

# MegaRAC® SP-X Firmware Release Document

---

Document Created: March 29, 2013

## Firmware Overview

Firmware Revision	1.08.00.39310
Hardware Platform(s)	Hornet (Pilot-III Evaluation Board) AST2300 Evaluation Board
Hosting Motherboard	AMI's 876 Motherboard and Intel DG41RQ
SVN Revision Label	RR6
Project Tag Name	SP-X

**Note:** RR6 project at SVN revision level 39310 is used for the final official release.

## Contents

1. Release Notes (This document)
2. Evaluation Firmware Image(s)
  - *spx-1080039310-hornet.ima* – Version 1.08.00.39310 on Hornet (Pilot-III) board
  - *spx-1080039310-hornet\_dualimge.ima* – Version 1.08.00.39310 image for Dual image support on Hornet (Pilot-III) board
  - *spx-1080039310-hornet\_mbmc.ima* – Version 1.08.00.39310 image with MBMC support on Hornet (Pilot-III) board
  - *spx-1080039310-hornet\_smbmc.ima* – Version 1.08.00.39310 image with SMBMC support on Hornet (Pilot-III) board
  - *spx-1080039310-hornet\_ssi.ima* – Version 1.08.00.39310 image with SSI support on Hornet (Pilot-III) board
  - *spx-1080039310-ast2300evb.ima* – Version 1.08.00.39310 on AST2300 board
  - *spx-1080039310-ast2300evb\_dualimge.ima* – Version 1.08.00.39310 image for Dual image support on AST2300 board
  - *spx-1080039310-ast2300evb\_mbmc.ima* – Version 1.08.00.39310 image with MBMC support for AST2300EVB
  - *spx-1080039310-ast2300evb\_smbmc.ima* – Version 1.08.00.39310 image with SMBMC support for AST2300EVB
  - *spx-1080039310-ast2300evb\_ssi.ima* – Version 1.08.00.39310 image with SSI Compute Blade support for AST2300 SoC
  - *spx-1080039310-ast2300evb\_nm.ima* – Version 1.08.00.39310 image with extended Node Manager support for AST2300 SoC
3. Source (If applicable)
4. Tools – MDS, YAFU applications, VMCLI application, Standalone Java KVM client application, Firmware Logo & Copyright override application, Key files used to create
5. MDS web workspace
6. Server Engines Matrox Video Driver
7. Documentation

## Firmware Image(s) Size

Hardware Platform	Size
Hornet (Pilot-III Evaluation Board)	32MB
AST2300 Evaluation Board	32MB

## Important Note

The generic SP-X firmware image is for Evaluation purpose only.

As listed above, the hardware platforms used for validation purposes are Pilot-III and AST2300 Evaluation boards. However, this release continues to have support for same family SoCs like Pilot-II and AST2050/AST2150.

Please refer to the remainder of this document for more details on this firmware release.

## Notes

This generic MegaRAC SP-X firmware release has support for the following SoC(s).

1. Pilot-III, Pilot-II
2. AST23000, AST2150, AST2050

Please refer to “CIM Element Mapping” document for profiles’ support in this release.

Please refer to **MDS Details** section of this rolling release for more details on the differences between new and previous versions.

## RR6 Release

### *New Features*

This list includes the features that are newly available in this rolling release, RR6.

1. Software based runtime level Licensing support
2. HPM support
3. MCTP support – I2C based
4. Automation Engine support
5. RADIUS User Privilege support
6. Time zone (Location-based) support
7. Web support for system firewall
8. Support for a different run-level upon warm reset
9. Support for IPv6 only (with disabled IPv4)
10. Improved network availability by delaying network restart in case of multiple Set
11. Firmware support for two NTP servers with project level NTP configuration
  - Web UI support

12. Multiple GPIOs processing support
13. 64MB SPI flash support
14. Multiple Login Fails Audit support
15. YAFU
  - Signed Image upgrade support with an option to replace the existing SignImage key
  - Support to upgrade both the images in the dual-image setup
  - Non-interactive support for all the options for easier script-ability
16. IPMI
  - Support for more than 255 sensors
  - UDS session information support
  - Support for separate message handling on each interface (IPMI on Steroids)
  - Priority configuration support for individual IPMI interfaces – *Caution must be used when changing the priority of individual interfaces as it may affect the overall command handling within the IPMI stack and will reflect in the response time for the affected interfaces.*
  - File Merging upon Delete SEL Entry in Reclaim SEL Space
  - “SEL flushing in the background” support – *Not available when “File Merging upon Delete SEL Entry in Reclaim SEL Space” support is enabled.*
  - IPMI commands for Active Session Information Availability
17. KVM
  - Sharing to allow more than two max sessions
  - BSOD capture as JPEG file
  - Menu option for user-initiated HID override request during concurrent KVM sessions
  - Auto-resizing of the client window based on server's resolution
18. vMedia
  - "ISO-13346 UDF" format support in image redirection
  - Scriptable VMCLI
19. “Power Save Mode” support for USB devices
20. LMedia/RMedia
  - Folder based support for LMedia area
  - Concurrent Local Media and Remote Media support
21. Latest BSP support on Pilot-III
22. DNS-TSIG Authentication Support
23. CIM
  - IPv6 support
  - CPU Profile support
  - System Memory Profile support
  - Time Zone support in TimeService Provider
  - Invoke methods and set operations support in SM CLP
  - Profiles dependency check in case of disabled core profiles

## ***MDS details***

This release includes MDS 3.7.5. Please refer to the readme.txt and release notes documents inside MDS directory for more details. Please follow the MDS installation guide to install it in Windows or Linux platform. Each MDS plug-in has its own user's guide to provide step by step details. All this documentation is part of this MDS release.

## MDS Device (DDF)/Configuration (CDF) support

This section provides details for the differences in devices (DDFs) and/or configuration (CDFs) support in the current and previous versions of MDS.

This is to ensure that an appropriate CDF/DDF can be selected and used in the MDS version being used in a particular project development. Also, the corresponding firmware modules are listed against each CDF/DDF change to avoid any build issues.

### Note:

1. If an active project requires migrating to the new MDS version (with no firmware changes) but doesn't need the CDF/DDF changes, the modified CDF/DDF files can be overridden by importing the same files from the previous MDS version. This is a typical case where the project codebase has been frozen.
2. If an active project migrates to the new rolling release base, it should also migrate to the new MDS version in that release. This is a case of a project based on SP-X bleeding edge codebase.

## Differences between MDS 3.7.5 (RR6) and 3.7.4 (RR5)

Configuration Changes		
CDF Name	Reason for change	Associated firmware packages
UserConfig.cdf	Added new field "UserPasswdConfigured" to find out whether password is configured for a particular User ID.	libipmimsghdlr-2.644.0-src.spx libipmipar-ANY.spx
PEFRecordDetailsConfig.cdf	New CDF file which contains LastBMCPProcessedEventID, LastSWProcessedEventID, LastSELRecordID, LastProcessedTimestamp and LastActionDone fields which are removed from PEFConfig.cdf file as these fields are updated frequently when SEL entries are added.	libipmimsghdlr-2.644.0-src.spx libipmipef-2.36.0-src.spx libipmipar-ANY.spx ipmimain-2.211.0-src.spx libipmistack-2.156.0-src.spx ipmi_dev-2.341.0-src.spx ipmipdk_dev-2.108.0-src.spx
PEFConfig.cdf	Removed fields LastBMCPProcessedEventID, LastSWProcessedEventID, LastSELRecordID, LastProcessedTimestamp and LastActionDone.	libipmimsghdlr-2.644.0-src.spx libipmipef-2.36.0-src.spx libipmipar-ANY.spx ipmimain-2.211.0-src.spx libipmistack-2.156.0-src.spx ipmi_dev-2.341.0-src.spx ipmipdk_dev-2.108.0-src.spx
Bondcfg.cdf	New CDF file for network bonding configurations	libipmimsghdlr-2.644.0-src.spx libipmipar-ANY.spx ipmimain-2.211.0-src.spx ipmi_dev-2.341.0-src.spx
IPMIConfig.cdf	New field to Enable/Disable Internal Power Good Monitoring Support	libipmipar-ANY.spx ipmi_dev-2.341.0-src.spx ipmimain-2.211.0-src.spx libipmichassis-2.23.0-src.spx
LANConfig1.cdf LANConfig2.cdf LANConfig3.cdf LANConfig4.cdf	Added support to configure MTU, PHY speed and IPV6 address.	libipmimsghdlr-2.644.0-src.spx libipmipar-ANY.spx
Device Changes		
DDF Name	Reason for change	Associated firmware packages
Watchdog2.ddf VoltageR.ddf	Added support to choose the LUN field from option list to support more than 256 sensors in SPX firmware.	libipmimsghdlr-2.644.0-src.spx libipmipar-ANY.spx

## MegaRAC® SP-X Firmware Release Document

VoltageE.ddf Voltage.ddf TempR.ddf TempNM.ddf TempE.ddf Temp.ddf SystemEvent.ddf Storage.ddf Service State.ddf SEL_Sensor.ddf ROMB Battery.ddf pwrsupplystatus.ddf Processor.ddf Presence.ddf PowerUnit.ddf PowerSupply.ddf PowerFault.ddf PMBVoltage.ddf PMBTemp.ddf PMBPower.ddf PMBFAN.ddf PMBCurrent.ddf PhysicalSecurity.ddf PeciCtrl.ddf Peci.ddf Mezz.ddf Interruptsensor.ddf Interrupt Sensor.ddf FRU State.ddf FANPWM.ddf FAN.ddf EventLogDisabled.ddf EntityPresence.ddf Drive.ddf Discrete.ddf DigitalVolt.ddf DigitalE.ddf Digital.ddf Current.ddf CpuStatusNC.ddf CpuStatus.ddf CMOS Battery.ddf ChassisIntrusion.ddf AggregatedThermal.ddf AggregatedFault.ddf	<b>This change is required if support for more than 256 sensors is needed.</b>	
FRUDeviceLocatorFile.ddf	The default FRU size is changed from 0xff to 0xffff	libipmipar-ANY.spx
AST2400.ddf	New DDF file for AST2400 SOC	libipmipar-ANY.spx
list.cfg	New field for user ID list	libipmipar-ANY.spx

### Documentation

The documents provided as part of this release are to help start the development based on SP-X tree. Also included are SP-X and MDS user's guides.

## Library API changes between RR6 and RR5

This section lists out all the changes in the existing structures and APIs between the current (RR6) and the previous (RR5) release. Please note that this list doesn't include the newly added structures or APIs since RR5.

