MEGARAC®
...a Generation Ahead

American Megatrends

Data Sheet

5/27/2008

MegaRAC® Development Studio

Integrated Development Environment for Server Management

Eclipse-based IDE

Integrated Web-development environment for system management design

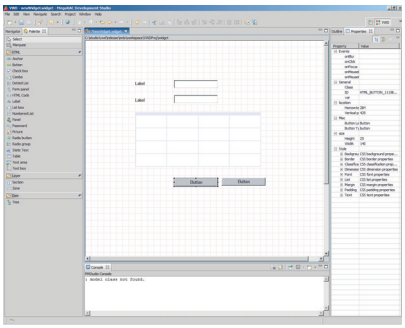
- Visual Web Developer (WVD) Plug-in
- PMCP Plug-in
- CIM SDK Plug-in

Highlights

- Graphical design of web pages and web-sites
- Web page generator automatically creates HTML and style files
- Native CIMOM Configuration
- SMASH CLP configuration
- WSMAN Resource Development and Configuration
- Easy graphical layout of platform sensors
- Can integrate Platform Development Kit

MEGARAC® DS DEVELOPMENT STUDIO

MegaRAC® Development Studio is a revolutionary Integrated Development Environment including powerful tools for platform porting of servers integrating MegaRAC latest generation Service Processors. DS is utilized for development and customization of the platform management structure and visual interface. Structured as a set of Eclipse Plug-ins compliant with the CIM model, MegaRAC DS enables OEMs to develop with ease the web management interface for their products, as well as port and customize the MegaRAC SP management structure for their platform according to advanced standards as CIM, SMASH and WSMAN.

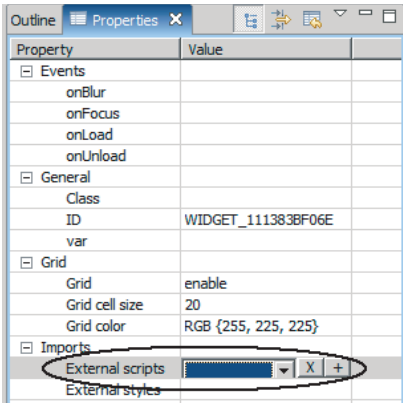


Graphical Interface Design

Visual Web Designer (VWD) provides a graphical interface and tools to model web pages and sites, generating related HTML, JavaScript and style sheet (CSS) code.

Developed as a collection of Eclipse plug-ins, VWD extends Eclipse views to create and navigate web site projects, configure web components and component properties, select controls from a palette and to report errors and warnings.

A graphical Layout Editor is used for creating a site layout using nested levels and regions.



PMCP Plug-ins

The IPMI PDK plug-in provides a graphical environment for designing the sensor layout with a drag-and-drop method, using existing schematic information. The graphical utility allows an OEM to "drag-n-drop" devices from a repository or library onto a workspace and connect the device pins to mirror the device connections as done on the motherboard. From this device map, the software creates sensor monitoring C code. The user can create a binary image that describes how to access the different sensors connected to the BMC. This virtually eliminates a lengthy porting process.

CIM SDK plug-in (on roadmap)

The CIM SDK helps OEM developers to easily expose their management data from the management card. The CIM repository provides a cross-platform toolkit enabling to browse OEM existing management data and add new management data based on the SMASH-CLP and WSMAN architecture.

Defined by the DMTF, CIM (Common Information Model) provides a common definition of management information for systems, networks, applications and services, and allows for vendor extensions. CIM's common definitions enable vendors to exchange semantically rich management data between systems.

The CIM SDK allows defining custom profiles and facilitates adding them into the SMASH or WSMAN infrastructure. OEMs can create their own classes to describe managed entities.

Data Sheet

Visual Web Designer Plug-ins

Project creation wizard
File creation Wizard
HTML
DOMAPI
Layout Graphical Editor (extension .ilayout) for creating site layouts
Level - Horizontal division
Region - Vertical Division
Page/Site layouts can use nested levels and regions
Palette View
Selection tool
File component drawers (files HTML, DOMAPI)
Property View
Import Javascript Libraries
Import web page from existing project
Export web page from existing project
Console View

CIM SDK Plug-in (on roadmap)

CIM Client
Provider Development Tool
WSMAN Resource Development Tool
SMASH Configuration
CIM Configuration
MOF Editor
Profile Development Tool
CIM SDK Wizards
CIM Class Explorer
CIM Class Properties
CIM Class Methods
CIM Class Hierarchy
Profile Explorer
WSMAN Resource Wizard
Provider Wizard
SMASH Command Wizard
SMASH Target Wizard
Profile File Wizard

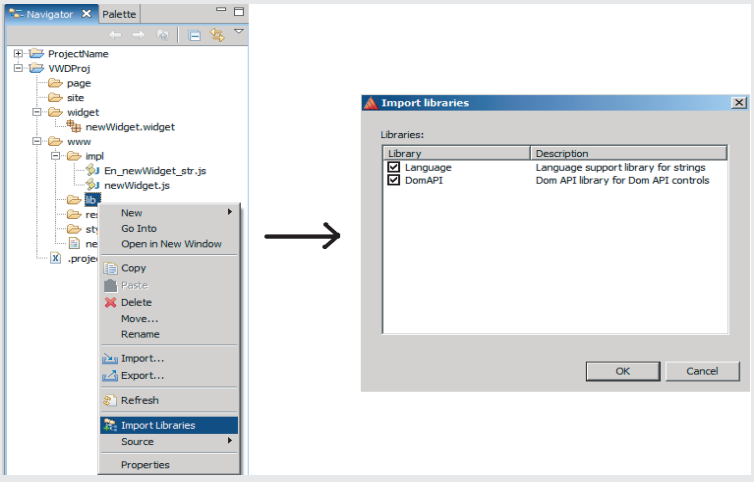
IPMI-PDK Plug-in

Easy to use GUI
Add /remove device to IPMI project by dragging and dropping
Device Description Files (DDFs) for adding future devices to library
Automatic SDR records creation without the need of user manual input
Sensor Monitoring and Device Control Information
GPIO Configuration
Select between GPIO alternate functions
Select direction of the GPIO
Select which devices are connected to what GPIO
Select not to use a GPIO (will not be configured)
Firmware Parameters Configuration
Device Support
- Add Micro-controller
- Add I2C Multiplexer
- Add I2C Controller
- Add Hardware Monitors (LM78, LM85 etc)
- Add LEDs, LCDs, Voltage, Temperature sensors, Fans etc.
- Create SDR Records
- Change Firmware Configuration
- Add FRU Information

Platform Development Kit

Integrate OEM commands
Implement platform specific actions, such as: blink LEDs, beep codes, toggle GPIOs, etc.
Over-ride existing IPMI commands
Add drivers for OEM devices, e.g. LCDs
Create interrupt handlers and associate them with a BMC interrupt pin
Customize hooks provided during BMC power-on to initialize OEM hardware.

IMPORTING JAVASCRIPT LIBRARIES



American Megatrends

American Megatrends Inc.
5555 Oakbrook Parkway, Suite 200,
Norcross, GA 30093 | t: 770.246.8600
Sales & Product Information
sales@ami.com | t: 800.828.9264
Technical Support
support@ami.com | t: 770.246.8645
www.ami.com