

IBM Tivoli License Compliance Manager



# Readme File for Fix Pack 2.3.0-TIV-TLCM-FP0001

*Version 2.3*



IBM Tivoli License Compliance Manager



# Readme File for Fix Pack 2.3.0-TIV-TLCM-FP0001

*Version 2.3*

**Note**

Before using this information and the product it supports, read the information in "Notices" on page 31.

This edition applies to the fix pack 2.3.0-TIV-TLCM-FP0001 for version 2, release 3 of IBM Tivoli License Compliance Manager (program number 5724-D33).

IBM welcomes your comments.

Address your comments to:

IBM License Management Information Development  
Rome Tivoli Lab  
IBM Italia S.p.A.  
Via Sciangai, 53  
00144 Rome  
Italy

Fax Number: (+39) 06 5966 2077

Internet ID: LMPUBS@IT.IBM.COM

When you send information to IBM, you grant IBM a non-exclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© Copyright International Business Machines Corporation 2004, 2007. All rights reserved.

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

---

# Contents

<b>Readme File for fix pack 2.3.0–TIV-TLCM–FP0001</b> . . . . .	<b>1</b>
About this fix pack . . . . .	1
Hardware inventory collection . . . . .	1
Agent installation size check. . . . .	23
Support for partitioning technologies. . . . .	23
Product fixes . . . . .	23
Documentation fix history . . . . .	25
Backward compatibility . . . . .	27
Other changes as a result of this fix pack . . . . .	27
Installation, migration, upgrade, and configuration information . . . . .	28
Hardware and software requirements. . . . .	28
Installing the fix pack . . . . .	28
Upgrading agents . . . . .	30
Reverting to the previous version . . . . .	30
<b>Notices</b> . . . . .	<b>31</b>
Trademarks . . . . .	32



---

## Readme File for fix pack 2.3.0–TIV-TLCM–FP0001

This readme provides important information about the fix pack 2.3.0-TIV-TLCM-FP0001 for IBM® Tivoli License Compliance Manager, version 2.3. This readme is the most current information for the fix pack and takes precedence over all other documentation.

*Please review this readme thoroughly before installing or using this fix pack.*

This readme includes the following topics:

- “About this fix pack”
- “Installation, migration, upgrade, and configuration information” on page 28

**Note:** To install the fix pack, you must be logged on as Administrator (on Windows®) or root (on UNIX®) to the computer where Tivoli License Compliance Manager is installed .

---

### About this fix pack

Fix pack 2.3.0-TIV-TLCM-FP0001 includes support for the new Hardware Inventory scan functionality and for additional agent platforms. It also includes fixes for reported APARs and defects.

This section includes the following topics:

- New functionality. See “Hardware inventory collection.”
- “Agent installation size check” on page 23
- “Support for partitioning technologies” on page 23
- “Product fixes” on page 23
- “Problems and workarounds” on page 25
- “Documentation fix history” on page 25
- “Backward compatibility” on page 27
- “Other changes as a result of this fix pack” on page 27

### Hardware inventory collection

Hardware inventory collection is an optional feature that enables Tivoli License Compliance Manager to collect and store hardware configuration information for hardware reconciliation processing by IBM Maximo Asset Management for IT.

You can use commands, available from the administration command line interface, to perform the following tasks:

- Enable the hardware inventory scan feature. This feature is disabled by default. See “Enabling the hardware inventory scan feature” on page 2.
- Define rules for the generation of a Globally Unique Identifier (GUID) for each monitored computer. Use of a GUID simplifies the reconciliation of hardware information with the system to which it relates. See “Defining the rule for generating GUIDs” on page 2.
- Configure hardware inventory scanning. This includes scheduling scans and determining whether to collect information in virtualized environments. See “Configuring the hardware inventory scan” on page 3.

- Check the current values for the hardware inventory scan configuration. See “Checking the configuration” on page 5.
- Check the status of hardware inventory scans. See “Checking the scan status” on page 5.

**Notes:**

1. You cannot activate hardware inventory collection if you are using either Tivoli License Compliance Manager for IBM software or IBM Tivoli® Configuration Manager License Manager extension.
2. Hardware inventory scanning is not implemented on i5/OS® platforms or on Linux® platforms running on zSeries®.

An initial hardware scan is automatically performed as soon as the agent is deployed. Changes that imply a migration of the agent, for example changing disks or upgrading the CPU do not trigger an automatic scan. To obtain consistently updated information about changes in hardware configuration, you must schedule scans to run on a regular basis or request a one-off scan after you have made changes.

To issue any of these commands, start the administration server command line interface, as follows:

**Windows**

Open the **Start** menu and click **Tivoli License Compliance Manager >Admin CLI**.

**UNIX**

1. Open a shell window and change to the directory:  
`<INSTALL_DIR>/admin/cli`
2. Enter the command:  
`./tlmcli`

**Enabling the hardware inventory scan feature**

Following installation of the fix pack, the hardware inventory scan feature is disabled by default. You can enable the feature using the **hwscanenable** command.

**Command syntax**

**hwscanenable** {-d | -e}

To enable the feature, issue the command:

**hwscanenable -e**

**Defining the rule for generating GUIDs**

You can configure Tivoli License Compliance Manager to generate a GUID for each monitored computer using a specified rule. The following options are available:

**None** Do not generate a GUID

**CSPProduct**

Generate a GUID by concatenating the values for manufacturer, model, and serial number.

**PrimaryMACAddress**

Generate a GUID using the Media Access Control (MAC) address of the first network card discovered on the computer.

**Note:** Neither method is able to generate unique GUIDs for systems running inside virtual containers or LPARs.



Use the **hwscanguidrule** command to set rule for generating a GUID. The default value is **None**.

#### Command syntax

```
hwscanguidrule {None | CSProduct | PrimaryMACAddress}
```

For example, to set the rule to CSProduct, , issue the command:

```
hwscanguidrule CSProduct
```

**Note:** Any change you make to the rule takes effect the next time a hardware scan is performed on each agent.

## Configuring the hardware inventory scan

Configuration of the hardware inventory scan includes specifying whether agents running in virtualized environments are to perform the scan and defining the scan schedule. Both types of configuration can be applied to all the agents for a specified organization or limited to those within a specified division of the organization.

#### Command syntax

The command for configuring hardware inventory scans is **hwscanconf**. It has the following syntax:

```
hwscanconf {-organization | -division} {-n <name> | -i <ID>} [-S  
<start_time> -N {1 | 7 | 30} -R <repetitions>] [{-e | -d}]
```

where:

#### **-organization**

specifies that the type of target for which the configuration is being defined is an organization. The organization name is specified using the **-n** parameter.

#### **-division**

specifies that the type of target for which the configuration is being defined is a division. The division name is specified using the **-n** parameter. If you have multiple organizations that include divisions with the same name, you can uniquely identify the division by specifying the ID, using the **-i** parameter.

#### **-n name**

specifies the name of the organization or division.

**-i ID** specifies the unique ID of the division.

#### **-S start\_time**

Specifies the date and time when the first or only occurrence of a hardware scan is start. The format *start\_time* must be YYYY-MM-DD-hh.mm.ss. When the command is used to schedule a hardware scan, this parameter is required.

**-N** Specifies the number of days included in the repeating period for a regular scan. Possible values are 1, 7, and 30. This parameter can only be specified if **-S** and **-R** are also specified.

#### **-R repetitions**

specifies the number repeating period that separate consecutive occurrences of the scan. The length of each repeating period is defined by the **-N** parameter. This parameter can only be specified if **-S** and **-N** are also specified.

- e specifies that the hardware scan is enabled in virtualized environments.
- d specifies that the hardware scan is disabled in virtualized environments.

### Specifying the rules for running the hardware scan in virtualized environments

Once the hardware inventory scan feature is enabled, the default configuration enables collection of information from both physical and virtual environments. However, when the hardware scan runs in a virtualized environment, it collects information related to the virtual machine rather than information about the physical hardware on which the virtual machine is running and this information is not required by IBM Maximo Asset Management for IT. You can collect and store this information for your own use or you can use the configuration command to define rules that disable the scan on agents running in virtualized environments.

