Tivoli Business Systems Manager

1.5-BSM-0035 Release Notes

Tivoli Business Systems Manager

1.5-BSM-0035 Release Notes

Tivoli Business Systems Manager Patch 35 Release Notes

Contents

Preface

Tivoli Business Systems Manager provides a robust, system management functionality. In support of the operational perspectives of that functionality, Tivoli Business Systems Manager installs components on both the OS/390 and NT platforms.

Accessing Publications Online

Publications in the product libraries are included in PDF or HTML formats, or both, on the product CD. To access the publications using a Web browser, open the infocenter.html file, which is located in the appropriate publications directory on the product CD.

When IBM publishes an updated version of one or more online or hardcopy publications, they are posted to the Tivoli Information Center. You can access updated publications in the Tivoli Information Center from the following Tivoli Customer Support Web site:

http://www.tivoli.com/support/documents/

The Tivoli Information Center contains the most recent version of the books in the product library in PDF or HTML formats, or both. Translated documents are also available for some products.

Note: If you print PDF documents on other than letter-sized paper, select the **Fit to page** check box in the Adobe Acrobat Print dialog (which is available when you click **File --> Print**) to ensure that the full dimensions of a letter-sized page are printed on the paper you are using.

Providing Feedback about Publications

We are very interested in hearing about your experience with Tivoli products and documentation, and we welcome your suggestions for improvements. If you have comments or suggestions about our products and documentation, contact us in one of the following ways:

- Send an e-mail to pubs@tivoli.com.
- Complete our customer feedback survey at the following Web site:

  http://www.tivoli.com/support/survey/

Contacting Customer Support

If you have a problem with any Tivoli product, you can contact Tivoli Customer Support. See the *Tivoli Customer Support Handbook* at the following Web site:

http://www.tivoli.com/support/handbook/

The handbook provides information about how to contact Tivoli Customer Support, depending on the severity of your problem, and the following information:

- Registration and eligibility
- Telephone numbers and e-mail addresses, depending on the country in which you are located

- What information you should gather before contacting support

Chapter 1. What's in this Document

This document provides important information about Patch 1.5-BSM-0035 for Tivoli Business Systems Manager Version 1, Release 5.

Please review these notes thoroughly before installing or using this product.

Each chapter in this document describes the changes to Tivoli Business Systems Manager v1.5 documentation as a result of Patch 35. Chapters include the following:

Chapter 2. Changes to the Installation and Configuration Guide

Servers Note

[Please add the following to Chapter 1.]

Note: All Tivoli Business Systems Manager servers should be located in the same Windows Domain to ensure proper communication between the servers.

Additionally, Tivoli Business Systems Manager communicates to the console using the IP address or Domain Name Server (DNS) name of the console as provided when the console initiates the session. The host file in C:\winnt\system32\drivers\..... can also be used for name resolution. Tivoli Business Systems Manager does not resolve host names using Windows Internet Name Services (WINS). Appropriate measures should be taken to ensure that the IBM Tivoli Business Systems Manager servers can communicate to the consoles using the customer selected network methodology.

To enable communication between Tivoli Business Systems Manager servers and the console environment through a firewall or router, refer to Appendix A.

If you decide to change the log on for Tivoli Business Systems Manager services from the default of LocalSystem to a userID, then that userID must have administrative rights for the Windows operating environment.

Setting Parameters for Language Support

[Please add the following to Chapter 4.]

The following parameters must be set to enable language support:

Windows NT

[HKEY_LOCAL_MACHINE\SOFTWARE\Accessible Software, Inc.\Access1\1.0\Settings]

"DefaultLanguage"=<language id> - The valid language ids are the directory names available in tivolimanager\lang

"LocalCodePage"=<NT code page id> - For example, 1252 for Danish

"RemoteCodePage"=<mainframe code page> - For example, 20277 for Danish

Also, if you are using a code page other than 437 on your Tivoli servers, you should review the comments in the cconv.tbl in tivolimanger\lang\<language id>. The console code page is set by the SetConsoleCodePage command.

Issue 'SetConsoleCodePage' from a command prompt and note the value of 'Current Console Code Page:'.

OS/390

The following 3 Source/390 Object Pump Startup Parameters are used to setup language support on the host. Refer to Chapter 6 of the Installation and Configuration Guide for more detail on usage.

- LANGUAGE_ID = ENU | <language id>
- SBCS_CODEPAGE=0037 | <SBCS codepage id>
- DBCS_CODEPAGE = <DBCS codepage id>

OMEGAMON Defaults

[Please add the following to the Setting Exceptions and Creating the OMEGAMON AS Profile section.]

The following steps enable you to set exception thresholds and save them within the Omegamon AS profile.

1. Log on to the classic Omegamon with an existing or default profile. For example: `LOGON APPLID( applid) DATA( USER= xx)` (omit the DATA(USER=xx) part to use the default profile)
2. For each exception that you want to set, issue the XACB LIST= cccc command where cccc is the exception name you want to modify. (See the Candle Omegamon documentation for the available exceptions, or contact your Candle Support Representative.)
3. Change the exception settings as required.
4. When all the exceptions are set, issue the Omegamon command: PPRF SAVE AS.
5. Reply Y to confirm Save. This saves the current exception settings in a profile with the AS suffix.

Pump 'OMEGAMON_EXCEPTION_CNTRL=ALL|DELTA' Parameter

OMEGAMON_EXCEPTION_CNTRL=ALL | DELTA was previously documented as "specifies whether the source/390 object pump should send an exception message every time an exception is detected (), or only when new exceptions are detected."

This function is permanently set to ALL and cannot be changed via the Object Pump startup parameters. Customer utilizing this option should note it is ignored, and may remove the parameter from their system if desired, as keeping it has no effect.

Maintaining customized URLs during Attach/Detach processing

Customers that have no customized URL hostnames are not affected by this section, this only applies if you utilize customized URL hostnames in the database.

The following comment lines have been added to the detachdatabases.sql for those that have some of the URL hostnames customized and set to hostnames other than websvr_host name specified in the attachdatabases.sql

Uncomment and modify the following lines to map the customized hostnames to an internal hostnames. Update the 'USER_DEFINED_HOSTNAMEx' to the customized URL hostnames used in the control_id field of the Object..MenuItem table. Leave the second column 'INTERNAL_NAMEx' unchanged. There is no need to map websvr_host name specified in the attachdatabases.sql URL hostname will be updated from USER_DEFINED_HOSTNAMEs to INTERNAL_NAMEs during databases detaching. URL hostname will be updated from INTERNAL_NAMEs to USER_DEFINED_HOSTNAMEs during databases attaching.

```
-- INSERT INTO #mappingURLHost (definedHost, pseudoHost) VALUES
('USER_DEFINED_HOSTNAME1', 'INTERNAL_NAME1')
```

```
-- INSERT INTO #mappingURLHost (definedHost, pseudoHost) VALUES
('USER_DEFINED_HOSTNAME2', 'INTERNAL_NAME2')

-- INSERT INTO #mappingURLHost (definedHost, pseudoHost) VALUES
('USER_DEFINED_HOSTNAME3', 'INTERNAL_NAME3')
```

