Point of Sale Subsystem

# UnifiedPOS

# Retail Peripheral Architecture
# IBM Defined Management Services

**Version 1.9.2**
**October 2006**

# Table of Contents

# IBM Defined Management Services

## Introduction

This document explains the high level design of the UnifiedPOS Management Services Subsystem and related components. This strategy conforms to the Common Information Model (CIM) from the Distributed Management Task Force (DMTF). The CIM mode for Retail devices has been proposed to UnifiedPOS Committee, and it is currently being reviewed by UnifiedPOS committee. The IBM management services for Retail devices is based off the CIM schema for Retail devices.

## UnifiedPOS Changes

To seamlessly support the integration of UnifiedPOS management services, some changes are required to the UnifiedPOS specification, as well as the device controls provided by members of the committee.

Each component, the control and the service will have the capability to expose the device to UnifiedPOS Management Services. A read/write Boolean property, **AllowManagement** at control, will allow the application to determine if the device should participate in systems management. The default value is true for **AllowManagement** property. The property is initialized at open time.

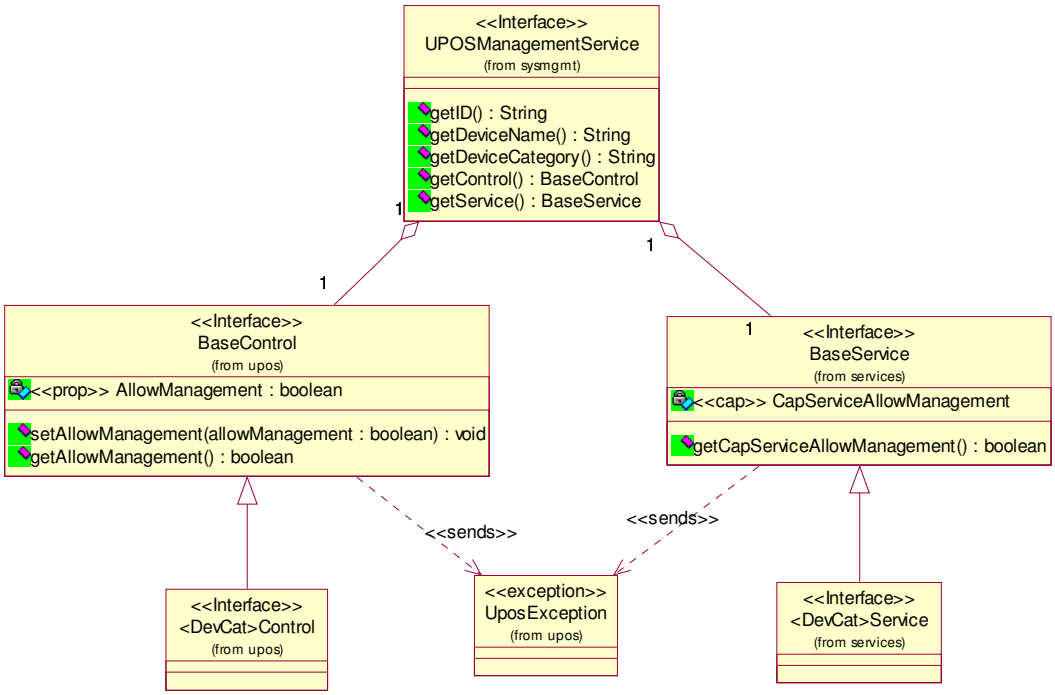
Also, each Service should implement the **UPOSManagementService** interface. The interface, from UPOS Management Services, it is what the component passes to Management Services when it registers with it. This interface serves as the connection point to Management Services and eventually the CIMOM. Registration occurs when the device is enabled, and un-register when disabled.

When the device is enabled, it should check with its corresponding control and service to determine if it will handle systems management, checking the **CapServiceSupportsManagement** capability and **AllowManagement** property. If **CapServiceSupportsManagement** is true, the Service will accept the responsibility to interact with systems management and register its own **UPOSManagementService** with UPOS Management Services and handle the systems management interface on behalf of the named device. When **CapServiceSupportsManagement** is false and the **AllowManagement** is true the control of the device will register the **UPOSManagementService** with UPOS Management Services. Finally when both are false the device will not participate in systems management.

