HP Business Service Management

for the Windows and Linux operating systems

Software Version: 9.12

Using Transaction Management Version 9.10

Document Release Date: November 2011
Software Release Date: November 2011

Legal Notices

Warranty

The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

The information contained herein is subject to change without notice.

Restricted Rights Legend

Confidential computer software. Valid license from HP required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Copyright Notices

© Copyright 2005 - 2011 Hewlett-Packard Development Company, L.P.

Trademark Notices

TransactionVision® is a registered trademark of the Hewlett-Packard Company.

Adobe® and Acrobat® are trademarks of Adobe Systems Incorporated.

AMD and the AMD Arrow symbol are trademarks of Advanced Micro Devices, Inc.

Google™ and Google Maps™ are trademarks of Google Inc.

Intel®, Itanium®, Pentium®, and Intel® Xeon® are trademarks of Intel Corporation in the U.S. and other countries.

iPod is a trademark of Apple Computer, Inc.

Java is a registered trademark of Oracle and/or its affiliates.

Microsoft®, Windows®, Windows NT®, Windows® XP, and Windows Vista® are U.S. registered trademarks of Microsoft Corporation.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates.

UNIX® is a registered trademark of The Open Group.

Acknowledgements

This product includes software developed by the Apache Software Foundation (**http://www.apache.org**).

This product includes software developed by the JDOM Project (**http://www.jdom.org**).

This product includes software developed by the MX4J project (**http://mx4j.sourceforge.net**).

Documentation Updates

The title page of this document contains the following identifying information:

- Software Version number, which indicates the software version.
- Document Release Date, which changes each time the document is updated.
- Software Release Date, which indicates the release date of this version of the software.

To check for recent updates, or to verify that you are using the most recent edition of a document, go to:

http://h20230.www2.hp.com/selfsolve/manuals

This site requires that you register for an HP Passport and sign-in. To register for an HP Passport ID, go to:

http://h20229.www2.hp.com/passport-registration.html

Or click the **New users - please register** link on the HP Passport login page.

You will also receive updated or new editions if you subscribe to the appropriate product support service. Contact your HP sales representative for details.

Support

Visit the HP Software Support web site at:

http://www.hp.com/go/hpsoftwaresupport

This web site provides contact information and details about the products, services, and support that HP Software offers.

HP Software online support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valued support customer, you can benefit by using the support web site to:

- Search for knowledge documents of interest
- Submit and track support cases and enhancement requests
- Download software patches
- Manage support contracts
- Look up HP support contacts
- Review information about available services
- Enter into discussions with other software customers
- Research and register for software training

Most of the support areas require that you register as an HP Passport user and sign in. Many also require a support contract. To register for an HP Passport ID, go to:

http://h20229.www2.hp.com/passport-registration.html

To find more information about access levels, go to:

http://h20230.www2.hp.com/new_access_levels.jsp

Table of Contents

Welcome to This Guide 15
How This Guide Is Organized 15
Who Should Read This Guide 16
Transaction Management Documentation 16
How Do I Find the Information That I Need? 18
Additional Online Resources 20
Documentation Updates 21

PART I: TRANSACTIONVISION ADMINISTRATION

Chapter 1: Introducing TransactionVision 25
TransactionVision Overview 26
Managing User Permissions 30
Tips for Using Browsers 31
The TransactionVision Demo 31
How to Administer TransactionVision 32
How to Set Up the TransactionVision Demo 34

Chapter 2: Processing Servers 39
Processing Servers Overview 40
Key Configuration Settings for Processing Servers 43
How to Create a Processing Server 48
How to Remove a Processing Server 49
How to Modify a Processing Server 49
How to Synchronize a Processing Server's Configuration 50
How to Recover a Processing Server 50
Processing Server User Interface 51
Troubleshooting and Limitations 75

Chapter 3: Analyzers 83
Analyzers Overview 84
Key Configuration Concepts for Analyzers 85
Correlation For DataPower Events 96
How to Add an Analyzer to the Deployment Environment 106
How to Assign a Communication Link to an Analyzer 107
How to View Analyzer Status 108
How to Stop and Restart Event Collection 108
How to Stop and Restart an Analyzer 109
How to Remove an Analyzer from the Deployment Environment 110
How to Recover an Analyzer 111
How to Enable Failure Mode on an Analyzer 112
Analyzers User Interface 117
Troubleshooting and Limitations 142

Chapter 4: Communication Links 155
Communication Links Overview 156
Message Queues in Communication Links 157
Message Queue Configurations 160
Message Queue Requirements 162
Default Communication Links 171
How to Create a Communication Link 174
How to Assign a Communication Link to an Analyzer 175
How to Test a Communication Link 175
How to Modify a Communication Link 177
How to Assign a Data Collection Filter to a Communication Link 177
How to Monitor a Communication Link 178
How to Configure DataPower Communication Links for Multiple Analyzers 178
How to Remove a Communication Link 180
Communication Links User Interface 181
Troubleshooting and Limitations 205

Chapter 5: Data Collection Filters 207
Data Collection Filters Overview 208
Categories of Filter Criteria 209
About Building the Data Collection Filter 210
Guidelines for Good Data Collection Filtering 211
How to Create a Data Collection Filter 213
How to Assign a Data Collection Filter to a Communication Link 215
How to Modify a Data Collection Filter 215
How to Remove a Data Collection Filter 216
Data Collection Filters User Interface 218

Chapter 6: Queries 259
Queries Overview 260
Optimizing Query Performance 260
How to Add a Query 262
How to Modify a Query 264
How to Test a Query 264
Queries User Interface 265

Chapter 7: Jobs 315
Built-in Jobs 316
Custom Jobs 317
How to Modify the Built-in Jobs 318
How to Add a Custom Job 318
Jobs User Interface 320

Chapter 8: Advanced Configuration 331
About Advanced Configuration 332
How to Modify the XDM Files 334
How to Modify the Built-In SonicMQ Broker Settings 334
How to Modify the Maximum File Size for User Data Display 336
How to Enable Session Tracking 336
Advanced Configuration User Interface 340

Chapter 9: Working with Other Applications in BSM 347
Service Health 348
MyBSM 353
Service Level Management 353
End User Management 355
HP Diagnostics 357
Business Process Insight 358
TransactionVision Data in the BSM Profile Database 361
TransactionVision Data in the RTSM 363
Exporting TransactionVision Data to BSM 363
How to Verify or Modify Communication Between BSM and TransactionVision 364
How to Enable Communication Between RUM and TransactionVision 364
How to Enable Communication Between BPI and TransactionVision 365
Troubleshooting and Limitations 365

Chapter 10: Administration Utilities 367
AnalyzerManager 368
ConfigurationTool 371
CorrelationUtil 373
CreateSqlScript 384
DataUtil 388
DeleteEvents 391
FlushStatusUtil 399
ManageQueue 400
MigrateDB 408
nanny 408
PartitionUtil 411
PassGen 413
rebind_agent 414
rebind_tux_sensor 415
runSupportSnapshot 416
SetupModule 419
TimeServer 420
ValidateXml 421

Chapter 11: Agent Error Messages 423
Error Messages 424

Chapter 12: APIs Monitored by Agents 467
CICS Agent 468
WebSphere MQ Agent 468
WebSphere MQ-IMS Bridge Agent 469
Servlet Technology in the Java Agent 469
JMS Technology in the Java Agent 471
EJB Technology in the Java Agent 473
JDBC Technology in the Java Agent 474
Tuxedo Agent 475
NonStop TMF Agent 476
.Net Agent 476
DataPower Agents 477

Chapter 13: Troubleshooting....479
Enabling SHLIB_PATH on HP-UX....481
Displaying Perl and Java Program Names....481
Null Pointer Exception After BSM Login....482
Monitoring WebSphere MQ Triggered Programs....482
Monitoring Client Applications....483
Optimizing Performance....484
Missing Events....484
Queue Manager or Event Broker Channel Limitations....485
Servlet and EJB Technology Problems....485
JMS Technology Problems....486
Monitored Applications Troubleshooting....486
Event Queue Cleanup....487
Invalid Schema and Communication Link IDs....487
Problems Importing System Tables into a Database....490
Connection to Database Is Lost....494
Problems Using International Characters....494
Agent Configuration Queue Manager or Event Broker....496
Analyzer Configuration Queue Manager or Event Broker....498
Schemas Missing New Tables....499
Database Deadlocks in IBM DB2 Databases....499
Unable to Communicate with Processing Server after Enabling SSL500

PART II: BUSINESS TRANSACTION ADMINISTRATION

Chapter 14: Introducing Transaction Management....505
Transaction Management Overview....506
Business Transactions Overview....507
How To Choose a Transaction Management Workflow....511

Chapter 15: Business Transaction Tracing 513
Transaction Tracing Overview 514
Transaction Tracing Rules 515
Importing/Exporting Transaction Tracing Rules 519
Match Conditions 520
How to Set Up a Business Transaction for Tracing 521
How to Create a Business Transaction CI 525
How to Create a Match Condition 525
How to Assign a Rule to the Appropriate Analyzers 531
How to Copy a Rule to Create a New One 532
How to Group/Ungroup Rules 532
How to Stop Tracing on a Business Transaction 533
How to Send Events to BPI 534
How to Create Custom Alerts 536
Business Transaction Tracing User Interface 544
Troubleshooting and Limitations 572

Chapter 16: Business Transaction Monitoring 573
Business Transaction Monitoring Overview 574
How to Set Up a Business Transaction for Monitoring 576
Transaction Configuration Page, Monitoring Tab 578

Chapter 17: Business Transaction Data Model 581
Business Transaction Data Model Overview 582
How to Add a Column of Data to the Transaction Data Model 584
Transaction Data Model User Interface 585

Chapter 18: Custom Correlation 589
Custom Correlation Overview 590
About Custom Event-to-Event Correlation 591
About Custom Tracking ID Correlation 593
About Modification Rules 594
Managing Custom Correlation and Modification Rules 597
How to Create Custom Tracking ID Rules 599
How to Replace a Standard Tracking ID With a Tracking ID Rule 601
How to Clear a Standard Tracking ID Rule 603
How to Create Event-to-Event Correlation Rules 604
How to Create Modification Rules 607
How to Test a Modification Rule's Regular Expression 612
How to Use Event Modification With Custom Correlation - Work Flow 613
Custom Correlation User Interface 617
Troubleshooting and Limitations 632

PART III: REPORTS AND TOPOLOGIES

Chapter 19: Introducing Transaction Management Reports and Topologies....635
Reports and Topologies Overview....636
How the Reports and Topologies Are Related....637
Drilldown Workflows....637
Key Information Reported For a Transaction....641
Reports and Topologies Workflows....645

Chapter 20: Transaction Management Reports....647
Reports Overview....648
How to View Metrics Aggregated by Transaction Type....649
How to View Metrics Aggregated by Web Session....649
How to View Metrics for a Particular Transaction Type....649
How to View Metrics at the Event Level....650
How to View Metrics Aggregated by Application Server....650
Reports User Interface....651
Troubleshooting and Limitations....715

Chapter 21: Transaction Topologies....717
Transaction Topology Overview....718
A Closer Look at the Flow Map....720
How to View the Aggregated Topology of a Business Transaction...738
How to View the Instance Topology of a Business Transaction....739
Transaction Topologies User Interface....740
Troubleshooting and Limitations....753

Chapter 22: Component Topology....755
Component Topology Overview....756
Static Mode Versus Dynamic Mode....758
A Closer Look at the Graph....759
How to View the Component Topology....770
How to Change the Layout of the Graph....772
How to Modify Component Groupings....776
How to Configure Arrow and Node Labels....781
How to Print the Component Topology Graphs....787
Component Topology Page....792
Troubleshooting and Limitations....806

Index....809

Welcome to This Guide

This guide describes how to administer TransactionVision and how to use the Transaction Management application in HP Business Service Management (BSM).

This chapter includes:

- How This Guide Is Organized on page 15
- Who Should Read This Guide on page 16
- Transaction Management Documentation on page 16
- How Do I Find the Information That I Need? on page 18
- Additional Online Resources on page 20
- Documentation Updates on page 21

How This Guide Is Organized

The guide contains the following parts:

Part I TransactionVision Administration

Describes how to initially configure and maintain the TransactionVision deployment environment.

Part II Business Transaction Administration

Describes how to define and maintain the Business Transaction CIs monitored by TransactionVision or HP Diagnostics.

Part III Reports and Topologies

Describes how to view and customize reports and topologies of information related to the Business Transactions.

Who Should Read This Guide

This guide is intended for the following users of BSM:

- Transaction Management administrators
- Transaction Management end users

Readers of this guide should be knowledgeable about navigating and using enterprise applications, and be familiar with BSM and enterprise monitoring and management concepts.

Transaction Management Documentation

Transaction Management documentation provides information on using the Transaction Management application of BSM and deploying and administering the TransactionVision components in the BSM deployment environment.

The Transaction Management documentation includes:

- The *TransactionVision Deployment Guide* describes the installation and configuration of the TransactionVision-specific components in the HP Business Service Management deployment environment. This guide is available as a PDF in the BSM Online Documentation Library.
- The *Using Transaction Management* guide describes how to set up and configure TransactionVision to track transactions, and how to view and customize reports and topologies of business transactions. This guide is available as the Transaction Management Portal or as a PDF in the BSM Online Documentation Library.

- The *TransactionVision Planning Guide* contains important information for sizing and planning new installations of TransactionVision. This guide is available on the HP Software Product Manuals web site at http://h20230.www2.hp.com/selfsolve/manuals.
- The *TransactionVision Advanced Customization Guide* contains information for how the TransactionVision platform can be extended and customized to achieve further control over its various functions. This guide is available as a PDF in the BSM Online Documentation Library: **Transaction Management Portal** > **TransactionVision Administration** > **Advanced Customization Using the APIs** topic.

Note: Updates to these guides sometimes occur independently of the software. See "Documentation Updates" on page 21 belowfor information on how to get the most current documentation.

Additional Transaction Management documentation can be found in the following areas of the Documentation Library:

- **Release notes**. Provides a list of version limitations and last-minute updates. From the HP Business Service Management DVD root directory, double-click **BSM_<version_number>_Release_Notes.pdf**. You can also access the most updated release notes file from the HP Software Support Website.
- **What's New**. Provides a list of new features and version highlights. In HP Business Service Management, select **Help** > **What's New**.
- **Online Documentation Library**. The Documentation Library is an online help system that describes how to work with HP Business Service Management and the Transaction Management application. You access the Documentation Library using a Web browser. For a list of viewing considerations, see "Viewing the HP Business Service Management Site" in chapter 6 of the *HP Business Service Management Deployment Guide* PDF.

To access the Documentation Library, in HP Business Service Management, select **Help** > **Documentation Library**. Context-sensitive help is available from specific HP Business Service Management pages by selecting **Help** > **Help on this page** and from specific windows by clicking the Help button. For details on using the Documentation Library, see "Working with the HP Business Service Management Documentation Library" in Platform Administration.

How Do I Find the Information That I Need?

This guide is part of the HP Business Service Management Documentation Library. This Documentation Library provides a single-point of access for all Business Service Management documentation.

You can access the Documentation Library by doing the following:

- In Business Service Management, select **Help** > **Documentation Library**.
- From a Business Service Management Gateway Server machine, select **Start** > **Programs** > **HP Business Service Management** > **Documentation**.

Topic Types

Within this guide, each subject area is organized into topics. A topic contains a distinct module of information for a subject. The topics are generally classified according to the type of information they contain.

This structure is designed to create easier access to specific information by dividing the documentation into the different types of information you may need at different times.

Three main topic types are in use: **Concepts**, **Tasks**, and **Reference**. The topic types are differentiated visually using icons.

Topic Type	Description	Usage
Concepts	Background, descriptive, or conceptual information.	Learn general information about what a feature does.
Tasks	**Instructional Tasks.** Step-by-step guidance to help you work with the application and accomplish your goals. Some task steps include examples, using sample data. Task steps can be with or without numbering: ➤ **Numbered steps.** Tasks that are performed by following each step in consecutive order. ➤ **Non-numbered steps.** A list of self-contained operations that you can perform in any order.	➤ Learn about the overall workflow of a task. ➤ Follow the steps listed in a numbered task to complete a task. ➤ Perform independent operations by completing steps in a non-numbered task.
	Use-case Scenario Tasks. Examples of how to perform a task for a specific situation.	Learn how a task could be performed in a realistic scenario.

Topic Type	Description	Usage
Reference	**General Reference**. Detailed lists and explanations of reference-oriented material.	Look up a specific piece of reference information relevant to a particular context.
	User Interface Reference. Specialized reference topics that describe a particular user interface in detail. Selecting **Help on this page** from the Help menu in the product generally opens the user interface topics.	Look up specific information about what to enter or how to use one or more specific user interface elements, such as a window, dialog box, or wizard.
Troubleshooting and Limitations	**Troubleshooting and Limitations**. Specialized reference topics that describe commonly encountered problems and their solutions, and list limitations of a feature or product area.	Increase your awareness of important issues before working with a feature, or if you encounter usability problems in the software.

Additional Online Resources

Troubleshooting & Knowledge Base accesses the Troubleshooting page on the HP Software Support Web site where you can search the Self-solve knowledge base. Choose **Help** > **Troubleshooting & Knowledge Base**. The URL for this Web site is http://h20230.www2.hp.com/troubleshooting.jsp.

HP Software Support accesses the HP Software Support Web site. This site enables you to browse the Self-solve knowledge base. You can also post to and search user discussion forums, submit support requests, download patches and updated documentation, and more. Choose **Help** > **HP Software Support**. The URL for this Web site is www.hp.com/go/hpsoftwaresupport.

Most of the support areas require that you register as an HP Passport user and sign in. Many also require a support contract.

To find more information about access levels, go to:

http://h20230.www2.hp.com/new_access_levels.jsp

To register for an HP Passport user ID, go to:

http://h20229.www2.hp.com/passport-registration.html

HP Software Web site accesses the HP Software Web site. This site provides you with the most up-to-date information on HP Software products. This includes new software releases, seminars and trade shows, customer support, and more. Choose **Help** > **HP Software Web site**. The URL for this Web site is www.hp.com/go/software.

Documentation Updates

HP Software is continually updating its product documentation with new information.

To check for recent updates, or to verify that you are using the most recent edition of a document, go to the HP Software Product Manuals Web site (http://h20230.www2.hp.com/selfsolve/manuals).

To search for TransactionVision documentation, choose **Transactionvision**, the desired product version and operating system, and click **Search**.

Part I

TransactionVision Administration

1

Introducing TransactionVision

This chapter includes:

Concepts

- TransactionVision Overview on page 26
- Managing User Permissions on page 30
- Tips for Using Browsers on page 31
- The TransactionVision Demo on page 31

Tasks

- How to Administer TransactionVision on page 32
- How to Set Up the TransactionVision Demo on page 34

Concepts

TransactionVision Overview

TransactionVision is a product that delivers a transaction tracing solution to the Transaction Management application. TransactionVision non-intrusively records individual events related to transactions flowing through a system. Then, TransactionVision's patented “Transaction Constructor” algorithm assembles those events into coherent business transactions. The reports and topologies of Transaction Management present the interaction of a business transaction among all the components of your system in tables and graphs.

This section includes:

- "TransactionVision Architecture" on page 26
- "Event Flow" on page 28
- "What Events are Collected?" on page 29
- "Data Collection Filters" on page 29
- "What TransactionVision Can Tell You About Your Business Transactions" on page 30

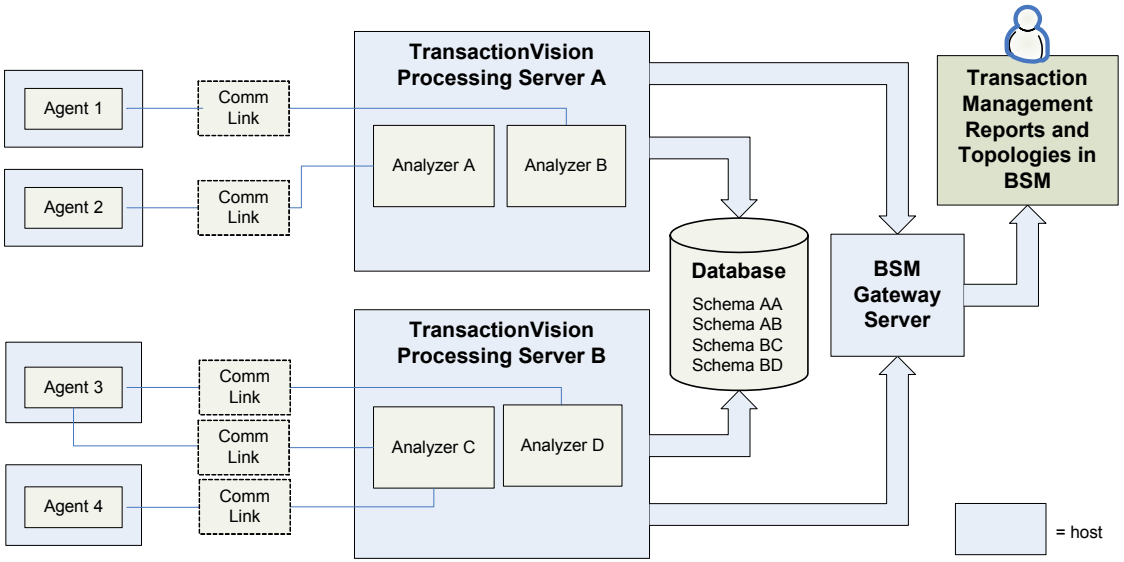
TransactionVision Architecture

TransactionVision consists of the following major components:

- **Processing Servers.** Container for the TransactionVision components that deliver the core functionality of transaction tracing.
- **Analyzers.** Processes incoming events collected by the agents and constructs corresponding business transactions.