Example 1: If you have a customized URL hostname NCTBSMSERVER1 in the control_id field of the Object..MenuItem table that you want to preserve during databases detaching/attaching, modify the following section of the code in the detachdatabases.sql from:

```
-- URL hostname will be updated from INTERNAL_NAMEs to
USER_DEFINED_HOSTNAMEs during databases attaching.

-- INSERT INTO #mappingURLHost (definedHost, pseudoHost) VALUES
('USER_DEFINED_HOSTNAME1', 'INTERNAL_NAME1')

-- INSERT INTO #mappingURLHost (definedHost, pseudoHost) VALUES
('USER_DEFINED_HOSTNAME2', 'INTERNAL_NAME2')

-- INSERT INTO #mappingURLHost (definedHost, pseudoHost) VALUES
('USER_DEFINED_HOSTNAME3', 'INTERNAL_NAME3')
```

to:

```
-- URL hostname will be updated from INTERNAL_NAMEs to
USER_DEFINED_HOSTNAMEs during databases attaching.

INSERT INTO #mappingURLHost (definedHost, pseudoHost) VALUES
('NCTBSMSERVER1', 'INTERNAL_NAME1')

-- INSERT INTO #mappingURLHost (definedHost, pseudoHost) VALUES
('USER_DEFINED_HOSTNAME2', 'INTERNAL_NAME2')

-- INSERT INTO #mappingURLHost (definedHost, pseudoHost) VALUES
('USER_DEFINED_HOSTNAME3', 'INTERNAL_NAME3')
```

There are two changes made in the code.

1. USER_DEFINED_HOSTNAME1 has been changed to the customized hostname NCTBSMSERVER1.
2. The leading -- has been removed.

Example 2: If you have four hostnames TBSMNORTH, TBSMSOUTH, TBSMWEST, TBSMEAST to preserve, modify the code to the following:

```
-- URL hostname will be updated from INTERNAL_NAMEs to
USER_DEFINED_HOSTNAMEs during databases attaching.

INSERT INTO #mappingURLHost (definedHost, pseudoHost) VALUES ('TBSMNORTH',
'INTERNAL_NAME1')

INSERT INTO #mappingURLHost (definedHost, pseudoHost) VALUES ('TBSMSOUTH',
'INTERNAL_NAME2')

INSERT INTO #mappingURLHost (definedHost, pseudoHost) VALUES ('TBSMEAST',
'INTERNAL_NAME3')

INSERT INTO #mappingURLHost (definedHost, pseudoHost) VALUES ('TBSMWEST',
'INTERNAL_NAME4')
```

The following changes are made in the code:

1. All the leading comment characters -- (double dashes) has been removed.

2. USER_DEFINED_HOSTNAME1 to USER_DEFINED_HOSTNAME3 have been changed to TBSMNORTH, TBSMSOUTH, TBSMEAST.
3. An extra line has been added for updating the hostname TBSMWEST. The second column is set to INTERNAL_NAME4.

Changes to Installing and Configuring the History Server

The following section replaces the History Server information in the IBM Tivoli Business System Manager V1.5 Installation and Configuration Guide.

In this section, the following terms will be used:

- *History server* refers to the secondary server that contains the history database, to which events from the Tivoli Business Systems Manager primary database server are forwarded. The data contained on the history database is used to run Historical Reporting.
- *Primary server* refers to the server to which events are forwarded from a variety of data sources. This data is used for real-time monitoring of an enterprise's critical resources.

Installing the History Server

There are two different approaches to the installation of the history server. They are the following:

- Linked Server
- Bulk Copy Program (BCP)

These approaches use different methods to transfer data from the database server to the history server (assuming they are installed on separate machines). You should install only one of them.

The Linked Server approach

- uses MS SQL Linked Server implementation to move data directly from the database server to the history server using Microsoft's MSSQL Server OPENQUERY command.
- does not need to have Tivoli Business Systems Manager installed on the history server (unlike the BCP approach, which does require Tivoli Business Systems Manager).
- requires the reporting system to be configured to retrieve historical data from the database server. (This has some performance impact on the primary server because the reporting queries are run against it. Therefore it is not recommended for large Tivoli Business Systems Manager implementations.)

The BCP approach

- uses the bcp command to copy data (events) out of the primary server to the history server.
- requires that the history server databases be synchronized with the database on the primary server. This is accomplished by jobs on the history server and on the primary server that transfer the necessary databases.
- The reporting system is set up to retrieve historical data from the history server. (There is less of a performance impact on the database server than with the Linked Server approach.)

Note: If you already have installed the Linked Server approach, and wish to migrate to the BCP approach, follow the instructions in the section below entitled *Converting from a Two-Server Linked History to a Two-Server BCP Approach Setup*.

Installing the History Server Using the Linked Server Approach

The history server can be configured on the same machine as the primary Tivoli Business Systems Manager SQL Server (Single Server configuration) or on a different machine (Two Server configuration).

Prerequisites:

- A system that has the primary server role must have Tivoli Business Systems Manager 1.5 Base installed.
- When using the two server configuration, the network connections and the necessary permissions with the history server must already be configured properly.

Single Server Configuration: To set up the history server locally on the primary server, issue the shell command:

```
sh HistoryServerSetup.ksh -T LINKED
```

Two Server Configuration: To set up the history server, issue the following shell command on the primary server:

```
sh historyserversetup.ksh –T LINKED –N <historyservername> -R
<historyserver_user> -W <historyserver_password>
```

where <historyservername> is the name of the history server, <historyserver_user> is the history server DB administrator userid, usually 'sa', and <historyserver_password> the password assigned to the DB administrator userid.

A dialog box will appear, asking you to verify the names of the history server and primary server. Click "YES"to continue installation, or click "NO"to cancel installation.

Note: If the log on fails due to an incorrect password for the remote history server, delete the linked server using SQL ServerEnterprise Manager->Security->Linked Serve, try the previous command again, and restart the server.

Verification Points for Two Server Installations:

1. The Distributed Transaction Coordinator (MSDTC) service on the primary and history servers needs to be configured to start automatically.
2. Run the following command on the primary server to determine if the event_table copy is working properly:

   ```
   osql -E -Q "use master exec asisp_copyEventHistory
   <history_server_hostname>"
   ```

   where <history_server_hostname> is the name of your history server. If you see a "rows affected"message, then it is working. If you see a "Msg 8501, Level 16, State 3" message or something similar, contact Tivoli Customer Support.
3. Run the following command on the history server to verify that the event_table data is available:

   ```
   osql -E -Q "use EventHistory select * from event_table order by ctime
   DESC
   ```

   If you see event data, it is working properly.

Configuring the Reporting System:

Prerequisite: You must install the Arial Unicode font on any console used to run the reporting system. Without this font, the browser may or may not correctly display the content of the reporting system pages. As of 7/18/2002, the Arial Unicode font can be downloaded from the following web site:
http://office.microsoft.com/downloads/2000/aruniupd.aspx

Use the Reporting System Database Configuration program found in the Tivoli Business Systems Manager programs folder to set up the DefaultData databases with the following parameters:

Table 1. Default data database parameters

Parameter	DefaultData	WebServer
Connection Name	DefaultData	WebServer
Driver	(SQL server)	(SQL server)
Server Name	<SQLServerName>	<SQLServerName>
User ID	sa	sa
Password	<sa_SQLServerPassword>	<sa_SQLServerPassword>
Database Name	Object	WebServer

Installing the History Server Using the BCP Approach

After you install the history server using this approach you can perform the following tasks:

1. Move BCP events from the primary server to the history server at five (tunable) minute intervals.
2. A periodic restoration of the Tivoli Business Systems Manager Object, Meta and RODM databases on the history server from the primary server.
3. Perform cleanup on events transferred from the Object database on the primary server to the history server.

