STAT_RC_ASYN_NOTIFICATION from mpt_display_event_info.
- In mptctl.c, changed the severity of the prink following mpt_verify_adapter, from KERN_ERR to KERN_DEBUG.
- Hotplug handling improvements which decrease the impact of keyboard freeze when many drives are added and removed.   The calling of mptsas_discovery_work from mptsas_hotplug_work was removed, and devices can be added from the discovery thread. The handling of raid events for hidden raid components was broken out into MPTSAS_XXX_PHYSDISK and MPTSAS_XXX_PHYSDISK_REPROBE events. Some improvements in mptsas_send_raid_event reduce the number times raid events are scheduled.   The ssleep() in mptsas_test_unit_ready was moved so its only called when the device is known to be busy, also support for sending test_unit_ready to hidden raid component was added.   Fix bug in mptsas_probe_expander_phys where the "struct device" parent pointer was set to NULL, resulting in device not being reported to transport layer after hot delete/add events.   Hot delete/add functions were broken out into new functions mptsas_add_end_device and mptsas_del_end_device.

Major Changes For Version 04.00.05.00-1
Release Date: 03/26/2007

General Changes

- mptfc_SetFcPortPage1_defaults into mptfc_rescan_devices which executes in a work queue thread to allow board hard reset processing to complete. mptfc_SetFcPortPage1_defaults was being called by POST_RESET processing with diagPending set. Hangs were observed due to mpt_config() not completing due to the config message timing out. The

hard reset handler couldn't execute due to diagPending being set. This change allows another hard reset which will clean up the outstanding config message.
- Fix oops in mptsas_ioc_reset. This would occur if a timeout occurs at driver load time, prior to scsi host registration. For example, at driver load time, config pages are read to obtain defaults, and a timeout could occur if the request didn't complete. Added are sanity checks in mptsas_ioc_reset checking for NULL pointers.
- Added verbose messages to track when devices and expanders are hot added or deleted.
- Added support to handle hidden raid component removal when the MPI_PD_STATE_FAILED is received.
- Changes surrounding hidden raid components, such as preventing task management request and sync cache when the device is known to be a hidden raid component
- Add support to prevent sync cache for deleted devices when they
- known to be physically removed; fix in mptsas and mptfc.

Defect fixes

- **Issue**: Source RPMs failed while building the binaries.
- **Reported by**: test
- **To Reproduce**: install the SRPM, then go to /usr/src/package, and try `rpmbuild –bb mptlinux.spec`
- **Description of Change**: Added fusion.mptctl in Sources section of spec file.

- **Issue:** Pulling a drive from a RAID1 volume will result in volume being remounted as "read-only".
- **Reported by:** Customer
- **To Reproduce:** Install SLES10 onto RAID1 volume with a driver supplied from LSI (not the built-in SLES10 driver), then send some IO to the volume, pull one of the drives, then monitor errors sent to /var/log/messages
- **Description of Change:** The reason the volume was being re-mounted as 'read-only" was when the drive was pulled, IO to volume was accidentally being terminated with DID_NO_CONNECT, which means there is no device. The end result is the filesystem thinking the device was removed, thus takes less risk by remounting it with write access disabled.   This problem manifested due to hidden raid components being exposed. The per device object retains the target id as the physdisk id in the object and it so happens that the physdisk id is matching the volume id.   The fix was to add sanity check insuring that when the hidden raid component is torn down, it will only terminate IO to that device, and not the volume

Major Changes For Version 04.00.04.00-1
Release Date: 03/06/2007

General Changes
- MPI header update version 1.5.15
- New "on the fly" enabling of debug logging.   This can be modified in SysFS, located in /sys/class/scsi_host/host#/debug_level.   In addition, there is a new command line option in mptbase.c that will enable debug logging at driver load time. The command line option is called mpt_debug_level. Example:

insmod mptbase.ko mpt_debug_level 0x408

This enables SCSI reply info and events logging.