The following supported environments are considered to be virtualized:

- HP Integrity VM guests
- HP vPar
- VMware guests
- Microsoft® Virtual Server guests
- Solaris Containers (Zones) guests

**Note:** LPAR is not treated as a virtualized environment.

To specify that agents of the Sales division that are running in virtualized environments are not to perform the scan, issue the following command:

**hwscanconf -division -n Sales -d**

If your environment includes multiple organizations and more than one organization has a Sales division, a message is shown, listing each Sales division with the name of the organization to which it belongs and its unique division ID. Reissue the command, using the division ID, as follows:

**hwscanconf -division -i <ID> -d**

To reenable scanning of virtualized environments, issue the command replacing the **-d** parameter with **-e**.

### Scheduling hardware inventory scans

Using the configuration command, you can schedule a single hardware inventory scan or set up a series of regularly repeating scans.

To set up the schedule, you must specify the date and time on which the first or only scan is to be performed. The scheduled time is the agent system time at which the scan is to be completed.

When setting up a repeating schedule you must also define the frequency of scans.

To schedule a single occurrence of a hardware inventory scan for agents in the Sales division, at the agent system time of midnight on 1st June 2007, issue the following command:

**hwscanconf -division -n Sales -S 2007-06-01-00.00.00**

Scans can be scheduled to repeat at regular intervals that are multiples of 1, 7, or 30 days. To schedule a series of hardware inventory scans for agents in the Sales division, to start at agent system time of midnight on 1st June 2007 and repeat every 4 weeks, issue the following command:

```
hwscanconf - division -n Sales -S 2007-06-01-00.00.00 -N 7 -R 4
```

If there is more than one Sales division, the command fails. Reissue the command, specifying **-i** parameter and the division ID.

## Checking the configuration

You can use the **hwscanshow** report to request a report of the rules and schedules you created using the **hwscanconf** command. The report can include an entire organization or can be limited to a specified division. It can be produced in either XML or CSV format.

### Command syntax

The **hwscanshow** command has the following syntax:

```
hwscanshow <format> <filename> {-organization | -division} {-n <name> | -i <ID>}
```

where:

*format* is the format in which the report is to be produced. Possible values are CSV and XML.

*filename* is the name of the output file to be produced.

#### **-organization**

specifies that the type of target for which the configuration is being shown is an organization. The organization name is specified using the **-n** parameter.

#### **-division**

specifies that the type of target for which the configuration is being shown is a division. The division name is specified using the **-n** parameter. If you have multiple organizations that include divisions with the same name, you can uniquely identify the division by specifying the ID, using the **-i** parameter.

#### **-n name**

specifies the name of the organization or division.

**-i ID** specifies the unique ID of the division.

## Checking the hardware inventory scan configuration for a specified organization

To produce a report, in XML format, of the rules and schedules set up for the SafeBank organization, issue the following command:

```
hwscanshow XML invrules.xml -organization -n SafeBank
```

## Checking the scan status

You can use the **hwscanstatus** command to request a report detailing when the last scan was scheduled and when the scan was actually last performed for each agent. The report can include agents for an entire organization or can be limited to a specified division. It can be produced in either XML or CSV format. You can request that the report include only those agents where the last scheduled scan was not performed, that is, agents where the last scan time is earlier than the last scheduled scan time.

## Command syntax

The **hwscanstatus** command has the following syntax:

```
hwscanstatus <format> <filename> [-p] {-organization | -division} {-n  
<name> | -i <ID>}
```

where:

*format* is the format in which the report is to be produced. Possible values are CSV and XML.

*filename* is the name of the output file to be produced.

**-p** If specified, the report includes only agents where the last scheduled scan was not performed.

**-organization** specifies that the type of target for which the status is being shown is an organization. The organization name is specified using the **-n** parameter.

**-division** specifies that the type of target for which the status is being shown is a division. The division name is specified using the **-n** parameter. If you have multiple organizations that include divisions with the same name, you can uniquely identify the division by specifying the ID, using the **-i** parameter.

**-n name** specifies the name of the organization or division.

**-i ID** specifies the unique ID of the division.

## Checking the hardware inventory scan status for a specified organization

To produce a report, in CSV format, of the agents in the SafeBank organization that did not successfully perform the last scheduled hardware inventory scan, issue the following command:

```
hwscanstatus CSV hwinvstatus.csv -p -organization -n SafeBank
```

## Database changes

The hardware inventory scan feature adds the following tables to the administration server database:

- “The TLMHW.COMPONENT table” on page 7: Stores common system information about monitored systems.
- “The TLMHW.COMPUTER table” on page 8: Stores common system information about monitored systems.
- “The TLMHW.COMPUTER\_SYS\_MEM table” on page 9: Stores information about installed memory on monitored systems.
- “The TLMHW.HDISK table” on page 10: Stores information about storage devices on monitored systems.
- “The TLMHW.INST\_MODEM table” on page 10: Stores information about modems installed on monitored systems.
- “The TLMHW.INST\_MOUSE table” on page 11: Stores information about pointing devices on monitored systems.
- “The TLMHW.INST\_PARTITION table” on page 11: Stores information about disk partitions on monitored systems.
- “The TLMHW.INST\_PRINTER table” on page 11: Stores information about printers attached to monitored systems.

- “The TLMHW.INST\_PROCESSOR table” on page 12: Stores information about processors on monitored systems.
- “The TLMHW.INST\_SMBIOS\_DATA table” on page 12: Stores general SMBIOS information for monitored systems.
- “The TLMHW.INST\_USB\_DEV table” on page 13: Stores information about USB devices on monitored systems.
- “The TLMHW.INST\_VID\_CARD table” on page 13: Stores information about video cards on monitored systems.
- “The TLMHW.IP\_ADDR table” on page 13: Stores information about IP addresses on monitored systems.
- “The TLMHW.IPX\_ADDR table” on page 14: Stores information about IPX addresses on monitored systems.
- “The TLMHW.MEM\_MODULES table” on page 14: Stores information about all memory modules installed in a monitored system.
- “The TLMHW.MODEM table” on page 15: Stores the details for types and models of modem.
- “The TLMHW.MONITOR table” on page 15: Stores information about all monitors connected to monitored systems.
- “The TLMHW.MOUSE table” on page 16: Stores the details for types and models of pointing device.
- “The TLMHW.NET\_ADAPTER table” on page 16: Stores information about network adapters on monitored systems.
- “The TLMHW.PCI\_DEV table” on page 17: Stores information about peripheral component interconnect (PCI) devices on monitored systems.
- “The TLMHW.PC\_SYS\_PARAMS table” on page 17: Stores BIOS information for monitored PCs.
- “The TLMHW.PRINTER table” on page 18: Stores the details of types and models of printer.
- “The TLMHW.PROCESSOR table” on page 18: Stores the details of types and models of processor.
- “The TLMHW.SMBIOS\_SYS\_DATA table” on page 20: Stores information SMBIOS configurations.
- “The TLMHW.STORAGE\_DEV table” on page 20: Stores information about storage devices on monitored systems.
- “The TLMHW.UNIX\_SYS\_PARAMS table” on page 21: Stores system parameter information for monitored UNIX systems.
- “The TLMHW.USB\_DEV table” on page 21: Stores information about types of USB device.
- “The TLMHW.VID\_CARD table” on page 22: Stores information about types of video card.

Additions have been made to several tables in the administration server database to support the hardware inventory scan feature. See “Changed tables” on page 22.

**The TLMHW.COMPONENT table:** Stores information about hardware components.

Columns	Description	Type	Null
ID	The component ID	bigint	no
MODEL	The component model.	varchar (128)	yes

Columns	Description	Type	Null
SERIAL_NUMBER	The serial number of the component.	varchar (64)	yes
MANUFACTURER	The manufacturer of the component	varchar (64)	yes
VERSION	The component version.	varchar (64)	yes
TYPE	The component type.	varchar (64)	yes
MODEL_CLASS	The model class of the component	varchar (64)	yes
CREATE_TIME	Date and time when the entry was added to the database.	timestamp	no
UPDATE_TIME	Date and time when the entry was last updated.	timestamp	no

**The TLMHW.COMPUTER table:** Stores common information about a computer system. One entry exists for each system scanned.