**CapServiceSupportsManagement** capability has as default value false

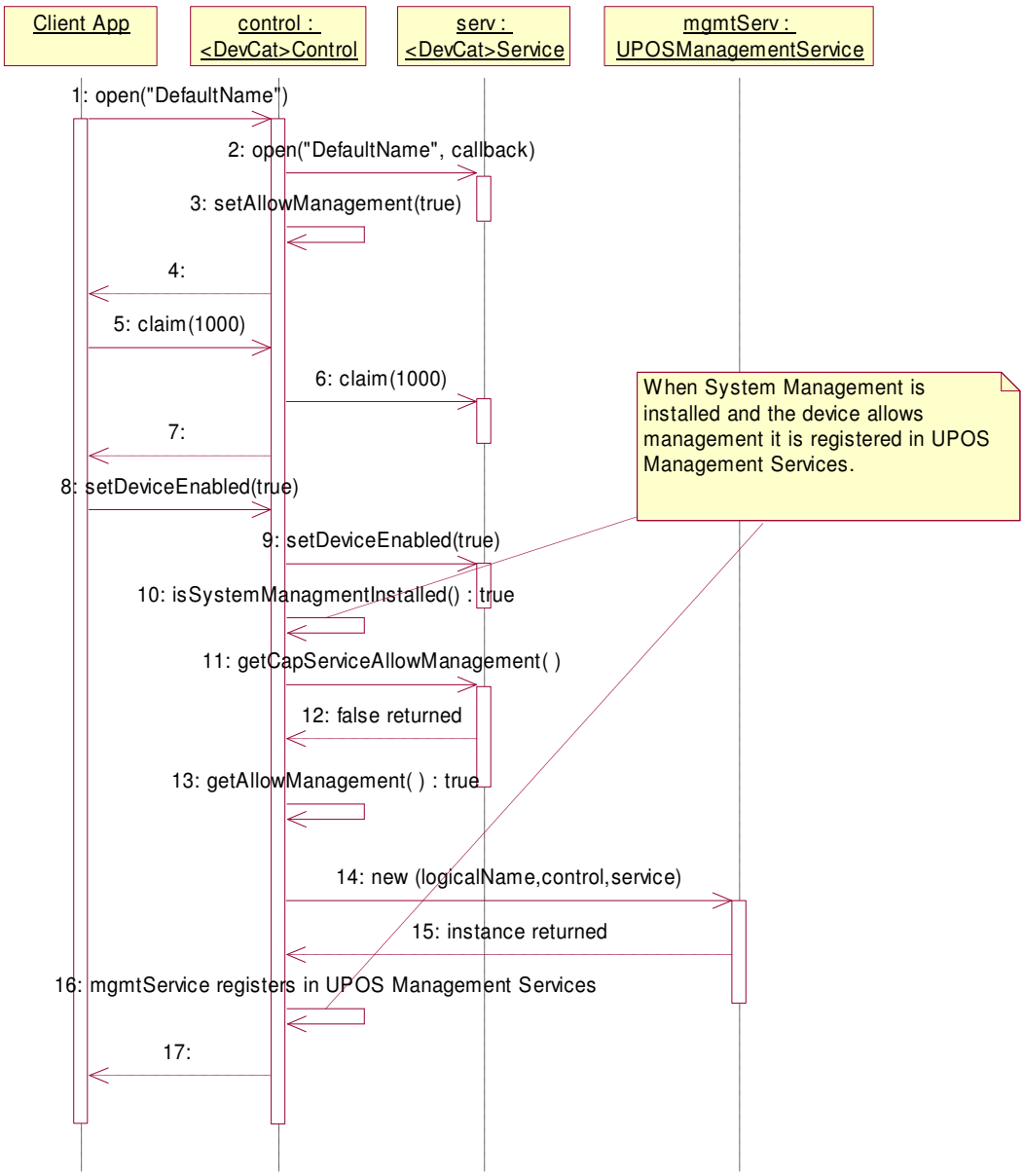
**The following sections describe the class diagrams and sequence diagrams created by IBM UnifiedPOS 1.9.1 release.**

# Controls Class Diagram for Management Service



# Device Registration Sequence Diagram

The following sequence diagram show the new sequences added to the Device Control to register the device with IBM Management Services at setDeviceEnabled() time.