Library package	RR5 API/Structure	RR6 API/Structure	Description
Libactivedircfg-src	typedef struct __tag_ADCONFIG { uint8 ADEnable; uint8 SSLEnable; uint32 ADTimeout; uint8 ADRACDomainStrlen; uint8 ADRACDomainStr[256]; uint8 ADType using uint8 ADFilterEnable; uint8 ADDCFilter1Len; uint8 ADDCFilter1[256]; uint8 ADDCFilter2Len; uint8 ADDCFilter2[256]; uint8 ADDCFilter3Len; uint8 ADDCFilter3[256]; uint8 ADRACUserNameStrlen; uint8 ADRACUserName[65]; uint8 ADRACPasswordStrlen; uint8 ADRACPassword[128]; }PACKED AD_Config_T;	typedef struct __tag_ADCONFIG { uint8 ADEnable; uint8 SSLEnable; <b>uint8 Reserved1</b> <b>uint8 Reserved2</b> uint32 ADTimeout; uint8 ADRACDomainStrlen; uint8 ADRACDomainStr[256]; uint8 ADType; uint8 ADDCFilter1Len; uint8 ADDCFilter1[256]; uint8 ADDCFilter2Len; uint8 ADDCFilter2[256]; uint8 ADDCFilter3Len; uint8 ADDCFilter3[256]; uint8 ADRACUserNameStrlen; uint8 ADRACUserName[65]; uint8 ADRACPasswordStrlen; uint8 ADRACPassword[128]; }PACKED AD_Config_T;	Added structure members Reserved1 and Reserved 2 to avoid alignment trap.
Libactivedircfg-src	typedef struct __tag_ADCONFIG { uint8 ADEnable; uint8 SSLEnable; uint32 ADTimeout; uint8 ADRACDomainStrlen; uint8 ADRACDomainStr[256]; uint8 ADType; <b>uint8 ADFilterEnable;</b> uint8 ADDCFilter1Len; uint8 ADDCFilter1[256]; uint8 ADDCFilter2Len; uint8 ADDCFilter2[256]; uint8 ADDCFilter3Len; uint8 ADDCFilter3[256]; uint8 ADRACUserNameStrlen; uint8 ADRACUserName[65]; uint8 ADRACPasswordStrlen; uint8 ADRACPassword[128]; }PACKED AD_Config_T;	typedef struct __tag_ADCONFIG { uint8 ADEnable; uint8 SSLEnable; uint8 Reserved1 uint8 Reserved2 uint32 ADTimeout; uint8 ADRACDomainStrlen; uint8 ADRACDomainStr[256]; uint8 ADType; uint8 ADDCFilter1Len; uint8 ADDCFilter1[256]; uint8 ADDCFilter2Len; uint8 ADDCFilter2[256]; uint8 ADDCFilter3Len; uint8 ADDCFilter3[256]; uint8 ADRACUserNameStrlen; uint8 ADRACUserName[65]; uint8 ADRACPasswordStrlen; uint8 ADRACPassword[128]; }PACKED AD_Config_T;	Removed structure Member ADFilterEnable.
Libadvisercfg-src	typedef struct { unsigned int mouse_mode char keyboard_layout[KEYBOARD_LAN_SIZE]; } AdviserCfg_T;	typedef struct { unsigned int mouse_mode char keyboard_layout[KEYBOARD_LANG_SIZE]; <b>unsigned int hostlock_feature;</b> } AdviserCfg_T;	Added Hostlock feature, structure member.
libAssocDynamicConditions-src	typedef struct {  const CMPIBroker * _broker; const CMPIContext * ctx; const CMPIResult * rslt; const CMPIObjectPath * ref; CMPIStatus * rc;	typedef struct {  const CMPIBroker * _broker; const CMPIContext * ctx; const CMPIResult * rslt; const CMPIObjectPath * ref; CMPIStatus * rc;	Added Structure members  AssociationDB_T *pDBAssoc; char Operation; char assocClass[MAX_LINE_LEN]; char IsValidRoleResultRole; char IsValidAssociation;

	<pre> int recordID, objs, assoc, inst; char * leftClassName, * rightClassName; char ** ResultClassList; int ResultClassCount; const char * Role, * ResultRole; const char ** PropertyList; char IsRefLeft; char IsRefRight; Properties_T properties[MAXPROPCOUNT]; int properties_Count; Status_T status;  }Parameters_T; </pre>	<pre> int recordID, objs, assoc, inst; char * leftClassName, * rightClassName; char ** ResultClassList; int ResultClassCount; const char * Role, * ResultRole; const char ** PropertyList; AssociationDB_T *pDBAssoc; char Operation; char IsRefLeft; char IsRefRight; char assocClass[MAX_LINE_LEN]; Properties_T properties[MAXPROPCOUNT]; int properties_Count; Status_T status; char IsValidRoleResultRole; char IsValidAssociation; }Parameters_T; </pre>	
libAssociation-src	<pre> typedef struct {  const CMPIBroker * _broker; const CMPIContext * ctx; const CMPIResult * rslt; const CMPIObjectPath * ref; CMPIStatus * rc; int recordID, objs, assoc, inst; char * leftClassName, * rightClassName; char ** ResultClassList; int ResultClassCount; const char * Role, * ResultRole; const char ** PropertyList; char IsRefLeft; char IsRefRight; Properties_T properties[MAXPROPCOUNT]; int properties_Count; Status_T status;  }Parameters_T; </pre>	<pre> typedef struct {  const CMPIBroker * _broker; const CMPIContext * ctx; const CMPIResult * rslt; const CMPIObjectPath * ref; CMPIStatus * rc; int recordID, objs, assoc, inst; char * leftClassName, * rightClassName; char ** ResultClassList; int ResultClassCount; const char * Role, * ResultRole; const char ** PropertyList; AssociationDB_T *pDBAssoc; char Operation; char IsRefLeft; char IsRefRight; char assocClass[MAX_LINE_LEN]; Properties_T properties[MAXPROPCOUNT]; int properties_Count; Status_T status; char IsValidRoleResultRole; char IsValidAssociation; }Parameters_T; </pre>	Added structure members AssociationDB_T *pDBAssoc; char Operation; char assocClass[MAX_LINE_LEN]; char IsValidRoleResultRole; char IsValidAssociation;
libCIM_AMIHooks-src	<pre> int AMI_TCPGetPortsList(Ports_list **, unsigned long *); </pre>	<pre> int AMI_TCPGetPortsList(Ports_list **, unsigned long *,char **pStrIP,int nIPCount); </pre>	Added arguments to Get the number of instances for TCPProtocolEndpoint for all Currently listening and established connections.
libCIMAPIHooks-src	<pre> int CIM_TCPGetPortsList(Ports_list **, unsigned long * ); </pre>	<pre> int CIM_TCPGetPortsList(Ports_list **, unsigned long *,char **pStrIP, int nIPCount); </pre>	Added arguments to Get the number of instances for TCPProtocolEndpoint for all Currently listening and established connections.
libCIMAPIHooks-src	<pre> enum Functions { Create_IPMI20_Session = 0, Close_Session, PowerUp, SoftOff, PowerOff, PowerCycle, HardReset, DiagnosticInt, GetDeviceID, GetSystemBootOptions_BootFlags, SetSystemBootOptions_BootFlags, </pre>	<pre> enum Functions { Create_IPMI20_Session = 0, Close_Session, PowerUp, SoftOff, PowerOff, PowerCycle, HardReset, DiagnosticInt, GetDeviceID, GetSystemBootOptions_BootFlags, SetSystemBootOptions_BootFlags, </pre>	Added Enum Functions Set TimeZone, SetBondEnabl GetAllIPv6Addresses,GetIPv6Netw orkCfg,SetIPv6Gateway,SetIPv6Ena ble,SetIPv6Prefix,SetIPv6Address,S etIPv6Source, e,GetBondEnable,

	GetMaxNumUsers, AddUser, ModUser, DelUser, GetUser, GetUserAccess, ClearSEL, GetSensorEventStatus, GetSensorReading, GetNumOfSELEntries, GetAllSELEntries, GetSELEntry, GetSELInfo, GetSELPolicy, GetMaxPossibleSELEntries, GetSDRRepositoryAllocInfo, GetSDRRepositoryInfo, GetCompleteSDR, GetFRUDData, IPMI_GetDeviceID, GetChassisStatus, IsBondEnabled, GetChannelNum, Get IPv4NetworkCfg, SetIPv4Address, SetIPv4NetMask, Set IPv4Gateway, GetLANCount, GetLANIndex, GetLANE nable, SetLANEnable, SetIPv4Source, GetIPv4Source, GetVlanStatus, Get_MACAddress, SetServiceConf, GetServiceConf, GetPHYConfig, GetNWExtEthCfg, GetNWInterfaceStatus, GetNwActIPCfg, SetNWExtIPCfg, SetDNSHostSetting, GetDNSHostSetting, SetDNSRegister, GetDNSRegister, SetDNSDomainSetting, GetDNSDomainSetting, SetDNSDomainName, GetDNSDomainName, SetDNSSetting, GetDNSSetting, SetDNSIPAddress, GetDNSIPAddress, SetDNSRestart, GetDHCPServerIP, GetPortFromNetstat, IsConnectionEstablished, TCPGetPortByName, TCPGetPortsList , FailOver_RedundancySet, GetDualImageFWBootSelector, GetDualImageFWUploadSelector, GetDualImageRebootStatus, GetDualImageCurActiveImg, GetDualImageGetFWVersion, SetDualImageFWBootSelector, SetDualImageFWUploadSelector, IsDualImageSupport, GetIMGFWInfo, Prepare_FlashArea, Restart_DeviceWithNewFirmware, Verify_FirmwareImage, Start_ImageFlash };	GetMaxNumUsers, AddUser, ModUser, DelUser, GetUser, GetUserAccess, ClearSEL, GetSensorEventStatus, GetSensorReading, GetNumOfSELEntries, GetAllSELEntries, GetSELEntry, GetSELInfo, GetSELPolicy, GetMaxPossibleSELEntries, GetSDRRepositoryAllocInfo, GetSDRRepositoryInfo, GetCompleteSDR, GetFRUDData, IPMI_GetDeviceID, GetChassisStatus, IsBondEnabled, SetBondEnable, GetBo ndEnable, GetChannelNum, GetIPv4Ne tworkCfg, GetIPv4Address, SetIPv4Address, SetIPv4NetMask, SetI IPv4Gateway, GetAllIPv6Addresses, Ge tIPv6NetworkCfg, SetIPv6Gateway, SetIPv6Enable, SetIPv 6Prefix, SetIPv6Address, SetIPv6Sourc e, GetLANCount, GetLANIndex, GetLANE nable, SetLANEnable, SetIPv4Source, GetIPv4Source, GetVlanStatus, Get_MACAddress, SetServiceConf, GetServiceConf, GetPHYConfig, SetTimeZone, GetNWExtEthCfg, GetNWInterfaceStatus, GetNwActIPCfg, SetNWExtIPCfg, SetDNSHostSetting, GetDNSHostSetting, SetDNSRegister, GetDNSRegister, SetDNSDomainSetting, GetDNSDomainSetting, SetDNSDomainName, GetDNSDomainName, SetDNSSetting, GetDNSSetting, SetDNSIPAddress, GetDNSIPAddress, SetDNSRestart, GetDHCPServerIP, GetPortFromNetstat, IsConnectionEstablished, TCPGetPortByName, TCPGetPortsList, FailOver_RedundancySet, GetDualImageFWBootSelector, GetDualImageFWUploadSelector, GetDualImageRebootStatus, GetDualImageCurActiveImg, GetDualImageGetFWVersion, SetDualImageFWBootSelector, SetDualImageFWUploadSelector, IsDualImageSupport, GetIMGFWInfo, Prepare_FlashArea, Restart_DeviceWithNewFirmware, Verify_FirmwareImage, Start_ImageFlash };	
--	--	--	--