Table 1. The TLMHW.COMPUTER table

Columns	Description	Type	Null
COMPUTER_SYS_ID	Unique identifier of the system.	bigint	no
COMPUTER_SCANTIME	The date and time of the most recent hardware scan in Greenwich Mean Time.	timestamp	yes
COMPUTER_MODEL	The model of the system.	varchar (128)	yes
COMPUTER_BOOT_TIME	The date and time that the system was last restarted.	timestamp	yes
COMUTER_ALIAS	The host name of the system.	varchar (255)	yes
SYS_SER_NUM	The serial number of the system.	varchar (64)	yes
OS_NAME	The specific operating system that is installed, for example Windows Server 2003 Standard Edition.	varchar (128)	yes
OS_TYPE	The type of operating system that is installed, for example Windows Server 2003.	varchar (32)	yes
OS_MAJOR_VERS	The major version of the operating system.	integer	yes
OS_MINOR_VERS	The minor version of the operating system.	integer	yes
OS_SUB_VERS	The sub-version of the operating system.	varchar (32)	yes
OS_INST_DATE	The date when the operating system was installed.	varchar (32)	yes
REGISTERED_OWNER	The name of the owner of the system	varchar (255)	yes
REGISTERED_ORG	The name of the organization to which the system belongs.	varchar (255)	yes
KEYBOARD_TYPE	A description of the type of keyboard assigned to the system.	varchar (255)	yes
FUNCTION_KEYS	The number of function keys on the keyboard.	integer	yes
TZ_LOCALE	The locale of the time zone in which the system is located.	varchar (32)	yes
TZ_NAME	The name of the time zone in which the system is located.	varchar (64)	yes

Table 1. The TLMHW.COMPUTER table (continued)

Columns	Description	Type	Null
TZ_DAYLIGHT_NAME	The name of the daylight saving time zone in which the system is located.	varchar (64)	yes
ON_SAVINGS_TIME	Indicates whether the system is on daylight saving time.	char (1)	yes
TZ_SECONDS	The number of seconds difference between the system time zone and Greenwich Mean Time.	integer	yes
TIME_DIRECTION	Indicates whether the time zone is early or later than Greenwich Mean Time.	char (1)	yes
OS_ARCH	The operating system architecture for the system.	varchar (24)	yes
OS_KERNEL_MODE	The operating system kernel mode	varchar (32)	yes
OS_LANG_VERS	The operating system language version identifies the default language for the system.	varchar (64)	yes
OS_LCID	The operating system locale identifier	varchar (64)	yes
CURRENT_LCID	The current locale identifier	varchar (64)	yes
VM_CHAIN_HASH	The hash of the virtual layer that hosts the agent.	char (24)	yes
VIRTUAL_COMP_TYPE	Indicates whether the monitored system is virtualized. Possible values are 0 (not virtualized) or 1 (virtualized).	smallint	no
COMPUTER_GUID	The Globally Unique Identifier (GUID) of the system.	char (32)	yes
COMPUTER_GUID_TYPE	Indicates the rule for generation of the GUID that was in use when the current information was collected. Possible values are:  <b>1</b> Concatenate the values for manufacturer, model, and serial number.  <b>3</b> Use the Media Access Control (MAC) address of the first network card discovered on the system.	integer	yes
CREATE_TIME	Date and time when the entry was added to the database.	timestamp	no
UPDATE_TIME	Date and time when the entry was last updated.	timestamp	no

**The TLMHW.COMPUTER\_SYS\_MEM table:** Stores information about the physical and virtual memory installed on a system. One entry exists for each system scanned.

Columns	Description	Type	Null
COMPUTER_SYS_ID	Unique identifier of the system.	bigint	no
PHYSICAL_TOTAL_KB	The total physical memory in KB.	bigint	yes
PHYSICAL_FREE_KB	The amount of free physical memory in KB	bigint	yes
TOTAL_PAGES	The total number of physical memory pages	bigint	yes
FREE_PAGES	The number of free physical memory pages.	bigint	yes



Columns	Description	Type	Null
PAGE_SIZE	The size of a page.	bigint	yes
VIRT_TOTAL_KB	The total amount of virtual memory in KB	bigint	yes
VIRT_FREE_KB	The amount of free virtual memory in KB.	bigint	yes
CREATE_TIME	The date and time when the entry was added to the database.	timestamp	no
UPDATE_TIME	Date and time when the entry was last updated.	timestamp	no

**The TLMHW.HDISK table:** Stores information about storage devices, for example hard disks and CD ROM on monitored systems. One record exists for each hard disk discovered on each system scanned.

Columns	Description	Type	Null
HDISK_ID	Unique identifier of the storage device.	varchar (32)	no
HDISK_CYLINDERS	The number of cylinders in the device.	integer	yes
HDISK_SECTORS	The number of sectors in the device.	integer	yes
HDISK_HEADS	The number of heads in the device.	integer	yes
HDISK_SIZE_MB	The size of the device in MB.	bigint	yes
CREATE_TIME	The date and time when the entry was added to the database.	timestamp	no
UPDATE_TIME	Date and time when the entry was last updated.	timestamp	no

**The TLMHW.INST\_MODEM table:** Stores information about the type of modem installed and the settings that relate to it. One record exists for each modem discovered on each system scanned.

Columns	Description	Type	Null
COMPUTER_SYS_ID	Unique identifier of the system.	bigint	no
MODEM_ID	The modem identifier.	varchar (32)	no
INST_MODEM_ID	Identifies the specific modem installed on the system.	integer	no
PORT	The port used by the modem.	varchar (16)	yes
PORT_SPEED	The speed of the port that the modem is using.	integer	yes
PORT_SETTINGS	The port settings of the modem.	varchar (16)	yes
USER_INIT	The user-specified initialization string for the modem.	varchar (128)	yes
CREATE_TIME	Date and time when the entry was added to the database.	timestamp	no
UPDATE_TIME	Date and time when the entry was last updated.	timestamp	no



**The TLMHW.INST\_MOUSE table:** Stores information about the type of pointing device installed and the settings that relate to it. One record exists for each pointing device discovered on each system scanned.

Columns	Description	Type	Null
COMPUTER_SYS_ID	Unique identifier of the system.	bigint	no
MOUSE_ID	The mouse identifier.	varchar (32)	no
INST_MOUSE_ID	Identifies the specific mouse installed on the system.	integer	no
CREATE_TIME	The date and time when the entry was added to the database.	timestamp	no
UPDATE_TIME	Date and time when the entry was last updated.	timestamp	no

**The TLMHW.INST\_PARTITION table:** Stores information about a disk partition on a drive on the system. One record exists for each partition for each system scanned.

Columns	Description	Type	Null
COMPUTER_SYS_ID	Unique identifier of the system.	bigint	no
FS_ACCESS_POINT	The location where the partition is mounted.	varchar (254)	no
DEV_NAME	The device name.	varchar (64)	yes
PARTITION_TYPE	Identifies the type of partition.	varchar (32)	yes
MEDIA_TYPE	The media type of the partition.	varchar (32)	yes
PHYSICAL_SIZE_KB	The physical size of the partition in KB	bigint	yes
FS_TYPE	The type of file system in the partition.	varchar (64)	yes
FS_MOUNT_POINT	The mount point of the file system.	varchar (254)	yes
FS_TOTAL_SIZE_KB	The total size of the file system in KB	bigint	yes
FS_FREE_SIZE_KB	The amount of free space in the file system in KB	bigint	yes
CREATE_TIME	Date and time when the entry was added to the database.	timestamp	no
UPDATE_TIME	Date and time when the entry was last updated.	timestamp	no

**The TLMHW.INST\_PRINTER table:** Stores information about the type of printer installed, the driver software, and port settings. One record exists for each printer for each system scanned.