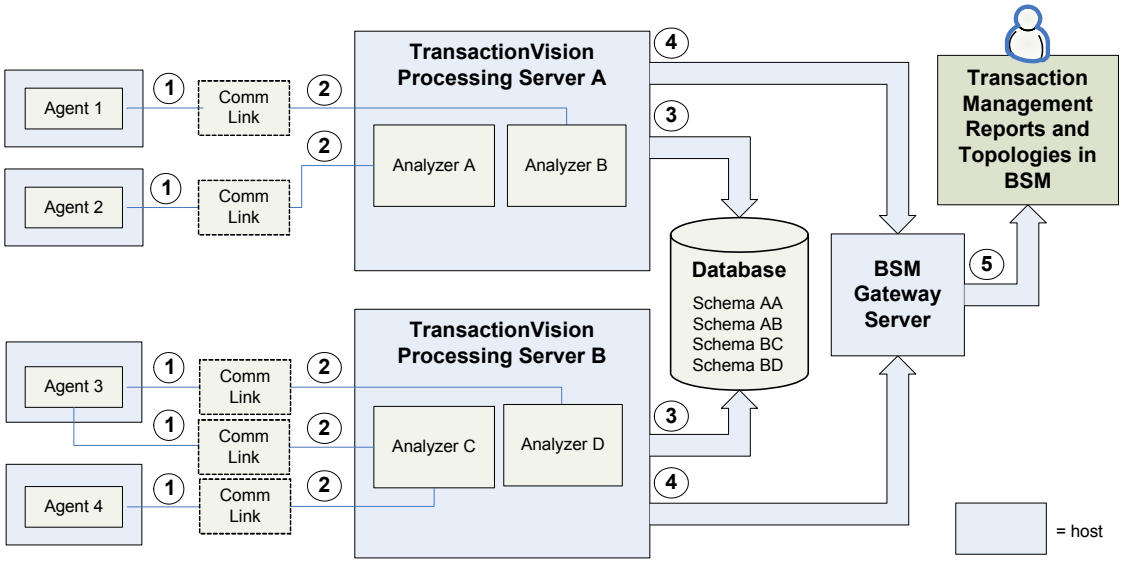
- **Agents.** Monitors transactional events in applications and collect those that meet certain criteria. Events are API level interactions between an application and the middleware it uses to carry out a transaction.
- **Communication Links.** Enables agents to communicate from the host on which an application is being monitored to the Analyzer. Communication links contains message queues which are managed by a messaging middleware product; either the built-in one can be used or one from a third-party.
- **Database.** Contains a schema for the event information collected by each Analyzer. This database is independent from the Business Availability Center databases.

The following diagram shows these components in a sample deployment environment. This environment has two Processing Servers that are each running two Analyzers. Each Analyzer is being sent events collected by agents. Agents are installed on the same host as the application they are monitoring:



Event Flow

Events flow through each component in the architecture as follows:



The numbered labels in the diagram have the following meaning:

1	Agents collect event data based on criteria and populate event queues in the communication links.
2	Analyzers pull event data from their associated event queues – this is known as raw event data. Analyzers process the raw event data and correlate events into business transactions.
3	This business transaction information as well as event level data and correlation data is stored in the TransactionVision database.
4	The Processing Servers send business transaction metrics to HP Business Service Management, where they get stored in the Profile database. This process is managed by a built-in TransactionVision job named BSM Aggregate TV Data Sample.
5	Transaction Management reports and topologies use this data, as well as other HP Business Service Management applications that report on the KPIs such as Service Health.

What Events are Collected?

An agent's job is to watch transactional events in applications and collect those that meet certain criteria. Transactional events are API-level interactions between an application and the middleware it uses to hand off control or data to another component, for example:

- Messaging API calls, such as WebSphere MQ, TIBCO EMS.
- J2EE Servlet or JSP invocations, JDBC calls, JMS calls, or EJB invocations (note: for J2EE there is a common install with the HP Diagnostics Agent).
- CICS API calls (File control, program control, and so forth).
- End user experience data. Real User Monitor can feed end user performance data to TransactionVision, capturing the timing data for the front end user experience part of a web transaction. This data is correlated together with other transactional event data from the TransactionVision Agents to form an end to end picture of the transaction.

For more information, see "APIs Monitored by Agents" on page 467.

Data Collection Filters

Data collection filters assigned to a communication link determine the amount and type of information collected by each agent. Data collection filters specify criteria such as the following:

- About which technologies, hosts, programs, or APIs is information collected.
- About which CICS regions, transactions, or job names is information collected.
- About which queues or queue managers is information collected.
- About which servlets, WebSphere applications, WebSphere servers and URIs is information collected.
- For which time range is information collected.
- What is the level of detail to be collected, such as API name only, API name and call arguments, or API name, call arguments, and data buffer segment. For the Servlet agent, the default data buffer size is 1K.

For more information, see "Data Collection Filters" on page 207.

What TransactionVision Can Tell You About Your Business Transactions

Transaction Management reports and topologies can tell users the following:

- The state of the transaction. For example, a stock buy transaction would fail if the account balance was less than $10,000, which is required to do the purchase, or a margin buy would fail if the account ratio is below a minimum balance threshold.
- A business value associated with the transaction.
- Whether the transaction has exceeded its threshold.

For more information about the questions TransactionVision can answer, see "Reports and Topologies Workflows" on page 645.

Managing User Permissions

User permissions determine what operations a user can perform in the Transaction Management application and administration pages. User permissions are initially defined during deployment of the TransactionVision components. For information about user permissions, see the "Security" chapter in the *HP TransactionVision Deployment Guide* PDF.

Tips for Using Browsers

The following information provides useful tips when using browsers to view your reports. For a list of supported browsers, see the *HP TransactionVision Deployment Guide* PDF.

- Users who have browser pop-up blockers must disable them for the TransactionVision user interface. Otherwise, certain reports may open windows with Session Expired error messages, and help pages are inaccessible. See the documentation for your pop-up blocker to determine how to add TransactionVision to the list of web applications for which pop-ups are enabled.
- Many TransactionVision reports and topologies contain session information and are temporary. Therefore, creating browser bookmarks to return to these pages at a later time may not work.
- If you set up the TransactionVision host and port setting in BSM to be **https://**, then you must login to BSM with **https://**.

The TransactionVision Demo

To help you become familiar with TransactionVision's features and capabilities, TransactionVision provides a sampling of trade events that can be processed by the Analyzer and viewed on the reports and topologies. The Trade sample contains simulated transaction data spanning over one month. This simulated data is for demonstration purposes only.

For details, see "How to Set Up the TransactionVision Demo" on page 34.

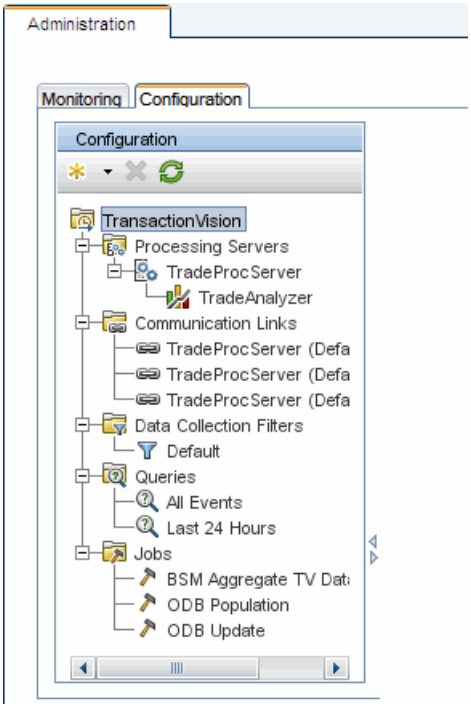
Tasks

How to Administer TransactionVision

The user acting in the role of TransactionVision Administrator performs the following tasks:

- Deploys and maintains one or more TransactionVision Processing Servers in the deployment environment. See Chapter 2, "Processing Servers."
- Deploys and maintains one or more TransactionVision Analyzers on the Processing Servers in the Deployment Environment. See Chapter 3, "Analyzers."
- Deploys and maintains a primary and backup Job Manager in the Deployment Environment. See "How to Create a Processing Server" on page 48.
- Deploys and maintains a primary and backup Query Manager in the Deployment Environment. See "How to Create a Processing Server" on page 48.
- For each agent in the deployment environment, defines a communication link between the agent and an Analyzer. See Chapter 4, "Communication Links."
- Defines transaction tracing rules and data to include with each transaction. See Chapter 15, "Business Transaction Tracing."
- Defines the correlation and modification rules to expand or modify event collection criteria. See Chapter 18, "Custom Correlation."
- Defines data collection filters. See Chapter 5, "Data Collection Filters."
- Monitors the built-in jobs and creates and maintains custom jobs. See Chapter 7, "Jobs."
- Keeps the TransactionVision appropriately secured, including controlling which users can access the Administration tab or the Application tab. See "Security" in the *HP TransactionVision Deployment Guide* PDF.

These tasks correspond to the items in the Configuration tab of the Transaction Management application. Only users with administrative privileges have access to this tab. This tab shows the full set of TransactionVision components currently defined in your deployment environment, for example:



These tasks are dependent on the TransactionVision components being installed. For installation information, see the *HP TransactionVision Deployment Guide* PDF.

How to Set Up the TransactionVision Demo

This task describes how to prepopulate an Analyzer schema with the **TRADE** demo set of events.

The steps include:

- "Locate the Trade Demo Sample Files" on page 34
- "Import the Transaction Definitions" on page 34
- "Import the Transaction Tracing Rules" on page 36
- "Import the Trade Demo Sample Events" on page 37
- "View the Demo Data in Reports and Topologies" on page 37

Locate the Trade Demo Sample Files

The following Trade Demo sample events files are in the **<TVISION_HOME>/samples/trade** directory.

File	Description
tvision_trade_events_91x.zip	Zip file jar file containing the Trade Demo Sample Events.
TradeTransaction.xdm	Trade demo XML database mapper transaction definition file defining trade-specific business transaction attributes.
TransactionDefinition.xml	The transaction tracing rule file.

Import the Transaction Definitions

1. In **Admin** > **Transaction Management** > **Configuration**, select the **TransactionVision** root node.
2. Navigate to **Configuration** > **XDM** in the right pane.

3 Create a new XDM file as follows.

a In the XDM properties area, click the **Create a custom XDM file** button.

b Enter **TradeTransaction** for Name, and choose **Transaction** for Document Type.

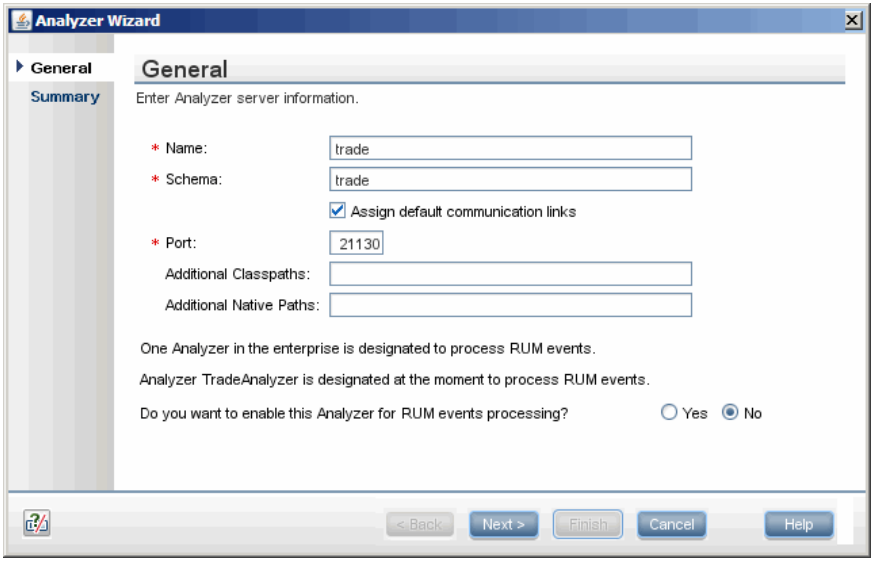
c Click **OK**.

4 Update the new **TradeTransaction** XDM file as follows:

a Delete the Mapping definition that has been pre-populated into the file, and copy the contents of **<TVISION_HOME>/samples/trade/TradeTransaction.xdm** into the Text window.

b Click **Apply** and choose **Yes** to the next dialog.

5 (left pane) Access the Analyzer wizard to create a new Analyzer with both a name and schema name of **Trade**. Specify **Assign default communication links** option and do not enable this analyzer for RUM events processing:



You can create the Analyzer on an existing Processing Server or on a new one. See "How to Create a Processing Server" on page 48.

For information about accessing the Analyzer wizard, see "How to Add an Analyzer to the Deployment Environment" on page 106.

Import the Transaction Tracing Rules

1. Navigate to **Admin** > **Transaction Management** > **Monitoring**.
2. In the Monitoring tab, click the **Transaction Configuration** button in the left pane.
3. From the drop-down list, choose **Business Transactions**.
4. Click the **New Application** button and create a new application CI named **Trade App**.
5. Under **Trade App**, create three new Business Transaction CIs: Equity, Bond, and Fund Transfer.
6. Click the **Transaction Tracing Rules** button.
7. Click **Import Transaction Rule Definition XML from a file**. Select the file **<TVISION_HOME>/samples/trade/TransactionDefinition.xml** and import it.
8. Click **Refresh**. Several Rules appear under Transaction Rules.
9. In the Monitoring tab, click the **Transaction Configuration** button.
10. Select **Bond**.
11. Click the **Tracing** tab in the right pane.
12. Select **Trace this transaction using an existing Rule Definition** and click **Configure Transaction Tracing**.
13. From the drop-down list, choose **Bond**. (Be sure you do not choose **Bond Page** instead.)
14. Set **Threshold** value to 5000 and assign the **Trade** analyzer. Click **Apply**.
15. Repeat the above steps for Equity and Fund Transfer. Choose a threshold value of 10000 for Equity, and 4000 for Fund Transfer.

Import the Trade Demo Sample Events

1. Copy and unzip the **tvision_trade_events_91x.zip** file to the file system of the host that contains the Processing Server with the Trade Analyzer.
2. Copy the file **trade_raw_events.jar** to <**TVISION_HOME**>/**logs**.
3. Navigate to and select **Admin** > **Transaction Management** > **Configuration** > **Processing Servers** > <processing server> > **Trade.**
4. On the **Configuration** > **Events** > **General** area, set **Read From Jar** in the right pane.
5. With the Analyzer still selected, click the **Status** tab, then click the **Start** button in the Analyzer status area.

   The jar file contains around 78155 events, after which the Analyzer processing can be stopped by clicking the **Stop** button.

View the Demo Data in Reports and Topologies

To view the imported data, run any report or topology specifying the Bond, Equity or Fund Transfer business transaction CIs on the Transaction Filter dialog. For a detailed overview of the reports and topologies and when to use them, see Chapter 19, "Introducing Transaction Management Reports and Topologies."

You may need to adjust the date range in the report settings before running the reports.

2

Processing Servers

This chapter includes:

Concepts

➤ Processing Servers Overview on page 40

➤ Key Configuration Settings for Processing Servers on page 43

Tasks

➤ How to Create a Processing Server on page 48

➤ How to Remove a Processing Server on page 49

➤ How to Modify a Processing Server on page 49

➤ How to Synchronize a Processing Server's Configuration on page 50

➤ How to Recover a Processing Server on page 50

Reference

➤ Processing Server User Interface on page 51

Troubleshooting and Limitations on page 75

Concepts

Processing Servers Overview

A TransactionVision Processing Server is a container for the TransactionVision components that deliver the core functionality of business transaction tracing to BSM. A Processing Server typically runs on its own host, separate from other BSM components.

The deployment environment can contain multiple Processing Servers. Each Processing Server can contain any of the following TransactionVision components:

- **Job Manager**. Manages the built-in and custom jobs that are used by TransactionVision.

  Job Managers are designated as either primary or backup. At least one Processing Server in the deployment environment must contain the primary Job Manager.

- **Query Engine.** Manages the queries that are used to populate some of the Transaction Management reports and topologies.

  Like Job Managers, Query Engines are designated as either primary or backup. At least one Processing Server in the deployment environment must contain the primary Query Engine.

- **Analyzers.** The Analyzers communicate with TransactionVision agents and process event data collected by the agents into transactions. At least one Processing Server in the deployment environment must contain an Analyzer.

Database for the TransactionVision Data

Event and transaction level data is stored in a database. Each Processing Server is assigned a particular database to which it writes the data. The reports and topologies of Transaction Management use data from all of the databases in the deployment environment.

You choose which type of database the deployment environment uses. In a production environment, the database is one of these third-party databases: Oracle, Microsoft SQL Server, or IBM DB2.

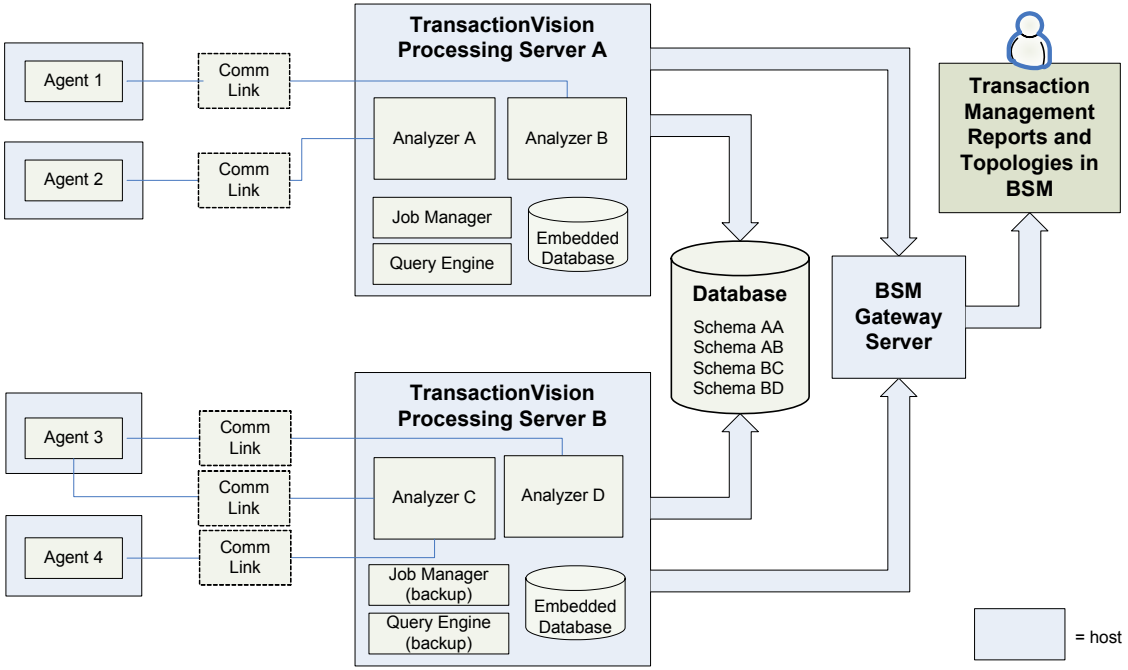
In a test environment where the volume of event collection is in the range of 100,000 to 200,000 events, the embedded database can be used instead of a third-party database.

The embedded database is also used to store configuration information for the Processing Servers.

For more information about setting up the database, see the *HP TransactionVision Deployment Guide* PDF.