- Change a majority of the debug logging to KERN_DEBUG priority.
- Display board rev level and chip type when driver loads in kernel log.
- Several new SySFS host attributes added in /sys/class/scsi_host/host#. There is new script in source that will dump these new attributes. It's found in subfolder scripts, the file called hba_properties. These new attributes will display various versions of bios/driver/firmware/mpi/nvdata, as well as the board assembly/name/tracer, in addition to io_delay and device_delay.
- CSMI IOCTL get_controller_config – return base_io and base_memory address from the values setup when driver loads. It was found with 1078 that IO address range was not being returned. This was due to the expected IO range not being found in the first resource bar, which is what the code was expecting.
- CSMI IOCTL get_controller_config – removed function csmisas_get_manufacture_pg0, then instead obtaining the board serial number saved off when driver initialized.
- Added support for upto 256 SGEL's for mptfc.
- Added support for displaying SAS_EXPANDER_STATUS_CHANGE event in the kernel logging.
- Mptlan – fix for setting RequestNB to zero.
- Fix memory leak in mptsas and mptspi, which would occur at driver load time if pci_register_driver failed.
- Increased mpt_cmd_retry_count from 60 to 144 seconds.
- In mptscsih_io_done, removed the SAM_STAT_BUSY if/then check that would return DID_BUS_BUSY, instead return the sam status as is.   The scsi mid layer no longer does infinite retries when busy is returned.
- Cleanup scsiio reply logging; renaming the function from mpiscshi_iocstatus_info_scsiio to mptscsih_info_scsiio.
- Picked up some upstream janitorial driver cleanup, most changes in documentation.

Major Changes For Version 04.00.03.00-3
Release Date: 03/06/2007

General Changes
- Re-spin dkms rpm with a modification in spec file of adding xen support.

Major Changes For Version 04.00.03.00-2
Release Date: 03/1/2007

General Changes
- Removed application source code from package.
- RHEL5 support

Major Changes For Version 04.00.03.00-1
Release Date: 02/28/2007

General Changes
- Changed memory allocation method in csmi ioctl stp pasthru to use the get_free_page API, due to some request failing.

Defect fixes

- **Issue:** Panic when csmi ioctl GET_PHY_INFO is called.
- **Reported by:** Customer
- **To Reproduce:** This is seen on RHEL5 32bit, using csmitool, option #7.
- **Description of Change:** Root problem was stack overflow. Change code to allocate memory for input/output buffer from kernel slab rather than the stack. The requested buffer was 2kb, which is too large for the stack.

Major Changes For Version 04.00.02.00-1
Release Date: 02/16/2007

General Changes
- Sorting volumes in ascending order before they are reported to upper scsi layers.

Major Changes For Version 04.00.01.00-1
Release Date: 01/08/2007

General Changes
- Merge in upstream kernel changes from 2.06.20 kernel; including kernel-doc changes, and INIT_WORK(work task) API change.
- Memory allocation in csmisas.c was changed from kmalloc to get_free_pages where more that 128 KB of memory may be requested. This allows the IOCTL to proceed normally rather than failing due to a memory allocation failure. This was noticed in the ssp_passthru routine that was being used by an application to flash disk drive FW that was about 750 KB.
- Sorting volumes so they are reported in ascending order. The driver is reporting volumes based in ioc page 2, and firmware can fill in this config page with volumes listed in descending order.

Defect fixes

- **Issue:** Panic when ioctl passed thru mptctl.ko times out.
- **Reported by:** Test.
- **To Reproduce:** This was replicated in creating an IME volume.
- **Description of Change:** Change mpt_HardResetHandler in passing CAN_SLEEP, instead of NO_SLEEP.

- **Issue:** Target reset occurs during creation of an IME volume.
- **Reported by:** Test.
- **To Reproduce:** This was replicated in creating an IME volume.
- **Description of Change:** The MPI_EVENT_RAID_RC_VOLUME_CREATED firmware event was issued by firmware too soon, and the target would timeout the first inquiry command, which lead to target reset.   The fix was to add a delay via mptsas_test_unit_ready, which by the time the inquiry was passed, it was accepting commands.