Columns	Description	Type	Null
COMPUTER_SYS_ID	Unique identifier of the system.	bigint	no
PRINTER_ID	Identifies the type of printer.	varchar (32)	no
INST_PRINTER_ID	Identifies the specific printer.	integer	no
PRINTER_NAME	The name of the printer.	varchar (254)	yes
PRINTER_LOCATION	The physical location of the printer.	varchar (254)	yes

Columns	Description	Type	Null
PRINTER_IS_LOCAL	Specifies whether or not the printer is local.	char (1)	yes
DRV_NAME	The printer driver name.	varchar (254)	yes
DRV_VERS	The printer driver version.	varchar (254)	yes
PORT_NAME	The name of the printer port.	varchar (254)	yes
CREATE_TIME	Date and time when the entry was added to the database.	timestamp	no
UPDATE_TIME	Date and time when the entry was last updated.	timestamp	no

**The TLMHW.INST\_PROCESSOR table:** Stores information about the type of processor or processors installed. One record exists for each processor for each system scanned.

Columns	Description	Type	Null
COMPUTER_SYS_ID	Unique identifier of the system.	bigint	no
PROCESSOR_NUM	The processor number	integer	no
PROCESSOR_ID	Identifies the type of processor	varchar (32)	no
SER_NUM	The processor serial number.	varchar (64)	yes
PROCESSOR_BOARD	The number of processor boards.	integer	yes
PROCESSOR_MODULE	The number of processor modules.	integer	yes
IS_ENABLED	Indicates whether the processor is enabled.	char (1)	yes
CREATE_TIME	Date and time when the entry was added to the database.	timestamp	no
UPDATE_TIME	Date and time when the entry was last updated.	timestamp	no

**The TLMHW.INST\_SMBIOS\_DATA table:** Stores general SMBIOS information for monitored systems. One entry exists for each system scanned.

Columns	Description	Type	Null
COMPUTER_SYS_ID	Unique identifier of the system.	bigint	no
SMBIOS_ID	The SMBIOS ID	varchar (32)	no
BIOS_DATE	The date the system BIOS was created.	varchar (16)	yes
SYS_SER_NUM	The serial number of the system	varchar (64)	yes
SYS_UUID	The universal unique ID of the system	varchar (36)	yes
BOARD_SER_NUM	The serial number of the system board.	varchar (64)	yes
CASE_SER_NUM	The serial number of the case for the system.	varchar (64)	yes
CASE_ASSET_TAG	The asset tag number of the case for the system.	varchar (64)	yes
POWERON_PASSWORD	Determines whether the power on password is set.	varchar (64)	yes

Columns	Description	Type	Null
CREATE_TIME	Date and time when the entry was added to the database.	timestamp	no
UPDATE_TIME	Date and time when the entry was last updated.	timestamp	no

**The TLMHW.INST\_USB\_DEV table:** Stores information about the type of USB devices installed. One record exists for each USB device for each system scanned.

Columns	Description	Type	Null
COMPUTER_SYS_ID	Unique identifier of the system.	bigint	no
USB_ID	Identifies the type of USB device.	varchar (32)	no
INST_USB_ID	Identifies the specific USB device.	integer	no
HOST_CNTRL	The host controller for the USB device.	varchar (64)	yes
DEV_ADDR	The device address for the USB device.	varchar (16)	yes
SER_NUM	The serial number for the USB device.	varchar (64)	yes
PORT_NUM	The port number used by the USB device.	varchar (16)	yes
PARENT_ADDR	The parent address used by the USB device.	varchar (16)	yes
CREATE_TIME	Date and time when the entry was added to the database.	timestamp	no
UPDATE_TIME	Date and time when the entry was last updated.	timestamp	no

**The TLMHW.INST\_VID\_CARD table:** Stores information about the type of video card installed and the operating system settings. One record exists for each video card for each system scanned.

Columns	Description	Type	Null
COMPUTER_SYS_ID	Unique identifier of the system.	bigint	no
VID_CARD_ID	Identifies the type of video card.	varchar (32)	no
INST_VID_CARD_ID	Identifies the specific video card.	integer	no
VID_HORIZNTL_RES	The horizontal resolution setting of the installed video card.	integer	yes
VID_VERTICAL_RES	The vertical resolution setting of the installed video card.	integer	yes
VID_COLORS	The color setting of the installed video card.	integer	yes
CREATE_TIME	Date and time when the entry was added to the database.	timestamp	no
UPDATE_TIME	Date and time when the entry was last updated.	timestamp	no

**The TLMHW.IP\_ADDR table:** Stores information about the IP address for the monitored system. One record exists for each IP address for each system scanned.

Columns	Description	Type	Null
COMPUTER_SYS_ID	Unique identifier of the system.	bigint	no
IP_ADDR	The IP address of the system.	varchar (40)	no
INST_IPADR_ID	Identifies the specific IP address	integer	no
IP_HOSTNAME	The IP host name for the system.	varchar (64)	yes
IP_DOMAIN	The IP domain name for the system.	varchar (64)	yes
IP_SUBNET	The IP subnet for the system.	varchar (40)	yes
IP_GATEWAY	The IP gateway name for the system.	varchar (254)	yes
IP_PRIMARY_DNS	The primary domain name service (DNS) for the system.	varchar (40)	yes
IP_SECONDARY_DNS	The secondary DNS for the system.	varchar (40)	yes
IS_DHCP	Indicates whether the Dynamic Host Configuration Protocol (DHCP) is in use.	char (1)	yes
PERM_MAC_ADDRESS	The permanent MAC address of the system.	varchar (64)	yes
IPV6_ADDRESS	The IPV6 address of the system	varchar (64)	yes
CREATE_TIME	Date and time when the entry was added to the database.	timestamp	no
UPDATE_TIME	Date and time when the entry was last updated.	timestamp	no

**The TLMHW.IPX\_ADDR table:** Stores information about the IPX address for the monitored system. One record exists for each IPX address for each system scanned.

Columns	Description	Type	Null
COMPUTER_SYS_ID	Unique identifier of the system.	bigint	no
IPX_ADDR	The IPX address of the system.	varchar (40)	no
NET_NUM	The network number of the system.	varchar (40)	yes
NODE_ADDR	The node address of the system.	varchar (40)	yes
LINK_SPEED	The link speed of the system.	integer	yes
MAX_PACKET_SIZE	The maximum package size that the system can handle.	integer	yes
CREATE_TIME	Date and time when the entry was added to the database.	timestamp	no
UPDATE_TIME	Date and time when the entry was last updated.	timestamp	no

**The TLMHW.MEM\_MODULES table:** Stores the details of memory modules installed. One entry exists for each memory module for each system scanned.

Columns	Description	Type	Null
COMPUTER_SYS_ID	Unique identifier of the system.	bigint	no

Columns	Description	Type	Null
INST_MEM_ID	The index number for the memory module.	integer	no
MODULE_SIZE_MB	The size of the installed memory module in MB.	integer	yes
MAX_MODULE_SIZE	The maximum supported memory module size in MB.	integer	yes
SOCKET_NAME	The name of the socket in which the memory module is installed	varchar (24)	yes
PACKAGING	The physical packaging of the memory, such as single in-line memory module (SIMM) or dual in-line memory module (DIMM).	varchar (16)	yes
MEM_TYPE	The type of memory installed.	varchar (48)	yes
CREATE_TIME	Date and time when the entry was added to the database.	timestamp	no
UPDATE_TIME	Date and time when the entry was last updated.	timestamp	no

**The TLMHW.MODEM table:** Stores the details for one particular type and model of modem. It contains one record for each unique modem in use in the monitored environment.

Columns	Description	Type	Null
MODEM_ID	The modem identifier.	varchar (32)	no
MODEM_DESC	The description of the modem.	varchar (64)	no
MANUFACTURER	The manufacturer of the modem.	varchar (64)	yes
PROVIDER_NAME	The maker of the software driver for the modem.	varchar (32)	yes
MODEM_TYPE	The type of modem.	varchar (32)	yes
INF_FILE	The name of the description file for the modem driver.	varchar (32)	yes
INF_SECTION	The details of the driver file for the modem.	varchar (32)	yes
CREATE_TIME	Date and time when the entry was added to the database.	timestamp	no
UPDATE_TIME	Date and time when the entry was last updated.	timestamp	no

**The TLMHW.MONITOR table:** Stores the details of monitors connected to systems in the monitored environment. One entry exists for each monitor for each system scanned.