# MegaRAC® SP-X Firmware Release Document

libCimNotification-src	<pre> struct cimn_class { char classname[CIMN_CLASS_LEN]; filename[CIMN_FILENAME_LEN]; cimn_listenerfp callback; struct cimn_class *next; next struct, }; </pre>	<pre> struct cimn_class { char classname[CIMN_CLASS_LEN]; char filename[CIMN_FILENAME_LEN]; <b>int delay;</b> cimn_listenerfp callback; struct cimn_class *next; }; </pre>	Added Structure Member for delay before sending notification (in seconds)
Libdbx-src	<pre> typedef struct db_static_ip_assignment_setting_data { char interface_name[MAX_CHAR_LEN]; char ipv4_address[MAX_CHAR_LEN]; char gateway[MAX_CHAR_LEN]; char subnet_mask[MAX_CHAR_LEN]; }db_static_ip_assignment_setting_data_t; </pre>	<pre> typedef struct db_static_ip_assignment_setting_data { char interface_name[MAX_CHAR_LEN]; char ipv4_address[MAX_CHAR_LEN]; char gateway[MAX_CHAR_LEN]; char subnet_mask[MAX_CHAR_LEN]; <b>char ipv6_address[MAX_CHAR_LEN];</b> <b>int ipv6_addr_type;</b> <b>int ipv6_subnetprefix;</b> <b>char ipv6_gateway[MAX_CHAR_LEN];</b> }db_static_ip_assignment_setting_data_t; </pre>	Added structure Members ipv6_address[MAX_CHAR_LEN]; int ipv6_addr_type; int ipv6_subnetprefix; char ipv6_gateway[MAX_CHAR_LEN];
Libfeaturedefine-src	<pre> typedef struct { int global_ipv6; int bond_support; int pam_reorder; int service_config; int capture_bsod; int single_port_app; int snmp_support; int lmedia_support; int dynamic_dns; int phy_support; int preserve_config; int rmedia_support; int web_preview; int fail_safe_config; int node_manager; int peci_over_ipmi; int sel_clock_sync; int circular_sel; int kcs_obf_bit; int ipmi_ipv6; int internal_sensor; int ncsi_cmd_support; int java_sol_support; int web_ssl_sha1_support; int web_ssl_tlsv1_support; int ssi_support; int ssi_event_forward; int ipmi_ver_check; int ipmi_res_timeout; int dcmi_1_5_support; int dual_image_support; int userpswd_encryption; int vlan_priorityset; int send_msg_cmd_prefix; int disable_pef_for_sel_entry; int cmm_support; int global_ssh_user; int global_ssh_operator; int global_telnet_user; </pre>	<pre> typedef struct { int global_ipv6; int bond_support; int pam_reorder; int service_config; int capture_bsod; int single_port_app; int snmp_support; int lmedia_support; int dynamic_dns; int phy_support; int preserve_config; int rmedia_support; int web_preview; int fail_safe_config; int node_manager; int peci_over_ipmi; int sel_clock_sync; int circular_sel; int kcs_obf_bit; int ipmi_ipv6; int internal_sensor; int ncsi_cmd_support; int java_sol_support; int web_ssl_sha1_support; int web_ssl_tlsv1_support; int ssi_support; int ssi_event_forward; int ipmi_ver_check; int ipmi_res_timeout; int dcmi_1_5_support; int dual_image_support; int userpswd_encryption; int vlan_priorityset; int send_msg_cmd_prefix; int disable_pef_for_sel_entry; int cmm_support; int global_ssh_user; int global_ssh_operator; int global_telnet_user; </pre>	Added structure members for Host Lock feature and Time Zone Support, Multiple user for vmedia Access, allow empty password login..  int time_zone_support; int disable_empty_passwd_login; int ntp_server_support; int allow_default_empty_passwd_login; int runlevel_support_on_warmreset; int solssh_alternate_escape_seq; int sd_server_support; int ncsi_over_mctp_support; int mctp_support; int mctp_set_slave_addr; int auto_resize_kvm_client_window; int ifc_specific_msg_handling; int power_consumption_virtual_device_usb; int interface_thread_priority; int delayed_lan_restart_support; int sel_write_background; int cached_sensor_reading; int capture_bsod_jpeg; int capture_bsod_raw; int system_firewall_support; int tsig_support; int more_than_256_sensors; int automation_engine_support; int hpm_rollback_support; int ncsi_keep_phy_linkup_support; int ipmi_command_handle_during_flash

	<pre> int global_telnet_operator; int global_telnet_authorization; int global_ssh_authorization; int global_telnet_authenticate; int web_user_support; int web_operator_support; int web_auth_support; int web_ssl_md5_support; int web_ssl_v3_support; int web_javasol_max_tab; int web_enc_hash_support; int online_flashing_support; int timeoutd_sess_timeout; int lmedia_medium_type_sd; int fwupdate_protocol_select; int vdd_power_check_get_video; int save_sel_log_via_web; int global_cim_caching; int fwupdate_section_based_flash; int mbmc_single_nic; int ipmi_thread_monitor_support; int auto_video_recording; int auto_video_recording_remote; int slpd; int runtime_dbgmsg_support; int key_board_language_select;     int extended_privilege;     int del_sel_reclaim_support; } CoreFeatures_T; </pre>	<pre> int global_telnet_operator; int global_telnet_authorization; int global_ssh_authorization; int global_telnet_authenticate; int web_user_support; int web_operator_support; int web_auth_support; int web_ssl_md5_support; int web_ssl_v3_support; int web_javasol_max_tab; int signed_hashed_image_support; int online_flashing_support; int timeoutd_sess_timeout; int lmedia_medium_type_sd; int fwupdate_protocol_select; int vdd_power_check_get_video; int save_sel_log_via_web; int global_cim_caching; int fwupdate_section_based_flash; int mbmc_single_nic; int ipmi_thread_monitor_support; int auto_video_recording; int auto_video_recording_remote; int slpd; int runtime_dbgmsg_support; int key_board_language_select; int extended_privilege; int del_sel_reclaim_support; int time_zone_support; int disable_empty_passwd_login; int ntp_server_support; int allow_default_empty_passwd_login; int runlevel_support_on_warmreset; int solssh_alternate_escape_seq; int sd_server_support; int ncsi_over_mctp_support; int mctp_support; int mctp_set_slave_addr; int auto_resize_kvm_client_window;     int ifc_specific_msg_handling;     int power_consumption_virtual_device_ usb;     int interface_thread_priority; int delayed_lan_restart_support; int sel_write_background; int cached_sensor_reading; int capture_bsod_jpeg; int capture_bsod_raw; int system_firewall_support; int tsig_support; int more_than_256_sensors; int automation_engine_support; int hpm_rollback_support; int ncsi_keep_phy_linkup_support; int ipmi_command_handle_during_flash ing;     int multiple_user_vmedia_access;     int dedicated_device_lmedia_rmedia; int host_lock_feature; int host_auto_lock; </pre>	<pre> shing;     int multiple_user_vmedia_access;     int dedicated_device_lmedia_rmedia; int host_lock_feature; int host_auto_lock;  Added Structure members for CIM profile  int cim_base_server_profile; int cim_boot_control_profile; int cim_clp_profile; int cim_cpu_profile; int cim_dhcp_client_profile; int cim_dns_client_profile; int cim_elec_profile; int cim_ethernet_port_profile; int cim_fan_profile; int cim_indications_profile; int cim_ip_interface_profile; int cim_phy_asset_profile; int cim_psm_profile; int cim_power_supply_profile; int cim_prof_reg_profile; int cim_record_log_profile; int cim_rba_profile; int cim_sensor_profile; int cim_sp_profile; int cim_simple_id_profile; int cim_smash_coll_profile; int cim_sm_admin_domain_profile; int cim_sw_inventory_profile; int cim_sys_mem_profile; int cim_sw_update_profile; int cim_ssh_svc_profile; int cim_telnet_svc_profile; int cim_tcr_profile; int wdt_flush_support; int runtime_license_support; </pre>
--	--	--	---

# MegaRAC® SP-X Firmware Release Document

		<pre> int cim_base_server_profile; int cim_boot_control_profile; int cim_clp_profile; int cim_cpu_profile; int cim_dhcp_client_profile; int cim_dns_client_profile; int cim_elec_profile; int cim_ethernet_port_profile; int cim_fan_profile; int cim_indications_profile; int cim_ip_interface_profile; int cim_phy_asset_profile; int cim_psm_profile; int cim_power_supply_profile; int cim_prof_reg_profile; int cim_record_log_profile; int cim_rba_profile; int cim_sensor_profile; int cim_sp_profile; int cim_simple_id_profile; int cim_smash_coll_profile; int cim_sm_admin_domain_profile; int cim_sw_inventory_profile; int cim_sys_mem_profile; int cim_sw_update_profile; int cim_ssh_svc_profile; int cim_telnet_svc_profile; int cim_tcr_profile; int wdt_flush_support; int runtime_license_support; } CoreFeatures_T; </pre>	
	<pre> typedef struct {     unsigned char     section_name[SECTION_NAME_LEN];     unsigned char Imgmod_ver_Major;     unsigned char Imgmod_ver_Minor;     unsigned char Imgmod_ver_Aux[2];     unsigned char     uploadmod_ver_Major;     unsigned char     uploadmod_ver_Minor;     unsigned char     uploadmod_ver_Aux[2];     unsigned char is_upgrade; }SectionInfo_T;  typedef struct StructFlasherCmdResponse {     unsigned char Command;     unsigned char Status;     unsigned char Data[256];     unsigned short DataLen; } FlasherCmdResponse;  typedef struct StructImageVerificationInfo {     unsigned char     CurrentImageName[16];     unsigned char NewImageName[16];     unsigned long </pre>	<pre> typedef struct {     unsigned char     section_name[SECTION_NAME_LEN];     unsigned char     Imgmod_ver_Major[2];     unsigned char     Imgmod_ver_Minor[2];     unsigned char     Imgmod_ver_Aux[2][4]; unsigned char uploadmod_ver_Major;     unsigned char     uploadmod_ver_Minor;     unsigned char     uploadmod_ver_Aux[4];     unsigned char is_upgrade; }SectionInfo_T;  typedef struct StructFlasherCmdResponse {     unsigned char Command;     unsigned char Status;     unsigned char Data[512];     unsigned short DataLen; } FlasherCmdResponse;  typedef struct StructImageVerificationInfo {     unsigned char     CurrentImageName[2][16]; </pre>	<p>Changed array dimention from 1d t0 2d</p> <p>Increased array size from 256 to 512</p> <p>Changed array dimention from 1d t0 2d</p>