Prerequisites:

- A daily backup of the Object, Meta and RODM databases on the primary server.
- You have installed the Internet Information Server 4.0 or higher and the Tivoli Business Systems Manager historical reporting system on the history server.
- The network connections and the necessary permissions between the primary server and the history server are configured properly.
- Ensure the SQL agent start/login accounts on the primary server have the write permission to the history server.
- Tivoli Business Systems Manager is installed on the history server.

Note: The BCP approach requires a two server configuration. The single server option is not supported.

Installing the History Server:

1. Insert the Tivoli Business Systems Manager Base Services CD in the history server computer. Double-click the BaseServices folder.
2. Double-click the Setup.exe icon.

3. From the Choose Setup Language dialog, confirm the language to use for your installation. Click OK.
4. When the System File Upgrade dialog opens, read the text and click Next.
5. From the Choose Destination Location dialog, we recommend that you place these files in a temporary directory. Click the Browse button to select a directory. Once a desired directory is selected or to use the default, click Next.
6. When the System File Upgrade is complete, click Finish. If you are prompted to reboot the computer, do so now.
7. From the Choose Setup Language dialog, confirm the language to use for your installation. Click OK.
8. The Welcome to the Setup program dialog opens. The text in this dialog is basic information about the Setup program, including how to cancel the installation. Read the information and click Next.
9. The Software License Agreement dialog opens. Read the agreement and accept the agreement to continue the installation process. Click Yes.
10. When the Choose Destination Location dialog opens, select a directory to install the Tivoli Business Systems Manager application and click the Browse button. The selected path is displayed in the Destination Folder. Once the desired path is selected or to choose the default destination folder, click Next.

    Attention: We recommend that you install the application on a drive other than the operating system default drive of C:\ to avoid the possibility of the log and working files becoming full and stopping the Windows operating system.
11. From the Setup Type dialog, select Custom and click Next.
12. From the Select Components dialog, select History Server. Click Next.

    Note: If you have previously installed components by selecting them from the Select Components dialog, do not clear them from the list.
13. Enter the hostname of the local server machine. (By default, the host name should be in the host name field, If it is not, enter it.) Click Next.
14. Enter the SQL Administrator Username, and SQL Administrator Password. Click Next.
15. The Select Program Folder dialog opens. Select Program Folder to host all applications. (The default is Tivoli) Click Next.
16. From the Setup Complete dialog, if you are prompted to restart the computer, select No and click Finish. If the Setup Complete dialog does not prompt you to restart, click Finish.

Configuring the History Server Database:

1. Apply <install_dir>\SQL \SchemaTypes.sql to the master database.
2. Apply <install_dir>\SQL \SchemaTypes.sql to the model database.
3. Apply <install_dir>\SQL \SchemaTypes.sql to the tempdb database.
4. Apply <install_dir>\SQL \MasterDBChanages.sql to the master database.
5. Apply <install_dir>\SQL \SchemaMessages.sql to the master database.
6. Create the following databases if they do not exist:
   - History - Use the SQL Enterprise Manager to create an empty database.
   - Object - Same as Live Server. Either copy the Object_Data.MDF file from the live server or extract the file from the \DBDevices\ DBDevices_V1.5.exe file.
   - Meta - Same as Live Server. Either copy the Meta_Data.MDF file from the live server or extract the file from the \DBDevices\ DBDevices_V1.5.exe file.

- RODM - Same as Live Server. Either copy the RODM.MDF file from the live server or extract the file from the \DBDevices\ DBDevices_V1.5.exe file.
- WebServer - Same as Live Server. Either copy the WebServer_Data.MDF file from the live server or extract the file from the \DBDevices\ DBDevices_V1.5.exe file.

BCP Setup: To setup the history server, issue the following shell command from the \sql directory on the primary server:

```
sh historyserversetup.ksh –N <HistoryServerName>-R <History_db_userid>-W
<History_db_password>
```

During the running of `historyserversetup.ksh`, two dialogs are displayed entitled "History Server Setup using BCP Approach."The first asks the installer to verify that the server names for the primary and history servers are correct. Select "No"to stop the installation.

The second dialog asks the installer to verify the server directories that are used during installation. If you select "No"the history server installation continues, but you must alter one or more of the SQL jobs found in the "Job Configurations"section below to complete the installation. If it is necessary to end the installation at this point, click Cancel.

Completing the History Server Installation: You need to follow these steps before using the reporting system for the first time:

1. In the Microsoft Enterprise Manager, check to see if the following jobs are on the history server:
   - `Copy Backups to History Server`
   - `Update Object and Meta Databases`

   Delete these jobs if they are present. These jobs may be searched for on the history server using SQL Enterprise Manager to expand the Management folder, then expanding "SQL Server Agent"and "Jobs."

   Note: Do not confuse these job names with the job names that were just installed (`Backups from Live DB Server` and `Databases for Reporting System`).
2. Run a database backup job on the primary server, if one has not already been run.
3. Insure that the User ID under which the SQLServerAgent on the primary and history servers runs has Administrator authority. If this is not done, the `Move Event to History Database` job on the primary server will fail with a "Unable to BCP EventBCP Table Out' message, and then `Copy Backups from Live DB Server` job on the history server will fail with a "Failure: Directory [$src_dir] does not exist or access authority is insufficient"message.
4. Run the `Copy Backups from Live DB Server` job on the history server. Verify that the database backups have arrived on the history server. Currently these are the backup files of the Meta, Object, and RODM databases.
5. Run the `Restore Databases for Reporting System` job on the history server.

Configuring the Reporting System:

Prerequisite: You must install the Arial Unicode font on any console used to run the reporting system. Without this font, the browser may or may not correctly display the content of the reporting system pages. As of 6/1/2002, the Arial Unicode font can be downloaded from the following web site:
http://office.microsoft.com/downloads/2000/aruniapd.aspx

Use the Reporting System Database Configuration program found in the Tivoli Manager programs folder to set up the DefaultData and PrimaryData databases with the following parameters:

Table 2. Default data database parameters

Parameter	DefaultData	LiveData	WebServer
Connection Name	DefaultData	LiveData	WebServer
Driver	(SQL server)	(SQL server)	(SQL server)
Server Name	<HistorySQLServerName>	<PrimaryTBSMSQLServerName>	<HistorySQLServerName>
User ID	sa	sa	sa
Password	<sa_HistorySQLPassword>	<sa_PrimarySQLPassword>	<sa_HistorySQLPassword>
Database Name	Object	Object	WebServer

Job Configurations: The following section describes jobs that are scheduled during the installation of the History server using the BCP approach. These jobs are accessed using the following steps:

1. In Enterprise Manager, select the appropriate server
2. expand the Management folder
3. expand the SQL Server Agent folder
4. click Jobs and locate the job name
5. right-click on the job name and select Properties
6. select the Steps tab
7. edit the appropriate step's command

History Server jobs:

1. `Copy Backups from Live DB Server` – This job copies the latest available backup files from the primary server to the history server for use by the reporting system. Currently these are the backup files of the Meta, Object, and RODM databases.

   Schedules Tab: Default is daily at 12:40 AM. Because the Object database can be extremely large, we recommend that `Copy Backups to History Server` be scheduled off-hours. In addition, the interval of this job should be determined based upon how the topology has changed in the monitored configuration. The less dynamic, the longer the interval that is necessary. For example, once every other day or once a week.