Major Changes For Version 04.00.00.00-1
Release Date: 01/08/2007

General Changes
- Bump Version

Major Changes For Version 03.03.20.00-1

Release Date: 01/08/2007

General Changes

Functionality
- SAS Loginfo updates
- lsiutil updates – version 1.48.11
- endian bug fix's for ppc
- renamed mpt_sp_ioc_info to mpt_iocstatus_info, and added some updated printks
- CSMI = GET_RAID_CONFIG (Device ID and Additional Data) were not working, and it involved bug fix's in both driver and application.
- CSMI = GET_LOCATION - two hot spares was not being displayed, only one
- MPT_DEBUG_REPLY – cosmetic clean up for this debug level

Defect fixes

- **Issue:** Upgrading from an older to newer RPM doesn't upgrade properly when using –Uvh options.   You will see several 'mv' errors.
- **Reported by:** Test.
- **To Reproduce:** Install an rpm using 'rpm –ivh', then try upgrading to a newer version using 'rpm –Uvh'.
- **Description of Change:** Corrected several bugs in the rpm spec file.

- **Issue:** Hidden raid component exposed to block layer after resysnc
- **Reported by:** Development.
- **To Reproduce:** Create an IM volume, and resycn one of the drives. After the drive completed, see drive assigned block dev node, such as /dev/sda.
- **Description of Change:** Added flag ev->hidden_raid_component to track whether a drive needs exposing during hot plug handling.

- **Issue:** panic in mptspi_is_raid
- **Reported by:** Development.
- **To Reproduce:** Load mptspi (53c1030) driver when there is IT firmware.
- **Description of Change:** Added sanity NULL pointer check for ioc_page_2. In previous version of driver, the volume pages are not cached when IR firmware is not detected.

Major Changes For Version 03.03.19.00-1
Release Date: 12/20/2006

General Changes

Functionality
- CSMI IOCTL – set_phy_info , changed return code from CSMI_SAS_STATUS_FAILED to CSMI_SAS_STATUS_BAD_CNTL_CODE to comply to the csmi spec for returning proper code when ioctls are not implemented.
- CSMI IOCTL - get_raid_element and set_raid_operation, added framework for these ioctls, but they are not implemented
- CSMI IOCTL - csmisas_raid_id_to_num – changed return value to signed because failure case is a negative number, and with it being unsigned, it could be interpreted incorrectly.
- CSMI IOCTL - csmisas_get_number_hotspares – changed to return zero for failure case. Before it was returning negative number when mpt_config failed.
- CSMI IOCTL - csmisas_get_ioc_pg5 – added data size as input parameter for the return buffer. In the testing of two global hotspares, it was found the data size was not calculated properly, which resulted in the 2nd hotspare info was not returned.
- CSMI IOCTL - csmisas_get_raid_config and mptspi_is_raid– added check to validating if ioc page2 was non-zero. In previous release, a change prevents reading associated RAID pages when its determined that the firmware is IT.
- mpt_send_handshake_request – was reverted back to previous implementation so the data size was passed as input param.
- MPTSAS_MAX_LUN = changed the max_lun from 511 to 16895. According to SAM3 section 4.9, for luns greater than 255, you must use the flat space address model, which means bit 6 of the 1st byte needs to be set. 16895 is the equivalent to 511.
- Fix'ed the reporting of the proper sas_address for the hba phys that form a wide port. This is reported in the SySFS under /sys/class/sas_phy/phy-x:y/sas_address
- Added sanity checks in mptsas_ioc_reset to prevent possible opps when the list hd->target_reset_list is checked. This could occur if host reset occurs before this list is initialized.   Also moved the spin locks down in mptsas_probe to after the list is initialized.
- mptscsih_io_done – If MPI_IOCSTATUS_SUCCESS and MPI_SCSI_STATUS_BUSY is returned, avoid returning DID_BUSY_BUSY for SPI. This requested by vmware.