# MegaRAC® SP-X Firmware Release Document

	CurrentImageStartAdd; unsigned long NewImageStartAdd; unsigned long CurrentImageSize; unsigned long NewImageSize; unsigned char CurrentImageVersion[16]; unsigned char NewImageVersion[16]; unsigned char Reserved[20]; unsigned long Status; } ImageVerificationInfo;	unsigned char NewImageName[16]; unsigned long CurrentImageStartAdd[2]; unsigned long NewImageStartAdd; unsigned long CurrentImageSize[2]; unsigned long NewImageSize; unsigned char CurrentImageVersion[2][16]; unsigned char NewImageVersion[16]; unsigned char Reserved[20]; unsigned long Status; } ImageVerificationInfo;	
Libfmh-src	typedef struct { unsigned char; Module_Name[8]; unsigned char; Module_Ver_Major; unsigned char ; Module_Ver_Minor; unsigned short; Module_Type; unsigned long Module_Location; unsigned long; Module_Size; unsigned short Module_Flags; unsigned longModule_Load_Address; unsigned long Module_Checksum; unsigned char Module_Ver_Aux[2]; unsigned char; Module_Reserved[6]; } PACKED MODULE_INFO;	typedef struct { unsigned char Module_Name[8]; unsigned char Module_Ver_Major; unsigned char Module_Ver_Minor; unsigned short Module_Type; unsigned long Module_Location; unsigned long Module_Size; unsigned short Module_Flags; unsigned long Module_Load_Address; unsigned long Module_Checksum; unsigned char Module_Ver_Aux[4]; unsigned char Module_Reserved[4]; } PACKED MODULE_INFO;	Module info structurs increased array size. Decreased array size for Module_Reserved from 6 to 4.  Increased Module_Ver_Aux array size from 2 to 4.
Libfwinfo-src	typedef struct { unsigned char Section_Name[10]; unsigned char Section_Ver_Major; unsigned char Section_Ver_Minor; unsigned char Section_Ver_Aux[2]; }Section_Info;	typedef struct { unsigned char Section_Name[10]; unsigned char Section_Ver_Major; unsigned char Section_Ver_Minor; unsigned char Section_Ver_Aux[4]; }Section_Info;	Increased Array size from 2 to 4
Libiniparser-src	typedef struct _dictionary_ { int n ; int size ; char val ; char key ; unsigned hash ; } dictionary ;	Removed	Removed Structure
Libuserauth-src	int GetReqUserInfo(void *LSession, char* pReqUser,char* pReqPassword,struct passwd *pwd,char *buffer,size_t buflen,uid_t uid,int getUserInfo);	int GetReqUserInfo(void *LSession, char* pReqUser,char* pReqPassword,struct passwd *pwd,char *buffer,size_t buflen,uid_t uid, int getUserInfo, usrpriv_t *Privilege);	Added Arguments int getUserInfo, usrpriv_t *Privilege.

# MegaRAC® SP-X Firmware Release Document

Libtoken-src	typedef struct { int        Status; char websession_token[WEBTOKEN_SIZE }; } __attribute__((packed)) websession_info_t;	typedef struct { unsigned short    cmd; unsigned short    datalen; } __attribute__((packed)) cmd_info_t;	Changed Structure membes.
Libsmclpdpkt-src	typedef struct smclpd_packet { SMP_TYPE type; SMP_PROC_STATUS status; union packet_data{ SMP_CS_Req_Data *cs_req; SMP_Cmd_Req_Data *cmd_req; SMP_Cmd_Resp_Data *cmd_resp; SMP_Jobs_Req_Data *jobs_req; SMP_Jobs_Resp_Data *jobs_resp; SMP_Job_Req_Data *job_req; SMP_Job_Resp_Data *job_resp; SMP_Alt_Session_Req_Data *alt_session_req_data; SMP_Alt_Session_Resp_Data *alt_session_resp_data; SMP_GAS_Req_Data *gas_req SMP_GAS_Resp_Data *gas_resp; SMP_GS_Req_Data *gs_req; SMP_GS_Resp_Data *gs_resp; SMP_MS_Req_Data *ms_req SMP_DS_Req_Data *ds_req SMP_CLPConfig_Req_Data *clp_req; SMP_CLPConfig_Resp_Data *clp_resp; }data; }SMPACKET;	typedef struct smclpd_packet { SMP_TYPE type; SMP_PROC_STATUS status; union packet_data{ SMP_CS_Req_Data *cs_req; SMP_Cmd_Req_Data *cmd_req; SMP_Cmd_Resp_Data *cmd_resp SMP_Jobs_Req_Data *jobs_req; SMP_Jobs_Resp_Data *jobs_resp; SMP_Error_Req_Data *error_req; SMP_Error_Resp_Data *error_resp; SMP_Job_Req_Data *job_req; SMP_Job_Resp_Data *job_resp; SMP_Alt_Session_Req_Data *alt_session_req_data; SMP_Alt_Session_Resp_Data *alt_session_resp_data; SMP_GAS_Req_Data *gas_req SMP_GAS_Resp_Data *gas_resp SMP_GS_Req_Data *gs_req; SMP_GS_Resp_Data *gs_resp SMP_MS_Req_Data *ms_req; SMP_DS_Req_Data *ds_req; SMP_CLPConfig_Req_Data *clp_req; SMP_CLPConfig_Resp_Data *clp_resp; SMP_DeleteJob_Req_Data *deletejob_req; }data}SMPACKET;	Added new structure members SMP_DeleteJob_Req_Data *deletejob_req; SMP_Error_Req_Data *error_req; SMP_Error_Resp_Data *error_resp;
Libradiusconf-src	typedef struct { unsigned char Enable; unsigned char IPAddr[32]; unsigned short PortNum; unsigned char Secret[32]; unsigned int Timeout; unsigned int ExtendedPrivilege; } RADIUSCONFIG;	typedef struct { unsigned char Enable; unsigned char IPAddr[256]; unsigned short PortNum; unsigned char Secret[32]; unsigned int Timeout; unsigned int ExtendedPrivilege; unsigned int Privilege; } RADIUSCONFIG;	Added New structure member unsigned int Privilege; ; for additional setting for IPMI LAN privilege and Increased array size to 256 from 32.
Libqom-src	struct qom_config { char qom_initialized unsigned short max_threads; unsigned short min_threads; char prov_libpath[MAX_PATH_LEN]; char class_repos_loc[MAX_PATH_LEN]; unsigned int max_thread_idle_time;  unsigned int default_prov_unload_time; char	struct qom_config { char qom_initialized; unsigned short max_threads; unsigned short min_threads; char prov_libpath[MAX_PATH_LEN]; char class_repos_loc[MAX_PATH_LEN];  unsigned int max_thread_idle_time; unsigned int default_prov_unload_time; char	Added structure member for mutex lock for Pam authenyication.

	<pre>default_ns[QOM_MAX_NAMESPACE_LEN]; char qom_terminating; int auth_enabled; char auth_lib[MAX_PATH_LEN]; void *auth_handle; char auth_func[MAX_PATH_LEN]; int (*auth_func_ptr) (pam_handle_t **, char *, char *, usrpriv_t *, char *, char *, char *); int http_enabled; int http_port; int https_enabled; int https_port; char key_file[MAX_PATH_LEN]; char cert_file[MAX_PATH_LEN]; };</pre>	<pre>default_ns[QOM_MAX_NAMESPACE_LEN]; char qom_terminating; int auth_enabled; char auth_lib[MAX_PATH_LEN]; void *auth_handle; char auth_func[MAX_PATH_LEN]; int (*auth_func_ptr) (pam_handle_t **, char *, char *, usrpriv_t *, char *, char *, char *); int http_enabled; int http_port; int https_enabled; int https_port; char key_file[MAX_PATH_LEN]; char cert_file[MAX_PATH_LEN]; pthread_mutex_t auth_lock;};</pre>	
Libopenradius-src	<pre>typedef struct radius_server_t {     struct radius_server_t *next;     struct in_addr ip;     u_short port;     char *hostname;     char *secret;     int timeout;     int accounting; } radius_server_t;</pre>	<pre>typedef struct radius_server_t {     struct radius_server_t *next;     struct in_addr ip;     struct in6_addr ip6;     u_short port;     char *hostname;     char *secret;     int timeout;     int accounting;     int family; } radius_server_t;</pre>	Added structure members int family; struct in6_addr ip6;
Libntpconf-src	<pre>extern int libami_getntpServer(char *); extern int libami_setntpServer(char *);</pre>	<pre>extern int libami_getntpServer(char *server1, char *server2, unsigned int length); extern int libami_setntpServer(unsigned char param, char *server);</pre>	Changed the Arguments.
Libnetwork-src	<pre>typedef struct netstat_est_data {     char localAddress[16];     char remoteAddress[16];     unsigned long ulLocalPort;     unsigned long ulRemotePort;     struct netstat_est_data * next; } NetstatEst_data;  typedef struct netstat_listen_data {     char localAddress[16];     unsigned long ulLocalPort;     struct netstat_listen_data * next; } NetstatListen_data;</pre>	<pre>typedef struct netstat_est_data {     char localAddress[128];     char remoteAddress[128];     unsigned long ulLocalPort;     unsigned long ulRemotePort;     struct netstat_est_data * next; } NetstatEst_data;  typedef struct netstat_listen_data {     char localAddress[128];     unsigned long ulLocalPort;     struct netstat_listen_data * next; } NetstatListen_data;</pre>	<p>Increasded array size from 16 to 128</p> <p>Increasded array size from 16 to 128</p>
Libnetwork-src	<pre>int AddToPortsList(Ports_list ** ppPortsList, unsigned long ulPort, char * pStrServiceName, char cType, char * pstrIP, unsigned long ulLocalPort, char * pstrLocalIP);</pre>	<pre>int AddToPortsList(Ports_list ** ppPortsList, unsigned long ulLocalPort, char * pStrServiceName, char cType, char * pstrLocalIP, unsigned long ulRemotePort, char * pstrRemoteIP);</pre>	Changed the arguments.
Libnetwork-src	<pre>typedef struct {     INT8U DNSIP1[IP6_ADDR_LEN];     INT8U DNSIP2[IP6_ADDR_LEN];     INT8U DNSIP3[IP6_ADDR_LEN];</pre>	<pre>typedef struct {     INT8U DNSIP1[IP6_ADDR_LEN];     INT8U DNSIP2[IP6_ADDR_LEN];     INT8U DNSIP3[IP6_ADDR_LEN];</pre>	Added structure member DNSEnable.