   Steps Tab: If changes to the default configuration are necessary, edit Step 1, "Copy Live Backups to History."The default command is:

   ```
   sh -c
   '//<HistoryServerName>/<TBSMInstallationDriveShare>/TivoliManager/bin/co
   pylivebackupstohist.ksh
   //<DBServerName>/<MSSQLInstallationDriveShare>/MSSQL7/backup
   //<HistoryServerName>/<MSSQLInstallationDriveShare>/MSSQL7/backup'
   ```

   a. Change //<HistoryServerName>/<TBSMInstallationDriveShare> to the history server's shared drive name where the Tivoli Business Systems Manager system files were installed.

b. Change //<DBServerName>/<MSSQLInstallationDriveShare> to the primary server's shared drive name where MSSQL was installed.

c. Change //<HistoryServerName>/<MSSQLInstallationDriveShare> to the history server's shared drive name where MSSQL was installed.

Note: You must observe the use of the forward slash ("/") here because of its use in a "sh"command. For example:

```
sh -c '//TBSMHIST/D$/TivoliManager/bin/copylivebackupstohist.ksh
//TBSMMAIN/D$/MSSQL7/backup //TBSMHIST/D$/MSSQL7/backup'
```

2. `Restore Databases for Reporting System` –This job updates the databases on the history server from the backup files retrieved by the `Copy Backups from Live DB Server` job.

Schedules Tab: The default is daily at 2:40 AM. This job depends upon the `Copy Backups from Live DB Server` job. The start time should adjusted based upon the scheduled interval and run length of the `Copy Backups from Live DB Server` job.

Steps Tab: If changes to the default configuration are necessary, edit step 2, "Restore Databases for Reporting System."The default command is:

```
asisp_historyinit "<HistoryServerMSSQLBackupPath>\", "BAK"
```

a. Change <HistoryServerMSSQLBackupPath> to specify the history server's SQL backup directory.

Note: You must observe the use of the backward slash ("\") here because of its use in a MSSQL procedure. For example:

```
asisp_historyinit "D:\MSSQL7\backup\", "BAK"
```

Primary Server Jobs:

1. `Move Events to History Database` –This job copies events from the primary server to the history server on a periodic basis.

Schedules Tab: The default is daily every five minutes.

Steps Tab: If changes to the default configuration are necessary, edit step 1: "Move Events to History Database."The default command is:

```
asisp_moveevent '<HistoryServerName>',
'\\<DBServerName>\<MSSQLInstallationDriveShare>\MSSQL7\backup',
'History', 'eventhist'
```

a. Change <HistoryServerName> to the name of the history server.

b. Change \\<DBServerName>\<MSSQLInstallationDriveShare> to the primary server's shared drive name where MSSQL was installed.

Note: You must observe the use of the backward slash ("\") here because of its use in a MSSQL procedure. For example:

```
asisp_moveevent 'TBSMHIST', '\\TBSMMAIN\D$\MSSQL7\backup',
'History', 'eventhist'
```

2. `Delete old Entries in Event Table` –This job deletes event table entries older than seven days which have been copied to the history server.

Schedules Tab: The default is daily at 1:00 AM. It maintains a local history of the past seven days on the primary server, while deleting older entries which have already been copied to the history server.

Steps Tab: If a change to the default configuration is necessary, edit step 1: Delete old Entries in Event Table. The default command is:

```
asisp_deleteoldevent
```

a. Add or change the @daystokeep=n parameter. For example:

```
asisp_deleteoldevent @daystokeep=14
```

Converting from a Two-Server Linked History to a Two-Server BCP Approach Setup

Prerequisites:

- a linked server for history has been installed
- existing historical events residing on the History server are retained
- the Object, Meta, and RODM databases on the Tivoli Business System Manager SQL Server are being backed up periodically
- there are successful backup files of the Object, Meta, RODM databases present

The following variables are defined:

- <HistoryServer> - The host name of the server that contains the history database
- <HistsaUID> - The sa User ID for the history database
- <HistsaPWD> - The sa password for the history database

1. Export the existing historical events: The events residing on the history server are exported by running the following BCP command:

   ```
   bcp "EventHistory..event_table"out <drive>\dbE entHistory.txt –w -q -
   S<HistoryServer>-U<HistsaUID>-P<HistsaPWD> -e<drive>:\dbE
   entHistoryERR.txt
   ```

   Use the appropriate values for <HistsaUID>, <HistsaPWD>, and <HistoryServer> on the history server database.

2. Uninstall the linked server: Run the following Kornshell command on the primary server:

   ```
   sh HistoryServerUninstall.ksh -N <HistoryServer>-R <HistsaUID> -W
   <HistsaPWD>
   ```

3. Install the history server using the BCP approach: Follow the instructions in the section above entitled Installing the History Server Using the BCP Approach.

4. Import the historical events: The saved events from step 1 are imported into the new history server setup by running the following BCP command:

   ```
   bcp "History..eventhist"in <drive>\dbEventHistory.txt –w -b10000 -q -
   S<HistoryServer>-U<Hist_saUID>-P<Hist_saPWD> -e<drive>:\dbE
   entHistoryERR.txt
   ```

   Use the appropriate values for <HistsaUID>, <HistsaPWD>, and <HistoryServer> on the History server database.

5. Configure the Reporting System: Follow the instructions in the section above entitled Configuring the Reporting System.

Chapter 3. Changes to the Administration Guide

[Please add the following additions/changes to the Tivoli Business Systems Manager 1.5 Administration Guide.]

Enhancements to CICS TDQ Processing

Description

Currently, Tivoli Business Systems Manager collects DFHAC2236 messages from the CICS transient data queue (TDQ). These events are collected when a transaction abends. These events are categorized as messages within the Tivoli Business Systems Manager database and have a default alerts state/priority of Red/High. There is no equivalent clearing message (Green/High) from the TDQ which results in a high volume of red alerts that must be manually cleared. The messages get applied to the transaction object which because of the incoming message(s) is set to a state of Abended. This may not really be the case because of the nature of the transactions of the same name running simultaneously and restarting and so on. Therefore, the additional problem is that this object is constantly in a state of Abended.

Solution

1. Reduce the Red in the views. This will be accomplished by reducing the severity of the incoming DFHAC2236 from High to Low. When a transaction receives a Red Low message it will turn Red and generate a child event with a Red/Low color severity combination which will be sent to the parent CICS. The default threshold for Red/Low child events on a CICS region is 200, which means that 201 such events would have to be received by the CICS object before the CICS turns red itself. Such child events can be sent to a parent object from all of its children. In other words if the CICS object had 200 transactions as children, one child event of Red/Low color severity from each child could turn the CICS object Red. The default threshold can be increased or decreased as the customer sees fit.
2. Remove the abended current state in the transactions. Change the incoming DFHAC2236 message to be stateless.
3. Remove stale Transaction abend messages. The last item is to resolve the problem of the Abend messages not having an equivalent automatic clearing message. A SQL procedure is now provided that sends an equivalent clearing message to all transactions that have outstanding DFHAC2236 messages posted on them that are older than a configurable number of hours.