Processing Servers in the Deployment Environment

The following diagram shows a sample deployment environment with two Processing Servers writing to the same third-party database. Each Processing Server has two Analyzers running. One Processing Server is dedicated to running the primary Job Manager and the Query Engine; the other Processing Servers is running the backup Job Manager and Query Engine.



Key Configuration Settings for Processing Servers

Processing Servers are configurable. This section describes some of the common configuration areas.

This section includes the following topics:

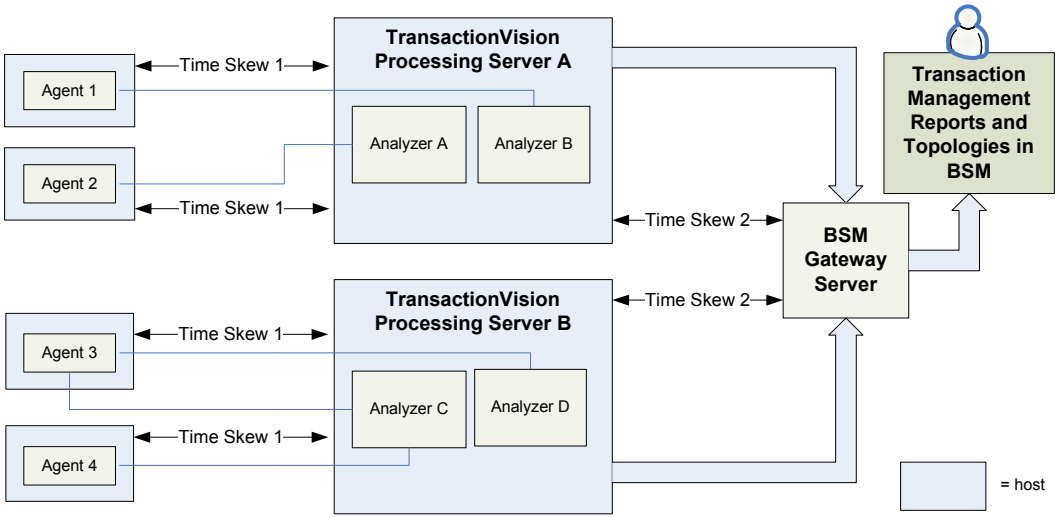
- "Communication with the BSM Gateway Server" on page 43
- "Time Skew Information" on page 44
- "Database Connection Settings" on page 45
- "Configuration Message Expiry Setting" on page 46

Communication with the BSM Gateway Server

See "Configuration Tab, BSM Settings Tab" on page 343.

Time Skew Information

Because of the time sensitivity of the collected event data, it is critical that all hosts in the deployment environment are synchronized with respect to time. There are two areas where differences in time, known as time skews, occur:



- Time Skew 1

  Time differences between the agent hosts and the Processing Server host are managed by the following configuration settings on the Processing Server:

  time skew check interval
  time skew history size
  time skew latency threshold
  time skew message priority
  time skew reply wait interval
  time skew retry threshold

- Time Skew 2

  Time differences between the BSM Gateway Server and the Processing Server are managed by the BSM time server. You set the **Time skew Web URL** property to identify the location of the time server.

Note: If the Processing Server is running on a VMware system, the system time of the VMware system is automatically used instead of the host system time in calculating all time skews.

For information about setting time skew settings, see "Processing Server Page, Configuration Tab" on page 66.

Some time skew settings can also be specified at the Analyzer level. See "Configuration Tab, General Tab" on page 125.

Database Connection Settings

The Processing Server requires constant access to the database. The Processing Server connects to the database via JDBC. When you create a Processing Server, you are prompted for the database type, host name, database name, database port, user name, and password which are used for establishing the JDBC connection to the database.

Processing Servers use third-party JDBC drivers from DataDirect for accessing the database. If there is a need to use the original vendor JDBC drivers instead, specify the JDBC driver in the JDBC property field as described in "Database Connection Parameters Page" on page 56.

If the database becomes unavailable to a Processing Server, the Processing Server logs all exceptions to the logs directory and does not process any more events. You can configure persistence for the event transport which saves the events and process them later when the processing server is connected to the database. Event persistence is specified as a communication link option.

The following properties control the Processing Server in the event the database connection is lost:

- Reconnect interval
- Reconnect timeout
- Query timeout

For more information about these settings, see "Database Connection Parameters Page" on page 56.

For more information about setting event persistence, see "Event Options Page" on page 196.

Configuration Message Expiry Setting

One of the configuration options for a Processing Server is Configuration Message Expiry. This setting controls when event collection stops in the event the Analyzers cannot send a configuration message. This happens if an Analyzer cannot connect to the queue manager, or if the Processing Server on which the Analyzer is running unexpectedly stops. For more information about configuration messages, see "Message Queues in Communication Links" on page 157.

By default, the setting is 60 minutes. In this case, if an agent does not receive a configuration message after 60 minutes the last configuration message no longer applies and event collection stops.

Depending on your environment, you may need to adjust the setting to be lower or higher. The following table summaries the reasons for high and low configuration message expiry values:

Reasons for a high Configuration Message Expiry setting	If it is critical that your agents do not miss collecting any events, set the configuration expiry to -1. The configuration messages will not expire. You can also set the configuration expiry to a value higher than the default, such as 120 minutes. This means that agents will continue to collect events and place messages on the queues until the time is reached. Make sure you have the necessary system resources to handle the backed up events in the event queue. See the documentation for the event transport provider for details.
Reasons for a low Configuration Message Expiry	If keeping all events is not important or you cannot accommodate a growing event queue, set your configuration expiry to a lower value. Agents stop collecting when the time is reached and do not collect again until they reconnect and the Analyzers sends out new messages. This prevents events from filling up the event queues.

These settings are on the "Processing Server Page, Configuration Tab" on page 66.

Tasks

How to Create a Processing Server

This task describes how to create a new Processing Server in the deployment environment.

This task includes the following steps:

- "Prerequisites" on page 48
- "Launch the Processing Server Wizard" on page 48
- "Results" on page 49

1 Prerequisites

A Processing Server must be installed on a host accessible by BSM.

You need to know the host name on which the Processing Server is installed.

If you are not using the embedded database, you need to know the about the third-party database to be used: database name and connection information, including login credentials.

For information about installing a Processing Server, see the *HP TransactionVision Deployment Guide* PDF.

2 Launch the Processing Server Wizard

a Select **Admin** > **Transaction Management**.

b Click the **Configuration** tab in the left pane.

c Click and then select **New Processing Server**.

d The Processing Server Wizard appears. Use it to define the new Processing Server. For details about this interface, see "Processing Server Wizard" on page 51.

3 Results

The new Processing Server name appears in the Configuration tab.

To assign an Analyzer to the new Processing Server, see "How to Add an Analyzer to the Deployment Environment" on page 106.

How to Remove a Processing Server

This task describes how to remove a Processing Server from the deployment environment.

1. Make sure any Analyzers created on the Processing Server are not needed. They will be deleted.
2. Select **Admin** > **Transaction Management**.
3. (left pane) Click the **Configuration** tab.
4. Select **Processing Servers** > <processing server>.
5. Right-click <processing server> and select **Delete**.
6. If you want to permanently remove the Processing Server, uninstall it. See the *HP TransactionVision Deployment Guide* PDF.

How to Modify a Processing Server

This task describes how to modify the properties of a Processing Server.

Most properties of a processing server can be changed without stopping the Processing Server. The Processing Server processes will stop and restart automatically upon configuration as needed. In rare cases, you may need to stop and restart the Processing Server processes manually. To do this, use the **nanny** utility. See "Administration Utilities" on page 367.

1. Select **Admin** > **Transaction Management**.
2. (left pane) Click the **Configuration** tab.
3. Select **Processing Servers** > <processing server>.

The processing server properties appear on the right. For details about this interface, see "Processing Server Page, Configuration Tab" on page 66.

4 (right pane) Click the **Configuration** tab.

How to Synchronize a Processing Server's Configuration

The configuration information you specify on the **Administration** > **Transaction Management** > **Configuration** tab is stored both locally in the Processing Server and in the BSM Management database.

The data in the two locations is synchronized automatically as needed. In rare cases, such as after Analyzer recovery, you may need to force the Analyzer to retrieve the current configuration from the BSM database.

To synchronize the Analyzer configuration with the BSM database:

1 Select **Admin** > **Transaction Management.**

2 (left pane) Click the **Configuration** tab.

3 Expand **Processing Servers**.

4 Right-click <processing_server> and select **Synchronize Configuration**.

How to Recover a Processing Server

If the host on which a Processing Server is running shuts down, the Processing Server services (Windows) and daemons (UNIX) are stopped. When the host starts again, the Processing Server services and daemons automatically restart. For information about the service and daemon names, see the *HP TransactionVision Deployment Guide* PDF.

The Analyzers that were running on the Processing Server will need to be recovered. See "How to Recover an Analyzer" on page 111.

Reference

Processing Server User Interface

This section describes:

- ➤ Processing Server Wizard on page 51
- ➤ Processing Servers Summary Page, Status Tab on page 61
- ➤ Processing Servers Summary Page, Job Manager Tab on page 64
- ➤ Processing Servers Summary Page, Query Engine Tab on page 65
- ➤ Processing Server Page, Status Tab on page 66
- ➤ Processing Server Page, Configuration Tab on page 66
- ➤ Processing Server Page, Log Files Tab on page 73

Processing Server Wizard

This wizard enables you to add a new Processing Server to the deployment environment.

To access	**1** Select **Admin** > **Transaction Management.** **2** (left pane) Click the **Configuration** tab. **3** Click the **New** button and select **New Processing Server.**
Important information	➤ A Processing Server must be installed before it can be added through this wizard. See the *HP TransactionVision Deployment Guide* PDF. ➤ After a Processing Server is created from this wizard its properties can be modified.
Relevant tasks	"How to Create a Processing Server" on page 48

Wizard map	This wizard contains: General Information Page > Job Manager Parameters Page > Query Engine Parameters Page > Database Selection Page > Database Connection Parameters Page > Database Advanced Parameters Page
See also	"Processing Server Page, Configuration Tab" on page 66

General Information Page

This wizard page enables you to name the Processing Server and designate a Job Manager or Query Engines to run in it.

Important information	General information about this wizard is available here: "Processing Server Wizard" on page 51.
Wizard map	The Processing Server Wizard contains: **General Information Page** > Job Manager Parameters Page > Query Engine Parameters Page > Database Selection Page > Database Connection Parameters Page > Database Advanced Parameters Page

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Elements	Description
Processing Server Name	A name for the Processing Server that is unique to the deployment environment. You may want to choose a name that represents the host on which the Processing Server is running or the database it is using.
Processing Server Host	The name of the host on which the Processing Server is installed.
Processing Server Port	The port on which the Processing Server listens. **Default value**: 21100

UI Elements	Description
Create Job Manager	Runs a Job Manager in this Processing Server. One and only one Processing Server in the deployment environment must host the primary Job Manager.
Create Query Engine	Runs a Query Engine in this Processing Server. One and only one Processing Server in the deployment environment must host the primary Query Engine.

Job Manager Parameters Page

This wizard page enables you to configure a Job Manager.

Important information	General information about this wizard is available here: "Processing Server Wizard" on page 51.
Wizard map	The Processing Server Wizard contains: General Information Page > **Job Manager Parameters Page** > Query Engine Parameters Page > Database Selection Page > Database Connection Parameters Page > Database Advanced Parameters Page

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Elements	Description
Job Manager Port	The port of the Job Manager. **Default value:** 21010
Job Manager SSL Port	The SSL port of the Job Manager. Specify this value if you are enabling SSL in the deployment environment. **Default value:** 21011
Select as Primary Job Manager	Designates this Job Manager as the primary Job Manager. If another Job Manager was designated as the primary Job Manager, it now becomes a backup Job Manager.

UI Elements	Description
Additional Classpaths	Specify the classpaths to Jar files for custom path settings or any custom bean/code.
Additional Native Paths	Specify the native paths to Jar files for custom path settings or any custom bean/code.

Query Engine Parameters Page

This wizard page enables you to configure a Query Engine.

Important information	General information about this wizard is available here: "Processing Server Wizard" on page 51.
Wizard map	The Processing Server Wizard contains: General Information Page > Job Manager Parameters Page > **Query Engine Parameters Page** > Database Selection Page > Database Connection Parameters Page > Database Advanced Parameters Page

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Elements	Description
Query Engine Port	The port of the Query Engine. **Default value:** 21000
Query Engine SSL Port	The SSL port of the Query Engine. **Default value:** 21001
Select as Primary Query Engine	Designates this Query Engine as the primary Query Engine. If another Query Engine was designated as the primary Query Engine, it will become a backup Query Engine.
Additional Classpaths	Specify the classpaths to Jar files for custom path settings or any custom bean/code.
Additional Native Paths	Specify the native paths to Jar files for custom path settings or any custom bean/code.

Database Selection Page

This page enables you to choose the database type of the database in which the Processing Server stores transaction and event data.

Important information	➤ General information about this wizard is available here: "Processing Server Wizard" on page 51. ➤ In production environments, TransactionVision requires a 3rd-party database. In test or demonstration environments where event collection volumes are low, the embedded database can be used.
Wizard map	The Processing Server Wizard contains: General Information Page > Job Manager Parameters Page > Query Engine Parameters Page > **Database Selection Page** > Database Connection Parameters Page > Database Advanced Parameters Page > Summary Page
See also	For information about which database versions are supported, see "Reviewing System Requirements" in the *HP TransactionVision Deployment Guide* PDF.

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
Select the database type in which the Processing Server will store transaction and event data	Embedded Database (Derby) Oracle IBM DB2 Microsoft SQL

Database Connection Parameters Page

This page enables you to specify the database connection information.

Important information	➤ General information about this wizard is available here: "Processing Server Wizard" on page 51. ➤ Some of the UI elements below are database specific. See the description.
Wizard map	The Processing Server Wizard contains: General Information Page > Job Manager Parameters Page > Query Engine Parameters Page > Database Selection Page > **Database Connection Parameters Page** > Database Advanced Parameters Page

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Elements	Description
Database host	Fully-qualified domain name of the host on which the database is installed. Applies to: IBM DB2, Microsoft SQL, Oracle.
Database name	Name of the database. Applies to IBM DB2.
Database sid	Name of the database. Applies to Oracle.
Database instance	Name of the database instance. Applies to Microsoft SQL.
Use Custom JDBC Driver	Some environments, such as Oracle 10g RAC, require a custom URL to access the database. Enter a custom JDBC URL and class to use with the JDBC driver manager.

UI Elements	Description
Database port	The port number where the Processing Server listens for RMI calls. Applies to: IBM DB2, Microsoft SQL, Oracle as follows. ➤ **IBM DB2** This is the TCP/IP port the database server is listening on. To identify the correct value, select the following menu items from the DB2 Control Center while the DB2 administration server is running: **All Systems** > **system_name** > **Instances** > **instance_name** Right-click and select **Setup Communications** from the context menu. Click **Properties** to get the port number your database instance is using. **Default value:** 50000 ➤ **Microsoft SQL** The port number to connect to on the SQL Server. **Default value:** 1433 ➤ **Oracle** This is the TCP/IP port the database server is listening on. To identify the correct value, run the **lsnrctl** command on the server side. Use the **status** command to find out the port number. **Default value:** 1521

UI Elements	Description
Database user	The user name that is used to connect to the database. Applies to: IBM DB2, Oracle. The user must have these privileges for database access: ➤ **IBM DB2** GRANT CREATETAB,CONNECT ON DATABASE TO USER <user>; ➤ **Oracle** grant resouce,connect to <user>; grant create user to <user>; grant create any table to <user>; grant create any index to <user>; grant select any table to <user>; grant insert any table to <user>; grant update any table to <user>; **Default value:** the currently logged in user is used to make the database connection.
Database password/ Confirm password	The password associated with the user name above. Applies to IBM DB2, Oracle.
User as Schema	Creates a TVISION schema based on a user instead of a role. Applies to Microsoft SQL.
Test Connection	Tests the connection to the database. If the connection fails, check the database connection information: host, port, user name, and password and try again.
Windows NT User	The Windows Server user (and password) are used to access the database. Applies to SQL Server.

Database Advanced Parameters Page

This page enables you to specify some advanced configuration information for the database access.

Important information	General information about this wizard is available here: "Processing Server Wizard" on page 51.
Wizard map	The Processing Server Wizard contains: General Information Page > Job Manager Parameters Page > Query Engine Parameters Page > Database Selection Page > Database Connection Parameters Page > **Database Advanced Parameters Page**

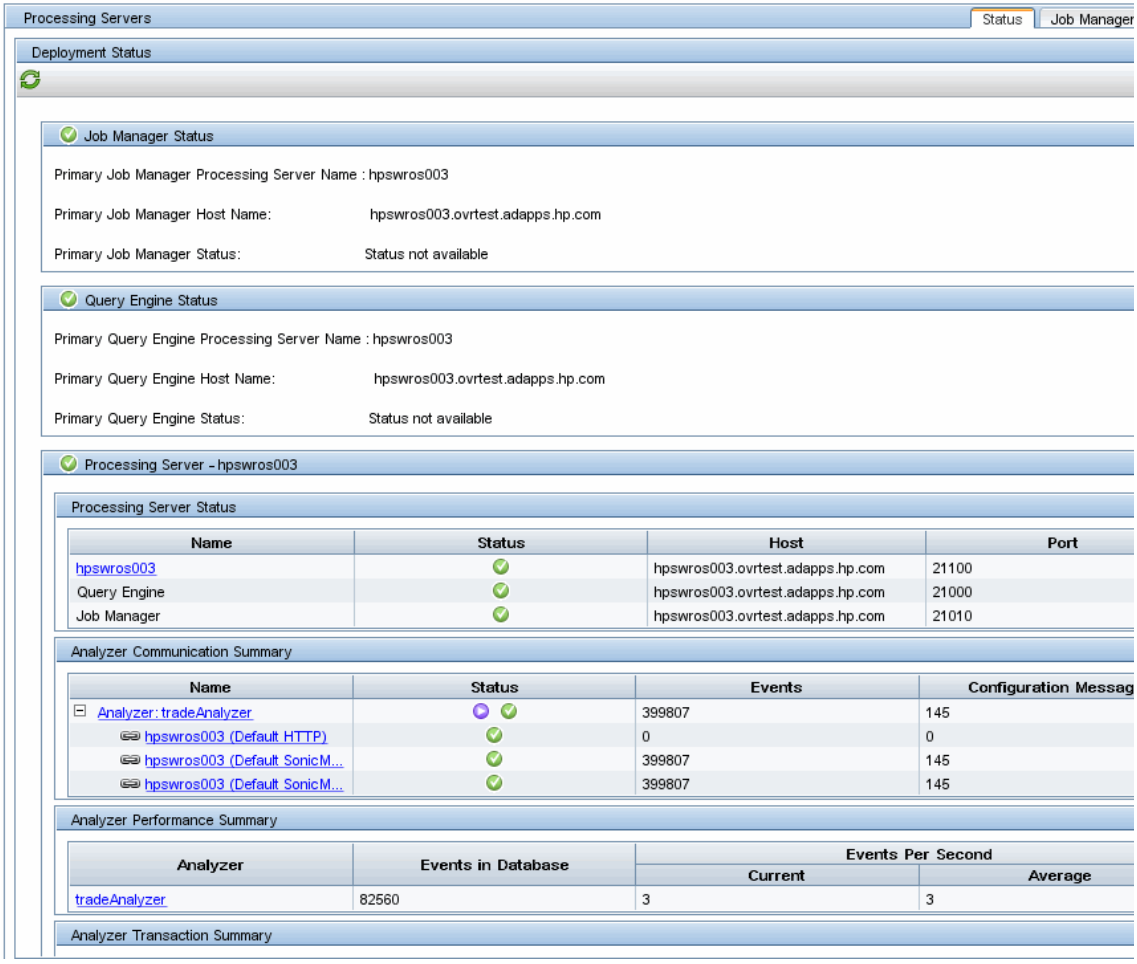
User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Elements	Description
Reconnect interval (sec)	The number of seconds the Processing Server waits before attempting to connect to the database in the event the database connection is lost. **Default value:** 10
Reconnect timeout (sec)	The total number of seconds after which the Processing Server stops attempting to connect to the database in the event the database connection is lost. **Default value:** 600
Query timeout (sec)	The number of seconds after which the Processing Server should abort a query that has not returned any data. **Default value:** 600
Excluded object types	If you have determined an object type that does not need to be specified in the query or data collection filter configuration pages, you can specify that object type here.