	<pre> INT8U  DNSDHCP; INT8U  DNSIndex; INT8U  IPPriority; }PACKED DNSCONF; </pre>	<pre> <b>INT8U  DNSEnable;</b> INT8U  DNSDHCP; INT8U  DNSIndex; INT8U  IPPriority; }PACKED DNSCONF; </pre>	
Libnetwork-src	<pre> typedef struct{ unsigned char CfgMethod; unsigned char MAC[6]; unsigned char Local_MAC[6]; unsigned char IPAddr[4]; unsigned char Broadcast[4]; unsigned char Mask[4]; unsigned char Gateway[4]; unsigned char BackupGateway[4]; unsigned char enable; unsigned char IFName[10]; unsigned short VLANID; unsigned char Master; unsigned char Slave; unsigned char BondIndex; } PACKED NWCFG_STRUCT; </pre>	<pre> typedef struct{ unsigned char CfgMethod; unsigned char MAC[6]; unsigned char Local_MAC[6]; unsigned char IPAddr[4]; unsigned char Broadcast[4]; unsigned char Mask[4]; unsigned char Gateway[4]; unsigned char BackupGateway[4]; unsigned char enable; unsigned char IFName[10]; unsigned short VLANID; <b>unsigned char vlanpriority;</b> unsigned char Master; unsigned char Slave; unsigned char BondIndex; } PACKED NWCFG_STRUCT; </pre>	Added new structure member unsigned char vlanpriority; for Virtual Lan priority.
Libnetwork-src	<pre> typedef struct netstat_data { char localAddress[16]; char remoteAddress[16]; unsigned long ulLocalPort; unsigned long ulRemotePort; struct netstat_data * next; } Netstat_data; </pre>	<pre> typedef struct netstat_data { char localAddress<b>[64]</b>; char remoteAddress<b>[64]</b>; unsigned long ulLocalPort; unsigned long ulRemotePort; struct netstat_data * next; } Netstat_data; </pre>	Increased Array size from 16 to 64.
Libnetwork-src	<pre> int TCP_GetPortsList(Ports_list ** ppPortsList, unsigned long * pulPort); </pre>	<pre> int TCP_GetPortsList(Ports_list ** ppPortsList, unsigned long * pulPort, char **ppStrIP, int nIPCount); </pre>	Added Argument char **ppStrIP, int nIPCount.
Libncsiconf-src	<pre> typedef struct { char InterfaceName[8]; int AutoSelect; int Packageld; int ChannelId; } NCSIConfig_T; </pre>	<pre> typedef struct { char InterfaceName[8]; int AutoSelect; int Packageld; int ChannelId; <b>int Speed;</b> <b>int Duplex;</b> <b>int AutoNeg;</b> <b>int VetoBit;</b> } NCSIConfig_T; </pre>	Added structure members int Speed; int Duplex; int AutoNeg; int VetoBit;
Libldapconf-src	<pre> typedef struct { unsigned char Enable; unsigned char SSLEnable; unsigned short PortNum; unsigned char IPAddr[256]; unsigned char Password[MAX_LDAP_BW_PASSWD_LEN]; unsigned char BindDN[64]; unsigned char SearchBase[128]; unsigned char DefaultRole; } LDAPCONFIG; </pre>	<pre> typedef struct {unsigned char Enable;unsigned char SSLEnable;unsigned short PortNum;<b>unsigned short PortNumSecondary;</b>unsigned char IPAddr[256];<b>unsigned char IPAddrSecondary[256];</b>unsigned char Password[MAX_LDAP_BW_PASSWD_ LEN];unsigned char BindDN[64];unsigned char SearchBase[128];unsigned char DefaultRole;} LDAPCONFIG; </pre>	Added Structure Members. unsigned short PortNumSecondary; unsigned char IPAddrSecondary[256];
Libmodhapi-src	<pre> typedef struct tag_WebSessionInfo { char fUsed; char SessionCookieString[WEBSSES_SESSI ON_COOKIE_STRLEN+1]; </pre>	<pre> typedef struct tag_WebSessionInfo { char fUsed; SessionCookieString[WEBSSES_SESSI ON_COOKIE_STRLEN+1]; </pre>	Added new structure member CSRFTokenString[WEBSSES_SESSION_COOKIE_STRLEN+1]; a string for CSRF token, The length is 10

	<pre> int ViewerCount; int ActivityCountdown; int InactivityTimeout_Seconds; int NeverTimeout; int CurFlashMode; char TimerTickleDisallow; char UserName[WEBSSES_UNAMEPWD_ MAXLEN+1]; char Password[WEBSSES_UNAMEPWD_M AXLEN+1]; char IPAddr[64]; void* pCallerData; int PortTIMER_CALLBACK_FN* p_fTimerCallback; } WEBSSES_SESSION_INFO; </pre>	<pre> char CSRFTokenString[WEBSSES_SESSION_ CSRF_TOKEN_STRLLEN+1]; int ViewerCount; int ActivityCountdown;. int InactivityTimeout_Seconds; int NeverTimeout; int CurFlashMode; char TimerTickleDisallow; char UserName[WEBSSES_UNAMEPWD_M AXLEN+1]; char Password[WEBSSES_UNAMEPWD_MA XLEN+1]; char IPAddr[64]; void* pCallerData; int Port; TIMER_CALLBACK_FN* p_fTimerCallback; } WEBSSES_SESSION_INFO; </pre>	
libiProviderHelper-src	<pre> PH_MakeInst_HostComputerSystem (_broker, ctx, ref, properties, &amp;pHostComputerSystemData[i], oemInst, rc); </pre>	<pre> PH_MakeInst_HostComputerSystem( _broker, ctx, ref, properties, &amp;pHostComputerSystemData[i], oemInst, <b>EL</b>, rc); </pre>	To create instance with property values
libiProviderHelper-src	<pre> PH_OEM_MakeInst_HostComputerS ystem(_broker, ctx, ref, properties, pInst, oemInst, rc); </pre>	<pre> PH_OEM_MakeInst_HostComputerSy stem(_broker, ctx, ref, properties, pInst, oemInst, <b>oprFlag</b>, rc); </pre>	To attach OEM desired properties to the new instance

## Important Notes

This release includes the licensing feature support. Please refer to the listed items for few specifics about this feature.

1. KVM, vMedia, and CIM services will not work by default, if the licensing feature is enabled. The license key needs to be uploaded to enable KVM, vMedia, and CIM services on the BMC. The default license key is **WFS0-3J4Y-RH0D-R1ND-X6R2-M188-PR** and is also provided along with the release deliverables.
2. A licensed feature may not work if a license key for the feature is uploaded when the existing key already has the license for the same feature.

## Known issues

Please refer to the published “Known Issues” listing for the known issues in this release, RR6.



## RR5 Release

### ***New Features***

This list includes the features that are newly available in this rolling release, RR5.

1. KVM
  - Low bandwidth video capture support
  - Auto-recorded video files
    - Remote share support
    - Standalone Java client support for playback
  - Auto resizing to fit the client resolution
  - Privilege support in KVM/VMCLI
  - IPMI Raw command support
  - Single JAR for standalone app
  - Keyboard mapping in KVM to send the correct codes as per host
  - KVM localization using menu option in the client
  - Recorded videos to be downloaded & playable in AVI format
  - RMedia configuration using IPMI commands
  - BSOD download as JPEG
2. IPMI
  - IPMI threads management support
  - Support for reusing the space upon a SEL entry deletion
  - LAN channel mapping via MDS. Setting override support using PDK hook
3. Dual Image
  - Online Firmware Update
  - Pilot-III support
  - Conf migration
4. Memory test support in u-boot
5. Section based flashing support via Web
6. Support for auto reboot in case of abrupt cancellation during YAFU based firmware update
7. Encrypted password support for AD/LDAP server authentication
8. Run-time support to enable debug messages at various levels
9. BSOD capture support on applicable host platforms
  - AST2300 based
10. Sensor threshold setting support from Web UI
11. SEL download (as a text file) support from Web UI
12. NCSI support model (Auto Failover and Manual Switch)
13. Scripting support for YAFUFLASH and YAFUKCS
14. MDS
  - IPMI.conf based configuration support via MDS
  - Multi BMC configuration support
  - An automatic new platform project creation support
15. MBMC
  - Support for single out-of-band interface support for all the instances
  - Performance optimization
  - Threads' minimization
  - Support for single set of PAR/PDK package for all the running instances

## 16. CIM/WSMAN

- GetClass() support for getting local properties of a class
- Privilege support
  - No Access - No actions are permitted
  - User - El/EIN/GI
  - Operator - El/EIN/GI/ModifyInstance/Invoke methods
  - Administrator/OEM - El/EIN/GI/ModifyInstance/Invoke methods/CreateInstance/DeleteInstance
- Removal of non-clp jobs after expiry.
- Added support for CIM\_Error class.
- Added support for GetErrors () invoke method in CIM\_ConcreteJob class.
- Added support for Job Control Profile.
- Modify instance support for CIM\_CLPProtocolEndpoint properties.
- Added additional configuration to CIM\_CLPSettingData and respective associations based on the configuration.
- Added PRJ configuration for enabling CIM caching.
- Added support for CIM\_RedundancySet class.
- Added support for RequestStateChange () invoke method for CIM\_EthernetPort class.
- Additional features in Physical Asset Profile
  - o Added CIM\_PhysicalAssetCapabilities instances for each physical element and corresponding associations.
  - o Added additional properties SKU, HostingBoard, Serial and PartNumber to Physical Asset Profile providers.
- Additional features in Boot Control Profile
  - o Added additional instances to CIM\_BootSourceSetting class.
  - o ChangeBootOrder() method support for both Host Computer System and SP Computer System
  - o Added support for Dual Image ChangeBootOrder() for SP Computer System
- Added additional instances for CIM\_SoftwareIdentity class in case of dual image configuration
- CIM\_ProtocolService instance support for all running services and support for RequestStateChange () method for all instances.
- Added support for ManageTime () in CIM\_TimeService class.
- CIM Schema is upgraded to the latest version 2.32
- Added support for Indications Profile

## 17. SMASHCLP

- Added support for customizing targets and properties
  - Property customization in SM CLP
    - The properties defined in OEM schema will be displayed with OEM configured prefix.
    - The prefix can be configured from PRJ menu item
    - Smash Configuration -> OEM SMASH Properties Prefix, Options OEM Prefix. The default value will be "CIM"

- Target customization in SM CLP  
OEM defined targets can be added in SM CLP target tree. The relevant changes need to be done in oem\_smashclp\_configuration package. The changes include adding target details in oem\_smash-cim.ini and adding symbols in SM\_OEM\_bnf.h files.

The customization steps are explained in sections 5.1 & 5.3 in CIM Customization Design document.

- Privilege support
  - No Access - No actions are permitted
  - User - Read only
  - Operator - Read/Write/Execute
  - Administrator/OEM - Read/Write/Execute/Create/Delete

## ***MDS details***

This release includes MDS 3.7.4. Please refer to the readme.txt and release notes documents inside MDS directory for more details. Please follow the MDS installation guide to install it in Windows or Linux platform. Each MDS plug-in has its own user's guide to provide step by step details. All this documentation is part of this MDS release.

## **MDS Device (DDF)/Configuration (CDF) support**

This section provides details for the differences in devices (DDFs) and/or configuration (CDFs) support in the current and previous versions of MDS.

This is to ensure that an appropriate CDF/DDF can be selected and used in the MDS version being used in a particular project development. Also, the corresponding firmware modules are listed against each CDF/DDF change to avoid any build issues.

### **Note:**

3. If an active project requires migrating to the new MDS version (with no firmware changes) but doesn't need the CDF/DDF changes, the modified CDF/DDF files can be overridden by importing the same files from the previous MDS version. This is a typical case where the project codebase has been frozen.
4. If an active project migrates to the new rolling release base, it should also migrate to the new MDS version in that release. This is a case of a project based on SP-X bleeding edge codebase.