Columns	Description	Type	Null
COMPUTER_SYS_ID	Unique identifier of the system.	bigint	no
INST_MONITOR_ID	Identifier of the installed monitored.	integer	no
MANUFACT_CODE_NAME	Identifies the manufacturer of the monitor.	varchar (64)	yes
NAME	The name of the manufacturer	varchar (64)	yes

Columns	Description	Type	Null
WEEK_MANUFACTURED	The week in which the monitor was manufactured.	integer	yes
YEAR_MANUFACTURED	The year in which the monitor was manufactured.	integer	yes
SERIAL_NUMBER	The serial number of the monitor.	varchar (64)	yes
SIZE_INCHES	The size of the monitor in inches	integer	yes
MIN_V_SCAN_FREQ	The minimum vertical scan frequency of the monitor.	integer	yes
MAX_V_SCAN_FREQ	The maximum vertical scan frequency of the monitor.	integer	yes
MIN_H_SCAN_FREQ	The minimum horizontal scan frequency of the monitor.	integer	yes
MAX_H_SCAN_FREQ	The maximum horizontal scan frequency of the monitor.	integer	yes
PIXEL_CLOCK_MHZ	The rate at which pixels are drawn on the screen.	integer	yes
CREATE_TIME	Date and time when the entry was added to the database.	timestamp	no
UPDATE_TIME	Date and time when the entry was last updated.	timestamp	no

**The TLMHW.MOUSE table:** Stores the details for one particular type and model of pointing device. It contains one record for each unique pointing device in use in the monitored environment.

Columns	Description	Type	Null
MOUSE_ID	The mouse identifier.	varchar (32)	no
MOUSE_MODEL	The model of mouse.	varchar (254)	yes
MOUSE_TYPE	The type of pointing device for example mouse, trackball.	varchar (32)	yes
BUTTONS	The number of buttons on the mouse.	integer	yes
CREATE_TIME	Date and time when the entry was added to the database.	timestamp	no
UPDATE_TIME	Date and time when the entry was last updated.	timestamp	no

**The TLMHW.NET\_ADAPTER table:** Stores information about the network adapter installed on a system. One record exists for each network adapter for each system scanned.

Columns	Description	Type	Null
COMPUTER_SYS_ID	Unique identifier of the system.	bigint	no
INDEX	The net adapter index.	integer	no
PERM_MAC_ADDR	The permanent Media Access Control (MAC) address of the system.	varchar (64)	yes
CURRENT_ADDR	The current network address of the system.	varchar (64)	yes
ADAPTER_TYPE	The network adapter installed on the system.	varchar (32)	yes

Columns	Description	Type	Null
ADAPTER_MODEL	The model of the network adapter	varchar (254)	yes
MANUFACTURER	The manufacturer of the network adapter.	varchar (128)	yes
INST_DATE	Network adapter installation date.	varchar (32)	yes
CREATE_TIME	Date and time when the entry was added to the database.	timestamp	no
UPDATE_TIME	Date and time when the entry was last updated.	timestamp	no

**The TLMHW.PCI\_DEV table:** Stores information about a PCI device installed in or connected to a system. One record exists for each PCI device for each system scanned.

Columns	Description	Type	Null
COMPUTER_SYS_ID	Unique identifier of the system.	bigint	no
INST_PCI_ID	Identifies the installed peripheral component interconnect (PCI) device.	integer	no
PCI_DEV_NAME	The name of the PCI device.	varchar (180)	no
PCI_REVISION	The revision of the PCI device.	char (8)	yes
CREATE_TIME	Date and time when the entry was added to the database.	timestamp	no
UPDATE_TIME	Date and time when the entry was last updated.	timestamp	no

**The TLMHW.PC\_SYS\_PARAMS table:** Stores BIOS and other system information for a PC. One entry exists for each system scanned.

Columns	Description	Type	Null
COMPUTER_SYS_ID	Unique identifier of the system.	bigint	no
USER_NAME	The Windows user name for the system.	varchar (254)	yes
DOMAIN_NAME	The Windows domain name for the system.	varchar (128)	yes
WORKGROUP_NAME	The Windows workgroup name for the system.	varchar (128)	yes
BIOS_ID	Identifies the BIOS that is in use on the system.	varchar (128)	yes
BIOS_ID_BYTES	Hexadecimal values from the BIOS.	varchar (16)	yes
BIOS_DATE	The revision date of the BIOS.	varchar (32)	yes
BIOS_STRING	The BIOS string.	varchar (128)	yes
BIOS_MANUFACTURER	Identifies the manufacturer of the BIOS.	varchar (64)	yes
MANUFACTURER_ID	Identifies the manufacturer of the PC.	varchar (64)	yes
BIOS_MODEL	The BIOS model.	varchar (64)	yes
BIOS_SER_NUM	The BIOS serial number.	varchar (64)	yes

Columns	Description	Type	Null
UPTIME	The current uptime of the system. Indicates the level of reliability of the system.	bigint	yes
IE_VERS	The version of Microsoft Internet Explorer that is installed.	varchar (64)	yes
CREATE_TIME	Date and time when the entry was added to the database.	timestamp	no
UPDATE_TIME	Date and time when the entry was last updated.	timestamp	no

**The TLMHW.PRINTER table:** Stores the details for one particular type and model of printer. It contains only one record for each unique printer in use in the monitored environment.

Columns	Description	Type	Null
PRINTER_ID	Unique identifier of the printer type.	varchar (32)	no
PRINTER_MODEL	A description of the printer model.	varchar (254)	yes
CREATE_TIME	Date and time when the entry was added to the database.	timestamp	no
UPDATE_TIME	Date and time when the entry was last updated.	timestamp	no

**The TLMHW.PROCESSOR table:** Stores the details for one particular type and model of processor. It contains only one record for each unique processor in use in the monitored environment.

Columns	Description	Type	Null
PROCESSOR_ID	Unique identifier of the processor type.	varchar (32)	no
MANUFACTURER	The manufacturer of the processor.	varchar (128)	yes
PROCESSOR_MODEL	The processor model.	varchar (200)	yes
PROCESSOR_FEATURES	A bitmask within which each bit that is set identifies a feature that is available for the processor.	integer	yes
MAX_SPEED	The maximum speed of the processor in MHz.	integer	yes
CURRENT_SPEED	The current speed of the processor in MHz.	integer	yes
BUS_SPEED	The external bus speed of the processor in megahertz (MHz).	integer	yes
CPU_INTERFACE	The external central processing unit (CPU) packaging interface.	varchar (32)	yes
ECACHE_MB	The size of the processor cache in MB	varchar (16)	yes
CPU_IMPL	The implementation (type) of the processor	varchar (16)	yes
CPU_MASK	The CPU mask for the processor.	varchar (16)	yes
CHIP_FAMILY	The chip family of the processor.	integer	yes
CHIP_MODEL	The chip model of the processor.	integer	yes