UI Elements	Description
Unicode bytes per character	Character-based XDM columns with the attribute unicode=true are generated with a size using this value as a multiplier of the base value. This option takes effect only if **Unicode database** is also selected on this page. **Default value**: 2 **Note**: Values larger than 3 may cause database creation problems.
Use encryption for passwords	Encrypts the entered Database Password field value before storing it. All configuration data including this is stored both locally in the embedded database and the BSM Management database.
Unicode database	Indicates whether the database has Unicode data. If set, all character-based XDM columns with the attribute unicode=true set are generated with double the byte size to allow the specified number of characters to be stored in the database. For character sets requiring more than two bytes per character, set **Unicode bytes per character** to the required number of bytes per character. **Default value**: Off
SQL trace	Logs detailed information about SQL queries issued by the Processing Server to the database. For more information, see "Processing Server Page, Log Files Tab" on page 73. **Default value**: Off
Enable WebSphere	If WebSphere MQ is the messaging middleware provider used in the TransactionVision deployment environment, specify the installation location.
Enable TIBCO EMS	If TIBCO is the messaging middleware provider used in the TransactionVision deployment environment, specify the installation location.

Processing Servers Summary Page, Status Tab

This tab enables you to view summary information about all Processing Servers in the deployment environment.



To access	**1** Select **Admin** > **Transaction Management.** **2** (left pane) Click the **Configuration** tab. **3** Select **Processing Servers**. **4** (right pane) Click the **Status** tab.
Important Information	The Processing Server status is not updated automatically; click **Refresh** to update the status.
See also	"How to Synchronize a Processing Server's Configuration" on page 50

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Elements	Description
	Refresh the list and statuses of the Processing Servers.
Job Manager Status	The Processing Server name, status, and host on which the primary Job Manager is running and the Job Manager status: Running or Not Running.
Query Engine Status	The Processing Server name and host on which the primary Query Engine is running and the Job Manager status: Running or Not running.
<for each Processing Server>	
Processing Server Status	The **Name**, **Host**, and **Port** of the Processing Server. For a description of these values, see "Database Connection Parameters Page" on page 56. The **Status** of the Processing Server: The Processing Server is running. The Processing Server is not running.

UI Elements	Description
Analyzer Communication Summary	For each Analyzer on the Processing Server: **Name.** The name of the Analyzer. **Status.** The status of the Analyzer: The Analyzer status is COLLECTING EVENTS. Hover over the status to see details. The Processing Server is running. The Processing Server is not running. Status information cannot be obtained. This typically indicates the Analyzer is not running. **Events.** The number of events collected through the communication links since the Analyzer was last started. **Config Msgs.** The number of configuration messages sent through the communication links since the Analyzer was last started.
Analyzer Performance Summary	For each Analyzer on the Processing Server: **Analyzer.** The name of the Analyzer. **Events in Database.** The number of events processed and stored in the database since the Analyzer was last started. **Events Per Second (Current/Average).** The current and average rate of event collection since the Analyzer was started.
Analyzer Transaction Summary	For each Analyzer on the Processing Server: **Name.** The transaction tracing rule assigned to each Analyzer.

Processing Servers Summary Page, Job Manager Tab

This tab enables you to view summary information about the Job Manager in the deployment environment.

To access	**1** Select **Admin** > **Transaction Management.** **2** (left pane) Click the **Configuration** tab. **3** Select **Processing Servers.** **4** (right pane) Click the **Job Manager** tab.
See also	"Jobs" on page 315

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Elements	Description
Job Manager Status	See "Processing Servers Summary Page, Status Tab" on page 61.
Job Manager Configuration	Create a new Job Manager. Edit properties of the selected Job Manager. Deleted the selected Job Manager. For each Processing Server in the deployment environment: **Processing Server** and **Host.** The Processing Server name and host. **Job Manager Port.** Click to change the Job Manager port. **Default value:** 21010 **SSL Enabled.** Click to enable or disable SSL on the Job Manager. **Default value:** false **Enable Primary Job Manager.** Check to enable the primary Job Manager on this Processing Server.

Processing Servers Summary Page, Query Engine Tab

This tab enables you to view summary information about the Query Engine in the deployment environment.

To access	**1** Select **Admin** > **Transaction Management.** **2** (left pane) Click the **Configuration** tab. **3** Select **Processing Servers.** **4** (right pane) Click the **Query Engine** tab.

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Elements (A-Z)	Description
Query Engine Status	See "Processing Servers Summary Page, Status Tab" on page 61.
Query Engine Configuration	Create a new Query Engine. Edit properties of the selected Query Engine. Deleted the selection Query Engine. For each Processing Server in the deployment environment: **Processing Server** and **Host.** The Processing Server name and host. **Query Engine Port.** Click to change the Query Engine port. **Default value:** 21000 **SSL Enabled.** Click to enable or disable SSL on the Query Engine. **Default value:** false **Select Primary Query Engine.** Check to enable the primary Query Engine on this Processing Server.

Processing Server Page, Status Tab

This tab contains the same information that appears on the Processing Servers Summary page Status tab. See "Processing Servers Summary Page, Status Tab" on page 61.

Processing Server Page, Configuration Tab

This tab enables you to change the settings of the Processing Server.

To access	**1** Select **Admin** > **Transaction Management**. **2** (left pane) Click the **Configuration** tab. **3** Select **Processing Servers** > <processing_server>. **4** (right pane) Click the **Configuration** tab.
See also	"Key Configuration Settings for Processing Servers" on page 43

The Processing Server Configuration tab includes the following areas:

- "General Area" on page 67
- "Database Area" on page 71
- "Advanced Area" on page 72

General Area

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Elements (A-Z)	Description
Initialize	See Initialization Status below.
<Initialization Status>	Indicates whether the Processing Server component on the specified host is synchronized with the Processing Server configuration settings specified on this tab. Click **Initialize** to synchronize the Processing Server. You typically only need to do this if the Processing Server has been reinstalled since it was first configured, or the BSM LW-SSO settings have changed since the Processing Server was first configured.
Configuration message expiry (min)	The expiration time in minutes of configuration messages received by agents. When the configuration messages expire, agents stop collecting events. See "Key Configuration Settings for Processing Servers" on page 43 for more information. Specify **-1** if you do not want configuration messages to expire. This setting affects all Analyzers on the Processing Server. **Default value:** 60
Configuration sync interval (min)	Interval in minutes at which existing configuration messages are resent to the agents, so that they do not unintentionally expire. **Default value**: 15 This setting affects all Analyzers on the Processing Server.
Connection retry delay (sec)	The number of seconds to wait between attempts to connect to the event queue's queue manager. This setting affects all Analyzers running on this Processing Server. **Default value:** 10

UI Elements (A-Z)	Description
Database exception threshold	The number of exceptions from the database to accept in the event collection threads before attempting to reconnect to the database. **Default value:** 3
Database reconnect interval (sec)	See the description of this property on the "Database Advanced Parameters Page" on page 59.
Debug	Enables Debug logging for all Analyzers on this Processing Server. This option is intended for troubleshooting and should not be turned on in production environments. See "Processing Server Page, Log Files Tab" on page 73.
Event get wait interval (sec)	The number of seconds to wait for an event message. Increasing this number slows down the Analyzer response to commands. This setting affects all Analyzers running on this Processing Server. **Default value:** 5
Host	The host of the Processing Server.
Name	The name of the Processing Server.
PII Objects	Cache size for PII objects. If you have a large system model that contains a lot of ProgramInstance objects, some reports such as the Instance Topology may render more quickly by increasing the cache size. See "Configuring the Cache" in "Key Configuration Concepts for Analyzers" on page 85. **Default value:** 1000
Port	The port of the Processing Server. **Default value:** 21100
SSL Port	The port of the Processing Server when enabled for SSL. **Default value:** 21101
SystemModel Objects	The number of system model objects to store. **Default value:** 1000

UI Elements (A-Z)	Description
Time skew check interval (sec)	The number of seconds between sending messages to determine the time skew across communication links. At each specified interval, the Analyzer takes time skew samplings across the communication links. The time skew is reported per Analyzer. See "Analyzer Page, Configuration Tab" on page 124. **Default value:** 3600 (one hour)
Time skew history size	The number of previous time skew samplings to consider for the best time skew to use. Having a larger history size improves the chance of finding a time skew with smaller latency, but it also increases the chance of picking a time skew that may have occurred long ago to represent a time skew that may be more off due to clock drifting. A value of 0 causes the time skew that was taken with the smallest latency to be used. This time skew is not discarded until a new time skew with equal or less latency is taken. Using this value disregards clock drift, which occurs frequently. **Default value:** 24
Time skew latency threshold (msec)	The number of milliseconds to accept when sampling time skews. If more than the specified time passes when waiting for a time skew reply, another time skew sampling is taken in hopes it has a lower latency. Any time skew samples that surpass the threshold are still kept in the time skew history because it is possible no samplings are under the suggested latency threshold. This threshold affects time skew sampling across communication links. **Default value:** 100

UI Elements (A-Z)	Description
Time skew message priority	The WebSphere MQ Message Priority on the Confirmation of Arrival (COA) time skew messages. The default value causes the default priority to be used. Since configuration and event messages use the default priority, it is a good idea to set this value at a value higher than the default. Setting a higher value ensures that time skew messages travel through channels faster than other messages that may be backed up on transmission queues. Therefore, latency times when sampling time skews is reduced and results are more accurate. This value must not exceed the MaxPriority attributes of the queue managers. **Default value:** -1
Time skew reply wait interval (sec)	The number of seconds to wait for the message sent to determine the time skew across a communication link. If no response is received in this time, the time skew is disregarded. **Default value:** 30
Time skew retry threshold	The maximum number of time skew samples to take at a single time skew check interval. At each time skew check interval, the Processing Server continues to retry up to this number of times if the sampling surpasses the time skew latency threshold. **Default value:** 8
Time skew Web URL	The URL of the servlet used to determine the time skew between the Processing Server host and the BSM Gateway Server. **Default value:** the time server servlet in the BSM installation which is **http://<bsm_gateway_server_host>:80/topaz/services/technical/time?alt=text/plain**.

UI Elements (A-Z)	Description
Trace	Enables Trace logging for all Analyzers on this Processing Server. See "Processing Server Page, Log Files Tab" on page 73.
Use the Time Server Running in the Processing Server	Use the time server within the Processing Server to calculate all time skews. Otherwise, specify the name of the host running the Time Server and its port number. **Default value**: Checked **Time Server Host**. The host name on which the Time server is running. **Time Server Port**. The port number for the Time Server. For information about running the time server independent of a Processing Server, see "TimeServer" on page 420.

Database Area

These elements are described with the Processing Server Wizard. See the description on the "Database Selection Page" on page 55, "Database Connection Parameters Page" on page 56 and "Database Advanced Parameters Page" on page 59.

Note: Any changes to Database configuration settings for a Processing Server cause all Analyzers assigned to the Processing Server to be restarted.

Advanced Area

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Elements (A-Z)	Description
Enable SSL	Enables SSL communication for this Processing Server. Enabling SSL here requires that all components in the TransactionVision deployment environment are enabled for SSL, including the BSM Gateway Server. The workflow for enabling SSL is described in "Using SSL with TransactionVision" in the *HP Business Service Management Hardening Guide* PDF and in "Security" in the the *HP TransactionVision Deployment Guide* PDF. IF SSL is enabled, specify the following values as described in the above chapter: **Keystore Location**. The full path and file name of the keystore file containing the private keys and client certificate. The keystore file must be a java keystore file (JKS). Note: Configure this field only if you want to use client certificates. **Keystore Password**. The password for the keystore file; must be identical to the private key password.
Enable TIBCO EMS	See "Database Advanced Parameters Page" on page 59.
Enable WebSphere MQ	See "Database Advanced Parameters Page" on page 59.
Lightweight Single Sign-On Domain	Enter the Processing Server's LW-SSO domain when **Use Host DNS Domain Lightweight Single Sign-On Domain** is turned off (unchecked).

UI Elements (A-Z)	Description
Synchronize TransactionClass properties for SQL alert jobs	Enables the use of the SQL Alert job, which is a custom job you can add to the TransactionVision deployment environment. See "How to Add a Custom Job" on page 318.
Use Host DNS Domain Lightweight Single Sign-On Domain	Enables the Processing Server to automatically lookup the LW-SSO domain. Uncheck this option to manually enter the domain in the **Lightweight Single Sign-On Domain** text box. This is only necessary if the following occurs: ➤ Automatic lookup (which uses the Java networking API) is unable to lookup the domain. ➤ BSM and the Processing Server domains share a common nested domain. In this case, manually set the domain to the common nested domain. For example, if BSM is in **emea.hp.com** and the Processing Server is in **cnd.hp.com**, then the domain needs to be manually set to **hp.com**. **Default**: on

Processing Server Page, Log Files Tab

This tab enables you to view the log files of a Processing Server.

To access	**1** Select **Admin** > **Transaction Management.** **2** (left pane) Click the **Configuration** tab. **3** Select **Processing Servers** > <processing_server>. **4** (right pane) Click the **Log Files** tab.
Important information	The **<TVISION_HOME>/logs** directory on the host on which the Processing Server is installed contains all log files. Only the log files related to the Processing Server, Job Manager, and Query Engine are shown here. If the log file is larger than 5 megabytes, only the most recent 5 megabytes of the log file are loaded.
See also	"Analyzer Page, Log Files Tab" on page 141

Processing Server Area

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Elements	Description
Log files	The log file names and sizes of the Processing Server logs.
Contents of <log file name>	The log file entries. Look here if SonicMQ is not starting correctly or the Processing Server is having issues starting.

Query Engine Area

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Elements	Description
Log files	The log file names and sizes of the Query Engine logs.
Contents of <log file name>	The log file entries.

Job Manager Area

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Elements	Description
Log files	The log file names and sizes of the Job Manager logs.
Contents of <log file name>	The log file entries.

Troubleshooting and Limitations

This section includes the following topics:

➤ "Port Conflicts" on page 75

➤ "Firewall Configuration" on page 77

Port Conflicts

The Processing Server can fail to initialize a Job Manager, Query Engine, SonicMQ Broker or Analyzer due to a port conflict. That is, the port number that TransactionVision attempts to use is already in use by another application. If a port conflict occurs, the component cannot be created.

Review the following table to help you resolve port conflicts.

Component	Default Port	To change, access the following from Administration > Configuration > TransactionVision > :
The port on which the BSM Gateway Server listens for Processing Server calls.	80	Configuration > BSM Settings
The port on which the Processing Server listens for BSM	21100	Processing Server page, Configuration > General tab, General area
The SSL port on which the Processing Server listens for BSM	21101	Processing Server page, Configuration > General tab, General area
Job Manager Port	21010	Processing Servers Summary page, Job Manager tab
Job Manager SSL Port	21011	Not needed; the Job Manager runs in SSL if the associated Processing Server is SSL enabled.
Query Engine Port	21000	Processing Servers Summary page, Query Engine tab

Component	Default Port	To change, access the following from Administration > Configuration > TransactionVision > :
Query Engine SSL Port	21001	Not needed; the Query Engine runs in SSL if the associated Processing Server is SSL enabled.
The port on which the Processing Server connects to the database	IBM DB2: 50000 Microsoft SQL: 1433 Oracle: 1521	Processing Server page, Configuration > General tab, Database area
SonicMQ Domain Manager port (bundled)	21110	Sonic Management Console, and Default SonicMQ Communication Link page, SonicMQ Connection Info area
SonicMQ Broker port (bundled)	21111	Sonic Management Console, and Default SonicMQ Communication Link page, SonicMQ Connection Info area
SonicMQ Broker SSL port (bundled)	21112	Sonic Management Console, and Default SonicMQ Communication Link page, SonicMQ Connection Info area
SonicMQ Broker HTTP port (bundled)	21113	Sonic Management Console and Default HTTP and RUM Communication Links page, SonicMQ Connection Info area
SonicMQ Broker HTTPS port (bundled)	21114	Sonic Management Console and Default HTTP and RUM Communication Links page, SonicMQ Connection Info area

Firewall Configuration

In your TransactionVision deployment environment, if there are firewalls between various TransactionVision components, or between TransactionVision servers and other BSM components, you may need to open certain ports on your firewalls. Use the following network traffic tables for reference:

NetworkTraffic Between Users' Browser and BSM Servers

From	To	Default Ports	Purpose
Browser	BSM Gateway	80	Use the BSM user interface.
Browser	BSM Gateway	443	Use the BSM user interface (SSL).

NetworkTraffic Between BSM Servers and TransactionVision Processing Server Components

From	To	Default Ports	Purpose
BSM Gateway	Process Manager	21100	Get status and administrate TransactionVision Processing Server.
BSM Gateway	Process Manager	21101	Get status and administrate TransactionVision Processing Server (SSL).
BSM Gateway	Query Engine	21000	Get status and query transaction instance data.
BSM Gateway	Query Engine	21001	Get status and query transaction instance data (SSL).
BSM Gateway	Job Manager	21010	Get status and administrate Job Manager.
BSM Gateway	Job Manager	21011	Get status and administrate Job Manager (SSL).

From	To	Default Ports	Purpose
BSM Gateway	Analyzer 1-5	21120, 21130, 21140, 21150, 21160	Get status and administrate Analyzer.
BSM Gateway	Analyzer 1-5	21121, 21131, 21141, 21151, 21161	Get status and administrate Analyzer (SSL).
RUM Engine	SonicMQ Broker	21111	Send RUM events to TransactionVision.
RUM Engine	SonicMQ Broker	21112	Send RUM events to TransactionVision (SSL).
Job Manager	BSM Gateway	80	Publish CIs and send sample data.
Analyzer	BSM Gateway	80	Get time for calculating time skew.

NetworkTraffic Between TransactionVision Processing Server Components and Database Servers

From	To	Default Ports
Analyzers, Query Engine, Job Manager	Database Server	IBM DB2: 50000
Microsoft SQL Server: 1433		
Oracle: 1521	Access the database that stores the transaction instance data.	

NetworkTraffic Between TransactionVision Processing Server Components and Messaging Providers

From	To	Default Ports	Purpose
Analyzer	SonicMQ Domain Manager	21110	Query event queue depth.
Analyzer	SonicMQ Broker	21111	Send configuration messages and get events.
Analyzer	WebSphere MQ	1414	Send configuration messages and get events.

Note:

- A TransactionVision Analyzer may use and communicate with a TransactionVision SonicMQ server running on the same or different TransactionVision Processing Server.
- For WebSphere MQ, if an Analyzer and the queue manager are running on the same system, this port is needed only if the Analyzer uses the client connection mode to access the queue manager.

NetworkTraffic Between TransactionVision Agents and TransactionVision Processing Server Components/Messaging Components

From	To	Default Ports	Purpose
Java Agent	Process Manager (Time Server)	21104	Get time for calculating time skew.
Java Agent	SonicMQ Broker	21111	Get configuration messages and send events.

From	To	Default Ports	Purpose
Java Agent	SonicMQ Broker	21112	Get configuration messages and send events (SSL).
Java Agent	WebSphere MQ	1414	Get configuration messages and send events.
.NET Agent	Process Manager (Time Server)	21104	Get time for calculating time skew.
.NET Agent	SonicMQ Broker	21111	Get configuration messages and send events.
.NET Agent	SonicMQ Broker	21112	Get configuration messages and send events (SSL).
.NET Agent	WebSphere MQ	1414	Get configuration messages and send events.
NonStop TMF Agent	Process Manager (Time Server)	21104	Get time for calculating time skew.
NonStop TMF Agent	Analyzer 1-5	21120, 21130, 21140, 21150, 21160	Get configuration messages.
NonStop TMF Agent	Analyzer 1-5	21121, 21131, 21141, 21151, 21161	Get configuration messages (SSL).
NonStop TMF Agent	SonicMQ Broker	21113	Sent events.
NonStop TMF Agent	SonicMQ Broker	21114	Sent events (SSL).
Tuxedo Agent	Process Manager (Time Server)	21104	Get time for calculating time skew.
Tuxedo Agent	Analyzer 1-5	21120, 21130, 21140, 21150, 21160	Get configuration messages.

From	To	Default Ports	Purpose
Tuxedo Agent	Analyzer 1-5	21121, 21131, 21141, 21151, 21161	Get configuration messages (SSL).
Tuxedo Agent	SonicMQ Broker	21113	Send events.
Tuxedo Agent	SonicMQ Broker	21114	Send events (SSL).
DataPower Agent	SonicMQ Broker	21113	Send events.
DataPower Agent	SonicMQ Broker	21114	Send events (SSL).

Note: Each Java or .NET Agent needs only one message provider.