## **Differences between MDS 3.7.4 (RR5) and 3.7.3 (RR4)**

Configuration Changes		
CDF name	Reason for change	Associated firmware packages

UserConfig.cdf	Renamed “Flags” field to “ExtendedPrivilege” to take care of KVM privileges.	libactivedircfg-1.21.0-src.spx libipmimsghdlr-2.501.0-src.spx ipmi_dev-2.278.0-src.spx cdserverapp-1.40.0-src.spx libuserprivilegepdsk-1.12.0-src.spx adviser-1.107.0-src.spx fserverapp-1.41.0-src.spx libldapconf-1.24.0-src.spx libmodhapi-1.84.0-src.spx ipmimain-2.172.0-src.spx libradiusconf-1.5.0-src.spx libipmipar-ANY.spx libifc-1.203.0-src.spx webifc_dev-1.35.0-src.spx
SerialConfig.cdf	The default configurations of field “Data2” which belongs to “MUX Switch Control” is changed from ‘0’ to ‘1 << 3’	No changes need in source packages as there is change in default configuration.
LoginAuditCfg.cdf	Added new field “KVMEventMask” to configure the need for SEL events in case of KVM session failure.	
LANIfcConfig.cdf	New cdf file to take care of default IPMI LAN Configurations	libipmipar-ANY.spx ipmimain-2.172.0-src.spx
IPMIConfig.cdf	Added options to enable/disable Chassis timer, IPMI firewall.	ipmimain-2.172.0-src.spx ipmi_dev-2.278.0-src.spx libipmipar-ANY.spx libipmimsghdlr-2.501.0-src.spx
<b>Devices Changes</b>		
<b>DDF name</b>	<b>Reason for change</b>	<b>Associated firmware packages</b>
FRUDeviceLocatorFile.ddf	FRU file path length is changed from 40 to 64 bytes	No changes need in source packages as there is change in FRU path length.
SEL_Sensor.ddf	Sensor to find the SEL repository status. It helps in logging event when SEL is 75% and 100% full.	libipmimsghdlr-2.501.0-src.spx

## Documentation

The documents provided as part of this release are to help start the development based on SP-X tree. Also included are SP-X and MDS user’s guides.

## Library API changes between RR5 and RR4

This section lists out all the changes in the existing structures and APIs between the current (RR5) and the previous (RR4) release. Please note that this list doesn’t include the newly added structures or APIs since RR4.

Library package	RR4 API/Structure	RR5 API/Structure	Description
libuserprivilegepdsk	typedef struct { struct { int lanpriv:4; int serialpriv:4; int lan1priv:4; int lan2priv:4; } };	typedef struct { struct { int lanpriv:4; int serialpriv:4; int lan1priv:4; int lan2priv:4; unsigned short reserved; //for	Added extended privilege support for KVM/Media sessions.

# MegaRAC® SP-X Firmware Release Document

	<pre>int PreferredShell; }usrpriv_t;</pre>	<pre>padding revanth added }ipmi; int PreferredShell; int Extendedprivilege; }usrpriv_t;</pre>	
libsmtp	<pre>typedef struct {     char username[65];    /*! Username to authenticate with mail server */     char password[65];    /*! Password to authenticate with mail server */     char local_host[256]; /*! The local host name to say hi with mail server */     char smtp_server[INET6_ADDRSTRLEN]; /*! Vaild IP address of Mail server */     char to_addr[ADDR_SIZE]; /*! Email address */     char cc_addr[ADDR_SIZE]; /*! Email address */     char err_addr[ADDR_SIZE]; /*! Email address */     char from_addr[ADDR_SIZE]; /*! Email address */     char reply_addr[ADDR_SIZE]; /*! Email address */     char subject[ADDR_SIZE]; /*! Email subject */     char message_body[MESSAGE_BODY_SIZE]; /*! The message to be sent */     unsigned char Attach_File; /*Checks for File attachment */     char LogValues[FILE_ATTACHMENT_SIZE]; /*Holds the Attachment data*/     unsigned char UserID; /* User ID*/     int AuthEnable; } SMTP_STRUCT;</pre>	<pre>typedef struct {     char username[65];    /*! Username to authenticate with mail server */     char password[65];    /*! Password to authenticate with mail server */     char local_host[256]; /*! The local host name to say hi with mail server */     char smtp_server[INET6_ADDRSTRLEN]; /*! Vaild IP address of Mail server */     char to_addr[ADDR_SIZE]; /*! Email address */     char cc_addr[ADDR_SIZE]; /*! Email address */     char err_addr[ADDR_SIZE]; /*! Email address */     char from_addr[ADDR_SIZE]; /*! Email address */     char reply_addr[ADDR_SIZE]; /*! Email address */     char subject[ADDR_SIZE]; /*! Email subject */     char message_body[MESSAGE_BODY_SIZE]; /*! The message to be sent */     unsigned char Attach_File; /*Checks for File attachment */     char LogValues[FILE_ATTACHMENT_SIZE]; /*Holds the Attachment data*/     unsigned char UserID; /* User ID*/     int AuthEnable;     int resptimeout;     int smtp_retries;     int retryinterval;     int smtp_portno; } SMTP_STRUCT;</pre>	Extended the structure to include SMTP port number, response time out, retry intervals, SMTP retries count
libradiusconf	<pre>typedef struct {     unsigned char Enable;     unsigned char IPAddr[32];     unsigned short PortNum;     unsigned char Secret[32];     unsigned int Timeout; } RADIUSCONFIG;</pre>	<pre>typedef struct {     unsigned char Enable;     unsigned char IPAddr[32];     unsigned short PortNum;     unsigned char Secret[32];     unsigned int Timeout;     unsigned int ExtendedPrivilege; } RADIUSCONFIG;</pre>	Added extended privilege support for KVM/Media sessions.
libprocmanager	<pre>int ProcMonitorRegister(char *procexecv,void (*handler)(int),int procargcnt,...);</pre>	<pre>int ProcMonitorRegister(char *procexecv,int port,char *slpargs, void (*handler)(int),int procargcnt,...);</pre>	Modified to pass port number and slp arguments along with the existing arugments.
libphyconf	<pre>int setPHYConfig(char* interfaceName, int autoNegotiation, int speed, int duplex)</pre>	<pre>int setPHYConfig(char* interfaceName, int autoNegotiation, int speed, int duplex,int mtu)</pre>	Passing MTU size along with the existing arguments

# MegaRAC® SP-X Firmware Release Document

[illegible]

## MegaRAC® SP-X Firmware Release Document

libactivatedircfg	int SSADUserPrivilege(LDAP * Id,char* userName,unsigned char* userDomainName,unsigned int* prv);	int SSADUserPrivilege(LDAP * Id,char* userName,unsigned char* userDomainName,unsigned int* prv,unsigned int* Userflag);	Added argument 'Userflag' to support KVM/Media privilege
libfeaturedefine		Added following fields to CoreFeatures_T  int vdd_power_check_get_video; int save_sel_log_via_web; int global_cim_caching; int fwupdate_section_based_flash; int mbmc_single_nic; int ipmi_thread_monitor_support; int auto_video_recording; int auto_video_recording_remote; int slpd; int runtime_dbgmsg_support; int key_board_language_select; int extended_privilege; int del_sel_reclaim_support;	Fields added to support the new features added in RR5.
instgend-src	struct PROFILE_INFO_T	Removed	Removed wrt generic provider customization, as the ProfileNames and their shared libraries are not required anymore.
libCIM_AMIHooks-src	int AMI_IsConnectionEstablished (char *);	int AMI_IsConnectionEstablished (char *, <b>Netstat_data *</b> );	Data from linux applet "netstat" is retrieved, while searching for a specific Remote Address.
libdbx-src	typedef struct db_physical_asset { db_fru_t fru; int id_str_len; char id_str[MAX_CHAR_LEN]; }db_phy_asset_t;	typedef struct db_physical_asset { db_fru_t fru; int id_str_len; char id_str[MAX_CHAR_LEN]; <b>int sen_num;</b> }db_phy_asset_t;	Added sensor number as a PhysicalAsset element, inorder to form unique InstanceID for PhysicalElements.
libdbx-src	typedef struct db_power_mgt_capabilities { char instance_id[MAX_CHAR_LEN]; char element_name[MAX_CHAR_LEN]; char power_change_capabilities[MIN_CHAR_LEN]; char power_states_supported[MIN_CHAR_LEN]; }db_power_mgt_capabilities_t;	typedef struct db_power_mgt_capabilities { char instance_id[MAX_CHAR_LEN]; char element_name[MAX_CHAR_LEN]; char power_change_capabilities[MIN_CHAR_LEN]; char power_states_supported[MIN_CHAR_LEN]; /* Added for CIM-Schema upgradation */ char req_power_states_supported[MIN_CHAR_LEN]; }db_power_mgt_capabilities_t;	Added RequestedPowerStatesSupported wrt CIM Schema Upgradation as RequestPowerStateChange() method is supported in CIM_PowerManagementService.

libdbx-src	typedef struct { char name[MAX_CHAR_LEN]; int protocol; char element_name[MAX_CHAR_LEN]; int max_connections; int requested_state; int enabled_state; int health_state; char operational_status[MAX_CHAR_LEN]; }; }db_protocol_service_t;	typedef struct { char name[MAX_CHAR_LEN]; int protocol; char element_name[MAX_CHAR_LEN]; int max_connections; int requested_state; int enabled_state; int health_state; char operational_status[MAX_CHAR_LEN]; char otherprotocol[MAX_CHAR_LEN]; }; }db_protocol_service_t;	Added attribute " otherprotocol" to provide additional descriptions for other protocols such as ., HTTP, CD -MEDIA, FD - MEDIA , KVM etc.,
libdbx-src	typedef struct db_concrete_job { char InstanceID[MAX_CHAR_LEN]; char ElementName[MAX_CHAR_LEN]; int JobState; char JobStatus[MAX_CHAR_LEN]; int DeleteOnCompletion; char TimeBeforeRemoval[MAX_CHAR_LEN]; char OperationalStatus[MAX_CHAR_LEN]; }; }db_concrete_job_t;	typedef struct db_concrete_job { char InstanceID[MIN_CHAR_LEN]; char ElementName[MIN_CHAR_LEN]; int JobState; char JobStatus[MIN_CHAR_LEN]; int DeleteOnCompletion; char TimeBeforeRemoval[MIN_CHAR_LEN]; char OperationalStatus[MAX_CHAR_LEN]; int PercentComplete; long RemovalTime; }; }db_concrete_job_t;	Added new attributes PercentComplete & RemovalTime , for removing concrete job within specified interval and to describe the job status during execution.
libdbx-src	int db_set_concretejob(sqlite3* pdbhl, db_concrete_job_t* con_job);	int db_set_concretejob(sqlite3* pdbhl, db_concrete_job_t* con_job, char *pJobInstID);	JobInstanceID parameter is used to keep track of the job, when PH_CreateJobInstance is used throughout the module.
libdbx-src	int db_modify_concretejob(sqlite3* pdbhl, char * InstanceID);	int db_modify_concretejob(sqlite3* ptmpdbhl, char * InstanceID, char* JobStatus, char* OperationalStatus, int PercentComplete, int JobState);	Additional parameters are passed to ModifyJobInstance, as each concrete job may pass or fail. This function sets the exact reason for which the Job has failed.
libqom -src	struct property { char *name;/*!< name of the property*/ unsigned short type;/*!< type of the property*/ CMPIBoolean key;/*!< key property flag*/ struct list qualifiers;/*!< list of qualifiers for the property*/ };	struct property { char *name;/*!< name of the property*/ unsigned short type;/*!< type of the property*/ CMPIBoolean key;/*!< key property flag*/ CMPIBoolean override;/*!< Override property flag*/ struct list qualifiers;/*!< list of qualifiers for the property*/ };	Added new attribute override to indicate whether the property is overridden by OEM.