# CIM ClassNames for UnifiedPOS Device Category Names

The correlations of UnifiedPOS programmatic names and CIM class names are defined in the following table

| UnifiedPOS Device Programmatic Names | CIM Class Name | Supported since |
|---|---|---|
| **BumpBar** | **UPOS_BumpBar** | |
| **CashChanger** | **UPOS_CashChanger** | |
| **CashDrawer** | **UPOS_CashDrawer** | **1.9.1** |
| **CAT** | **UPOS_CAT** | |
| **CheckScanner** | **UPOS_CheckScanner** | **1.9.1** |
| **CoinDispenser** | **UPOS_CoinDispenser** | |
| **FiscalPrinter** | **UPOS_FiscalPrinter** | |
| **HardTotals** | **UPOS_HardTotals** | **1.9.1** |
| **Keylock** | **UPOS_Keylock** | **1.9.1** |
| **LineDisplay** | **UPOS_LineDisplay** | **1.9.1** |
| **MICR** | **UPOS_MICR** | **1.9.1** |
| **MotionSensor** | **UPOS_MotionSensor** | **1.9.1** |
| **MSR** | **UPOS_MSR** | **1.9.1** |
| **PINPad** | **UPOS_PINPad** | |
| **PointCardRW** | **UPOS_PointCardRW** | |
| **POSKeyboard** | **UPOS_POSKeyboard** | **1.9.1** |
| **POSPower** | **UPOS_POSPower** | |
| **POSPrinter** | **UPOS_POSPrinter** | **1.9.1** |
| **RemoteOrderDisplay** | **UPOS_RemoteOrderDisplay** | |
| **Scale** | **UPOS_Scale** | **1.9.1** |
| **Scanner** | **UPOS_Scanner** | **1.9.1** |
| **SignatureCapture** | **UPOS_SignatureCapture** | |
| **SmartCardRW** | **UPOS_SmartCardRW** | |
| **ToneIndicator** | **UPOS_ToneIndicator** | **1.9.1** |

Refer to “<installdir>\sysmgmt\UPOSMgmtSrvProv.mof” file for a complete class definition

# IBM UnifiedPOS Provider for System Management

The IBM Provider act as driver and interface between the abstract world of the Common Information Model (CIM) and the UnifiedPOS device characteristics of Retail Hardware.

Following describes the providers supported by IBM:

## Instance Provider

An instance provider supplies instances of one or more given classes. For example, an instance provider can supply information regarding a POSPrinter device.

| CIM Method | WMI Equivalent | Supported by | |
|---|---|---|---|
| | | Windows | IRES 2 |
| GetInstance | GetObjectAsync | Yes | Yes |
| ModifyInstance | PutInstanceAsync | No | No |
| DeleteInstance | DeleteInstanceAsync | No | No |
| EnumerateInstances | CreateInstanceEnumAsync | Yes | Yes |
| EnumerateInstanceNames | | No | Yes |
| ExecQuery | ExecQueryAsync | No | No |

Method Provider

A method provider allows CIMOM access to the methods of a class.

| CIM Method | WMI Equivalent | Supported by | |
|---|---|---|---|
| | | Windows | IRES 2 |
| InvokeMethod | ExecMethodAsync | No | No |

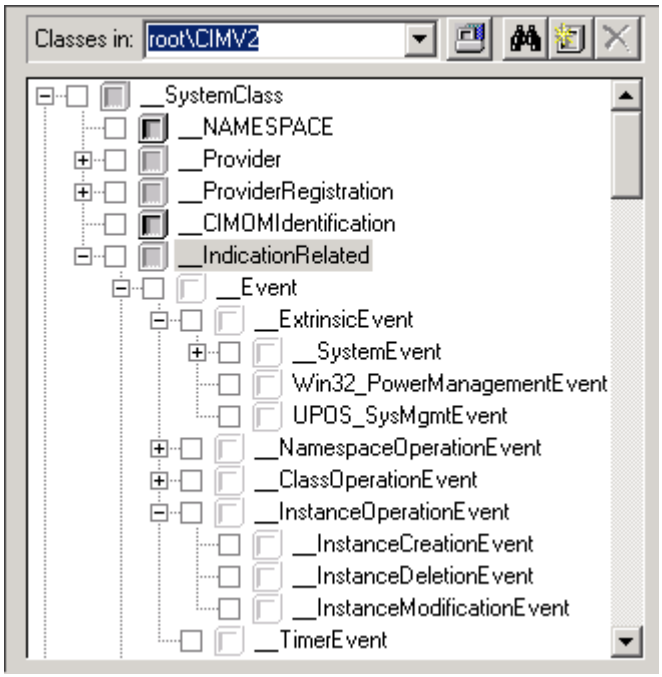
## Windows Event Provider (WMI)

An event provider is a COM object that supplies WMI notifications of intrinsic and extrinsic events. An intrinsic event reports an internal data change to WMI, while an extrinsic event reports a user-defined event not described by an intrinsic event.