- mptscsih_io_done – If MPI_IOCSTATUS_SCSI_IOC_TERMINATED returned for FC, avoid infinite retries by returning DID_ERROR instead of DID_RESET.
- mptscsih_IssueTaskMgmt – If handshake fail's or request times out, clear the tmPending and tmState flags. Since these flags were not cleared, the next task management request would not get issued since sanity checks of these flags would fail.   These flags were not being clear because FAILED was being returned, and the from the eh entry points, we were only clearing them if the return was not the "FAILED" return value.
- mptscsih_taskmgmt_complete - for task aborts, check the termination_count in the reply mf to verify whether the task was completed. Without this fix, its possible that the driver could return a FAILED status to scsi-mid layer, when actually the command was aborted. This could occur during port down retry, as the firmware returned MPI_IOCSTATUS_SCSI_TASK_MGMT_FAILED even though it completed the task. The firmware would do that because the tasks are in the internal pending Q, instead of the request Q, so the task abort never issued across the wire.
- mptscsih_get_scsi_lookup and mptscsih_set_scsi_lookup – change the spin looks to having irqsave/irqrestore API. During some target mode testing, was seeing periodic opps when this API is called.

Major Changes For Version 03.03.18
Release Date: 11/27/2006

General Changes

Functionality
- Merged CSMI IOCTLS that can be shared between 2.6, 3.2, and 3.3 path drivers
- Merged RHEL5 and SLES10 functionality
- Change ioc->DevicesPerBus max value based on port_facts->PortSCSIID, instead of ioc->MaxDevices.   This is to insure proper device scan at driver load time. The MaxDevice fields means max number of targets the firmare will enumerate, but with persistant ids, its does not reflect the maxium number of target_ids that firmare could assign.   The firmware reserves some target_ids for persistent_ids, but it will not exceed PortSCSIID, thus this required change.

Major Changes For Version 03.03.17.05
Release Date: 11/17/2006

General Changes

Defect fixes

- **Issue:** system hang at bootup after installing csmi agents.
- **Reported by:** Test.
- **To Reproduce:** Install csmi agents and reboot.
- **Description of Change:** Fixed a panic in csmisas_fill_location_data_raid() due to NULL pointer.

Major Changes For Version 03.03.17.04
Release Date: 10/25/2006

General Changes

Defect fixes

- **Issue:** cmsitest tool GET_LOCATION ioctl doesn't work
  Reported by: Customer.
  To Reproduce: Use the csmitest tool to issue GET_LOCATION ioctl
  Description of Change: Add helper routines that will convert OS to FW mapping. This ioctl input is OS mapping. This fix was also applied to GET_DEVICE_ADDR, GET_SCSI_ADDR, and ISSUE_TM_PASSTHRU, which also have mapping pass as the input.

Major Changes For Version 03.03.17.03
Release Date: 10/20/2006

General Changes

Defect fixes

- **Issue:** task aborts resulting from cable pull during heavy I/O
  Reported by: Customer.
  To Reproduce: Setup several targets with IO sent simultaneously to each device, then cable pull. Both IO Missing delay and device missing delay need to be enabled.

Description of Change: The worker threads were not invoked fast enough following device_not_responding event, thus scsi mid layer was timing I/O out. The change is to issue target reset immediately when the event arrives from the interrupt handler. A link list is used to queue up the commands.

- **Issue:** csmitest_tool : sas_address for failed drives is not retained in various configurations, most notably RAID0.
  Reported by: Customer.
  To Reproduce: Create a RAID0 volume, and pull a drive. The sas_address should be non-zero for the pulled drive.
  Description of Change: Added device link list to retain sas_address's when devices are added. Devices are not removed from the list after they have been deleted, however when a device is added, there are sanity checks to insure there are unique target id and/or sas address populated in the list.