Configuration Instructions

1. Adjust the Priority of the DFHAC2236 message from High to Low. This can be done by the Tivoli Business Systems Manager administrator by using Console option EDIT --> RESOURCE TYPES. Double Click the 'Transaction - TRAN' Object. On the 'States' tab, adjust the priority from High to Low
2. Configure the SQL Job to remove stale Transaction abend messages; A new SQL job called "Delete CICS Transaction Messages"(in the category "CICS Monitoring") is created to run the procedure to delete messages. The job is disabled when it is created so it will not run automatically. From Management/Jobs of SQL Enterprise Manager, you can do either of the following:

- Run the job on demand
- Enable the job to run according to its schedule
- Once you have enabled the job, a default schedule is provided for the job that causes the procedure to execute once every five minutes. Use SQL Enterprise Manager to change the schedule of the job if you desire:
  a. Go to Management/Jobs of SQL Enterprise Manager.
  b. Right-click the job.
  c. Select "properties"from the pop-up menu.
  d. Change to the "Schedules"tab.
  e. One schedule should be listed and it should be highlighted. Click the "Edit"button to change the schedule.
  f. Click the "Change"button.
  g. Modify the schedule.
  h. Press the "OK"button to save the recurring schedule.
  i. Press the "OK"button to save the schedule.
  j. Press the "OK"button to properties of the job.

The procedure deletes messages that are older than a threshold. The default threshold is five minutes. Use Enterprise Manager to change the threshold of the procedure if you desire:

1. Go to Management/Jobs of SQL Enterprise Manager.
2. Right-click the job.
3. Select "properties"from the pop-up menu.
4. Change to the "Steps"tab.
5. One step should be listed and it should be highlighted. Click the "Edit"button to change the step.
6. Click the "Change"button.
7. The threshold is the number that appears after "RemoveStaleTRANMsgs"in the "Command"area of the step and is expressed in minutes. Modify the threshold.
8. Press the "OK"button to save the step.
9. Press the "OK"button to properties of the job.

Removing 'Note Created' dialog when opening new notes

Use the following SQL statement to add a new row to the SystemConfiguration table (this applies to the NT console only):

Procedure Name

```
_SetSystemConfiguration
```

To Remove 'Note Created' dialog

EXEC _SetSystemConfiguration

@groupname = 'API_INTEGRATION_OPTIONS'

@propertyname = 'CONFIRM_NOTE_CREATION'

@propertyvalue = 'NO'

@propertydatatype = 'CHAR'

A value of 'NO' in the propertyvalue column for this option will suppress the display of the 'Note Created Successfully.' dialog following a user successfully creating a note. To turn on the display of this dialog, replace the value of 'NO' with 'YES' in the row for the option or delete the row.

Error deleting workspaces

You may receive the error GTM0558E: "An error occurred deleting workspace *workspace_name* while attempting to delete a workspace from the Console. This may occur if the workspace's name contains special characters, such as a single quote ('). If you receive this message, perform the following steps.

1. Stop all Consoles.
2. Stop the Console Server.
3. On the Console Server machine, delete the following file:

   c:\TivoliManager\JavaAppServer\w32-ix86\jre\lib\ext\xerces.jar

   where c:\TivoliManager is the directory where you installed the Console Server).
4. Restart the Console Server.
5. Restart the Console.
6. Delete the workspace.

asisp_definemenuitem

[Please add the following to the Command appendix.]

asisp_definemenuitem

Purpose

Adds a new menu item to a drop down menu on the Tivoli® Business Systems Manager console. If the menu item exists, it is updated with the new parameters.

Usage

asisp_definemenuitem '*item_name*', '*class_id*', '*object_id*', '*label*', '*parent_id*', '*previous_id*', '*action*', '*override*', '*security*', '*flags*'

Parameters

item_name
: Specifies the name of the menu item.

class_id Specifies the class ID to which the menu item applies.

object_id
: Specifies the object ID to which the menu item applies. Specify 0 if this menu item is for all members of the class.

label Specifies the label to display. This is what the user sees.

parent_id
: Specifies the ID of the parent menu item entry under which to place this menu item. Specify NULL if this is a top-level menu item.

previous_id

Specifies the ID of the menu item entry after which to place this menu item. Specify NULL if this menu item is to be appended to the end of the menu items under the parent menu item. Specify -1 if the menu item is to be placed first under the parent menu item; all subsequent menu items are shifted down.

action Specifies the action that is to be taken when the menu item is invoked.

override

Specifies which menu item to override. This is in case a user wants to define a different menu item that would replace an existing one but perform a different action. For example, a menu item may be defined on class 'A'. A user wants this menu item to do something different for class 'B' which derives from class 'A'. So this user would create the new menu item to override the one defined for class 'A' so that the one defined for class 'A' does not show up for class 'B'. Use the following format which uniquely identifies the menu item to be overridden:

"`<name>/<obj_cid>/<obj_id>`**"**

Specify NULL if this is a new menu item.

security

Specifies the feature level security key for which the caller must be authorized. This is needed for the user to be authorized to invoke the menu item. The Java client simply checks to see if there is a value in this field. The value can be any non-empty string ("dog","admin") If an empty string is used ("","") then the menu item will be available to both Operators and Administrators. If so, then the menu item will only show for Administrators.

flags Specifies the flag bits that control the type and actions of the menu item. You can enter the following values together with a logical OR (|):

Table 3. asisp_definemenuitem flags

Flag Bits	Type or Action
0x00000001	Separator
0x00000002	control_id specifies a command line with parameters for a process to be launched
0x00000004	Launch control in dialog box instead of view
0x00000008	Submenu
0x00000010	Built-in command to be invoked when an item is selected
0x00000020	Built-in command to be invoked when the menu is built
0x00000040	Add item to list in disabled mode if security check fails. Default is to remove the item.
0x00000080	This entry eliminates any entry that was overridden (instead of replacing it).
0x00000100	Invoke Web control with URL
0x00000200	Run a method on an object and display the results in a view
0x00000400	Launch ActiveX control in a view
0x00000800	HyperView
0x00001000	Run a method on an object
0x00002000	Used with MI_COMMANDLINE(0x00000002) to specify that the menu item will be disabled (instead of removed) if the command file cannot be run

Table 3. asisp_definemenuitem flags (continued)

Flag Bits	Type or Action
0x00004000	GraphView
0x00008000	Menu item will be displayed in the application menu (not as a popup menu)
0x10000000	Self. Enabled for the object.
0x20000000	Aggregate. Enabled for aggregates in which this object is located.
0x40000000	Link. Enabled for items linked to this object. Used with Self and Aggregate bits: • If Self is also set, this item is enabled for Business Systems that reference this object. • If Aggregate is also set, this item is enabled for Business Systems which contain a Business System that references this object.

Processing

The asisp_definemenuitem is an SQL stored procedure that can be run from the Microsoft SQL Server Query Analyzer GUI interface or the isql command line interface. This stored procedure resides in the Object database. Run this procedure against the Object database.

The Tivoli Business Systems Manager console server might need to be recycled for this stored procedure to take effect.

Examples

The following example adds the Command top-level menu item for all OS objects:

```
exec asisp_definemenuitem
 'Command','OS',0,'Command',NULL,NULL,'',NULL,'',1342177288
```

The following example adds the Launch menu item under the Command menu item, and invokes the Microsoft Internet Explorer application:

```
exec asisp_definemenuitem 'Launch','OS',0,'Launch',2524,NULL,
'C:\Progra~1\Plus!\Micros~1\iexplore.exe',NULL,'',1342177282
```

Assigning a name to a CICSplex CMAS folder

[The following should be added to the CPSM User information section.]

The CICSplex CMAS folder name is blank by design as it is just a collection of CMASs. If you want to assign a name to a folder, go to the property page of the folder and enter a name in the Name field.

"Log on Locally" information

[Please add the following to Chapter 11: Tivoli Business Systems Manager Security]

Tivoli Business Systems Manager Administrators AND Tivoli Business Systems Manager Operators must have the Windows User right to "Log on locally"or they will not be able to access Tivoli Business Systems Manager. (Note that the Administrators and Users groups generally have this right so if the Tivoli Business

Systems Manager administrators/operators are members of either of these two groups, they may already have this right.)