For example, an event in response to changes, creation, or deletion of the UnifiedPOS_POSPrinter class would classify as an intrinsic event. An event that is generated on the basis of something other than the modification, creation or deletion of an existing WMI object is an extrinsic event.

IBM UnifiedPOS provider provides events for creation, deletion and modification of instances (all the supported UnifiedPOS Status Update Events(SUE) are reported as instance modification events). Even though these events could be classified as intrinsic events, they are implemented as extrinsic events, since they are instances of

`UPOS_SysMgmtEvent`, which is subclass of `__ExtrinsicEvent` (See figure below). Intrinsic events are generated by the WMI, rather than by the provider. Intrinsic events that concern to UPOS System Management are `__InstanceOperationEvent` and its subclasses. In order to receive intrinsic events, as a client application, it is necessary to subscribe a consumer with WMI for the intrinsic events specific to the CIM classes that are intended to monitor.




*Detail of class hierarchy around `__Event` system class.*

*(This class hierarchy is shown using CIM Studio from the WMI Tools, see reference ahead)*

## Linux Indication Provider (IRES 2)

The indication provider is not currently supported in Linux.

# Installation and Configuration

The following sections provide details regarding the installation and configuration of IBM UnifiedPOS system management support on Windows and IBM Retail Environment for SUSE LINUX (IRES) V2.

Note:

The system management support is currently provided only on **JavaPOS**.

# Installation on Windows

During IBM UnifiedPOS installation, a check box is option is provided for selecting system management support. When this option is selected, the installation automatically installs and configures necessary system management components. However, the system management has dependencies on the existence of Windows core components that support system management.

### Windows WMI Component

The Microsoft WMI components are required to run the IBM UnifiedPOS Management Services on Windows. The WMI component is typically a part of Windows OS, and IBM installation does not install Microsoft WMI components during installation. In case it is missing, it can be downloaded from:

http://www.microsoft.com/downloads/details.aspx?familyid=013BB284-3946-44A9-AC3C-BF2A569EAA72&displaylang=en

Optionally, the WMI components can be obtained by installing the Microsoft .NET component.

### Installation on Windows 2000:

On Windows 2000, it is highly recommended that the WMI component is properly installed from one of the above methods. Otherwise you may see some failures when you try to enumerate POS devices in WMI due to missing dlls.

### Validating System management:

The WMI Tool can be used to verify the system management properties of Retail Devices. The tool can be downloaded from.

http://www.microsoft.com/downloads/details.aspx?FamilyID=6430f853-1120-48db-8cc5-f2abdc3ed314

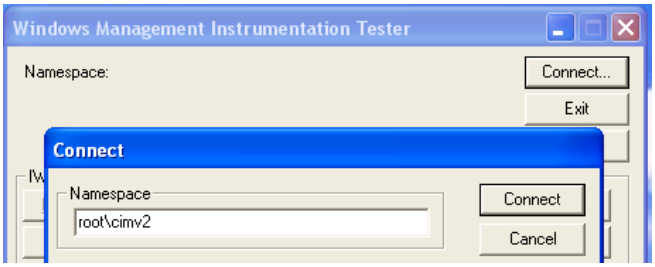
1. Install the WMI tool
2. Open, claim, enable JavaPOS device(s)
3. Launch WMI Tool: Start -> WMI CIM Studio
4. In Connect to namespace, enter “root\cimv2”, and log into CIM Studio
5. To view the properties, select the JavaPOS device on left pane, under UPOS_LogicalDevices

# Uninstallation on Windows

When the IBM UnifiedPOS software is uninstalled, it will automatically remove the system management components on Windows XP. However on Windows 2000, some manual clean is required as described below.

## Uninstallation on Windows 2000

IBM UnifiedPOS installer uses the "wmic" utility to delete the mof classes from WMI. Since this utility does not exist on Windows 2000, the mof classes must be manually deleted.




1.- Press Start-> run menu item.

2.- Type "wbemtest" at open field and press the "OK" button.