Major Changes For Version 03.03.17.01
Release Date: 10/13/2006

General Changes

Functionality
- All Debugging printk's in csmisas.c module are moved under MPT_DEBUG_CSMISAS. Also added more enhanced debugging for ssp_passthru.

Defect fixes

- **Issue:** csmitool – GET_RAID_CONFIG doesn't show fault tolerant volume with rebuilding status when it's rebuilding.
  Reported by: Customer.
  To Reproduce: Create a IM volume, and pull a drive, and reinsert. Use csmitool to display raid_config, and notice the RaidStatus.
  Description of Change: The driver wasn't looking at the correct member in the volume_page0 structure to determine if the array was rebuilding, or if the array was inactive.

- **Issue:** csmitool – GET_RAID_CONFIG doesn't show FAILED logical volumes.
  Reported by: Customer.

To Reproduce: Create a IS volume, and pull a drive. The volume becomes FAILED. Use csmitool to display raid_config.
Description of Change: Add support in driver for FAILED logical volumes.

- **Issue:** csmitool – SSP_PASSTHRU was sending passthru to incorrect target after a logical volume is created.
  Reported by: Development.
  To Reproduce: Create a volume, and send INQUIRY via SSP_PASSTHRU. Notice that one of the sas_address's will return data to the logical volume instead of the hidden physical disk.
  Description of Change: The change is to look at the phy_info->attached.phys_disk_num to determine if the device is a hidden raid component, and for RAID_PASSTHRU, send the correct target phys disk number for the target_id.

Major Changes For Version 03.03.17
Release Date: 10/09/2006

General Changes

Functionality
- LSIUTIL update – version 1.47
- Add command line option max_lun for mptsas and mptfc. The default is 511.   The scsi_mid layer parameter is max_report_luns, for changing the max supported luns. Both parameters will need to modify if greater than 512 is required.
- CSMI GET_LOCATION – add hot spare support when the target id is for a logical volume.
- CSMI SSP_PASSTHRU – move code to read in payload buffer for a WRITE to the top of function, where the entire buffer could be properly read in from user to kernel space.
- CSMI GET_RAID_CONFIG – fix volume size for greater than 2TB, but in raid component calculation and hot spare sanity checks
- CSMI GET_RAID_CONFIG – add support for displaying offline logical volumes, as well as adding support in in csmitest tool. Including a bug fix in csmitool which had caused an infinite loop when an ioctl returned unsuccessful return codes.
- CSMI – some little endian fix's (some 32bit fields were using 16bit le conversion).
- Ioctl status flags cleanup, in both mptctl and csmi.
- Fix driver panic when uloading mptfc. This was created with previous driver when support for inactive raid volumes was added. The problem was due to the inactive_list link list not being initialized for FC.

- In handshaking routines, changing mdelay to udelay, as it fail's when handshaking during interrupt time.
- Mptfc updates, support for new fibre channel attributes, merged in from upstream kernels.
- Mptfc updates, support for blocking error handling when devices are being removed, thus preventing panic situation. This code merged in from upstream kernels.
- Adding spin_locks when accessing ScsiLookup[] array, protecting this array as its touched during request, completion, and error handling time.
- Mptspi – domain validation fix's to insure the inquiry max speed test is negotiating at correct speed. This fix involves reading the inquiry data, and setup the proper min sync. This fix was corrected in 2.6.19 kernels in spi transport layer. Also added debug messages to properly debug, enabling MPT_DEBUG_DV.
- Mptspi – fix infinite domain validation when a command times out, resulting in host reset. The fix is to program last known parameters w/o having start domain validation all over again.
- Mptspi – sanity checks added to prevent panic which could occur if the reading of config pages within mpt_attach timeout when driver loads, thus resulting in a host reset. The host reset callback handler in mptspi was assuming hd pointer was non-zero.
- Adding channel support in mptscshi_search_running_cmnds , supporting greater than 255 targets. This would effect a host reset, when there were devices populated on more than one bus.
- Task aborts were returning FAILED to scsi mid-layer, even though the task management request succeeded. Thus was fixed in the hd->tm_iocstatus check in mptscsih_IssueTaskMgmt function, adding TASK_TERMINATED and IOC_TERMINATED.