Columns	Description	Type	Null
CHIP_STEPPING	The chip stepping code for the processor	integer	yes
VIRT_MODE_EXT	Indicates whether virtual mode extension is supported.	char (1)	yes
PAGE_SIZE_EXT	Indicates whether page size extension is supported.	char (1)	yes
TIME_STAMP_COUNTER	Indicates whether time stamp counter is supported.	char (1)	yes
MODEL_SPECIFIC_REG	Indicates whether model-specific registers are supported.	char (1)	yes
PHYSICAL_ADDR_EXT	Indicates whether physical address extension is supported.	char (1)	yes
MACHINECHECK_EXCPT	Indicates whether machine-check exception is supported.	char (1)	yes
CMPXCHG8B_SUPP	Indicates whether the compare exchange 8-byte instruction is supported.	char (1)	yes
ON_CHIP_APIC	Indicates whether the integrated advanced programmable interrupt controller (APIC) is supported.	char (1)	yes
MEM_TYPE_RANGE_REG	Indicates whether memory type range registers are supported.	char (1)	yes
PAGE_GLOBAL_ENABLE	Indicates whether the page global extension is enabled.	char (1)	yes
MACHINECHECK_ARCH	Indicates whether the machine-check architecture is supported.	char (1)	yes
COND_MOVE_SUPP	Indicates whether the conditional move instruction is supported.	char (1)	yes
MMX_TECHNOLOGY	Indicates the Intel® MMX features that are supported.	char (1)	yes
ON_CHIP_FPU	Indicates whether the floating processor unit is supported.	char (1)	yes
DEBUG_EXT_PRESENT	Indicates whether the debugging extension is supported.	char (1)	yes
FAST_SYS_CALL	Indicates whether fast system call is supported.	char (1)	yes
PAGE_ATTR_TABLE	Indicates whether the page attribute table extension is supported.	char (1)	yes
PAGE_SIZE_EXT36	Indicates whether the 36-bit page size extension is supported.	char (1)	yes
SER_NUM_ENABLED	Indicates whether the processor serial number is enabled.	char (1)	yes
FAST_FLOAT_SAVE	Indicates whether fast floating point save and restore instructions are supported.	char (1)	yes
SIMD_EXT_SUPP	Indicates whether the streaming single instruction, multiple data (SIMD) instruction set is supported.	char (1)	yes
NOW_3_D_ARCH	Indicates whether the AMD 3DNow! instructions are supported.	char (1)	yes

Columns	Description	Type	Null
CREATE_TIME	Date and time when the entry was added to the database.	timestamp	no
UPDATE_TIME	Date and time when the entry was last updated.	timestamp	no

**The TLMHW.SMBIOS\_SYS\_DATA table:** Stores general SMBIOS details. It contains only one record for each unique SMBIOS configuration that is in use in the monitored environment.

Columns	Description	Type	Null
SMBIOS_ID	SMBIOS ID.	varchar (32)	no
BIOS_VENDOR	The manufacturer of the system BIOS.	varchar (64)	yes
BIOS_VERS	The version number of the system BIOS.	varchar (128)	yes
BIOS_SIZE	The size of the system BIOS.	char (8)	yes
SYS_MANUFACTURER	The manufacturer of the system.	varchar (64)	yes
SYS_PRODUCT_NAME	The product name of the system.	varchar (64)	yes
SYS_VERS	The version number of the system.	varchar (64)	yes
BOARD_MANUFACTURER	The manufacturer of the system board.	varchar (64)	yes
BOARD_PRODUCT	The product name of the system board.	varchar (64)	yes
BOARD_VERS	The version number of the system board.	varchar (64)	yes
CASE_MANUFACTURER	The manufacturer of the case for the system.	varchar (64)	yes
CASE_TYPE	The type of case for the system.	varchar (64)	yes
CASE_VERSION	The version number of the case for the system.	varchar (64)	yes
CREATE_TIME	Date and time when the entry was added to the database.	timestamp	no
UPDATE_TIME	Date and time when the entry was last updated.	timestamp	no

**The TLMHW.STORAGE\_DEV table:** Stores the details for a particular storage device installed on a system. One record exists for each storage device for each system scanned.

Columns	Description	Type	Null
COMPUTER_SYS_ID	Unique identifier of the system.	bigint	no
INST_STORAGE_ID	The index number for the installed storage device	integer	no
STORAGE_TYPE	The type of storage device.	varchar (64)	yes
MANUFACTURER	The manufacturer of the storage device.	varchar (128)	yes
MODEL	The model of storage device.	varchar (254)	yes
SER_NUM	The serial number of the storage device	varchar (64)	yes
HDISK_ID	The hard disk ID	varchar (32)	no

Columns	Description	Type	Null
CREATE_TIME	Date and time when the entry was added to the database.	timestamp	no
UPDATE_TIME	Date and time when the entry was last updated.	timestamp	no

**The TLMHW.UNIX\_SYS\_PARAMS table:** Stores UNIX system parameters for a system. One entry exists for each system scanned.

Columns	Description	Type	Null
COMPUTER_SYS_ID	Unique identifier of the system.	bigint	no
BOOT_TIME	The date and time that the system was last restarted.	timestamp	yes
UPTIME	The current uptime of the system. Indicates the level of reliability of the system.	integer	yes
RUN_LEVEL	The run level of the system.	integer	yes
HOST_NAME	The host name of the system.	varchar (254)	yes
CREATE_TIME	Date and time when the entry was added to the database.	timestamp	no
UPDATE_TIME	Date and time when the entry was last updated.	timestamp	no

**The TLMHW.USB\_DEV table:** Stores the details for one particular type of USB device. It contains only one record for each unique USB device in use in the monitored environment.

Columns	Description	Type	Null
USB_ID	The USB ID	varchar (32)	no
USB_VERS	The USB version of the device.	varchar (32)	yes
DEV_CLASS	The device class of the USB device.	varchar (32)	yes
DEV_SUBCLASS	The device subclass for the USB device.	integer	yes
VENDOR_ID	The vendor ID of the manufacturer of the USB device.	integer	yes
PRODUCT_ID	The product ID for the USB device.	integer	yes
MANUFACTURER	The manufacturer for the USB device.	varchar (128)	yes
PRODUCT	The type of product the USB device is.	varchar (254)	yes
NUM_OF_PORTS	The number of USB ports present on the USB device.	integer	yes
DEV_IS_HUB	Indicates whether the USB device is a hub.	char (6)	yes
CREATE_TIME	Date and time when the entry was added to the database.	timestamp	no
UPDATE_TIME	Date and time when the entry was last updated.	timestamp	no

**The TLMHW.VID\_CARD table:** Stores the details for one particular type of video card. It contains only one record for each unique video card in use in the monitored environment.

Columns	Description	Type	Null
VID_CARD_ID	The video card ID	varchar (32)	no
VID_CARD_MODEL	The video card model.	varchar (254)	yes
VID_CARD_BIOS	The BIOS information for the video card.	varchar (254)	yes
VID_DAC_TYPE	The integrated digital-to-analog converter (DAC) for the video card.	varchar (254)	yes
VID_MEM	The amount of memory for the video card.	integer	yes
VID_BIOS_RELDATE	The BIOS release date	varchar (32)	yes
VID_CHIP_TYPE	The chip type for the video card.	varchar (254)	yes
CREATE_TIME	Date and time when the entry was added to the database.	timestamp	no
UPDATE_TIME	Date and time when the entry was last updated.	timestamp	no

**Changed tables:** Additions have been made to the following tables to support the hardware inventory scan feature:

- ADM.AGENT. See Table 2.
- ADM.CONTROL. See Table 3.
- ADM.DIVISION. See Table 4.

*Table 2. Additions to the ADM.AGENT table.*

Columns	Description	Type	Null
HWSCAN_TIME	The date and time of the last hardware scan.	timestamp	yes

*Table 3. Additions to the ADM.CONTROL table.*

Columns	Description	Type	Null
HARDWARE_SCAN_ENABLED	Indicates whether the hardware inventory scan feature is enabled.	char (5)	yes
COMPUTER_GUID_RULE	Indicates the rule used to calculate the computer GUID. Possible values are:  <ol style="list-style-type: none"> <li>1 Concatenate the values for manufacturer, model, and serial number.</li> <li>3 Use the Media Access Control (MAC) address of the first network card discovered on the system.</li> </ol>	smallint	yes

*Table 4. Additions to the ADM.DIVISION table.*

Columns	Description	Type	Null
HWINV_START_DATE	The date on which the hardware scan schedule starts.	timestamp	no

Table 4. Additions to the ADM.DIVISION table. (continued)

Columns	Description	Type	Null
HWINV_RATE_TYPE	Indicates the hardware scan repeating period. Possible values are:  <b>0</b> No repetition <b>1</b> The repetition period is 1 day. <b>2</b> The repetition period is 7 days. <b>3</b> The repetition period is 30 days.	smallint	no
HWINV_RATE_VALUE	The number of repeating periods that separate consecutive occurrences of the hardware scan.	integer	no
HWINV_VIRTUAL_DISABLED	Indicates whether the hardware scan is disabled in virtual environments. Possible values are 0 (scanning enabled), 1 (scanning disabled).	smallint	no

## Agent installation size check

For all agent deployment methods, a space check is now made to ensure that the installation will not start and then fail because of lack of sufficient space in the agent installation directory. Before starting the installation process, the following space requirement in the agent installation directory must be satisfied:

### Windows agent

38 MB

### Linux agent

50 MB

### Solaris, AIX® and HP-UX agents

65 MB

### GSKit component

30 MB

If the space available is insufficient, the installation fails with return code -17.