# MegaRAC® SP-X Firmware Release Document

libqom -src	<pre>struct ea_data{     struct list *prov_hook_list;/*!&lt;     provider hooks to be invoked*/     enum operation op;/*!&lt; operation -     action to be performed*/     union {         struct ein_data ein;         struct ei_data ei;         struct gi_data gi;         struct ci_data ci;         struct mi_data mi;         struct di_data di;         struct ai_data ai;         .....};</pre>	<pre>struct ea_data{     struct list *prov_hook_list;/*!&lt;     provider hooks to be invoked*/     struct list *class_name_list; /*!&lt; List     of sub class names in the hierarchy */     enum operation op;/*!&lt; operation -     action to be performed*/     union {         struct ein_data ein;         struct ei_data ei;         struct gi_data gi;         struct ci_data ci;         struct mi_data mi;         struct di_data di;         struct ai_data ai;         .....};</pre>	Retrieved ClassName for each hook present for the provider, wrt generic provider customization.
libwsman_dev-src	<pre>typedef struct Session {     char username[128];     char password[128];      char user_ip[32];     int user_port;     ..... };</pre>	<pre>typedef struct Session {     char username[128];     char password[128];     int user_priv;     char user_ip[32];     int user_port;     .....};</pre>	Added "user_priv" to provide special privileges for a user in SMASH CLP.
libnetwork-src	<pre>int tftp(const int cmd, const char *     hostsys, const char *remotefile, int     localfd, const int port, int     tftp_bufsize, const char     *progressfile);</pre>	<pre>int tftp(const int cmd, const char *     hostsys, const char *remotefile, int     localfd, const int port, int tftp_bufsize,     const char *progressfile, long int     filesize);</pre>	File size is passed as an additional argument for trivial file transfer.
libnetwork-src	<pre>1) void     ConvertIPnumToStr(unsigned char     *var, unsigned int len, char *string);</pre>	<pre>1) void ConvertIPnumToStr(unsigned     char *var, unsigned int len, unsigned     char *string);</pre>	1) Changed Data Type of the output string to unsigned char*.
libProfile-src	Each classes had different providers, the API to CIMOM is exposed as <pre>OSBase_&lt;CLASS-NAME&gt;CleanUp OSBase_&lt;CLASS-NAME&gt;EnumInstanceNames OSBase_&lt;CLASS-NAME&gt;EnumInstances OSBase_&lt;CLASS-NAME&gt;GetInstance OSBase_&lt;CLASS-NAME&gt;CreateInstance OSBase_&lt;CLASS-NAME&gt;SetInstance OSBase_&lt;CLASS-NAME&gt;InvokeMethod OSBase_&lt;CLASS-NAME&gt;ExecuteQuery OSBase_&lt;CLASS-NAME&gt;MethodCleanUp</pre>	Since Generic Provider is implemented, CIMOM always calls the following API, <pre>OSBase_ProfileCleanUp OSBase_ProfileEnumInstanceNames OSBase_ProfileEnumInstances OSBase_ProfileGetInstance OSBase_ProfileCreateInstance OSBase_ProfileSetInstance OSBase_ProfileInvokeMethod OSBase_ProfileExecuteQuery OSBase_ProfileMethodCleanUp</pre>	CIMOM server need not to keep track of several shared libraries for their corresponding providers, in RR5 it just loads one shared library.
libProviderHelper-src	Since all the classes has its own provider the Provider <-> Helper API is different for each provider.	<p>With respect to the generic provider implementation, provider helper exposes APIs in the following format,</p> <pre>void PH_EIN_&lt;ProviderName &gt; (const     CMPIBroker * _broker, const     CMPIContext * ctx, const     CMPIObjectPath * ref, const     CMPIResult *rslt, CMPIStatus * rc);</pre> <pre>void PH_EI_&lt;ProviderName&gt;(const     CMPIBroker * _broker, const     CMPIContext * ctx, const     CMPIObjectPath * ref, const char     **properties, const CMPIResult *rslt,     CMPIStatus * rc);</pre>	All the APIs of provider helper are generalized to a common signature.

		<pre>void PH_GI_&lt;ProviderName&gt;(const CMPIBroker *_broker, const CMPIContext * ctx, const CMPIObjectPath *ref, const char **properties, const CMPIResult * rslt, CMPIStatus *rc);</pre>	
		<pre>CMPIStatus PH_MI_&lt;ProviderName&gt;(const CMPIBroker *_broker (in), CMPIInstanceMI * mi, const CMPIContext * ctx, const CMPIObjectPath * cop, const CMPIInstance * ci, const char **properties, const CMPIResult * rslt);</pre>	
		<pre>CMPIStatus PH_IM_&lt;ProviderName&gt;(const CMPIBroker *_broker(in), const CMPIMethodMI * mi, const CMPIContext * ctx, const CMPIObjectPath * ref, const char * methodname, const CMPIArgs * in, const CMPIResult * rslt, CMPIArgs * out);</pre>	
		<pre>CMPIStatus PH_DI_&lt;ProviderName&gt;( const CMPIBroker *_broker, const CMPIInstanceMI * mi, const CMPIContext * ctx, const CMPIObjectPath * cop, const CMPIResult * rslt);</pre>	
		<pre>CMPIStatus PH_EQ_&lt;ProviderName&gt;( const CMPIBroker *_broker(in), const CMPIInstanceMI * mi(in), const CMPIContext * ctx(in), const CMPIObjectPath * ref(in), const char * lang(in), const char * query(in), const CMPIResult * rslt(out));</pre>	
		<pre>CMPIStatus PH_CI_&lt;ProviderName&gt;( const CMPIBroker *_broker, const CMPIInstanceMI * mi, const CMPIContext * ctx, const CMPIObjectPath * cop, const CMPIInstance * ci, const CMPIResult * rslt);</pre>	

## Known issues

The following is the list of known issues that exist in this release. The future releases will provide the fixes for these issues. The PATCH packages can also be provided for individual fixes, if needed.

1. U-boot environment variables may get corrupted when Bonded IP is used to update the firmware image using YAFUFLASH.
2. Configuration migration may not work as expected when firmware image is updated to an older version.
3. Duplicate user names across IPMI/AD/LDAP may result into no-access using that user account.

## RR4 Release

### ***New Features***

This list includes the features that are newly available in this rolling release, RR4.

1. Extended Node Manager support
  - This feature is currently added and tested on AST2300 based platform only.
2. SSI Compute Blade support
  - This feature is supported on both AST2300 and Pilot-III but is tested on AST2300 based live customer platform only.
3. KVM/Media redirection support using http/https
4. Dual Image support
  - This feature is supported on AST2300 only at this time. Pilot-III support is being added and will be available soon.
5. BSOD capture support on applicable host platforms
  - This feature is currently added and tested on Pilot-III based platform only.

### ***MDS details***

This release includes MDS 3.7.3. Please refer to the readme.txt and release notes documents inside MDS directory for more details. Please follow the MDS installation guide to install it in Windows or Linux platform. Each MDS plug-in has its own user's guide to provide step by step details. All this documentation is part of this MDS release.

### **MDS Device (DDF)/Configuration (CDF) support**

This section provides details for the differences in devices (DDFs) and/or configuration (CDFs) support in the current and previous versions of MDS.

This is to ensure that an appropriate CDF/DDF can be selected and used in the MDS version being used in a particular project development. Also, the corresponding firmware modules are listed against each CDF/DDF change to avoid any build issues.

#### **Note:**

5. If an active project requires migrating to the new MDS version (with no firmware changes) but doesn't need the CDF/DDF changes, the modified CDF/DDF files can be overridden by importing the same files from the previous MDS version. This is a typical case where the project codebase has been frozen.
6. If an active project migrates to the new rolling release base, it should also migrate to the new MDS version in that release. This is a case of a project based on SP-X bleeding edge codebase.

### **Difference between MDS 3.7.3 (RR4) and 3.7.1 (RR3)**

Configuration Changes		
CDF name	Reason for change	Associated firmware packages
UserConfig.cdf	<ul style="list-style-type: none"> <li>• The number of default user</li> </ul>	libUserPrivilege-1.15.0-src.spx

	<p>configuration has been changed from 10 to 16.</p> <ul style="list-style-type: none"> <li>Added new field “Flags” to take care of extended privilege needed for PAM authentication</li> </ul>	libuserprivilegepkg-1.4.0-src.spx
SysInfoCfg.cdf	Bug fixing for “Get/Set System Info Parameters” command	ipmi_dev-2.216.0-src.spx libipmimsgghndlr-2.368.0-src.spx
SOLConfig4.cdf SMTPConfig4.cdf RMCPlus4.cdf LANConfig4.cdf LANChannel4.cdf	New CDF files to take care of the 4 <sup>th</sup> NIC configurations.	ipmi_dev-2.216.0-src.spx
LANChannel3.cdf LANChannel2.cdf LANChannel1.cdf	The number of default user configuration has been changed from 10 to 16	ipmi_dev-2.216.0-src.spx
IPMIConfig.cdf	Added a new field “NM_IPMB_BUS” to configure IPMB bus used for Node Manager	libipmimsgghndlr-2.368.0-src.spx ipmimain-2.134.0-src.spx libipminmsupport-2.21.0-src.spx
EncUsrPswdConfig.cdf	Support added to encrypt the password stored in non-volatile memory of BMC.	libipmilan-2.55.0-src.spx libencryption-1.7.0-src.spx ipmi_dev-2.216.0-src.spx UserPswdEncryptionFeature-1.6.0-ANY.spx ipmimain-2.134.0-src.spx ipmipdk_dev-2.80.0-src.spx libactivedircfg-1.18.0-src.spx libipmimsgghndlr-2.368.0-src.spx libipmipdkcmds libipmistack-2.108.0-src.spx libipmiuds-2.20.0-src.spx libldapconf-1.18.0-src.spx sys_base-2.78.0-src.spx
DCMIConfig.cdf	Made changes to take care of DCMI_1.5 support handling in run time.	libipmipar
ChassisConfig.cdf	Added a new field SysPartitionScan to store service partition scan used in “Set System Boot options” command in non-volatile storage.	libipmimsgghndlr-2.368.0-src.spx ipmi_dev-2.216.0-src.spx
<b>Devices Changes</b>		
<b>DDF name</b>	<b>Reason for change</b>	<b>Associated firmware packages</b>
TempNM.ddf	Temperature sensor for Node Manager	libipminmsupport-2.21.0-src.spx
NM.ddf	Hardware Monitoring device of Node Manager	libipminmsupport-2.21.0-src.spx
list.cfg	Added LAN Channel 4 to channel type list.	NONE

## ***CIMSDK***

CIMSDK support is available in the current MDS release on Linux platforms.

1. Support to create new profiles with SPX architecture
2. Support to edit existing profiles

## ***CIM PDK***

CIM PDK in this release supports the followings.

- OEM Hooks to retrieve system specific information required by providers can be added
- Static values assigned to CIM Class properties can be modified.

## ***Documentation***

The documents provided as part of this release are to help start the development based on SP-X tree. Also included are SP-X and MDS user's guides.

## ***Known issues***

The following is the list of known issues that exist in this release. The future releases will provide the fixes for these issues. The PATCH packages can also be provided for individual fixes, if needed.

4. The followings are not currently enabled in "Dual Image" feature support. They are under development and can be provided as updates in the future, if needed.
  - Configuration Preservation
  - Online Firmware update
  - Fail-safe booting
5. SMASHCLPD is taking higher CPU (about 95%) for about two minutes when starting (at boot up) and when "show -l all" SMASHCLP command is given.
6. When Single port is enabled for KVM & Media, the following issues are observed.
  - o Web server may occasionally crash if a connection request is made during SMASCLPD's higher CPU usage as mentioned above in item 2.
  - o HD redirection may produce OOM issue.
  - o Linux based VMCLI tool doesn't work.
7. Web based firmware update doesn't show the progress update during TFTP based firmware update. The firmware upgrade completes successfully though.
8. LDAP user login may not work correctly, if user password encryption support is enabled.