SQL Job 'Cleanup Old Log Files'

[Please add to the Appendix for SQL Jobs]

Functionality

The job will cleanup all .log, .dat, .lu62, .msg and .txt files on all the applicable Tivoli Business Systems Manager servers specified in the job step older than the number of days also specified in the job step. This job excludes the makemvs*log files when deleting the old log files, since these might be needed months after the fact. The default schedule of this job has been set to run once a week.

Required Customizations

The `Cleanup Old Log Files` job MUST be customized before it is run, which is why it is disabled by default. In order to customize the job, the following must be done:

1. The <Database_Server>, <History_Server>, <Application_Server>, <Propagation_Server>, <Event_Handler> and <SNA_Server> slots in the 'Edit Job Step' window must be replaced with the server names of the database server, history server, application server, propagation server, event handler and sna server machines respectively. If you do not want old logs on a particular server to be deleted, then the appropriate server slot MUST be replaced with a NULL. NO SERVER SLOT SHOULD BE LEFT EMPTY.
2. The <Days_To_Retain> slot must be replaced with an integer that specifies the number of days of logs you want to keep on all Tivoli Business Systems Manager servers. For example, if <Days_To_Retain> is replaced by 10, all logs older than 10 days will be deleted off of all the Tivoli Business Systems Manager server machines that have been specified in the first 6 slots. If a number is not entered, the number of days of logs to keep will default to 7.

Note Editor changes

- Ownership notes were modified so that if an event is not pre-selected prior to opening the Note Editor (ex: from the resource's Properties page) then the description for the first event in the Event list will be placed into the Note Editor's Description field when the window is initialized.
- As user selects events in the Event list, the description of the first selected event will be placed into the Note Editor's replacing the existing text. However, once the user manually edits the field this behavior will cease and selecting events will no longer replace the text in the Description field (so that no user typed text is lost) {If an event was selected prior to opening the Note Editor, then selecting events does not have the behavior of replacing the text in the Description field}

Problem Ticket changes

- If invoked via the Note Editor (using the 'Create problem ticket' checkbox), then the Description field on the 'Create Problem Ticket' window will initially contain the text:

  `<CustomerID> - <Text_from_Note_Editor_Description_field>`
- If not invoked via the Note Editor then the Description field on the 'Create Problem Ticket' window will initially contain only the <CustomerID>

Description of 'Leaf Processing' and its usage within the Console

The Tivoli Business Systems Manager Event Viewer and Managed Objects views implement what is called 'Leaf Processing'. This means that when you build LOB's and use these views, the physical children of the LOB objects may be taken into account. This only occurs if you drag and drop an object into a LOB that has children in the Resource(eg. physical) view, and within the LOB you do NOT drag any of those children underneath the said resource.

For example, if we have the following configuration:

- Resource View:
  - —Batch Schedule ABC
    - Batch Job1
    - Batch Job2
    - Batch Job3
  - —Batch Schedule XYZ
    - Batch Job4
    - Batch Job5
    - Batch Job6

And we build an LOB such as:

- LOBXYZ
  - —Batch Schedule ABC
  - —Batch Schedule XYZ

The Event Viewer launched from LOB XYZ will show the events of Batch Jobs 1 through 6.

Alternatively, you may negate this 'Leaf' processing by placing the children in the LOB underneath their parent. Using the example above, if you now built your LOB as follows:

- LOBXYZ
  - —Batch Schedule ABC
    - Batch Job1
  - —Batch Schedule XYZ
    - Batch Job6

The event viewer will now only show events for Batch Job 1 and 6, and NOT 2 through 5.

This concept works for all Object Types that have physical children within the resource view.

In the current release of Tivoli Business Systems Manager 1.5, this concept also works with the Managed Objects View. **HOWEVER, in a future release, we plan to eliminate this function for this view when the objects are not physically linked.** The overhead of this processing is large, and customer have indicated that it is not what they would like, nor do they use it. They prefer to use this view similar to hyperview, where only the listed objects are displayed. The effect this change will have on the following sample LOB is described below:

- LOBXYZ

—Batch Schedule ABC

—Batch Schedule XYZ

Today: Managed Objects View on LOB XYZ will return both Batch Schedules ABC and XYZ and also the six batch jobs.

Future: Managed Objects View on LOB XYZ will return ONLY Batch Schedules ABC and XYZ. HOWEVER, this view called from the LOB representation of either Batch Schedule will return the appropriate children for each parent, as these objects are linked to a physical resource.

If there are any concerns with this planned design change, please contact Tivoli Support immediately.

Considerations for LOB building and usage when using Leaf Processing

Customers implementing 'Leaf' Processing should understand the following:

1. This process is for the two views mentioned only. One must understand that if the Event Viewer is launched off a tree view or hyperview, the status of the objects in those views may be different then the events in event viewer or status in managed objects. This is because the tree/hyper views are not considering children of the objects in the LOB, while the event viewer and managed objects view might. This is working as designed.
2. Leaf nodes are the same as children, and should be considered when designing LOBs as performance may be adversely affected by the number of children. Managed Objects view especially can take very long periods of time to return if the number of children exceeds 2000.

We always recommend that customers design their LOB views carefully, limited the number of children to those critical to their business. The larger the object count in a LOB, the longer database queries for those objects will take.

Console Server

Changes to Servers.Properties

Multiple new options have been added to the Java Console Server Properties files since the release of Tivoli Business Systems Manager 1.5. These options include:

- com.tivoli.tbsm.server.Servers.maxTraceFiles
- com.tivoli.tbsm.ui.wam.UIConsoleFrame.displayLOBTree
- com.tivoli.tbsm.ui.banner.UIBannerAreaManager.bannerAlias
- java.rmi.server.hostname
- com.tivoli.tbsm.ui.view.UIToolTips.tooltipInformation
- com.tivoli.tbsm.common.TMViewProperties.suppressNotificationRefresh

A sample Servers.Properties files has been included with this patch in the TivoliManager\Javaappserver directory as file 'Sample_Servers.properties'. The new parameters are described within this file, as well as documented below.

com.tivoli.tbsm.server.Servers.maxTraceFiles

Description: The maxTraceFiles property controls the number of log files that the server keeps. If you want to retain trace data longer at the expense of more disk space,

increase this value. Each log file is approximately 10MB in size. If you do not specify an integral value or specify one less than 3, the server keeps 3 10 MB log files.

Example: com.tivoli.tbsm.server.Servers.maxTraceFiles=3

com.tivoli.tbsm.ui.wam.UIConsoleFrame.displayLOBTree

Description: A flag to determine if the LOB Tree for the Java Console should be displayed. This flag applies to Tivoli Business Systems Manager Operators only. A value of true indicates that all Tivoli Business Systems Manager Operators will be able to see the LOB tree from the Java Console. A value of false will not allow Tivoli Business Systems Manager Operators to see the LOB tree. The default value is true. This value should be set to true if Tivoli Business Systems Manager Operators have not been given a workspace to use; otherwise, the Tivoli Business Systems Manager Operator will not be able to see any LOB views.

Example:com.tivoli.tbsm.ui.wam.UIConsoleFrame.displayLOBTree=true

The impact to the Java Console is relatively minor. If the new property does not exist within the Servers.properties file, then the Java Console will proceed as it has already proceeded in the past by displaying the LOB tree.