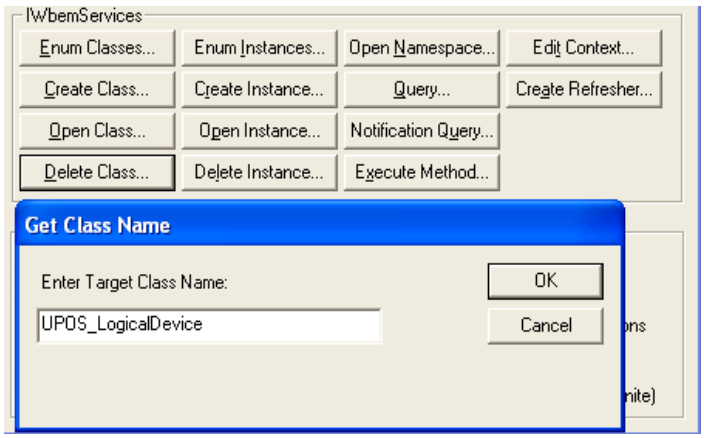
At "Windows Management Instrumentation Tester" window :

3.- Click on "connect…" button.

At Connect window:

4.- type "root\cimv2" at Namespace label.

5.- Click "connect" button.




6. Click on "Delete Class…"

At "Get Class Name" window

7.- Type "UPOS_LogicalDevice" and click "OK" button.

8.- This will delete the parent "UPOS_LogicalDevice" class and all its children classes from WMI.

9.- Repeat 6 and 7 steps for "UPOS_DecoupledProvider" and "UPOS_SysMgmtEvent" classes.

# Installation on IBM Retail Environment for SUSE LINUX (IRES)

The IRES 2 Image Builder Tool currently does not support the automatic installation of system management components. The system management components therefore must be installed manually. Below is a brief description of system management components along with installation instruction on branch server and client images.

Note that the system management support on IRES 2.1.2 and IRES 2.1.3 is provided on use as-is basis.

## IBM JavaPOS System Management Components:

The IBM UnifiedPOS support for System Management comprises of the following components:

rpms:

posIBM_sblim-cmpi-upos-client-<version>.i586.rpm

posIBM_sblim-cmpi-upos-server-<version>.i586.rpm

posIBM_XML4C-5.4.6-1.i586.rpm

jar files:

jpos1911.jar.sysmgmt:

jpos_sysmgmt.jar.sysmgmt

These files will be located in /opt/ibm/javapos/lib directory after installing the ibm-javapos-<version>.i386.rpm.

Pre-requisites:

- The UnifiedPOS system management support leverages the existing IRES system management infrastructure. Therefore the core IRES system management rpms, such as IBM_sblim, must be installed on branch server as well as clients.
- The Pegasus CIM server rpm must be installed on branch server. The Pegasus version tested is 2.4.1
- A working IBM JavaPOS configuration. In other words, the application should be able open/claim/enable JavaPOS devices.

## On the Branch Server:

- Ensure that the Pegasus CIM sever is running. You can check the status by issuing "/etc/init.d/tog-pegasus status". If server is not running, you can start the server by issuing "/etc/init.d/tog-pegasus restart" command
- Install "posIBM_sblim-cmpi-upos-server-<version>.i586.rpm" on the branch server

## On Client Images:

The following steps provide instructions for installation for the client image, assuming you are using IRES Image Builder tool.

1. Include the following rpms for the clint image
    a. "posIBM_sblim-cmpi-upos-client-<version>.i586.rpm"
    b. "posIBM_XML4C-5.4.6-1.i586.rpm"
2. On the client image, enable the JavaPOS Controls for system management. This can be done through copy operation on IRES Image Builder tool as follows:
    a. Select Configuration on left pane
    b. Under Options, click on Applications
    c. Click on Command
    d. In the text field on the right side, add the following copy commands
        - cp /opt/ibm/javapos/lib/jpos1911.jar.sysmgmt /opt/ibm/javapos/lib/jpos1911.jar
        - cp /opt/ibm/javapos/lib/jpos_sysmgmt.jar.sysmgmt /opt/ibm/javapos/lib/jpos_sysmgmt.jar
    e. Enable check box "Automatically Run Command"
    f. At the bottom, select "Run As" root

## Validating System management:

1. On Server: Verify that the Pegasus CIM server is running on branch server:
    - Check status: /etc/init.d/tog-pegasus status
    - Start server: /etc/init.d/tog-pegasus restart
2. On Server: Verify the /etc/hosts file has an entry for every client machine.
    - e.g. "1 <ip address>" where "1" is the <namespace-name> used by CLI command in step five.
3. On Client: Verify that the remote cmpi daemon is running on client
    - Check status: /etc/init.d/cmpird status
    - Start server: /etc/init.d/cmpird start
4. On client, open/claim/enable one or more devices using a JavaPOS application.
5. On server side, you can view the system management properties for a given device by issuing Pegasus CLI command. For example, to retrieve POSPrinter device properties.
    - cd /opt/tog-pegasus/bin
    - ./CLI ei -n root/posClient<namespace-name> UPOS_POSPrinter -u root -p <root password>