Major Changes For Version 03.03.16
Release Date: 09/07/2006

General Changes

Functionality
- Greater than 256 target and lun support, upto 32bit value, restricted by firmware max supported devices.
- Inactive RAID volume support; reporting its hidden raid components at driver load time, and when foreign configurations are added. When

inactive volume is deleted, the hidden raid components id's are reported to block layer. When inactive volume is activated, then the volume id is reported to block layer.
- Csmisas.c - get_raidconfig – adding support to report DriveType for hotspares.
- Mptspi.c - Cleanup of mptspi target_alloc and slave_alloc routines; adding mptspi_qcmd.
- Mptfc.c – when initiator protocol is disabled, return 0, instead ENODEV. Returning EDNODEV, prevented additional controllers to be probed.
- Mptbase.h - Delete hd->Targets[] memory array. Changes in VirtTarget, using id instead of target_id, and channel instead of bus_id, and delete luns[8].
- Mptbase.c - Cleanup mpt_findImVolumes; changed malloc calls to use GPF_KERNEL, instead of GFP_ATOMIC, added proper allocation, deallocation of pIocPg2

Major Changes For Version 03.03.15
Release Date: 09/01/2006

General Changes

Defect fixes

- **Issue:** csmitool – ioctl get_raid_config, doesn't display a sas address when drive is pulled.
  Reported by: Customer.
  To Reproduce: Create a IM volume, and pull a drive. Use csmitool to display raid_config, and notice the sas_address.
  Description of Change: Add a link list inside the volume vtarget instance that maintains the sas address, target id, and channel for all the hidden raid components. When a drive is pulled, the old data is retained. When a new target is added to existing volume, the new raid component info is added into the link list. If there is an target id/channel that match's the new device, the old data is replaced with the new data.

Major Changes For Version 03.03.14
Release Date: 08/25/2006

General Changes

Functionality

- Add support to display driver version when using the modinfo utility.
- vdev sanity check in mptfc_qcmd.
- Added lsiutil version 1.46.06
- Added updated csmitest_oem v90 support

Defect fixes

- **Issue:** csmitool displays hotspare in IS volumes.
  Reported by: Customer.
  To Reproduce: Create two volumes that are IM and IS, and a hostpare. Then using csmitool to display select GET_RAID_CONFIG, verify that hotspare is only listed for the IM volume.
  Description of Change: Added check for volume type.

Major Changes For Version 03.03.13-2
Release Date: 08/16/2006

General Changes

Functionality

- Removed xenpae configuration from the i686 build.
- Added Power PC `ppc64` driver update disks and rpms.

Defect fixes

- **Issue:** Unresolved symbols using driver update disks.
  Reported by: Software test.
  To Reproduce: Installing SLES10, with DUD, selecting F5 option. Cntrl^Alt^F3 will display unresolved symbols when the driver are loaded from DUD.
  Description of Change: The Yast installer was unloading media drivers in incorrect order when attempting to load the DUD drivers. The unresolved symbols came because the media drivers were never unloaded. The fix was adding a file on the dud called module.order, so the installer knows what order drivers need to be unloaded and loaded.