## ***RR3 Features***

This list includes the features that are newly available in this rolling release, RR3.

1. DCMI 1.5 support
2. Remote images/media storage/redirection support over NFS and Samba share
3. TFTP based firmware upgrade support
4. Java based SOL support
5. ECC support on Pilot-III
6. PCI function1 support on Pilot-III

7. 32MB SPI flash support
8. MBMC fixes to support multiple IPMI instances
9. KVM features
  - MACRO (Pre-defined and user-defined) support
  - Stand-alone KVM/Java client support
  - Virtual keyboard with multi-language support
  - Localization support – No translations available for non-English languages
10. SD support as a removable USB drive to the host
11. UEFI based YAFUFLASH support

## RR2 Features

This list includes the features that were made available in the last rolling release, RR2.

1. A single-point services configuration support for all out-of-band services
  - Single point encryption control support
  - NCML library support and Services library API (To find what all available services)
  - Web page support
  - Backend support API in case of service state change
2. Audit messages' logging throughout the stack
3. **Project configuration based selective web pages display** – This is available at build time. Some of the web pages are necessary and are part of the stack by default, if the web support is included. Many of the configurable feature based pages are included only when the feature is included in the project configuration.
4. LIGTHTPD web server support
5. **Faster web interface support** – Data extensive pages (like sensors, SEL, dashboard) and their backend processing have been improved upon for faster page display.
6. MDS
  - a. PMCP Simulator
7. Circular SEL support
8. Logo and Copyright override support in a prebuilt image using a command line tool
9. Bonding enabled by default
10. BT interface support in IPMI
11. KVM client features
  - Mouse mode change support
  - Active KVM users list display
  - Power control support
12. Sections based firmware update support using YAFU applications

## Other Features

This section lists out the other features, available in this firmware release.

1. Single Point Authentication
  - PAM based single point authentication support for all out-of-band services
  - Web based interface to order the authentication services to be tried on SP upon user login

2. Configuration migration support (forward and backward) with provision to select specific files overrides
3. Automatic video recording on SP
  - Configurable IPMI events and LPC reset based triggers
  - Web interface to download the recorded video files
  - Web interface to playback the recorded video files
4. Extended Net-SNMP support
  - Integration with existing users
  - Broader functional coverage
5. MDS
  - Complete Windows based SP-X plug-in support
  - SP-X package based debugging support
6. Multiple E-Mail format support for IPMI alerts
7. Web based support for Self-Signed certificate generation, and simultaneous upload of SSL certificate and the key
8. Additional Cipher suites (HMAC-SHA256, HMAC-SHA-256-128) support in IPMI
9. HID access negotiation support across simultaneous KVM sessions
10. Network Bonding support
11. FQDN support
12. Multiple hardware boards (with different sensors) support using a single firmware image
13. DHCP renewal support upon network link up
14. Slave address configuration support for both IPMB buses
15. Web based Forced Network Link Settings support (MDIO/NCI)
  - AMI OEM commands based backend implementation
16. IPMI fixes to be able to run on any enabled network interfaces, including virtual network
  - Including fix to support eth1 when only one LAN channel exists
  - Supporting eth0 and eth1 to use same MAC address in case when only one LAN channel can be used at any given time
17. Fix Copyright and Licensing issues
18. Watchdog support
19. SMBus 2.0 support (AST only)
20. RSA/DSA keys support for SSL/SSH
21. Telnet support
22. Services Configuration Support
23. IPv6 support for RMCP/RMCP+
24. LDAP IPv6 support
25. LDAP group extension
26. Support for on the fly Certificate exchange without BMC Reset
27. Add and replace certificates without BMC reboot
28. Logging log audit for RMCP/RMCP+
29. Logging log audit for Web Server
30. Logging log audit for Telnet/Serial/SSH from RM
31. ARP storm test
32. SSH/Telnet timeout
33. DCMI1.1 compliance
34. Hardware based cipher support
35. System admin support

36. Frame Buffer support (AST2300 only)
37. Wide screen support (AST2300 only)
38. External video support (AST serial)
39. Encrypted Flash support
40. Secondary SMTP server configuration support
41. Time zone support in web GUI Event log page
42. VLAN configuration support
43. SDR File override during firmware upgrade
44. OPMA support
45. SOLSSH support
46. Host lock support during remote KVM redirection
47. Hardware cursor support during remote KVM redirection
48. KVM sharing during remote KVM redirection
49. JViewer: Video recording in JViewer without JMF(AVI stream playable in windows media player)
50. KVM console in the Web preview at Dashboard page
51. Mouse synchronization option for RHEL (less than 6.0) in webpage
52. Toolbar support added to display status of functions/options selected from menu/toolbar in remote client
53. Higher resolution support during remote KVM session- 1920\*1200
54. Multiple instances support for CD/FD/HD during remote media redirection

Feature	Notes
Each feature available and built as a separate package	MDS based packaging support
SoC and Hardware platform based stack customization	MDS based
Package based building during porting/development	MDS based
Generate customized stack base package depending upon features' selection	MDS based
SMTP-AUTH support	Login, Plain, and CRAM-MD5 only
SMTP (No AUTH required) support	
Secured way to reset the password, if forgotten	
DCMI support	
Infrastructure (Secondary IPMB) support for ME	
Optional IPMI commands support as per NM2.0 requirements	
Required support in BMC as per NM 2.0 specification	
AMD APML support	
Process/Thread managers for auto recovery of the features	
Support to restore factory default configuration	
NTP support	
SLP support	
IPMI 2.0 based management	<ul style="list-style-type: none"> <li>BMC stack with a full IPMI 2.0 implementation</li> <li>Many optional commands supported</li> <li>Faster response to the required commands over SMM interface</li> <li>Interfacing library (LIBIPMI)</li> </ul>



Overwriting standard IPMI command support	With OEM version of command handler
Dynamically pluggable transport layers (channels)	LAN, KCS, IPMB, USB, Serial (Basic, Terminal)
Customization to the existing core – Web and Sensors	MDS based
FRU generation support	MDS based
Customization to the existing core – Platform porting	<ul style="list-style-type: none"> <li>• MDS based PDK hooks level porting</li> <li>• Default IPMI configuration changes</li> </ul>
Extensive device support in MDS	<ul style="list-style-type: none"> <li>• PMBus</li> <li>• PECE device support</li> <li>• Satellite sensors support</li> </ul>
GUI based mechanism to add support for new devices	MDS based
Addition of OEM features to the existing core	MDS based
Customization /Addition of OEM features on the binary	MDS based
Fan monitoring and control	<ul style="list-style-type: none"> <li>• PWM and TACHOMETER support</li> </ul>
SOL (Text Console Redirection)	<ul style="list-style-type: none"> <li>• Support in IPMI stack for SOL to remotely access BIOS and text console</li> </ul>
Event Log and Alerting	<ul style="list-style-type: none"> <li>• Read Log events</li> <li>• Sensor readings</li> <li>• SNMP traps</li> <li>• E-Mail alerts</li> </ul>
Support for industry standard discovery methods	<ul style="list-style-type: none"> <li>• UPNP</li> <li>• DDNS</li> <li>• SLP</li> <li>• RMCP Ping</li> <li>• WS-DISCOVER</li> </ul>
Support for industry standard name services	<ul style="list-style-type: none"> <li>• NMBD</li> <li>• WINS</li> </ul>
Sophisticated User Management	<ul style="list-style-type: none"> <li>• IPMI based user management</li> <li>• Added security with SSL (HTTPS)</li> <li>• Multiple user permission level</li> <li>• Extended security algorithms and cipher suites support</li> </ul>
Remote Server Power Control	<ul style="list-style-type: none"> <li>• Server's power status report</li> <li>• Support for remotely power-cycle, power-down, power-up, reset the server</li> </ul>

Support for industry standard name services	DDNS
Support for privilege based Windows Active Directory	
Privilege based Open LDAP support	
RADIUS support	
Configurable Audit and System Log information	
IPv6 support	
Web server with HTML5/JS based web UI	<ul style="list-style-type: none"> <li>• Full configuration using web UI</li> <li>• Fail-safe firmware upgrade</li> <li>• Multi-language support in Web interface with English as the currently supported language</li> </ul>
PS/2 Keyboard & Mouse support, if available in SoC	Available on Pilot-II/Pilot-III boards
<ul style="list-style-type: none"> <li>• Remote KVM and vMedia support on Windows, Linux, and MAC OS client system</li> <li>• Remote KVM and vMedia support for 32-bit and 64-bit client system (Windows, Linux)</li> </ul>	<ul style="list-style-type: none"> <li>• Absolute and Relative mouse mode support</li> <li>• Support for all standard video resolutions</li> <li>• Simultaneous floppy and CD/ DVD redirection</li> <li>• Efficient USB 2.0-based CD/DVD redirection</li> <li>• Support for USB key (Emulation types: Floppy, Hard Disk)</li> <li>• Dynamic USB device attachment support using “Auto Attach” mode</li> <li>• Completely secured (Authenticated/Encrypted) remote KVM/vMedia</li> <li>• CD/Floppy/USB Key driver and ISO image</li> </ul>
Local Media/Virtual USB Flash support (SPI flash based)	<p>Virtual Flash support available on Pilot-III</p> <p>The framework allows us to extend the support for other hardware types like SD or RAMDISK.</p>
Secondary Boot SPI and Backup SPI support	Pilot-III Only
Command line firmware upgrade: YAFU	<ul style="list-style-type: none"> <li>• IPMI based firmware upgrade</li> <li>• Support in U-Boot mode as well</li> <li>• DOS, Windows, and Linux based</li> <li>• 32-bit and 64-bit OS</li> </ul>

	support
VMCLI (Command Line Virtual Media Redirection) Client	<ul style="list-style-type: none"> <li>• Windows, and Linux based</li> <li>• 32-bit and 64-bit OS support</li> </ul>
Line resilient support	
Bad password threshold support in IPMI	
Support for KCS non-communication interrupt	Pilot-III Only
VLAN support	
Dynamic USB media device configuration support	To support multiple virtual CD/DVD, Floppy or hard disks
Multiple BMC support	
SSH support	
New KVM features support	Zoom, Video recording, Local video locking, context menu, and support for multi-viewer console application
Complete web based PEF support as per IPMI	
KCS support in u-boot	
Three KCS interfaces support	
Hardware Watchdog support	
FRU page support	
SP stack level firewall support	
Virtual network USB interface (RNDIS/CDC) core support	This includes the USB core modules support only for USB based virtual network.
SMASHCLP support with multiple concurrent sessions	
CIM/WSMAN with CIMPDK and CIMSDK support	
software modules availability in spx package format	
Set MAC address support in IPMI	
Hardware based Encryption support	<ul style="list-style-type: none"> <li>• Pilot-III only</li> <li>• Standalone encryption libraries are available at this time.</li> </ul>
Basic Net-SNMP support	
Fail-safe (Backup) configuration support	
Dynamic (Web based) NCSI configuration support	