When the LOB tree is not displayed, an Operator will not have the ability to navigate, open, or view any LOB unless an workspace has been created for the Operator. Workspaces can be created by Administrators and assigned to Operator(s). Within a Workspace view, an Operator will be able to perform normal LOB operations that are specific/designated for the Operator.

The purpose of this fix is to allow the ability to turn on/off the LOB tree for Operators, due to security requirements/restrictions of the customer. However, it should be noted that if an Administrator has placed a restricted LOB resource (that is, an LOB that an Operator should not see) into an LOB that an Operator is allowed to view, then the Operator will have access to the restricted LOB.

com.tivoli.tbsm.ui.banner.UIBannerAreaManager.bannerAlias

Description: The location of the banner files to load remotely from the Java Application Server. This value should point to the location of the banner files used by the Java Console and does not need to be on the same machine as the web server. The value should contain the format:
http://hostMachine.domain/virtualDirectory/locationToBannerFiles

Example: com.tivoli.tbsm.ui.banner.UIBannerAreaManager.bannerAlias=

java.rmi.server.hostname

Description: Server host name or IP address. Specify the fully qualified host name (example: www.ibm.com) or IP address (example: 129.42.17.19) that clients should use to connect to the console server. In most situations, you do not have to specify this value because the host name as returned by the operating system is sufficient. However, in cases where the computer hosting the console server has multiple IP addresses you may need to use this property to specify which address clients should use to connect. The default is the primary host name returned by the operating system.

Example: java.rmi.server.hostname=tbsmserver.yourcompany.com

com.tivoli.tbsm.ui.view.UIToolTips.tooltipInformation

Description: tooltipInformation

Eight different values are supported for this indicator:

- The number one (1) indicates ownership information will not be displayed in any resource flyover tooltips.
- The number two (2) indicates ownership information will only be displayed in tree view resource flyover tooltips.
- The number three (3) indicates state information will not be displayed in any resource flyover tooltips.
- The number four (4) indicates state information will only be displayed in tree view resource flyover tooltips.
- The number five (5) indicates ownership and state information will not be displayed in any resource flyover tooltips.
- The number six(6) indicates ownership and state information will be displayed in tree view resource flyover tooltips.
- The number nine (9) indicates resource flyover tooltips will not be displayed at all.
- The number ten (10) indicates all resource flyover tooltips will be displayed.

The default is 5: Display no ownership or state information in any resource flyover tooltips

Example: com.tivoli.tbsm.ui.view.UIToolTips.tooltipInformation=5

com.tivoli.tbsm.common.TMViewProperties.suppressNotificationRefresh

Description: Setting to control whether notifications of database changes can cause a view to refresh. If this property is set to false (default), open views on the console may automatically refresh the view when notification of certain changes is received from the database. When this property is set to true, views will only be refreshed as a result of a timed refresh set by the user or as a result of the user pressing the refresh button.

Note: This setting does NOT affect updates of the display with changes in attribute values, such as the alert state of an object. The database changes that can cause full refreshes typically involve the creation or deletion of resources in the database.

Example:
com.tivoli.tbsm.common.TMViewProperties.suppressNotificationRefresh=false

Note: Many customers have reported 'constant refreshing' within the list views despite auto refresh being disabled. This is likely due to high levels of new object creation via auto-discovery or drag and drop. If you have seen this problem, we recommend changing the above parameter to TRUE.

Recommended Default Changes

We recommend that Tivoli Business Systems Manager Administrators be familiar with all parameters in Servers.Properties and adjust them to meet their business need. However, since the release of Tivoli Business Systems Manager 1.5, we have noted that failure to adjust some of the defaults has caused performance issues at many customers, and would like to note our standard recommendation for certain

parameters as follows. The default Servers.properties in the next Tivoli Business Systems Manager release will reflect these changes, however, 1.5 customer will need to make these changes manually if they have not already done so.

com.tivoli.tbsm.server.DBManager.maxconnecti oncount

Connections are pooled and used on demand for each database query/operation and returned to the pool when that operation completes. It is possible that a system with 20 consoles and 4 event viewers per console could request as many as 80 connections for the event viewers at some point in time. Other views also can request connections, such as the views the event viewers are attached to. If this happens, you will see connection wait log entries. In practice having all of the requests line up at once is not going to occur that often, and if/when it does you probably want to throttle the ability of the consoles to overload the database. Given this, we recommend a value of 30 for this statement.

Description: Maximum number of connections open at a time. The default is 10 if you do not specify this property. Increasing this value increases responsiveness to consoles but also increases the load that the console server can place on the database engine.

Example: com.tivoli.tbsm.server.DBManager.maxconnectioncount=30

com.tivoli.tbsm.server.DBManager.activeTimeou t

Customers using large database queries are often timing out with the standard two minute default for this parameter. We recommend adjusting this parameter to 10 minutes.

Description: Allocated connection timeout (seconds). Database connections allocated to a request longer than this time are deemed timed out and may be closed automatically. Time required to retry a call due to deadlock or error recovery counts against this timeout. Decreasing this value speeds up the detection of hung or stalled database calls but increases the likelihood that normally long running database calls will be terminated prematurely. The default is 120 (2 minutes).

Example: com.tivoli.tbsm.server.DBManager.activeTimeout=600

Logging and tracing parameters

Typical server recommendations for trace levels when experiencing a problem are as follows:

com.tivoli.tbsm.server.Servers.traceLevel=debug

com.tivoli.tbsm.server.Servers.DatabaseTrace=true

These are recommendations for general problems, we do not typically recommend using the ALL level of trace. Additionally, if there is a possible communication problem between the application server and the console, the APITrace should be set, for example:

com.tivoli.tbsm.server.Servers.APITrace=true

Additionally, with this patch, the log file maximum size has been increased from 1MB to 10MB. Due to this, we will no longer recommend increasing the maxTraceFiles entry as the default of 3 should suffice. If you already increased this parameter, you may want to lower it when applying this patch to decrease disk space utilization.

com.tivoli.tbsm.server.Servers.maxTraceFiles=3

Note: These are the recommended settings for problem resolution. During normal operation, the default trace values should be used. These are error, false, false (respectively).

Miscellaneous Changes

- The RODM option 'Reregister' is no longer available in the console for the GMFF and SNNT objects as this is deemed a redundant function and only worked for previously registered objects. Users should use 'unregister' and 'register' instead of this option.
- The Performance Monitor option 'Force Logon' has been removed from the console as it is obsolete and performs no function.
- We now support 'multi-select' on the CICS Transaction screen as well as the MVS uncategorized resource view.
- The Key/Non-Key conversion menu item is not available on Batch jobs in a LOB. This function should be performed by the System Administrator in the Batch Management view. This option is only valid for CA7 Users.
- The apostrophe is not allowed in object names.
- The object type filter (there is one in the Managed Resources view) no longer contains duplicate listings of 'Operating System'. An OSYS object is now referred to as a 'Computer Operating System.'
- OW54008 changed gemdmmap.sh script to support a single-quote in -D parameter which takes in a DM profile name.

Chapter 4. Changes to Miscellaneous Documentation

Addition to the Patch 29 Release Notes

[Please add the following information to the How to Configure HyperView section on page 19 of the Patch 29 Release Notes.]

STORED PROCEDURE LOB_GETDESCENDENTS HAS A HARDCODED RETURNED MAXROWS OF 3000

To manually set the max return row number, using either isql or SQL Query Analyzer, issue the following:

exec _SetSystemConfiguration

@groupname='_GetDescendents',

@propertyname = 'RETURN_LIMIT',

@propertyvalue = <int value you want to set>,

@propertydatatype = 'INT'

Changes to the 1.5 User Guide

[Please make the following change to the 1.5 Tivoli Business Systems Manager User's Guide.]