3

Analyzers

This chapter includes:

Concepts

- Analyzers Overview on page 84
- Key Configuration Concepts for Analyzers on page 85
- Correlation For DataPower Events on page 96

Tasks

- How to Add an Analyzer to the Deployment Environment on page 106
- How to Assign a Communication Link to an Analyzer on page 107
- How to View Analyzer Status on page 108
- How to Stop and Restart Event Collection on page 108
- How to Stop and Restart an Analyzer on page 109
- How to Remove an Analyzer from the Deployment Environment on page 110
- How to Recover an Analyzer on page 111
- How to Enable Failure Mode on an Analyzer on page 112

Reference

- Analyzers User Interface on page 117

Troubleshooting and Limitations on page 142

Concepts

Analyzers Overview

The TransactionVision Analyzers process collected event data into business transactions. The Analyzer runs in the context (container) of a TransactionVision Processing Server.

The TransactionVision deployment environment can have multiple Analyzers. For more information about the TransactionVision deployment environment, see "Processing Servers Overview" on page 40.

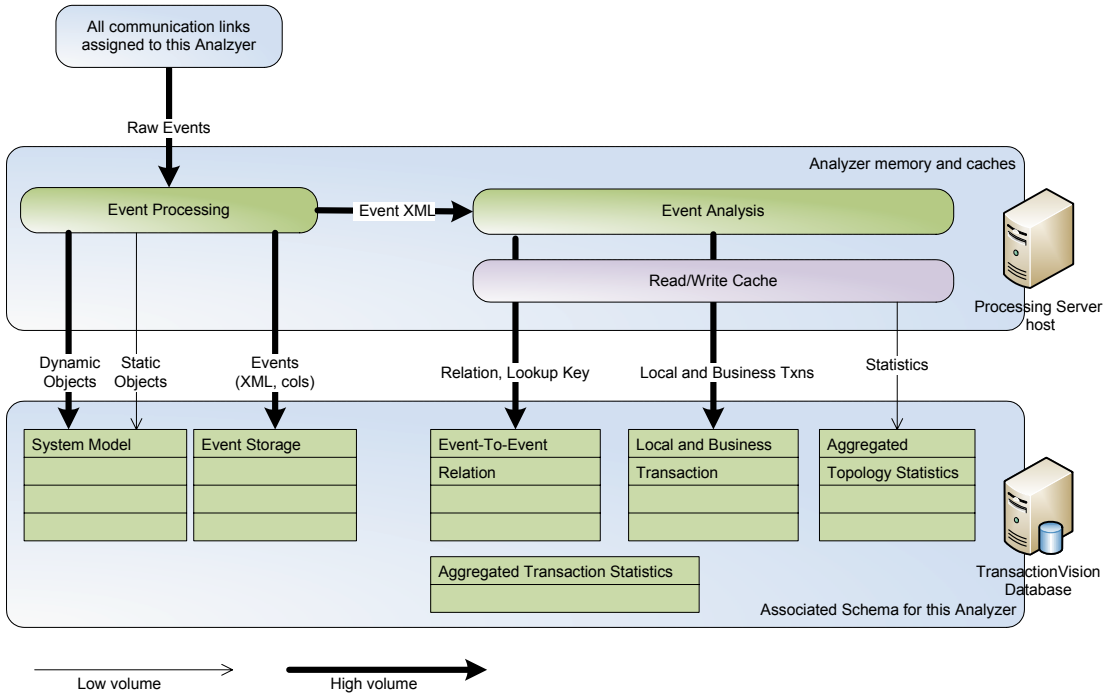
The processing of events by an Analyzer occurs in the phases below:

Phase 1: Event collection. Raw events arrive in the event queues, and the Analyzer pulls them off and forms an in-memory Event XML document to store and process them.

Phase 2: Correlation. The Analyzer processes the events correlating each event into the transaction instance to which it belongs.

Phase 3: Transaction Classification. The Analyzer executes the rules assigned to it to classify the transaction into a business transaction. It also assigns transaction attributes to the transaction.

Throughout these phases, there are several points at which the Analyzer writes its data to the associated database schema:



The writing of user data, event XML, and raw events to the database can be enabled or disabled. See the "Miscellaneous Information Page" on page 199 in Chapter 4, "Communication Links".

Key Configuration Concepts for Analyzers

Analyzers are highly configurable.

This section includes the main configuration concepts:

- "About Data Purging" on page 86
- "About Deleting Analyzers" on page 86
- "About Time Skews" on page 87
- "Configuring the Caches" on page 87

➤ "About Logging" on page 88

➤ "Choosing Standard Mode or Failure Mode" on page 89

➤ "Enabling Correlation For Certain Technologies" on page 93

➤ "Enabling an Analyzer for RUM" on page 94

➤ "Enabling an Analyzer for BPI" on page 96

About Data Purging

In the TransactionVision deployment environment, the data storage tables that contain the event and transaction data can grow to a very large size. You need to determine how long the data is to be retained to support the needs of the Transaction Management users and schedule data purging accordingly. A full database causes the Analyzers writing to it to stop event collection and also generates database exceptions.

By default, no data is purged. You can schedule the purging based on event age or other factors. See the Data Purge tab in "Configuration Tab, Data Purge Tab" on page 137.

About Deleting Analyzers

When you delete an Anlayzer, its data remains in the database. This allows you to access the data again by creating a new Analyzer and assigning it to that same database.

If you want to permanently remove the data from the database, you can manually do so by using the TransactionVision **CreateSqlScript** utility described in "CreateSqlScript" on page 384. The command is:

```
CreateSqlScript -d -s <schemaName> -e
```

You can also remove the database by using you database vendor administration tools.

For instructions on deleting Analzyers, see "How to Remove an Analyzer from the Deployment Environment" on page 110.

About Time Skews

The time difference between an agent host and the Analyzer host to which the agent is sending events is a time skew. Time skews are calculated by the Processing Servers and are automatically applied to the relevant event data, such as event start time or stop time. That way, differences in the time on the different hosts do not affect the event data.

Time skews rely on the time servers that are run by the Processing Server. If an agent cannot access the Processing Server host, it uses another time server as designated in the communication link definition.

Time skews are reported on the Communication Links Summary page. See "Communication Links Summary Page" on page 202.

Time skews also exist between a Processing Server and the BSM Gateway Server. See "Key Configuration Settings for Processing Servers" on page 43.

Configuring the Caches

The Analyzer uses several caches to hold data it is processing for event analysis. A bigger cache size minimizes database access and increases performance, but also increases the amount of memory used by the Analyzer.

Each analyzer has its own set of caches, since each Analyzer runs in its own JVM.

The cache is configurable as follows:

- **PII object:** Specifies the number of PII system model objects cached.
- **Events:** Specifies the size per each event's cache.
- **Transactions:** Specifies the size of each transaction correlation cache.

Typically, you do not need to modify the default cache sizes. However, if you are running the Analyzer in Failure mode, you do need to increase the caches sizes. See "Memory Cache Changes Needed for Failure Mode" on page 92.

The statistics cache is a special cache that is used to hold the data shown in the Component Topology. The statistic cache is structured based on a configurable timeslice value.

The timeslice is used to partition data into different pieces so that depending on what time frame you are interested in looking at, this data can be quickly retrieved. A timeslice represents a period of time during which all events belonging to the same program and MQ object are condensed into a single statistic.

For example, if you had a program putting messages to a particular queue, all the statistics data relating to the program and the queue (latency times, success counts, and so on) during a particular time slice are stored in the same database row. By changing the values of the timeslice interval you can control the size of the statistics table and how efficient it is to access the statistics table. The larger the time slice, the more efficient the storage of results will be; the trade-off you lose is that your timeslice interval is the smallest increment of time that you can view your data in for the Component Topology in Static mode. For example, if you were to set a time slice of one day, you would not be able to view these statistics on an hourly basis. By default it is 60 minutes.

To change the default time slice, go to **Admin** > **Transaction Management** > **TransactionVision (root)** > **Configuration** > **General.** Click **Server** and then set the **topologyStateTimeslice.** Click **Apply**.

All Analyzers use this value.

About Logging

These special types of logging can be enabled for an Analyzer:

- **Performance Logging**. Analyzer event processing rate and other associated metrics are logged. The log provides metrics for average events per second overall and over the last event sampling period.

  Log location: Navigate to **Admin** > **Transaction Management** > **Configuration** > **Processing Servers** > <my processing server> > < my analyzer> > **Log Files** > **PerfLog**.

  To enable: Navigate to **Admin** > **Transaction Management** > **Configuration** > **Processing Servers** > <my processing server> > < my analyzer> > **Configuration** > **Events** > **Performance.**

- **Trace Logging**. Verbose and detailed information about what an Analyzer is doing internally. This logging is used mainly to troubleshoot problems and should not be turned on in a production environment.

  Log location: Navigate to **Admin** > **Transaction Management** > **Configuration** >> **Processing Servers** > <my processing server> > <my analyzer> > **Log Files** > **Trace.**

  To enable: Navigate to **Admin** > **Transaction Management** > **Configuration** > **Processing Servers** > <my processing server> > < my analyzer> > **Configuration** > **General** > **Analyzer General.**

- **Debug Logging**. Highly detailed information about what an Analyzer is doing internally. This logging is used to troubleshoot problems and should not be turned on in a production environment.

  Log location: Navigate to **Admin** > **Transaction Management** > **Configuration** > **Processing Servers** > <my processing server> > <my analyzer> > **Log Files** > **DebugLog.**

  To enable: Navigate to **Admin** > **Transaction Management** > **Configuration** > **Processing Servers** > <my processing server> > < my analyzer> > **Configuration** > **General** > **Analyzer General.**

Note: If the log file is larger than 5 megabytes, only the most recent 5 megabytes of the log file are loaded.

For information about accessing log files, see the *HP TransactionVision Deployment Guide* PDF.

Logging also occurs at the Processing Server level. See "Processing Server Page, Log Files Tab" on page 73.

Choosing Standard Mode or Failure Mode

Business transaction-level data is always written to the database, regardless of the mode. By default, Analyzers run in standard mode. In standard mode, all event, correlation, and local transaction data is flushed continuously to the database.

The Analyzer can be configured to run in failure mode. In this mode, the Analyzer flushes only the event, correlation, and local transaction data of failed transactions or transactions that have violated their defined threshold to the database. Event data for successful transactions is not stored.

Failure mode is useful because it significantly reduces the data storage requirement for the Analyzer. This reduction in database access also increases the performance of the Analyzer.

However, an Analyzer running in failure mode cannot recover from a system crash or processing error because event data is kept only in memory.

Failure mode can be enabled on a per-Analyzer basis, allowing you to run certain Analyzers in failure mode while running others in standard mode.

A typical use case for the failure mode is for high-volume systems with short-lived transactions where statistics collection and failure alerting is the primary requirement.

When Data is Flushed to the Database in Failure Mode

In failure mode, the business transaction data is always flushed to the database. The remaining data (event XML, user data, correlation and local transaction data) is stored in memory caches until either:

- The transaction **result** attribute is set to **SUCCESS** by matching classification rules, in which case all data for this transaction kept in memory is discarded. Further events for this transaction are discarded as well.
- The transaction result is set to **FAILED** by the classification rule engine, in which case all data of this transaction is flushed to the database. Further events for this transaction are stored as well.
- The transaction violates its threshold. The response time of a transaction is continuously monitored and transaction data is flushed to the database if this response times exceeds the **Threshold** value specified in the business transaction tracing properties.

- The transaction result has still not been set even after a certain configurable time has passed, in which case the transaction is also considered to be “aged out”, and all data is flushed to the database. The SLA status column of the transaction is set to **Aged Out**. In case an event is processed later which marks the transaction as **successful**, the SLA status is reset and the transaction is treated in the same way as a successful transaction - all further events are discarded.

The time after which a transaction is considered to “age out” is dynamically computed from the sum of the transaction threshold plus **Process delay time** specified in the Analyzer configuration. This value specifies the maximum time the processing of any event can be ‘delayed’ within the Analyzer in respect to other events of the same transaction. There is a certain amount of lag between the time events in a transaction are generated at the agent and the time events for a transaction are completely processed by the Analyzer.

This lag is due to the effects of queuing in the WebSphere MQ infrastructure, event batching, multithreading, and context switches by the OS. So if for example, the transaction threshold is defined as 5 seconds, and the Process delay setting is 10 seconds, any transaction whose result has not been determined 15 seconds after the first event has been processed are aged out.

Successful transactions still appear together with their attribute values on the Transaction Tracking Report, but it is not possible any more to navigate to any other views or drill down to the event level, since all this data has not been written to the database. But for failed and aged out transactions all data is available, and the report offers the same functionality as if those transactions had been processed in Standard Mode.

Memory Cache Changes Needed for Failure Mode

A high system load, too small cache sizes, or keeping event data too long in the caches can all lead to cache overflows (indicated by a **FailureModeCacheOverflow** warning in the **AppLog** file). When this happens, the entire cache contents are written to the database slowing down the Analyzer event processing rate.

As mentioned previously, failure mode only works efficiently if the Analyzer is able to clear out the memory caches as quickly as new data is coming in. If a cache overflow occurs, the system slows down considerably since all the data has to be written to the database. A system with frequent cache overflows does not perform as expected.

The ability of the Analyzer to keep the caches from overflowing is determined by the multiple factors:

- Rate of which new events are coming into the system
- Database performance
- Configured size of the caches
- Time an open transaction is kept in the system (determined by threshold + process delay)

The Analyzer Failure Mode is intended for systems with short lived transactions, with threshold values in the range of seconds. As the threshold values get bigger, the transactions need to be kept in the caches much longer, and more data can accumulate in the caches. Such a system can still work properly if the cache sizes are large enough, or if the load on the system is small enough.

Failure Mode Considerations

There are certain issues with the Failure Mode about which you need to be aware. First, since all event data is initially only kept in memory, this mode does not offer any recoverability. If an Analyzer crash, database error, or general system failure occurs, the data in the caches is lost. In standard processing mode, the XML event is always written to the database, so that in case of such a failure the missing analysis data can be recovered. In Failure Mode, the XML event data might not have been written to the database, so there is nothing to recover from.

Another issue arises when events for a successful transaction are being processed (or even entering the system) after the data for the transaction has been discarded from the caches. If these events contain an event which increases the response time of the transaction beyond the threshold value, and if writing of threshold violations is enabled, then this transaction only has partial data stored in the database, because previous event data has already been discarded. Note that this issue does not occur if the transaction classification contains time rules, since successful transactions are not removed from the cache until the time rules have fired.

The second case in which late events can cause a problem occurs when the event is being processed so late that the correlation data for the corresponding transaction has already been pushed out of the lookup caches (which are, like the write caches, also limited in size). In this case, an event might not get correlated into the existing transaction, and a new transaction is created instead. Increasing the size of the caches (event_based) could remedy the problem, but it might also be an indicator that the Failure Mode is not well suited for this kind of transaction.

Enabling Correlation For Certain Technologies

For collected events, the Analyzer generates and stores lookup keys in the database to be used for correlation. These database storage operations affect the overall Analyzer performance. To avoid unnecessary lookup key generation and storage, some of the correlations that the Analyzer can perform have been disabled by default.

If you are collecting events of the technologies listed below, you must manually enable the correlation. This must be done before collecting the events you want to correlate.

Correlation is enabled by editing the **Beans.xml** file for each Analyzer. This file controls the beans loaded by the Analyzer framework for event processing. The **Beans.xml** file is accessed on each Analyzer's **Configuration** > **XML** tab.

Tip: To find text in the **Beans.xml** file, click anywhere in the text and then enter CRTL-F.

DataPower

If you collect DataPower events, you choose one of two beans to be enabled depending on the needs of your monitoring environment. For information about DataPower correlation, see "Correlation For DataPower Events" on page 96.

.NET

The following .NET correlation rules are available by default on the Transaction Management Admin user interface and can be enabled from the Event Customization Rules page. For user interface details, see "Event Customization Rules Page" on page 618. For details on enabling rules, see "How to Assign a Rule to the Appropriate Analyzers" on page 531.

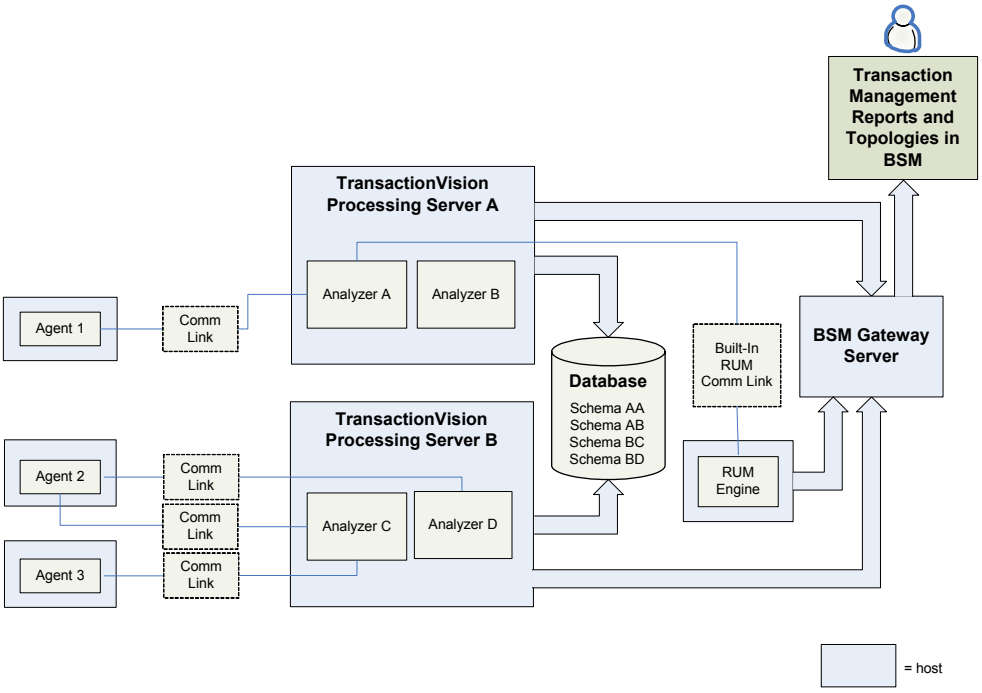
- .NETMSMQRule
- .NETRemotingRule
- .NETRule
- .NETWCFRule

Enabling an Analyzer for RUM

Transaction Management reports and topologies can be configured to include RUM data by integrating TransactionVision components with RUM components in the same deployment environment.

An Analyzer in the deployment environment must be enabled to collect the RUM events. That Analyzer uses the built-in RUM communication link to communicate with the RUM Engine.

In this example deployment environment, Analyzer A combines events from RUM with the events from Agent 1 resulting in transactions containing information from both:



To enable an Analyzer for RUM, set the corresponding check box on the General page of the Analyzer wizard. See "Analyzer Wizard" on page 117.

The Analyzer cache size for RUM objects is configurable. See "Configuration Tab, General Tab" on page 125.

The RUM Engine must also be configured to communicate with TransactionVision. See "How to Enable Communication Between RUM and TransactionVision" on page 364.

Enabling an Analyzer for BPI

One or more Analyzers can be configured to send events to BPI, where they are used as input for Business Process modeling.

To enable an Analyzer for BPI, see the BPI settings group in "Configuration Tab, Events Tab" on page 130.

For more information about the communication between BPI and TransactionVision, see "Business Process Insight" on page 358.

Correlation For DataPower Events

TransactionVision can correlate DataPower events with events collected from other technologies. This correlation is managed by enabling specific event relationship beans in the **Beans.xml** configuration file of the corresponding Analyzer. These event relationship beans allow all related events to be correlated into the same business transaction, regardless of the technology on which they are based.

Note: To optimize out-of-the-box performance, not all event relationship beans are enabled by default. You must enable the event relationship beans for the technologies that will be collected by each Analyzer. This is done in the Analyzer's **Beans.xml** configuration. See the sections that follow.

This section describes the different correlations for DataPower Events:

- "DataPower to DataPower Correlation" on page 97
- "DataPower to Servlet Correlation" on page 97
- "DataPower to WebSphere MQ Correlation" on page 98

DataPower to DataPower Correlation

DataPower services can call other DataPower services whether it is as a backend to the service or as a side service call. Although this case does not involve a different technology, it still requires an event relationship bean to correlate calls between multiple DataPower events.

Uncomment the following line in **Beans.xml** to enable DataPower to DataPower event correlation:

```
<Module type="Bean"
class="com.bristol.tvision.services.analysis.eventanalysis.DataPowerToDataPowerRel
ationshipBean"/>
```

The **Beans.xml** file is accessed on each Analyzer's **Configuration** > **XML** tab.