## Support for partitioning technologies

The fix pack introduces support for agents in Solaris Containers non-global zones with resource pools enabled.

When an agent is deployed in a non-global zone and resource pools are enabled, the agent details that are shown on the Web UI include information about the processor set associated with the resource pool for the zone where the agent is deployed.

**Note:** When agents are deployed in non-global zones, an agent must also be deployed in the global zone.

## Product fixes

This is the first fix pack issued for Tivoli License Compliance Manager, version 2.3. It fixes the following APARs:

**APAR IY95336**

On Linux 390 platforms, the manual agent deploy setup fails in silent mode.

The fix pack resolves the problem.

**APAR IY95183**

On Windows systems, a noise is heard when the agent checks the presence of the floppy disk.

The fix pack resolves the problem by installing a new version of the Common Inventory Technology (CIT) component.

**APAR IY94558**

The initial scan performed at agent start up takes too long.

The problem was related to the collection of system storage information. This is no longer required and so has been removed.

**APAR IY94123**

The Software Package Block (SPB) for agent configuration changes does not include the possibility to change the organization.

The fix pack adds the Organization parameter to the SPB `change_agt_configuration_platform.spb`.

The main objective of this change is to allow configuration errors to be corrected. If the name of the organization has been entered incorrectly, the agent is not able to plug in. You can use the software package block to distribute a correction to all affected agents.

You can also use the SPB to transfer agents that are already active from one organization to another. However, be aware that all installed software and software use information that has already been collected by the agents will no longer be available. Before distributing the SPB, use the **Manage Infrastructure > Agents** task on the Web UI to delete the agent. When the agent plugs in after the SPB has been applied, a new agent record is automatically created.

**APAR IY93924**

The time on the agents is not in synch with the time on the server.

The fix pack resolves the problem.

**APAR IY93548**

No software use is detected for WebSphere® Commerce 5.6.x when the J2EE application has the same name as the application server.

The fix pack resolves the problem.

**APAR IY93364**

If the administration and runtime servers are started using the Windows command line and, later on, the user logs off from Windows, the servers are also stopped.

The fix pack resolves the problem.

**APAR IY93328**

Abend code on an i5/OS agent because of a variable overflow caused by an http connectivity problem with the runtime server.

The fix pack resolves the problem.

#### APAR IY93251

The agent cannot handle correctly 'Manufacturer' strings longer than 31 characters.

The fix pack resolves the problem.

#### Problems fixed

The following problems were found since the general availability of Tivoli License Compliance Manager, version 2.3 and fixed with this fix pack:

##### **The generateAgentId command does not work if fix pack 3 is applied to WebSphere Application Server, version 6.1**

This problem is caused because the update to the Java™ SDK introduced in fix pack 3 is not backward compatible. At the GA version of Tivoli License Compliance Manager, version 2.3, the **generateAgentId** command will not work after you have applied fix pack 3 to WebSphere Application server.

When this fix pack has been applied, the command works both with and without fix pack 3.

##### **Exceptions are generated when a division that has many connected agents is deleted.**

When the fix pack has been applied, it is no longer possible to delete divisions that have connected agents.

#### Problems and workarounds

The following describes a problem affecting interaction with IBM Tivoli Configuration Manager.

##### **Incorrect agent installation return code reported in the Tivoli Configuration Manager logs.**

When the agent is deployed using Tivoli Configuration Manager and errors occur, incorrect agent installation return codes are written to the software package block log.

This problem is caused because the error return codes are negative integers and logging in Tivoli Configuration Manager does not handle them correctly. The correct return code is written in the file `s1mrc`, which is stored in the directory on the target computer from which the agent installation was launched.

##### **On HP-UX platforms, problems can occur accessing the Web UI after the server has been stopped and restarted.**

This problem is caused by a WebSphere transport problem that affects both the administration and runtime servers. The problem is resolved in fix pack 5 for WebSphere Application Server 6.1.

If you do not want to apply the WebSphere Application Server fix pack at this time, you can overcome the problem by stopping both the Tivoli License Compliance Manager server and the HTTP server and then restarting them.

## Documentation fix history

This fix pack documents solutions for the following problems or errors in the product documentation:

##### **Missing information about installing the agent on VMware environments (APAR IY96544 for Tivoli License Compliance Manager version 2.2)**

Problem (found in *IBM Tivoli License Compliance Manager: Planning, Installation, and Configuration*): you can install the agent on VMware

environments without running the CIT enabler. Agents installed in this manner will only be able to send inventory data: they will not send usage data, nor aggregated or hardware information. On agents running VMware ESX 3.0 (or later), APAR IY96549 for CIT version 2.3 must be installed. The CIT enabler in this fix pack includes the APAR.

**Missing CIT required libraries for VMware environments (APAR IY95943 for Tivoli License Compliance Manager version 2.2)**

**Problem:** In Chapter 1 of *IBM Tivoli License Compliance Manager: Planning, Installation, and Configuration*, the information about libraries required by CIT for VMware environments is documented in Table 4 "Supported agent platforms", but is missing from page 20 where the CIT enabler specifications are documented.

**Solution:** The table that documents the CIT enabler software on page 20 of Chapter 1 should include the following new paragraph: "On hosts running Red Hat Linux, the enabler requires the compatibility packs documented in Table 4 on page 11."

**Instructions for configuring the WebSphere Application Server virtual host for secure communications are not clear. (APAR IY94284 for Tivoli License Compliance Manager version 2.2)**

**Problem:** In Chapters 2 and 3 of *IBM Tivoli License Compliance Manager: Security Management*, the instruction that describes the configuration of a secure port for all virtual host aliases does not make clear which virtual host should be selected.

**Solution:** In both chapters, the instruction is changed as follows:

Ensure that the WebSphere Application Server has virtual aliases for the secure port.

Logon to the WebSphere Application Server console and do the following:

1. On the Virtual Host settings page, select the virtual host instance called **default\_host**.
2. Configure a secure port for the virtual host and all its aliases. You can create a single definition that applies to all possible host aliases, setting the **Host Name** to \* and **Port** to the secure port number (normally 443).
3. Generate the Web Server plug-in.
4. Propagate the Web Server plug.in.

**Clarification for differences in the steps for configuring Java 2 security depending on the WebSphere Application Server configuration.**

**Problem:** In Chapter 5 of *IBM Tivoli License Compliance Manager: Security Management*, the instructions for configuring Java 2 security do not reflect possible differences in the the WebSphere Application Server configuration.

**Solution:** The following differences need to be stated:

- In the document the directory where Java 2 policies are stored is given as:

```
<WAS_INSTALL_DIR>\profiles\default\config\cells\  
<host_cell_name>\applications\  
SLM_<server_type>
```

```
_Application.ear\deployments\  
SLM_<server_type>_Application\META-INF
```



The subdirectory `\default` should be documented as a variable. The subdirectory name is the name of the default profile created when WebSphere Application server is installed. It could be "default", but it could also have other values.

- If you have installed the Tivoli License Compliance Manager server with an existing installation of WebSphere Application Server, for which Java 2 security was already configured, you need only to complete the task of replacing the profile file with the file in which you have defined the location of the Tivoli Common Directory. Steps 7, 8, and 9 are not required.

**The prerequisite for space in the /tmp directory during installation of the runtime server is too low (APAR IY93190 for Tivoli License Compliance Manager version 2.2)**

**Problem:** In Chapter 1 of *IBM Tivoli License Compliance Manager: Planning, Installation, and Configuration*, the table of prerequisites for servers specifies a requirement for 250 MB of free space on UNIX platforms in the /tmp directory. For runtime server installations, this is insufficient. In addition, the table of prerequisites for the runtime server should say that the WebSphere Application Server temporary directory (`java.io.tmpdir`) by default is set to the /tmp directory.