Major Changes For Version 03.03.13
Release Date: 08/08/2006

General Changes

Functionality
- Added debug printks for new events
    MPI_EVENT_SAS_BROADCAST_PRIMATIVE
    MPI_EVENT_SAS_INIT_DEVICE_STATUS_CHANGE
    MPI_EVENT_SAS_INIT_TABLE_OVERFLOW
    MPI_EVENT_SAS_SMP_ERROR
- removed obsolete flag ioc->reload_fw

Defect fixes

- **Issue:** Path is not being added in a Controller Offline/Online, No REPORT_LUNS command was sent in the Ctlr Plug-back
  Reported by: Customer.
  To Reproduce: By adding/removing/adding cables, offline/online target mode controllers using serial console commands `rdacMgrAltCtlResetHold` and `rdacMgrAltCtlResetRelease`
  Description of Change: Fix's surrounding issuing TUR during device add. The driver needed to send the TUR's on one sec intervals, and only retry on sam_status_busy, unit_attention, and device becoming ready.

- **Issue:** The LSI HBA driver gives a Panic after a Cable Pull
  Reported by: Customer.
  To Reproduce:
  1. It's a 1x1 setup with one host connected to one array (ksystone)
  2. Install HBA driver & MPP
  3. Map 32 LUNs to the host
  4. Reboot the host
  5. After the host boots up, start I/O's to all the 32 LUNs
  6. Pull the Cable on CtlrB
  7. After the FW timer of 144secs expires, the HBA driver started removing the devices and that's when the HBA driver gave a Panic and the Panic stack is as follows with the EIP at "mptsas_qcmd+0x8/0x28 [mptsas]"
  Description of Change: Added a pointer sanity check in mptsas_qcmd.

- **Issue:** Deletion/Creation of volumes after reboots results in doubled volume entries in SG
  Reported by: Software Test
  To Reproduce:
  cfggen 0 create im 2000 0 1 qsync noprompt

cfggen 0 create ime 2000 2 3 4 5 noprompt
cfggen 0 hotspare 6

Reboot, and then run:
cfggen 0 delete noprompt

Then run the volume creation commands again.

View the SG device listing using "sg_map -i -x". Where the primary drives should be listed will be listings for the volumes, in additon to the normal entries for the volumes later down in the list.
Description of Change: Additional handling of MPI_EVENT_RAID_RC_PHYSDISK events.

- **Issue:** After deleting an inactive volume, the driver does not re-add the disks
  Reported by: Software Test
  To Reproduce: Delete an inactive volume, the drives that were in that volume are not added by the driver.
  According to 'dmesg', we receive the events for it, but they are just not added.
  /proc/scsi/scsi and 'sg_map -i -x' do not show any of the disks that were in the inactive volume.
  Description of Change: Additional handling of MPI_EVENT_RAID_RC_PHYSDISK events.

- **Issue:** Taking way too long to recover from FAULT STATE
  Reported by: Software Test
  To Reproduce: Make filesystems on both volumes and start copies/compares between the two.
  Pull the primary on the IM volume, wait 30 seconds after the volume has started resyncing with the hotspare and then pull the first drive of the IME volume. Wait for the IM volume to finish resyncing and then pull the third drive of the IME volume.
  30 seconds later, pull the old hotspare.   After pulling the old hotspare, the card blinks a 801f fault. Linux takes anywhere from 10-20 minutes to recover from the fault (entered as separate Linux issue #392). IO does not fail, and resumes after recovery from the fault.
  Description of Change: In error handling threads, issue hard reset when this is not operational.
- **Issue:** CSMI: SAS Address of hot spare drive is all zero's in the data returned from CSMI GET_RAID_CONFIG Ioctl call
  Reported by: Customer
  To Reproduce: Create an IM volume with a hot spare drive and using csmitest tool issue CC_CSMI_SAS_GET_RAID_CONFIG call to display Drives information and note the SAS Address field for the hot spare drive
  Description of Change: Moved memset of buffer before filling in SAS Address for a hot spare drive

- **Issue:** CSMI Some of the fields returned by GET_RAID_CONFIG CSMI Ioctl call are not filled-in or returns default values for hot spare drives
  Reported by: Customer
  To Reproduce: Create an IM volume with a hot spare drive and using csmitest tool issue CC_CSMI_SAS_GET_RAID_CONFIG call to display Drives information and note that usBlockSize, uDriveIndex are returned zero and bDriveType is returned UNKNOWN
  Description of Change: Filled in usBlockSize, uDriveIndex, and bDriveType fields for hot spare drives in GET_RAID_CONFIG Ioctl