DataPower to Servlet Correlation

DataPower can interact with J2EE Servlets. TransactionVision can correlate DataPower events to Servlet events when Servlets are being called as a backend to the service or as a side service call.

Uncomment the following lines in **Beans.xml** to enable DataPower and Servlet event correlation:

```
<Module type="Bean"
class="com.bristol.tvision.services.analysis.eventanalysis.ServletToDataPowerRelation
shipBean"/>
<Module type="Bean"
class="com.bristol.tvision.services.analysis.eventanalysis.DataPowerToServletRelation
shipBean"/>
```

The **Beans.xml** file is accessed on each Analyzer's **Configuration** > **XML** tab.

DataPower to WebSphere MQ Correlation

DataPower Multi-Protocol Gateways can interact with WebSphere MQ. A DataPower Multi-Protocol Gateway can be initiated by a message from a request queue with the response being delivered to a response queue on the frontend. Similarly, WebSphere MQ can also be called as a backend service to a Multi-Protocol Gateway. TransactionVision can correlate DataPower events with WebSphere MQ events for both frontend and backend relationships.

Two different correlation algorithms are provided to correlate with WebSphere MQ. One of the two different algorithms must be chosen for the Analyzer when there is a need to correlate between DataPower and WebSphere MQ events.

The preferred algorithm is used to correlate DataPower events with its WebSphere MQ API calls occurring within the WebSphere MQ Listener that the Multi-Protocol Gateway is connected to.

An alternative algorithm attempts to correlate the DataPower events directly to the WebSphere MQ application that is interacting with the Multi-Protocol Gateway. Each algorithm has its caveats and limitations which should be considered when deciding which one best fits the needs of the environment to be monitored by TransactionVision.

Option 1: WebSphere MQ Listener Relationship Algorithm Beans (Preferred)

The preferred event relationship algorithm to use to correlate DataPower events with WebSphere MQ events relies on the collection of WebSphere MQ Listener events (runmqlsr or amqrmppa). With this algorithm, the DataPower events are correlated to the WebSphere MQ API calls made in the Listener by the DataPower Multi-Protocol Gateway when putting and getting messages to and from the queue manager.

With this kind of event relationship, the TransactionVision topologies show the DataPower service node linked to the WebSphere MQ Listener program in a calling manner—much like the DataPower service node would be linked to a Servlet node. This relationship is preferred because of the statistical information (particularly latency) provided by DataPower events is more suited for call links rather than message path links used in WebSphere MQ.

This event relationship is also preferred, because the DataPower Agent does not have the ability to perform WebSphere MQ object resolution. This means that DataPower events do not contain queue type information. For example, if DataPower is putting to a remote queue, the event generated by the DataPower event will not contain the fact that the queue is a remote queue or what the remote queue resolves to. Since this event relation connects the DataPower service node to the WebSphere MQ Listener program, all WebSphere MQ object resolution and message path statistics are handled by the Listener WebSphere MQ events, and the Listener events are related and connected with the WebSphere MQ application that is interacting with DataPower on the frontend and/or backend.

Uncomment the following lines in **Beans.xml** to enable the WebSphere MQ Listener Relationship Beans (Option 1):

```
<Module type="Bean"
class="com.bristol.tvision.services.analysis.eventanalysis.MQListenerToDataPowerRel
ationshipBean"/>>
<!--<Module type="Bean"
class="com.bristol.tvision.services.analysis.eventanalysis.DataPowerToMQListenerRel
ationshipBean">
```

The **Beans.xml** file is accessed on each Analyzer's **Configuration** > **XML** tab.

WebSphere MQ Listener Relationship Algorithm Notes and Limitations

Relating DataPower events to the WebSphere MQ Listener is the preferred method as it allows proper WebSphere MQ object resolution to be handled by the WebSphere MQ Agent on both sides of the transaction. In addition, latency and other statistics are more accurate.

To use this algorithm, the WebSphere MQ Listener of the queue manager(s) that DataPower connects to must be monitored by the WebSphere MQ Agent. Since the WebSphere MQ Listener program can handle many connections and activities other than just DataPower connections and communications, great care should be taken to limit the amount of events produced from the Listener.

Some examples of additional traffic may include connections and communications from other WebSphere MQ clients that are of no interest and connections and communications from receiver channels-a remote queue definition from another queue manager that resolves to a queue on the monitored queue manager.

The following steps may be needed to limit the events collected from WebSphere MQ Listeners so that unwanted, unrelated events do not complicate the business transaction correlation.

1 Only collect events from the WebSphere MQ Listeners in which DataPower directly connects to.

For example, if DataPower is connecting to one queue manager to send messages across a remote queue definition or cluster queue to other queue managers, the WebSphere MQ Listener of the other queue managers should not be monitored by the WebSphere MQ Agent.

Monitoring the other WebSphere MQ Listeners could cause unwanted events from remote queue receiver channels to be correlated into the business transaction causing confusing results in TransactionVision topologies.

There are two ways of limiting collection to only the WebSphere MQ Listeners that DataPower is connecting to.

a Most efficiently, the Agent can be completely disabled for the WebSphere MQ Listeners on any given host by adding the following lines to **wmq_exit_agent.deny**:

```
runmqlsr*
amqrmppa*
```

b Alternatively, data collection filters can be designed to include WebSphere MQ Listener program names (runmqlsr*****, amqrmppa*****) on the queue managers of interest and exclude those that are not of interest. This would require creating two data collection filters-one that would include the WebSphere MQ Listener program and another that does not. Each data collection filter would be assigned to each WebSphere MQ communication link based on if the WebSphere MQ Listener program should be collected on that queue manager or not.

2 Only collect events from the WebSphere MQ Listener when they are involved with the queues being used by the monitored DataPower Multi-Protocol Gateways. Since the WebSphere MQ Listener could be serving other clients or receiver channels, these other activities may not be of interest or may add inaccurate correlated events to the business transaction of interest. In the data collection filter(s), filters including only those queue names in which the DataPower Multi-Protocol Gateways are interacting with can greatly reduce the unwanted events produced from the WebSphere MQ Agent monitoring the Listener.

Note: It is also suggested to always limit the WebSphere MQ APIs collected to only MQGET, MQPUT, and MQPUT1 in any new data collection filter created.

---

3 There may be cases where it is necessary to monitor other WebSphere MQ Listeners-those not directly connected to by DataPower. If such WebSphere MQ Listeners provide connectivity for remote queue definitions on the queue manager that DataPower connects to through receiver channels, additional filtering needs to be designed to prevent unwanted correlated events from being collected as a result of remote queue message delivery through the receiver channel to the Listener.

Considering the scenario where DataPower is communicating messages through an intermediate queue manager over remote or cluster queues to a backend queue manager where the interacting WebSphere MQ application is processing the messages from DataPower, activity on the local queue in which the WebSphere MQ application gets remote messages from DataPower needs to be collected from the application but not from the WebSphere MQ Listener receiving the message through the receiver channel. In such scenarios where it is desired to monitor the WebSphere MQ Listener for other reasons, filtering should be added to exclude MQPUT events in the Listener to the local queue in which the WebSphere MQ application is getting messages originating from DataPower.

To implement such complex filtering, multiple data collection filters would need to be designed and assigned to the communication link. One data collection filter would select specifically on the WebSphere MQ Listener program (runmqlsr*, amqrmppa*). In this data collection filter, the local queue names in which any WebSphere MQ applications getting messages from DataPower would be excluded. In a second data collection filter, all other programs would be selected to collect from. This is done by excluding the WebSphere MQ Listener program (runmqlsr*, amqrmppa*). Both of these two data collection filters need to be assigned to the communication link for the queue manager in question.

Note: It is also suggested to always limit the WebSphere MQ APIs collected to only MQGET, MQPUT, and MQPUT1 in any new data collection filter created.

Option 2: WebSphere MQ Relationship Algorithm Beans

An alternative WebSphere MQ relationship algorithm is provided to address the possible use case where it may not be possible or desirable to monitor WebSphere MQ Listeners. This alternative relationship algorithm attempts to correlate events from the DataPower Agent directly to the interacting WebSphere MQ applications processing messages delivered to and from the DataPower Multi-Protocol Gateways.

With this kind of event relationship, the TransactionVision topologies show the DataPower service node linked to the WebSphere MQ queues that the interacting application is linked to. Such a link through a queue in the topology shows message path metrics rather than calling metrics in which DataPower events are typically related.

Uncomment the following lines in **Beans.xml** to enable the alternative WebSphere MQ Relationship Beans:

```
<Module type="Bean"
class="com.bristol.tvision.services.analysis.eventanalysis.MQToDataPowerRelationshi
pBean"/>
<Module type="Bean"
class="com.bristol.tvision.services.analysis.eventanalysis.DataPowerToMQRelationshi
pBean"/>
```

The **Beans.xml** file is accessed on each Analyzer's **Configuration** > **XML** tab.

WebSphere MQ Relationship Algorithm Notes and Limitations

Since this alternative WebSphere MQ relationship algorithm is not the preferred algorithm, it includes several limitations and potential inaccuracies.

- Inaccurate Latency Statistics

  Being that this type of relationship links the DataPower service node to the queues in which messages are passed to and from the monitored interacting WebSphere MQ applications, the latency statistics represent time spent on the queue rather than a call latency (total time spent before control comes back to DataPower) which DataPower events provide. For accurate message path latency calculations, there needs to be timestamp information when the MQPUT occurs and when the resulting MQGET occurs.

  When matching a DataPower event to a WebSphere MQ event, the DataPower event only contains a general timestamp of when the service is called. It does not contain a timestamp of when exactly the MQPUT or MQGET occurs. This means there will be some inaccuracy of the calculated message path latency. For example, an MQPUT on the backend of a Multi-Protocol Gateway would include the time spent in the request rule when calculating the message path latency of the backend request message.

  In addition to this latency inaccuracy with this alternative algorithm, there is also the potential of timeskew differences which can further affect the latencies.

➤ DataPower Nodes Potentially Linked to Wrong Queues

Since this relationship links the DataPower service node to the queue that the related interacting WebSphere MQ application is linked to, there is the potential that the DataPower service is not linked to the appropriate queue. This scenario occurs when remote or cluster queues are used.

The DataPower Agent does not have the ability to lookup and resolve queues that it is interacting with. Because of this limitation, TransactionVision only has the queue resolved by the interacting WebSphere MQ application to link to. For example, if a Multi-Protocol Gateway is putting a request message through a remote queue definition, it would be expected that the DataPower service node in the topologies would link to that remote queue definition. However, the DataPower service is actually linked to the local queue that the remote queue definition resolves to in which the interacting WebSphere MQ application is getting the message from.

Also, there would be no indication of the remote queue existence within the topology since there was no resolution of the remote queue. This may or may not be a large concern as long as there is a connection shown between the DataPower service to the interacting WebSphere MQ application.

➤ No Correlation When WebSphere MQ Only On DataPower Service Frontend

There is one use case in which the DataPower event cannot be correlated to a WebSphere MQ event. When a Multi-Protocol Gateway is defined with WebSphere MQ on the frontend and some other technology on the backend, there is no way to correlate the DataPower event to the MQGET of the frontend interacting WebSphere MQ application receiving the response. In this use case, there is no available MQMD information for the frontend response for correlation.

Caution: Collection of WebSphere MQ Listener events should be completely avoided when using this relationship algorithm as it will cause unwanted events to be correlated into the business transactions. If there is a need to collect from the WebSphere MQ Listeners, serious consideration should be made to use the preferred WebSphere MQ Listener Relationship Algorithm Beans.

Tasks

How to Add an Analyzer to the Deployment Environment

This task describes how to add a new instance of an Analyzer to the deployment environment.

This task includes the following steps:

- "Prerequisites" on page 106
- "Launch the Analyzer Wizard" on page 107
- "Results" on page 107
- "Assign a Communication Link to the Analyzer" on page 107

1 Prerequisites

Determine the Processing Server on which to add the Analyzer. A Processing Server can have up to 5 Analyzers.

If a new Processing Server is needed to host the Analyzer, see "How to Create a Processing Server" on page 48.

Note that correlation cannot happen across Analyzers. Events for a single transaction must all be collected and sent to the same Analyzer so that correlation can occur.

An agent, on behalf of a single application, can send events to more than one Analyzer however. This can be a desirable configuration for partitioning the traffic into different Analyzers/transaction classes. For example, if the application is some sort of general purpose hub (like IBM WMB), different events coming from the same broker may have different business meanings and belong in different business transactions.

2 Launch the Analyzer Wizard

a Select **Admin** > **Transaction Management.**

b Click the **Configuration** tab in the left pane.

c Navigate to the Processing Server to which to add the new Analyzer, then right-click and select **New Analyzer.**

d The Analyzer Wizard appears. For details about this interface, see "Analyzer Wizard" on page 117.

3 Results

The new Analyzer appears in the Configuration tab.

4 Assign a Communication Link to the Analyzer

The way that an Analyzer gets events is by one or more communication links assigned to it. See "How to Assign a Communication Link to an Analyzer" on page 107.

How to Assign a Communication Link to an Analyzer

This task describes how to assign a communication link to an Analyzer. The Analyzer processes events from its assigned communication links only.

1 Select **Admin** > **Transaction Management**.

2 (left pane) Select the **Configuration** tab.

3 Navigate to and select the target Analyzer. You may have to expand the **Processing Servers** folder.

4 (right pane) Click the **Configuration** tab.

5 In the Communication Link and Data Collector Filter Assignment area, click the **Assign communication link** button.

6 In the Communication Link and Data Collector Filter Assignment group, choose the communication link to assign to the Analyzer and click **OK**.

7 Click **Apply**.

By default, the Collect All data collection filter is assigned to the communication link. To assign a different data collection filter, see "How to Assign a Data Collection Filter to a Communication Link" on page 177.

How to View Analyzer Status

This task describes how to view the status for an Analyzer.

1 Select **Admin** > **Transaction Management.**

2 (left pane) Click the **Configuration** tab.

3 Navigate to the Analyzer for which you want to view the status and select it. You may have to expand the Processing Server folder.

4 The Analyzer status is displayed on the Status tab. For details about this interface, see "Analyzer Page, Status Tab" on page 120.

How to Stop and Restart Event Collection

For some maintenance tasks in the TransactionVision deployment environment, you need to temporarily start or stop event collection of an Analyzer. These tasks include:

- Database imports or exports involving the database tables directly (not through the Transaction Management Administration pages).
- Analyzer configuration involving changing schema definition

To stop event collection for an Analyzer:

1 Select **Admin** > **Transaction Management**.

2 (left pane) Click the **Configuration** tab.

3 Select **Processing Servers** > <processing_server> > <analyzer>.

4 (right pane) Click the **Status** tab and then click the **Stop Analyzer** button. For more information about this user interface, see "Analyzer Page, Status Tab" on page 120.

The Analyzer shuts down all collection threads without waiting for them to clean out events in the event queue. While the Analyzer is shutting down all the threads, its status is set to QUIESCING. Once all collection threads are shut down, the status changes to STOPPED.

To start event collection for an Analyzer:

1 Select **Admin** > **Transaction Management.**

2 (left pane) Click the **Configuration** tab.

3 Select **Processing Servers** > <processing_server> > <analyzer>**.**

4 (right pane) Click the **Status** tab and then click the **Start Analyzer** button.

The Analyzer status becomes COLLECTING and details about event collection are displayed for each communication link associated with the Analyzer in the Analyzer and Communication Link Status area.

For more information, see "Analyzer Page, Status Tab" on page 120.

How to Stop and Restart an Analyzer

Some database maintenance tasks require that the Analyzer service or process be shut down.

A Processing Server can host up to 5 Analyzers. Each Analyzer has a unique service name from **tv_analyzer1** up to **tv_analyzer5**.

To completely stop the Analyzer process, use the **nanny** utility on the host on which the Processing Server is running, and specify the Analyzer service name:

```
nanny stopService tv_analyzer1
```

If you stop the analyzer by using the **nanny** utility, you need to use the same utility to restart the Analyzer. To restart the Analyzer, use the following nanny command:

```
nanny startService tv_analyzer1
```

For more information about the nanny command, see "nanny" on page 408.

How to Remove an Analyzer from the Deployment Environment

There are two ways of removing an Analyzer from the deployment environment.

- "Delete an Analyzer Permanently" on page 110
- "Archive an Analyzer" on page 111

Delete an Analyzer Permanently

When you delete an Anlayzer, its data remains in the database. To permanently remove the data, you need to use the TransactionVision **CreateSqlScript** utility shown in step 5.

1. Select **Admin** > **Transaction Management.**
2. Click the **Configuration** tab in the left pane.
3. Expand **Processing Servers**.
4. Navigate to the Analyzer, then right-click and select **Delete.**
5. To permanently remove the data from the database, run the following command:

   ```
   CreateSqlScript -d -s <schemaName> -e
   ```

   For more details, see "CreateSqlScript" on page 384.

Archive an Analyzer

You can set an Analyzer to Archive mode to deactivate it. A deactivated Analyzer cannot be started or stopped by the nanny and does not collect events, however its configuration is retained.

You may want to use Archive mode to retain the analyzer configuration, either for documentation purposes or if you want to make the Analyzer active again at some other time.

1 Select **Admin** > **Transaction Management**.

2 Click the **Configuration** tab in the left pane.

3 Expand **Processing Servers**.

4 Navigate to the Analyzer.

5 Click the **Configuration** tab in the right pane.

6 On the **General** tab, set **Archived**.

7 Click **Apply**.

How to Recover an Analyzer

Errors in Analyzer processing can happen in two different stages:

- Event processing - where events are unmarshalled and the **event.xml** file is created.

  In this case the events are put on the exception queues defined for the communication links and remain there. You use the **ManageQueue** utility to process them. For more information about this utility, see "ManageQueue" on page 400.

- Event analysis - processing that results in the local transaction being defined.

  In this case, the events are pulled off the queues and are longer available. The next time the Analyzer starts, it scans the associated database schema for unprocessed events and automatically recovers the statistics cache and resumes processing.

How to Enable Failure Mode on an Analyzer

This task describes how to run an Analyzer in Failure mode.

This task includes the following steps:

- "Prerequisites" on page 106
- "Enable Threshold Violation Tracking - Optional" on page 112
- "Set the Enable failure mode setting to ON." on page 113
- "Adjust Caches" on page 113
- "Set Other Related Properties" on page 114
- "Tune the Settings - Optional" on page 114

1 Prerequisites

Using failure mode requires that the **Result** attribute of the transaction be defined accurately. See "Data Rules" in "Transaction Tracing Rules Page, Rule Definitions Tab" on page 563.

2 Enable Threshold Violation Tracking - Optional

By default, failure mode captures data for transactions that have a FAILED result or have timed out.

Failure mode can also capture data for transactions that have violated their threshold setting. To enable threshold violation tracking, enable the LogSLAViiolationBean section in the Analyzer's **Beans.xml**:

```
<Module name="LogSLAViolationCtx" type="Context">
  <Module
class="com.bristol.tvision.services.analysis.eventanalysis.LogSLAViolationBean"
type="Bean">
    <Attribute name="WriteTxn" value="true"/>
    <Attribute name="LogLevel" value="Warning"/>
  </Module>
</Module>
```

The **Beans.xml** file is accessed through the **Analyzer** > **Configuration** > **XML** tab.

When this section is enabled, the response time of each transaction is continuously monitored against the threshold setting that is defined for this transaction class, and the status is stored in the **sla_status** for the transaction. If the response time exceeds the threshold, a value of 1 (=Violated) is stored, otherwise a value of 0 (=None). For transactions which age out during the analysis process (as mentioned above), a value of 2 (=AgedOut) is stored into the column. When the **WriteTxn** attribute is set to true, transactions violating the threshold are treated like failed transactions.

The **LogLevel** attribute specifies the severity level of the log message and can be adjusted to the level of the corresponding log appender. Valid values are **Info**, **Warning**, and **Error**. The default is **Warning**.

Note: If you are using the SMTP log appender to send SLA alerts via email, the **LogLevel** attribute has to be set to **Error**.

---

3 Set the Enable failure mode setting to ON.

To enable failure mode processing in the Analyzer, set the **Failure mode** check box on the **Analyzer** > **Configuration** > **Cache** tab. By default, failure mode is disabled.

The Analyzer does not need to be stopped to enable or disable failure mode. Once the mode is set and the Analyzer starts processing new events, those events are processed using the new processing mode.

4 Adjust Caches

Failure Mode requires increases to the default settings of the cache sizes. In general; you need to increase the sizes to at least twice the sizes used in the standard processing mode. See "Key Configuration Concepts for Analyzers" on page 85.