Opening a Resource's Properties

To open a resource's properties:

1. Right-click the resource to open its context menu.
2. Click Properties.

Additions to the 1.5 RMF Release Notes

Enabling RMF functionality changes

[Please add the following information to the "Enabling RMF Functionality" section in the 1.5 RMF Release Notes included with Patch 24.]

Use the object IDs of the OSs intended to be monitored to identify resources while creating relationship statements. To display the object ID in the java client, press the key combination CNTL-SHIFT-F12. Then bring up the properties page and select the 'ResourceDebugAttribute' tab. The cid and id will be found in the value column corresponding to attribute column 'cid' and 'id'.

RMF feed with a single DDS in a sysplex

[Please add the following information to the "Resource Management Facility - Source/390 Object Pump Startup Parameters" section in the 1.5 RMF Release Notes included with Patch 24.]

Source/390 Object Pump Startup Parameters

The following values are defaults and are used if no others are provided for RMF support within the source/390 object pump.

`RMF_SERVER_IP` = <IP address>

This supplies the IP address of the DDServer address space. This is the IP address of the system upon which the RMF Distributed Data server resides. If this is not provided, the default of the local MVS systems IP address is used. This can be used when a single instance of RMF DDServer is collecting data for multiple systems within a sysplex.

If the DDServer is not in the same LPAR as the source/390 pump, the startup parameter, RMF_SERVER_IP should be set before the source/390 pump is initialized.

Additions to the 1.5 OPC Release Notes

[Please add the following to page 8 of Chapter 2 of Tivoli Business Systems Manager OPC Release Notes Version 1.5.]

Monitoring OPC Using Exit 7

There are currently two ways for Tivoli Business Systems Manager to monitor OPC events and this depends on the number of OPC scheduled jobs in a given day. The two ways are:

- OPC Monitoring by "Exception"
- OPC Monitoring by "Exit 7"

Tivoli Business Systems Manager's OPC Monitoring by Exception is a Tivoli Services offering for customers with large OPC implementation (greater than 12,000 jobs per day). Contact your Tivoli Services team for additional information.

Tivoli Business Systems Manager's OPC Monitoring by Exit 7 actually makes use of two different mechanisms to keep track of any OPC-controlled job. One is the Exit 7 itself, which provides information to Tivoli Business Systems Manager of such states as Started, Ended, and CP Changed. The other mechanism is the trapping of the following EQQ messages:

- EQQE036I –"ENDED IN ERROR"
- EQQE037I –"LATE"
- EQQE038I –"LONG DURATION"
- EQQE039I –"LONG TIME ON INPUT QUEUE"
- EQQQ515W –"HAS WAITE(d) RESOURCE"

Exit 7 monitoring is enabled using the method described in *Tivoli Business Systems Manager OPC Release Notes Version 1.5*. For OPC Customers doing monitoring via OPC Exit 7 and EQQ messages, the following APAR PQ63193 is a prerequisite. APAR PQ63193 is available from Tivoli Business Systems Manager Level 2 Support and applies to OPC version 2.2 and 2.3.

To enable the trapping and monitoring using the EQQ Messages, the following is needed:

1. You should already have the required OPC Tracker and Controller tasks defined in Tivoli Business Systems Manager for the various Operating Systems where those Started Tasks run. If the Objects are not already defined, they should be created now.

2. Once created, you need to extract the Object (hex) "id" value assigned by the database to each of the OPC Tracker and Controller tasks. There are two ways to do this:
   - Traditional Console: Hold down the CTRL and Shift Keys while double clicking on the object to display the Properties page. The information will be displayed on the title line and will be in parentheses.
   - Java Console: With the Operating System expanded, locate the OPC Tracker and Controller tasks. While pressing the CTRL and Shift Keys, press the F12 key. The information will be displayed next to the name of the object.

   Write down the hexadecimal number displayed for each of the Tracker and Controller Tasks. The first six digits (reading from left to right) are all you need.
3. On the SQL Server machine, copy the sqi procedure `Sample_OPC_WTO_MessageDescription.sqi` to a new filename (e.g. `CustomOPCWTOMessages.sqi`). This procedure can be found in the directory TivoliManger\sql.
4. Open the new file for editing (e.g. using NOTEPAD.EXE). Follow the instructions within the file and select the appropriate option (depending on whether or not the OPC EQQ messages have been customized at your site). For example:

   `OPC_STC_MESSAGE_DESCRIPTIONS_01(0xDE6)` -- Default OPC message processing

   `OPC_STC_MESSAGE_DESCRIPTIONS_02(0xDE6)` -- Customized OPC message processing sample
5. Delete the unneeded `OPC_STC_MESSAGE_DESCRIPTIONS_` option and repeat the entries left for each OPC Controller and Tracker defined to Tivoli Business Systems Manager. The `OPC_STC_MESSAGE_DESCRIPTIONS_` option you select, must be between the BEGIN and END SAMPLE lines.
6. Substitute hexadecimal number between the parenthesis, e.g."0xDE6"with the ID of the OPC Controller and Tracker STC to receive the message (see step number 2 above). Each Controller and Tracker should have one `OPC_STC_MESSAGE_DESCRIPTIONS_` entry.
7. Once done, save the file.
8. Compile the SQI to SQL using the following command line:

   Syntax: `sh clsql <input_file.SQI>`

   Example: `D:\TivoliManager\sql> sh clsql CustomOPCWTOMessages.SQI`
9. Apply the generated SQL file to your database with `ISQL/w` or via the command line:

   Syntax: `isql -S<servername> -U<user> -P<password> -n -i<input_file.SQL>`

   Example: `D:\TivoliManager\sql> isql -Snjros1d61 -Usa -Punknown -n -iCustomOPCWTOMessages.SQI`
10. Verify correct insertion into the database by running the following ISQL: (substituting 0xDE6 with the appropriate Object ID) Example: `select * from MessageDescription_T where obj_cid='STC' and obj_id=0xDE6`
11. Recycle Propagation.
12. Once the Tivoli Business Systems Manager source/390 object pump is restarted or the OPC Controller/Tracker are re-registered, the Tivoli Business Systems Manager source/390 object pump will start trapping all of the above EQQ messages. Please note that the OPC Exit 7 implementation needs to be done either concurrently or prior to EQQ message trapping.

Addition to the 1.5 Messages Guide

[Please add the following information to the Administrator Response field of message GTM1940I in Chapter 1 of the Messages Guide.]

Administrator Response: If a console code page other then 437 is being used on the Tivoli Business Systems Manager servers, review the comments in the cconv.tbl file in TivoliManager\Lang\<language id>. The console code page can be determined via the SetConsoleCodePage command. Issue 'SetConsoleCodePage' from a DOS prompt and note the value of 'Current Console Code Page:'.

Change to the SA/390 Release Notes

[Please make the following changes (in bold) to the table on page 21 of the SA/390 Release Notes that were included in Patch 24.]

STOPPED	Inactive	Active	Inactive	Green	High	The application has been shut down by an external agent such as an operator console. SA cannot restart it and will not allow the MVS Automatic Restart Manager to restart it. It is anticipated that this message will not be received while in the desired state. If so, a schedule violation should be generated.
	Active	Active	Inactive	**Green**	**Ignore**	

Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.