Major Changes For Version 03.03.12
Release Date: 07/28/2006

General Changes

Functionality
- Added CSMI IOCTL Support, v90
  CSMI_SAS_GET_CNTLR_CONFIG – updated uControllerFlags
  CSMI_SAS_GET_RAID_INFO – updated uMaxDriverPerSet and implemented new fields
  CSMI_SAS_GET_RAID_CONFIG – updated bDriveCount, bStatus, bInformation, union.Drives and implemented new fields
  CSMI_SAS_GET_RAID_FEATURES – new IOCTL
  CSMI_SAS_GET_RAID_CONTROL – new IOCTL
- Updated CSMI_SAS_GET_CONNECTOR_INFO Ioctl by filling in data from Manufacturing Page 7

Major Changes For Version 03.03.11
Release Date: 07/14/2006

General Changes

Functionality

- Adding v1.5.13 MPI Headers Updates.
- Adding 1078 Support.
- Transport Layer Retries support added - OEM specified cdb only for tape devices.
- Adding sas nexus loss support, also known as port down retry support.
- Adding support for sas enclosures with smart drives.
- Adding mpt_cmd_retry_count - during hotadding device, the driver will issue test_unit_ready for waiting for end device to get into state to accept commands. The retry count is to specify how many retries.

- Adding mpt_qas command line option - for disabling Quick Arbitration Select feature for entire host (only mptspi).
- Adding new debug prints for new sas loginfo : internal device reset, abort task set internal, clear task set internal, query task internal. Also adding new strings in iop_code_str[] and pl_code_str[].
- Initializing aen_event_read_flag to zero after host reset.
- SAS hot plug remove handling: the target reset is moved from slave destroy to hotplug threads. The vtarget->deleted flag is used to prevent further IO to device.
- Removing 1066/1066E support pci device recoginization.
- Removing mpt_pq_filter command line option.
- Converting pci device id's using the MPI_MANUFACTURE_DEVICEID_XXX and MPI_MANUFACTURE_DEVID_XXX defined in lsi/mpi_cnfh.h.
- Remove MPI_TARGET_FLAGS_VALID_INQUIRY flag.
- Converting define SASDEBUG to MPT_DEBUG_SAS.
- Removing oem references.
- Returing IRQ_NONE from interrupt handler to let upper layers know when driver didn't process the interrupt.
- SendEventAck, called when firmware sends an asyn event to driver. This routine was cleaned up.
- Fix's to insure download boot could occur when either channel of 1030 is reset. Necessary in order for onboard controller in flashless environment to become operational.

Defect fixes

- **Issue:** Lack of Wide Port Support
  Reported by: Developer
  To Reproduce: Add wide target. The end device reported for each phy.
  Description of Change: Wide Port Support (detection of wide/narrow ports, reporting end device once for every phy), adding debug level MPT_DEBUG_SAS_WIDE, adding target alloc and destroy entry points.

- **Issue:** panic when loading mptctl at the same time as one of the fusion llds (mptsas/mptfc/mptspi)
  Reported by: Customer
  To Reproduce: Add wide target. The end device reported for each phy.
  Description of Change: Fix in mpt_device_driver_register, check for NULL pointer pcidev->driver.

- **Issue:** abort_task reported success, when it actually failed.
  Reported by: Developer
  To Reproduce: Added hd->tm_iocstatus flag in interrupt routine, and returns proper status

Description of Change: Additional sanity checks added to prevent scsi cmd from being double completed during error recovery.

Major Changes For Version 03.03.11
Release Date: 07/14/2006

General Changes

Functionality
- SAS Transport Support
- SPI Transport Support (using generic domain validation)
- FC Transport Support

  (This driver base stream created from 3.02.63)