5 Set Other Related Properties

The following other configuration parameters should also be changed:

- **Thread count.** Defines the number of flushing threads to use for Failure Mode. This number should be set high enough so that the cache data can be flushed fast enough, but low enough to not waste too many system resources. The default value is 2.
- **Process delay time.** Defines the amount of time, in milliseconds, which gets added to the threshold value for determining the age out timeout and is a crucial value for the Failure Mode. If this value is set too low, a lot of successful transaction data unnecessarily ages out and is written to the database, which results in poorer performance, and wasted database disk space. If this value is set too high, too much data is held in the memory caches. If the cache sizes are not large enough for the system load, cache overflows occur frequently and lead to very poor performance. The default value is 5 seconds.
- **Discard overflow.** In case of a cache overflow all the data in the caches is normally written to the database. If the overflow was caused, for whichever reason, by a longer lasting system slowdown, the constant flushing of the cache could further worsen the situation and slow down the Analyzer even more. If the main concern is to avoid any chance of filling up the event queue in such situations, you can set the above parameter to true and force the discarding of all cache data in case of an overflow. The default value is false.

6 Tune the Settings - Optional

The diagnostic status output of the Analyzer can assist in finding appropriate configuration values for the Failure Mode. The following steps show how to get a meaningful status.

a In the Analyzer configuration page, enable **Events** > **Performance Logging** > **Log Details**.

b Enable Failure Mode.

c Run the Analyzer in Failure Mode for a sufficient amount of time on typical data.

d Run **AnalyzerManager –n** <analyzer_id> **–diagstatus**.

e Look for the section beginning with **FailureMode Cache Manager statistics for schema**.

Sample Output

```
FailureMode Cache Manager statistics for schema TEST

[…]

Number of forced flushes:          0

FailureMode FlushThread#0: last flush: 4 rows 0 ms
  current flush time = 0 ms
  total flush cycles: 213
  cycles skipped: 3
  max scan depth: 0

FailureMode FlushThread#1: last flush: 0 rows 0 ms
  current flush time = 0 ms
  total flush cycles: 309
  cycles skipped: 309
  max scan depth: 0

Max txn processing time in schema TEST: 16188 ms (business txn 858)
Avg txn processing time in schema TEST: 4842 ms
Number of txns (success/failed/agedOut) :  0/129/6
Number of overflow cache writes in schema TEST: 0
```

The values of interest are:

- **Number of forced flushes.** This is the number of times the Analyzer processing had to wait for the caches to be flushed to disk. If there are no forced flushes, the event processing and data flush happen mostly asynchronously with maximum performance. If the number of forced flushes is high, the processing threads have to suspend the analysis process until the flushing threads have emptied the caches often, which results in poor Analyzer performance. To minimize forced flushes, increase the number of flushing threads, or the size of the caches.

- **total flush cycles / cycles skipped.** The ratio of both numbers gives an indication of how well the flushing threads are used. If the number of cycles skipped is very close to the total number of flush cycles, the threads are idle most of the time and the number of flushing threads can be reduced to save some system resources. If all flushing threads show a very low number of skipped cycles, it indicates that all threads are fully utilized, and additional flush threads might be necessary.
- **Maximum txn processing time.** This is the longest time a transaction has been processing inside the Analyzer (from first to last event). The Failure Mode process_delay value should be chosen at least so big that the maximum processing time is still less than the sum of transaction class threshold and process_delay value to avoid unnecessary aging out of transaction data. If the value is set too high or too low, it can lead to performance issues as described earlier.
- **Number of overflow cache writes.** The number of times the complete cache contents had to be flushed to the database because a cache overflow occurred. Since overflow cache writes have a severe impact on the performance, it should always be tried to keep this value at zero by configuring the parameters explained above as required. If this still won't succeed it might indicate that the system load or transaction structure may not be well suited to be processed in Failure Mode.

Reference

Analyzers User Interface

This section describes:

- Analyzer Wizard on page 117
- Analyzer Page, Status Tab on page 120
- Analyzer Page, Configuration Tab on page 124
- Analyzer Page, Log Files Tab on page 141

Analyzer Wizard

This wizard enables you to define an instance of the Analyzer.

To access	**1** Select **Admin > Transaction Management.** **2** (left pane) Click the **Configuration** tab. **3** Expand **Processing Servers.** **4** Right-click the Processing Server on which you want to deploy the Analyzer and select **New Analyzer.**
Important information	➤ This wizard collects only some of the Analyzer configuration parameters and uses defaults for the remaining ones. ➤ The full set of properties is available on the Configuration tab. See "Analyzer Page, Configuration Tab" on page 124.
Wizard map	This wizard contains: General Page > Summary Page
See also	➤ "Analyzer Page, Configuration Tab" on page 124 ➤ "Key Configuration Concepts for Analyzers" on page 85

General Page

This page enables you to configure the Analyzer.

Important information	General information about this wizard is available here: "Analyzer Wizard" on page 117.
Wizard map	The Analyzer Wizard contains: **General Page** > Summary Page

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
Name	The name of the Analyzer. The name must be unique to the deployment environment.
Schema	The schema name to be created in the database that will store the Analyzer's event data. The name must be unique to the deployment environment.
Assign default communication links	Assign the built-in communication links to this Analyzer. See "Default Communication Links" on page 171.
Port	The port number where the Analyzer listens for administrative and status requests. **Default value:** 21120, 21130, 21140, 21150, 21160
Additional Classpaths	Specify the classpaths to Jar files if your event transport provider requires custom path settings or any custom bean/code.
Additional Native Paths	Specify the native paths to Jar files if your event transport provider requires custom path settings or any custom bean/code.
Do you want to enable this Analyzer for RUM Events processing?	Choose **Yes** to have the Analyzer process RUM events as well as those captured by any TransactionVision Agents. If you want to include RUM event data in the Transaction Management reports, one Analyzer in the deployment environment must be enabled to process RUM events.

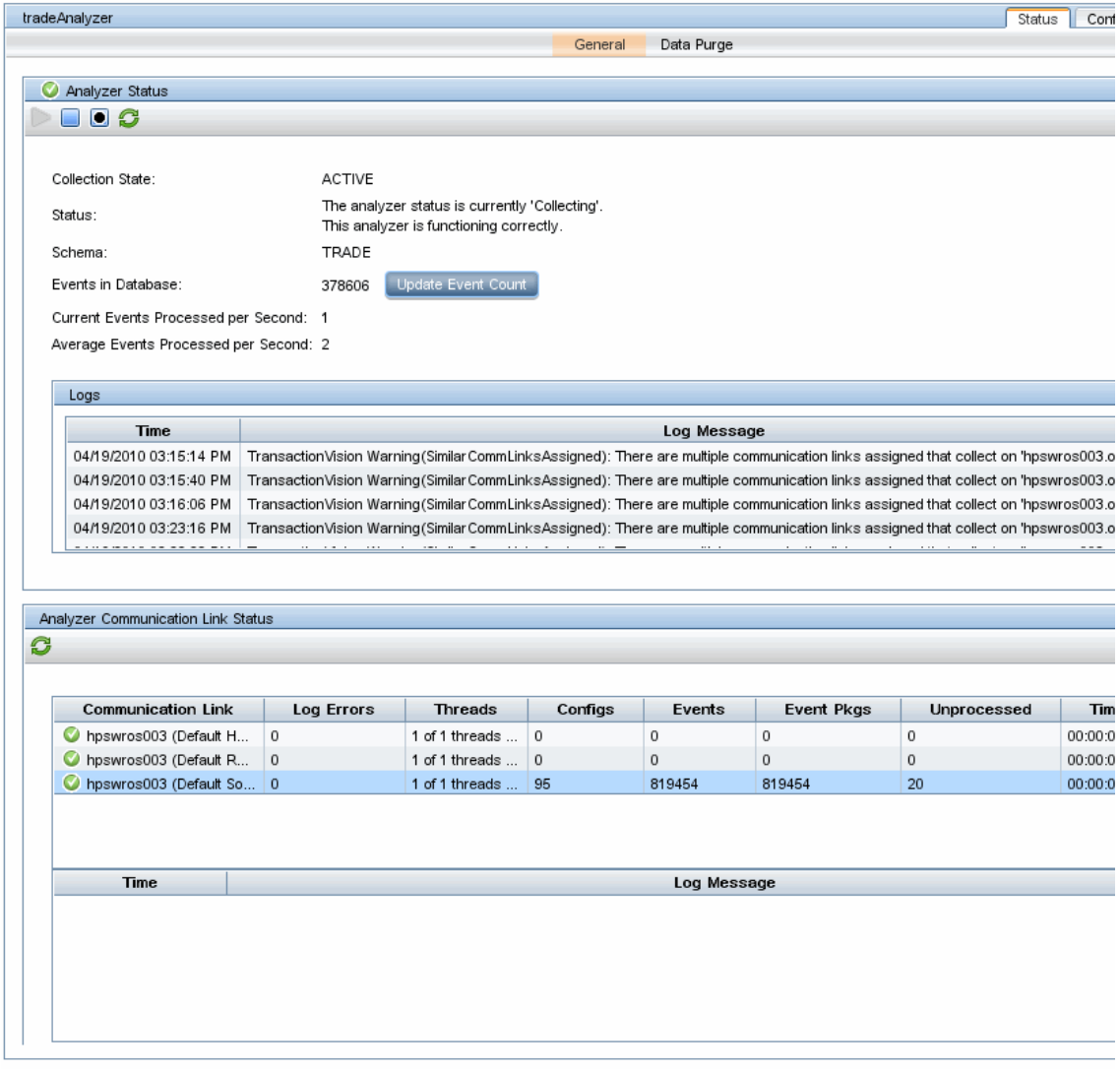
Summary Page

This page provides a summary of the configured Analyzer.

Important information	General information about this wizard is available here: "Analyzer Wizard" on page 117. Click **Finish** to add the new Analyzer to the Processing Server.
Wizard map	The Analyzer Wizard contains: General Page > **Summary Page**

Analyzer Page, Status Tab

The Status tab of the Analyzer page provides the current status of the selected Analyzer. The following is an example of the Status tab of the Analyzer page.



To access	**1** Select **Admin** > **Transaction Management.** **2** Click the **Configuration** tab in the left pane. **3** Navigate to **Processing Servers** > <processing_server> > <analyzer>. **4** Click the **Status** tab in the right pane.
See also	➤ "Processing Server Page, Status Tab" on page 66. ➤ "Configuration Tab, Data Purge Tab" on page 137.

General Tab

User interface elements are described below.

UI Element	Description
Analyzer Status	Start event collection for the Analyzer. Stop event collection for the Analyzer, allowing time for the event queue to be emptied. Stop the Analyzer, not allowing time for the event queue to be emptied. When the Analyzer restarts it will clean out the event queue. Refresh Analyzer status. **Average Events Processed per Second**. The rate of event processing. **Collection State**. Indicates whether the Analyzer is currently collecting events. **ACTIVE** indicates that the Analyzer is collecting events. **INACTIVE** indicates that the Analyzer is not currently collecting events. **Current Events Processed per Second**. The current rate of event processing. **Events in Database**. The number of events stored in the database schema for this Analyzer. Click **Update Events** to refresh the value. **Schema**. The database schema in which the Analyzer data is stored.

UI Element	Description
Analyzer Status (continued)	**Status**. The current status of the Analyzer: ➤ COLLECTING. The Analyzer is started and collecting events. ➤ COMMAND PENDING. The Analyzer is switching between statuses. ➤ FAILED. BSM cannot connect to the Analyzer; an error message is provided, and the exception and stack trace are listed. ➤ RECOVERY. The Analyzer is performing event analysis and statistics data recovery. See "Analyzer Recovery Mode" on page 143. ➤ QUIESCING. The Analyzer is shutting down all collection threads in response to a stop command. Once all collection threads are down, the status changes to STOPPED. ➤ STOPPED. The Analyzer has been stopped and is not collecting events. **Logs**. The log messages of the Analyzer.
Analyzer Communication Link Status	For each communication link assigned to the Analyzer, the following details display: **Communication Link**. Communication link name **Log Errors**. The number of error messages **Threads**. Number of event collection threads. This is controlled by the **Number of Event Collection Threads** value specified in the communication link. **Configs**. Number of configuration messages sent through this communication link since the Analyzer was last started. **Events**. Number of events processed since the Analyzer was last started.

UI Element	Description
Analyzer Communication Link Status (continued)	**Event Pkgs**. Number of event packages sent to the event queue since the Analyzer was last started. **Unprocessed**. Number of unpracticed events on the event queue. **Time skew**. Time skew across the communication link. **Last Event Received**. How long ago the most recent event was received by the Analyzer. **Last Event Processed.** How long ago the most recent event was processed by the Analyzer.

Data Purge Tab

User interface elements are described below.

UI Element (A-Z)	Description
	Start job. Start the data purge operation on this Analyzer. If the next scheduled time for the purge operation has already passed, it runs immediately; otherwise, it is placed in a waiting state until its scheduled time arrives. To set up the schedule for purging, see "Configuration Tab, Data Purge Tab" on page 137.
	Stop job. Stop the data purge operation on this Analyzer. If the data purge operation is currently running, a cancel command is first issued, followed by commands to stop and shut it down.
	Force Stop Job. Force stop the data purge operation. This button is only enabled if the data purge operation is in Quiescing mode.
	Cancel Job Run. Cancel the current data purge operation without stopping the data purge operation scheduled. The job changes to a waiting state until the next scheduled run time.

UI Element (A-Z)	Description
	Run Now. Run the data purge if the purge operation is in a waiting state.
	Cancel the current data purge operation without stopping the data purge operation scheduled. The job changes to a waiting state until the next scheduled run time.
	EditJob. Edit the job properties.
Job Logs	Any log or warning messages.
Next Run	The next scheduled run of the data purge operation.
Status	The status of the data purge operation.

Analyzer Page, Configuration Tab

Note: Some changes require the Analyzer to be automatically restarted. These changes include modifying the native path, classpath, ports and schema/xdm (which is handled elsewhere) changes.

The Configuration tab includes the following tabs:

- "Configuration Tab, General Tab" on page 125
- "Configuration Tab, Cache Tab" on page 127
- "Configuration Tab, Events Tab" on page 130
- "Configuration Tab, Data Purge Tab" on page 137
- "Configuration Tab, Properties Tab" on page 140

Configuration Tab, General Tab

The General tab of the Configuration tab enables you to view and modify Analyzer properties.

To access	**1** Select **Admin** > **Transaction Management.** **2** (left pane) Click the **Configuration** tab. **3** Expand **Processing Servers**. **4** Select <processing_server> > <analyzer>. **5** (right pane) Click the **Configuration** > **General** tab.
Important information	Some configuration changes require the Analyzer to be automatically restarted.
See also	➤ "Analyzer Wizard" on page 117 ➤ "Key Configuration Concepts for Analyzers" on page 85

User interface elements are described below:

UI Element (A-Z)	Description
Analyzer General	**Additional Classpath.** Classpaths to Jar files needed by your event transport provider. See the Additional Classpath description in "Analyzer Wizard" on page 117. **Additional Native Paths.** Classpaths to Jar files needed by your event transport provider. See the Additional Classpath description in "Analyzer Wizard" on page 117. **ApplyTimeSkew.** Disable if all the hosts in the deployment environment (Processing Servers, Queue Managers, agents) are already time synchronized, thereby guaranteeing that the time on all of the hosts is always the same. **Default value:** On **Archived.** Enable to stop the Analyzer and place it in archive mode. See "How to Remove an Analyzer from the Deployment Environment" on page 110. **Default value:** Off **Debug.** Enables debug logging. See "About Logging" on page 88. **Default value:** Off **Description.** Description of the Analyzer. **Name.** Name of the Analyzer. **Native path.** Native paths to Jar files needed by your event transport provider. See the Additional Native Path description in "Analyzer Wizard" on page 117. **Port.** The port number on which the Analyzer runs. See the Port description in "Analyzer Wizard" on page 117. **Schema.** Schema name to contain the Analyzer data. See the Schema description in "About Logging" on page 88. **SSL Port**. The port number on which the Analyzer runs when enabled for SSL. See "Security" in the *HP TransactionVision Deployment Guide* PDF. **Trace.** Enables trace logging. See "About Logging" on page 88. **Default value:** Off

UI Element (A-Z)	Description
Communication Link and Data Collection Filter Assignment	Assign a communication link to the Analyzer. Assign a data collection filter to a communication link. Delete the selected communication link. **Name**. The communication link assigned to this Analyzer, and the data collection filter assigned to it.

Configuration Tab, Cache Tab

The Cache tab of the Configuration tab enables you to view and modify Analyzer properties related to the caches.

To access	**1** Select **Admin** > **Transaction Management.** **2** (left pane) Click the **Configuration** tab. **3** Expand **Processing Servers**. **4** Select <processing_server> > <analyzer>. **5** (right pane) Click the **Configuration** > **Cache** tab.
Important information	Some configuration changes require the Analyzer to be automatically restarted.
See also	"Key Configuration Concepts for Analyzers" on page 85

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element (A-Z)	Description
General	**Event Objects.** The cache size for data cached per event. **Default value:** 1000 **Flush Interval(min).** How often the statistics cache is written out to the database. Until the data is written to the database, it won't be visible in reports or topologies. Depending on the volume of incoming requests, it might be useful to either flush more often, or less often. A longer flush time means fewer database writes, but will result in a larger cache size. **Default value:** 10 **RUM Objects.** The cache size for RUM data processing. **Default value:** 1000 **Transaction Objects.** The cache size for data cached per correlation/transaction. **Defeat value:** 1000

UI Element (A-Z)	Description
Database	**Discard Overflow.** In case of a cache overflow all the data in the caches will normally be written to the database. If the overflow was caused, for whatever reason, by a longer lasting system slowdown, the constant flushing of the cache could further worsen the situation and slow down the Analyzer even more. If the main concern is to avoid any chance of filling up the event queue in such situations, you can set this option to On and force the discarding of all cache data in case of an overflow. **Default value:** Off **Failure Mode.** Enables failure mode on the Analyzer. In failure mode, the Analyzer stores only event data of failed transactions or transactions violating their threshold. Failure mode is a significant change to the TransactionVision data flows and requires several configuration changes. See "How to Enable Failure Mode on an Analyzer" on page 112. **Default value:** off **Number of Threads.** The number of threads to use for writing the cached analysis data to the database. The best value is dependent on the hardware the Analyzer is running on and can only be determined by performance measuring with different values, but a good starting point would be to set this number to half the number of collection threads used for a particular schema. **Default value:** 2. **Process Delay Time.** The amount of time, in milli-seconds, which gets added to the threshold value for determining the age out timeout and is a crucial value for the Failure Mode. If this value is set too low, a lot of successful transaction data will unnecessarily age out and be written to the database, which will result in poorer performance, and wasted database disk space. If this value is set too high, too much data will be held in the memory caches. If the cache sizes are not large enough for the system load, cache overflows will occur frequently and lead to very poor performance. **Default value:** 5 seconds

UI Element (A-Z)	Description
Database (continued)	**Standard Mode.** Enables standard mode on the Analyzer. **Default value:** on **Skip Threshold (Idle Threads).** Skip threshold for idle flush threads for flushing the caches to the database. **Default value**: 50 ms **Thread Wait Time (msec).** Wait time for idle flush threads. **Default value**: 100 **Wait Time until Overflow (msec).** Wait time until cache overflow is reported. **Default value**: 60

Configuration Tab, Events Tab

The Events tab of the Configuration tab enables you to view and modify Analyzer properties related to event handling.