**Solution:** The requirement is to have at least 700 MB of free space on UNIX platforms.

**The syntax for the backupconf and restoreconf commands is incorrect.**

**Problem:** In *IBM Tivoli License Compliance Manager: Commands*, the syntax described for the command `backupconf` and `restoreconf` commands is incorrect. The documented syntax for the commands is:

```
backupconf [-d <directory_name>]
restoreconf [-d <directory_name>]
```

The correct syntax is:

```
backupconf [<directory_name>]
restoreconf [<directory_name>]
```

## Backward compatibility

Dependencies:

Tivoli License Compliance Manager, version 2.3

## Other changes as a result of this fix pack

This fix pack creates or updates a file called `<INST_DIR>\product.xml` (where `<INST_DIR>` is the directory where Tivoli License Compliance Manager is installed), which keeps track of the current Tivoli License Compliance Manager version.

Agents deployed after this fix pack has been applied to the runtime server will have a new version number of 2.3.0.10. The individual agents' software will be updated only when you specifically reinstall the agent software, or after you have used the agent self-update facility in the agent configuration file, as described in *IBM Tivoli License Compliance Manager: Planning, Installation, and Configuration*.

---

## Installation, migration, upgrade, and configuration information

This section includes the following topics:

- “Hardware and software requirements”
- “Installing the fix pack”
- “Upgrading agents” on page 30
- “Reverting to the previous version” on page 30

### Hardware and software requirements

This section includes the following topics:

- “Supported platforms”
- “System requirements”

#### Supported platforms

The following information supplements the supported platform information documented in *IBM Tivoli License Compliance Manager: Planning, Installation, and Configuration*:

**On the following platforms, changes are required to the required maintenance levels, service packs, patches or compatibility patches**

**HP-UX 11i v2 running on rx7640 and rx8640.**

**Required patch:** PHKL\_35174

This patch is required when agents are deployed in partitioned environments.

**HP-UX 11i v2 on PA-RISC**

**Required patch:** PHSS\_35381

**Limitation to the deployment of agents on Solaris x86 platforms (global and non-global zones)**

Agents cannot be deployed on hardware that supports hyper-threading functionality.

#### System requirements

The following are additions to hardware and software prerequisites detailed in *IBM Tivoli License Compliance Manager: Planning, Installation, and Configuration*.

**The space in the agent installation directory required for agent installation is changed**

**Windows agent**

38 MB

**Linux agent**

50 MB

**Solaris, AIX and HP-UX agents**

65 MB

**GSKit component**

30 MB

### Installing the fix pack

This section contains the instructions for installing the fix pack. Once you have installed the fix pack, you cannot uninstall it automatically. For details of how to revert to the previous version see “Reverting to the previous version” on page 30.

The fix pack includes the following files:

**2.3.0-TIV-TLCM-FP0001-servers-<platform>.zip**

For each supported platform, the compressed file contains the files for installing the fix pack on computers where an administration server, a runtime server, or an administration server database or runtime server database is installed.

**2.3.0-TIV-TLCM-FP0001-SPB.zip**

This file contains the software package blocks for each supported agent platform, to be used when deploying agents using IBM Tivoli Configuration Manager.

**2.3.0-TIV-TLCM-FP0001-agent-gateway.zip**

This file contains the software packages for each supported agent platform in PKT format.

**2.3.0-TIV-TLCM-FP0001-RSH-SSH-<platform>.zip**

This contains the files needed to deploy agents on UNIX platforms using the RSH/SSH deployment wizard.

**2.3.0-TIV-TLCM-FP0001-agentOS400.zip**

This contains the files needed to deploy agents on i5/OS platforms.

**2.3.0-TIV-TLCM-FP0001-ManualDeploy-<platform>.zip**

This contains the files needed to install agents using a local wizard.

**2.3.0-TIV-TLCM-FP0001-SPB-TOOLS.zip**

This contains the SPBs for agent configuration update that have been updated for APAR IY94123.

The fix pack must be applied to all server and database components of Tivoli License Compliance Manager.

**Notes:**

1. Before starting the installation, back up the administration and runtime server databases and the installation directories of administration server and runtime servers.
2. During the installation of the fix on the administration server or a runtime server, the wizard stops and starts the server. If the server is running in a WebSphere Application Server secure cell, you will be asked to provide the user ID and password for the secure cell. Ensure that you have this information before you start.

To install the fix pack, complete the following steps on each computer where a component is installed:

1. Log on to the computer where one or more of the Tivoli License Compliance Manager components is installed as Administrator (Windows) or root (UNIX).
2. Unpack the servers compressed file into a temporary directory.
3. Launch the setup file for the platform on which you are installing the fix pack. The installation wizard starts.

**Note:** No license agreement panel is displayed. The fix pack is subject to the same terms and conditions under which Tivoli License Compliance Manager, version 2.3 is licensed.

4. Specify a directory where the wizard can create a backup of your server configuration and other files that are affected by the fix pack and click **Next**.

If the wizard is unable to create the backup, the installation cannot continue. Ensure that you have the correct permissions to write to the specified directory and that there is sufficient space to create the backup.

**Note:** The wizard calculates the space required to create the backup based on the initial size of a server when it is installed. Under normal circumstances, the space occupied by the server directories should not increase very much. However, if other directories, for example directories required for manual agent deployment, have been created within the server file structure, the space required for the backup could be considerably more than anticipated. Under these circumstances, it is possible that the installation could fail because of lack of space. If this occurs, a message will be displayed informing you that the installation has failed, possibly because of insufficient permissions. You must then make more space available for the installation.

5. Check the summary panel for the installation and click **Next** to confirm that you want to install the fix pack.

If you are applying the fix pack to a server that is running in a WebSphere Application Server secure cell, you will be prompted to supply a valid user ID and password when the wizard stops and starts the runtime server.

6. When the installation is complete, click **Finish**.

## Upgrading agents

To apply the fixes to agents that are already deployed, you must redeploy or update the agent. For information about how to do this see the sections **Redeploying an agent** and **Agent self-update** in *IBM Tivoli License Compliance Manager: Planning, Installation, and Configuration*.

Agents are supported only when they are registered to a runtime server with the same, or a higher, Tivoli License Compliance Manager maintenance level.

## Reverting to the previous version

There is no automatic method for uninstalling this fix pack.

You must rollback the changes on each computer where the fix pack has been installed, as follows:

1. Stop the server.
2. Manually replace the files that were changed by the installation of the fix pack with the backup copies taken during the installation.

The backup directory specified during the installation of the fix pack contains a subdirectory 2.3.0-TIV-TLCM-FP0001 which contains a subset file structure for the Tivoli License Compliance Manager components to which the fix pack has been applied. You must copy the files manually from each subdirectory of the backup to the corresponding directory of the server installation.

3. If this is the first fix applied to the GA version of the code, delete the file `<INSTALL_DIR>\product.xml`.

If previous fixes have been applied, the `product.xml` file is one of the files that you have restored from the backup directory.

4. Restart the server.

---

## Notices

This information was developed for products and services offered in the U.S.A. IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing  
IBM Corporation  
North Castle Drive  
Armonk, NY 10504-178, U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation  
Licensing  
2-31 Roppongi 3-chome, Minato-ku  
Tokyo 106, Japan

**The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:**

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement might not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation  
2Z4A/101  
11400 Burnet Road  
Austin, TX 78758 U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurement may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

---

## Trademarks

The following terms are trademarks of the IBM Corporation in the United States or other countries or both:

AIX  
DB2  
DB2® Universal Database  
IBM  
i5/OS  
Tivoli  
WebSphere  
zSeries

Intel and Itanium are trademarks of Intel Corporation in the United States, other countries, or both.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S., and other countries. Microsoft and Windows are registered trademarks, of Microsoft Corporation in the U.S. and other countries.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product, and service names may be trademarks or service marks of others.









Program Number: 5724-D33