# Uninstallation on IBM Retail Environment for SUSE LINUX (IRES)

When the server rpm "posIBM_sblim-cmpi-upos-server-<version>.i586.rpm" is removed, it fails to clean up the UPOS Classes it created. Threfore the UPOS classes can be manually deleted as follows:

- cd /opt/var/tog-pegasus/repository/root#cimv2/classes
- For each UPOS class in the above directory, issue the following command to delete the class completely
  - /opt/tog-pegasus/bin/CLI –n root/cimv2 –u root –p <password> dc <UPOS_ClassName>.

# System Management Configuration File

To customize some of the system management functions, several properties are defined in sysmgmt.properties file. The details are described below.

File Name: systemgmt.properties

Location : <install directory>\sysmgmt directory (Windows)
/opt/ibm/javapos/etc (IRES)

**Property:** provider.eventSocket.Port

**Default value:** 42114

**Description:** Port number used by Windows Event CIM Provider.

**Property:** provider.response.timeout

**Default value:** 30000

**Description:** Timeout value from CIM Provider to UPOS Management Services..

**Property:** upos.requestSocket.Port

**Default value:** 42115

**Description:** Port number used by UPOS Management Services.

**Property:** provider.eventSocket.IP

**Default value:** 127.0.0.1

**Description:** Destination IP address for event UDP datagrams. Typically this would be the local machine, however due to limitations in IRES implementation, it might be necessary to report events directly to the Pegasus Server.

**Property:** provider.maxQueryThreads

**Default value:** 10

**Description:** Maximum number of threads that are allowed to connect to the Java drivers at the same time.

# Problem Determination

The IBM JavaPOS provides facility to gather trace information for JavaPOS and Provider components. You can selectively enable/disable traces as follows.

## Enabling Java Trace:

1. Go to IBM JavaPOS properties directory and open the "jutil.properties", for windows is "c:\pos\IBMJPOS\CONFIG" and for IRES is "/opt/ibm/javapos/etc"
2. Turn on the "com.ibm.jutil.tracing.TurnOnAllNamedTracers=ON" trace.
3. The location for these files is "<HOME>/.ibmjpos" where <HOME> is the absolute path to the user's home directory.

## Enabling Provider logging:

1. Create a system environment variable called = "UPOS_SYSMGMT_LOG" with value="1"
   a. As an example for Windows XP:
      - Right click on "My Computer" and select properties
      - Select "advanced" tab.
      - Click on "Environment variables".
      - In "System variables" select "new".
      - At "variable name:" write "UPOS_SYSMGMT_LOG".
      - At "variable value:" write "1";
      - Click on "OK" at "New System variables" window.
      - Click on "OK" at "environment variables" window.
      - Click on "OK" at "System Properties" window.
   b. An example for Linux:
      - Edit the "/etc/profile" file
      - Add the following line "export UPOS_SYSMGMT_LOG=1"
      - Restart the system
2. The log file "UPOS_SysMgmt.log" will be created in "c:\pos\log" directory on Windows and "/var/log" directory on IRES.

## Notes on IRES

- Ensure that the pegasus server is running on branch server. You can restart, /etc/init.d/tog-pegasus restart
- On clients, ensure that the cmpird is running. You can restart by entering - /etc/init.d/cmpird start

# References

Documents referenced and utilized for the implementation of UnifiedPOS Management Services:


- UnifiedPOS Retail Peripheral Architecture Version 1.9 http://www.nrf-arts.org/
- Common Information Model Version 2.2 http://www.dmtf.org/standards/cim
- Common Information Model Schema 2.9 http://www.dmtf.org/standards/cim/cim_schema_v29
- Java WBEM Services 1.0 API http://wbemservices.sourceforge.net/javadoc/api/index.html
- JSR 48: WBEM Services Specification http://jcp.org/en/jsr/detail?id=48
- SBLIM Project http://sblim.sourceforge.net/index.html
- CMPI Specification v1.3 http://www.wbemsource.org/doc.tpl?CALLER=documents.tpl&dcat=&gdid=3712
- Pegasus site http://www.openpegasus.org
- CIM Schema for Retail Devices: CIM-UPOS(6).pdf http://www.nrf-arts.org/