To access	**1** Select **Admin** > **Transaction Management.** **2** (left pane) Click the **Configuration** tab. **3** Expand **Processing Servers**. **4** Select <processing_server> > <analyzer>. **5** (right pane) Click the **Configuration** > **Events** tab.
Important information	Some configuration changes require the Analyzer to be automatically restarted.
See also	"Key Configuration Concepts for Analyzers" on page 85

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
General	**Create Client Host System Model Objects.** Whether client host system model objects are created and stored by the Analyzer. These objects are needed if the Web Client Grouping criteria is set to **Client Host** in the Component Topology. **Default value:** off **FailSafe Collection Shutdown.** Stop sending configuration messages to agents if a serious failure occurs in event collection and processing. **Default value:** on **Garbage Collection Sleep (sec).** Interval for forced garbage collection. **Default value:** 120 **No Events Processing.** Disables event processing. If you want to only measure the incoming rate of events to the event queue, you can enable this option. No event or transaction-level data is written to the database schema or therefore available for the reports and topologies. **Default value:** off **Read from Jar.** Reads raw events from a JAR file <TVISION_HOME>/logs/<SCHEMA>_raw_events.jar. See "Storing and Using Raw Events" on page 146. **Default value**: off **Write to Jar.** Stores raw events into a JAR file <TVISION_HOME>/logs/<SCHEMA>_raw_events.jar. See "Storing and Using Raw Events" on page 146. **Default value:** off

UI Element	Description
Database	**Batch Commit Count.** The number of processed events to batch before a commit is issued. **Default value:** 50 **Commit Bytes Threshold.** The number of bytes (of processed events) after which a commit is forced. **Default value:** 1,000,000 **Commit time threshold.** The number of seconds after which a commit (of processed events) is forced. **Default value:** 1 **JDBC Batch Count.** If **Use JDBC Batching** is specified, the number of SQL operations to batch in JDBC batching mode. This value should be equal to or a multiple of the **Batch Commit Count** value. **Default value:** 50 **Keep raw events.** Once a raw event has been processed, it is no longer needed. However if you want to process the events again (for example, with a custom Java bean for a different type of analysis), you can retain them by enabling this setting. **Default value:** off **Move user events to User Data table.** Once a raw event has been processed, it is no longer needed. However if you want to process the events again (for example, with a custom Java bean for a different type of analysis), you can retain them by enabling this setting. **Default value:** off **Read from buffer table.** Read raw events from the database in addition to reading from the queue. Use this property to test custom beans or troubleshoot issues. It should always be set to Off in a production environment. **Default value:** off **Use JDBC Batching.** Executes database statements in batch mode. **Default value:** on **Use Oracle batching.** Use Oracle specific update batching instead of JDBC batching. TransactionVision must be manually configured to use the Oracle JDBC driver. See the Use Custom JDBC URL property on "Database Connection Parameters Page" on page 56. **Default value:** off

UI Element	Description
Analysis	**Correlation Limit.** Limit the number of event relations being created for the same correlation key. **Default value:** -1 (no limit) **Event Matching Interval.** The time in seconds to wait between every invocation of event matching. **Default value:** 600. **Generate API only transactions.** By default the Analyzer does not perform any transaction analysis on events that have been collected with a Data Collection Filter setting of User Data Range = API Names Only. Set this property to **On** to change this behavior. **Default value:** off **Latency Resolution (1/sec).** The resolution used to calculate latency between events. Possible values are: ➤ 1 to specify seconds ➤ 10 to specify 1/10 seconds ➤ 100 to specify 1/100 seconds ➤ 1000 to specify milliseconds **Default value:** 100 **Local Transaction Limit.** The maximum number of local transactions. **Default value:** -1 (no limit) **Partial Event Handling.** Handle partial WebSphere MQ events from blocking MQGET API. With this option on, additial processing will be enabled in the Analyzer to handle potential partial events from blocking MQGET API. When turned off, ENTRY only events from blocking MQGET API will be discarded, and EXIT only events will be processed. Since, in most cases, the EXIT part of an MQGET contains everything needed, turning on this additional processing in the Analyzer is not needed. **Default value:** off

UI Element	Description
Analysis (continued)	**Partial Event Lifetime(min).** The maximum time in minutes a partial event entry can exist in the partial_event table. When this time limit is reached, the partial event is flushed. **Default value:** 10 minutes. **Separate Child Thread Transactions.** By default, if a servlet spins off a thread to make some JMS calls, the servlet passes tracking information to the child thread. The result is that both the servlet and JMS events belong to the same transaction. However, you may want to separate these events into different transactions. For example, a servlet may spin off a long-running thread that you do not want to be part of the same transaction as the servlet. If you do not want threads spun off by a servlet to be included in the same transaction as the servlet, list the servlet program name as the value for this property. Separate multiple program names with a servlet, list the servlet program name as the value for this property. Separate multiple program names with a comma: **program1, program2**.
RUM	**Lifetime**. The time in minutes that RUM events are kept before they are discarded from the cache (and no longer eligible for correlation). **Default value**: 1 **Cleanup Sleep Time Interval**. The time in millesconds to wait before cleanup of RUM events. **Default value**: 60000 **Sleep Time Interval**. The time in milliseconds to wait for RUM events arriving from the RUM Engine. **Default value**: 1000

UI Element	Description
Performance	**Count Interval.** The number of events after which performance data is logged. **Default value:** 500 **Enable Performance.** Enables performance logging. **Default value:** off **Enable Queue Depth Retrieval.** Retrieves the current event queue depth for each log interval. **Default value:** off **Log Details.** Generates detailed performance and statistics logs. **Default value:** off **Start at Event Count.** The number of events to be processed before performance logging starts. **Default value:** 500

UI Element	Description
BPI	**Enable BPI Integration.** Enables BPI for this Analyzer. **Default value:** off **Event Expiry Time(sec).** Time that delayed events (events that could not be delivered) are kept before being purged. **Default:** 600 **Event Thread Retry Interval(sec).** Time the Analyzers checks to clean up expired events and retry any events that failed to send. **Default:** 120 **Initial Context Factory.** The classname of the JNDI context factory. This value depend on which JMS vendor you use (see their documentation for details). Some examples are: com.sun.jndi.fscontext.RefFSContextFactory com.tibco.tibjms.naming.TibjmsInitialContextFactory. **Password.** Optional setting if authentication to the JMS provider is required. **Provider URL.** The URL to the JNDI repository, format is dependent on the JMS vendor. A RefFSContextFactory has a URL similar to file:/C:/jndi. For TIBCO, you might use something like tibjmsnaming:// host:7222. **Queue Connection Factory.** The name of the Queue Connection Factory JNDI object. If using the built-in Sonic MQ to send events to BPI, the queue and corresponding JNDI mapping (TV2BPI.EVENT.QUEUE) are created automatically. **Queue Name.** The name of the JNDI object (if JNDI is used), or the actual name of the queue (in the case of WMQ JMS). **Username.** Optional setting if authentication to the JMS provider is required.

Configuration Tab, Data Purge Tab

The Data Purge tab of the Configuration tab enables you to control when event data is deleted from the database.

To access	**1** Select **Admin** > **Transaction Management.** **2** Click the **Configuration** tab in the left pane. **3** Expand **Processing Servers**. **4** Select <processing_server> > <analyzer>. **5** Click the **Configuration** > **Data Purge** tab in the right pane.
Important information	➤ By default, no event data is purged. ➤ You can monitor the data purging operation on the Analyzer Status tab. ➤ Data purging can also be done through the deleteEvents command line utility. See "DeleteEvents" on page 391.
See also	"About Data Purging" on page 86

The user interface elements are grouped into three categories:

➤ "Purge Options" on page 138

➤ "Common Options" on page 138

➤ "Advanced Options" on page 139

Purge Options

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
Time	**Time Range.** Delete all events that were collected in the specified time period. **Newer Than.** Delete all events that were collected after the specified time. **Older Than.** Delete all events that were collected before the specified time. **Default value:** Delete all events older than 48 hours from the current time.
Event Count Limit	**Event Count Limit**. The number of events processed after which deletion occurs. The oldest events are deleted first. **Default value**: none
Query Name	**Query**. The query used to select the events to be deleted. **Default value**: none

Common Options

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
Purge Data Job Interval	Maximum number of events to process before cleaning out the system model and transaction tables. **Default value**: 1000
Disable Purge Job	Disables the purging operation. **Default value**: off

UI Element	Description
Force Option	Additional options that allow you to control which data is deleted. Specify the options in the Advanced Options area below. **Default value**: none
Topology Graph Stats	Delete statistics data as well as event data. Statistics data is the topology and business transaction statistics table. **Default value**: off

Advanced Options

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element (A-Z)	Description
	Create a new option.
	Edit an existing option.
	Deleted the selected option.
Clean Queue Objects	Delete queue system model objects that are no longer referenced. **Default value**: false
Commit Count	Commit database transactions after the specified COUNT. **Default value**: 1000
Disable Multi Threaded Delete	No deletion threads are used. By default 4 deletion threads are used for the deletion operation. **Default value**: false
Keep Local Transactions	Do not delete local transactions from the transaction tables. Business transactions are candidates for deletion. **Default value**: false

UI Element (A-Z)	Description
Min Age	Do not delete events earlier than this minimum number of hours. For example, if **MinAge** is 48 hours, then events in the past 48 hours are not deleted regardless of other settings.
Timeout	Force deletion to end after the specified time. **Default value**: 60 minutes

Configuration Tab, Properties Tab

The Properties tab of the Configuration tab enables you to view and modify Analyzer XML files.

To access	**1** Select **Admin** > **Transaction Management.** **2** (left pane) Click the **Configuration** tab. **3** Expand **Processing Servers**. **4** Select <processing_server> > <analyzer>. **5** (right pane) Click the **Configuration** > **Properties** tab.
Important information	Be sure that you maintain valid XML when making any modifications to the files. Some configuration changes require the Analyzer to be automatically restarted.
See also	See the *HP TransactionVision Advanced Customization Guide* PDF.

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element (A-Z)	Description
Name	The name of the XML file, which can be customized.
Text	The content of the file. Enter **CTRL-F** to launch a Find dialog. Click **Accept** to apply your changes, or **Cancel** to discard them.

Analyzer Page, Log Files Tab

The Log Files tab of the Analyzer page provides the current log files for the selected Analyzer.

To access	**1** Select **Admin** > **Transaction Management.** **2** (left pane) Click the **Configuration** tab. **3** Navigate to **Processing Servers** > <processing_server> > <analyzer>. **4** (right pane) Click the **Log Files** tab.
Important information	The <**TVISION_HOME**>**/logs** directory on the host on which the Processing Server is installed contains all log files. Only the log files related to the Analyzer are shown here. If the log file is larger than 5 megabytes, only the most recent 5 megabytes of the log file are loaded.
See also	➤ "Processing Server Page, Log Files Tab" on page 73 ➤ "About Logging" on page 88

User interface elements are described below.

UI Element	Description
Log Files	Refresh the log files. **Log name.** The name of a log file. **Log size**. The size of a log file.
Contents of log file <log file>	The log file content.

Troubleshooting and Limitations

This section provides the following troubleshooting and limitation information:

- "Common Log Entries" on page 142
- "Retrieving Events From the Exception Message Queue" on page 143
- "Analyzer Recovery Mode" on page 143
- "Memory Resource Problems" on page 145
- "Storing and Using Raw Events" on page 146
- "Correlated Event Threshold Exceeded" on page 149
- "Local Transaction Threshold Exceeded" on page 149
- "DataPower Event Correlation" on page 150
- "Batching Multiple Listener Events in a Transaction" on page 151
- "JDBC Batch Error Due to Incorrect Database Version" on page 153

Common Log Entries

TransactionVision Error(ErrorGettingSkewFromWebAppServer): Error getting time skew from web application server (http://OVRNTT44.ovrtest.adapps.hp.com:80/topaz/services/technical/time?alt=text/plain): Unexpected end of file from server

Check the time skew settings for the Processing Server. See "Key Configuration Settings for Processing Servers" on page 43.

Retrieving Events From the Exception Message Queue

Event messages that could not be placed on the event queue for any reason, are placed on the exception message queue defined in the communication link. See "Communication Links" on page 155, for more information.

Once the cause of the failure is resolved, use the ManageQueue utility to replay the events on the exception message queue to the event queue, as in the following example:

```
ManageQueue.bat -iq TVISION.EXCEPTION.QUEUE -oq TVISION.EVENT.QUEUE
-qm YOUR_QUEUE_ MANAGER
```

For more information about this utility, see "ManageQueue" on page 400.

If the exception message queue contains events that have failed after the XML data of the event has been written to the database (in the event analysis stage), those events cause a “Duplicate XML event” warning when they are replayed to the event queue. You can safely ignore this warning.

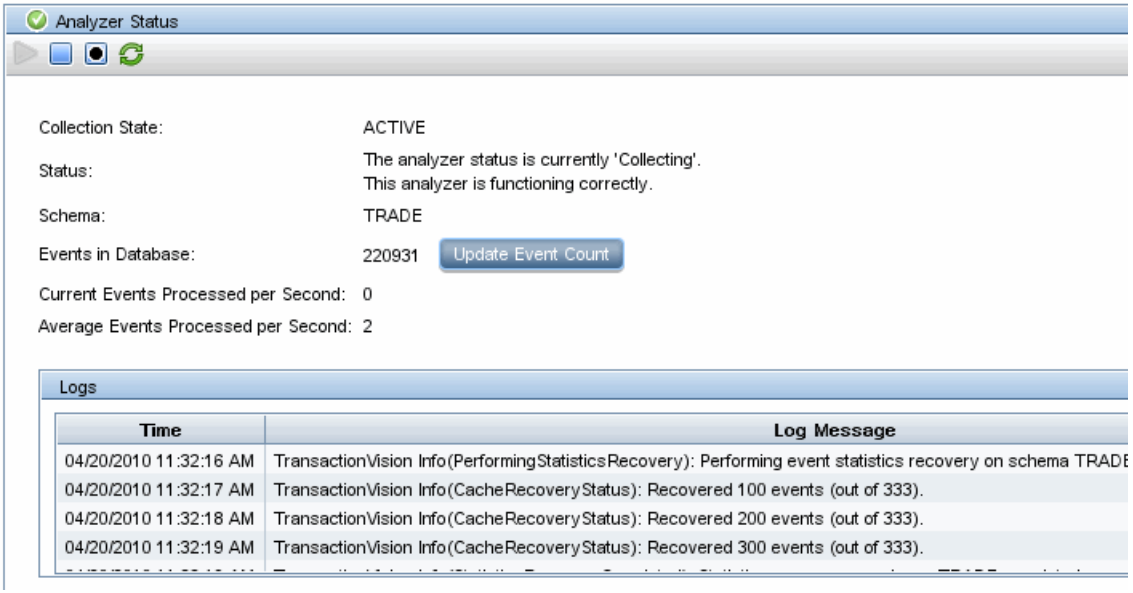
Analyzer Recovery Mode

The Analyzer uses in-memory caches to accumulate and store event analysis data as well as the statistics data for the Static Component Topology. This data is flushed and written to disk frequently by separate flush threads.

While the Analyzer is running, it keeps track of which events the analysis and statistics data has been flushed. If the writing of the cache data fails for any reason, the transaction and statistics data is incomplete, and the Analyzer schema is marked for RECOVERY mode. Failures that can lead to RECOVERY mode include database errors, event processing errors, and abnormal termination of the Analyzer process.

When an Analyzer whose schema has been marked for RECOVERY is started, the Analyzer goes into RECOVERY mode to regenerate the analysis and statistics data from the stored XML event. The time needed for the recovery process is proportional to the number of events that need to re-analyzed, and is usually slower than the normal event processing rate.

When the Analyzer is in RECOVERY mode, the Analyzer status displays the number of events processed and the total number of events to process (for example, “230 of 2140 events processed”). Click **Refresh** to update the number of events processed. A log entry in the **AppLog** file also shows how many events need to be recovered:



If Trace logging is enabled for the Analyzer, the progress of the recover process is also logged. To enable trace logging for the Analyzer, set the value of the Trace property to On on the Analyzer Configuration page.

Once the data has been flushed to disk successfully, the recovery mode of the schema is reset, and the Analyzer starts up normally the next time.

Note: If the data is not flushed because of a processing error which cannot be resolved, the Analyzer is not able to re-analyze the corresponding events and continue to start up in RECOVERY mode. In such a case, the recovery state can be cleaned out with the FlushStatusUtil utility. After the state has been cleaned, the Analyzer does not attempt to recover those events again. For more information about this utility, see "FlushStatusUtil" on page 399.

Memory Resource Problems

The amount of memory the Analyzer can use for event processing is bound by the JVM parameter "-Xmx" which defines the maximum JVM heap size. The default value is 256 Megabytes, but this value can be changed via the **JVMOPTS** parameter in the <TVISION_HOME>\bin\start_analyzer.bat or start_analyzer.sh script.

If the in-memory caches in the Analyzer are configured too large, the Analyzer tries to keep more data in memory than the JVM is capable of and at some point throws an OutOfMemory exception such as:

```
TransactionVision Fatal(OutOfMemoryError) : Please check your TVision cache size settings in file <TVISION_HOME>/config/services/CacheSize.properties and the JVM heap size setting.
```

In this case the Analyzer stops processing, and the error is logged to the **AppLog** file. To resolve this problem either reduce the cache sizes on the Analyzer **Configuration** > **Cache** tab or increase the JVM heap size in the <TVISION_HOME>\bin\start_analyzer.bat or start_analyzer.sh script.

Another possible memory related error is the overflow of the analysis caches. When this error occurs, the following error is logged to the **AppLog** file:

```
TransactionVision Error(EventAnalyzeFailed): Failed to analyze events from ('CSQ1' 'TVISION.EVENT.QUEUE') in database ('TVISION'): TransactionVision Error(CacheOverflow): Overflow of DB cache RelationLookup in schema MQIMS
```

This error can indicate two different issues: a) the DB caches are too small for the event volume (solution is to increase the cache sizes on the Analyzer Cache tab), or b) the event data cannot be written fast enough to the database (solution is to increase the throughput of the database).

A cache overflow means that the amount of data for one or more transactions that the Analyzer is trying to store in memory exceeds the configured capacity of the caches. The caches most likely to overflow are "LocalTransaction" (the size of which may be configured by the **Transaction Objects** setting), "RelationLookup", and "EventRelation" (both of which may be configured by the **Event Objects** setting).

The follow table summarizes possible causes of a cache overflow and includes instructions regarding how the problem may be corrected:

Cause of cache overflow	Corrective action required
The monitored transaction data is very complex and requires larger cache sizes.	Increase the appropriate cache size parameter in CacheSize.properties.
An erroneous custom correlation or transaction bean generates a huge amount of incorrect relational or transactional data for one event.	Check the logic of any custom beans, especially the generation of custom relation lookup keys (see "Correlated Event Threshold Exceeded" on page 149 for details).

Storing and Using Raw Events

The TransactionVision agents generate event messages and put them on the event queue. The Analyzer collects these event messages, processes them, and stores them in the database. The unprocessed event messages are called raw events.

TransactionVision provides two ways to store and re-process raw events in a JAR file: either using the ManageQueue utility or configuring the Analyzer during processing.

Using the ManageQueue Utility

- ManageQueue can get raw events from a queue (such as the event queue or exception queue) to a JAR file. This can be used to obtain events that the Analyzer failed to process (if those events are still available on the exception queue) for initial investigation.
- ManageQueue can read a JAR file and put the events back to a queue (such as the event queue).

For more information about using the ManageQueue utility, see "ManageQueue" on page 400. Specifically, see the following examples:

- Archive all messages on the event queue into a jar file while leaving them on the queue.
- Replay all events stored in a jar file for the TVISION schema to the **TVision–tv.qmgr** communication link.

Configuring the Analyzer

The Analyzer can be configured to:

- Automatically store raw events that are collected during Analyzer processing into a JAR file. This should be used after a problem in the Analyzer has occurred to obtain all events that were collected.

Note: Since writing the JAR file has a performance penalty it should never be enabled during production.

- Read raw events from the JAR file and process them as if they originated from the event queue.
- Enable JAR reading and writing during the Analyzer run time.

To configure a JAR file for storing raw events:

1. Select **Admin** > **Transaction Management.**
2. (left pane) Click the **Configuration** tab.
3. Expand **Processing Servers**.
4. Select <processing_server> > <analyzer>**.**
5. (right pane) Click the **Configuration** > **General** tab.
6. Set the **Write to jar** check box.
7. Stop and Start the Analyzer.

   Once you start an Analyzer and collect events, a JAR file with the name **SCHEMA_raw_events.jar** is created in <**TVISION_HOME**>**/logs**.

   Any events for this Analyzer are stored in the JAR file.
8. Stop the Analyzer to use the JAR file.

To configure a JAR file for processing raw events:

1. Select **Admin** > **Transaction Management.**
2. (left pane) Click the **Configuration** tab.
3. Expand **Processing Servers**.

4. Select <processing_server> > <analyzer>.
5. (right pane) Click the **Configuration** > **General** tab.
6. Set the **Read from jar** check box.
7. Stop and Start the Analyzer.
8. Copy the JAR file with the name <**SCHEMA**>**_raw_events.jar** into **$TVISION_HOME/logs.** Replace <**SCHEMA**> with the actual schema name for the Analyzer.

   The Analyzer gets the JAR file, processes all events in the file, and then renames the JAR file to <**SCHEMA**>**_raw_events.jar.processed.TIMESTAMP**. This avoids the Analyzer from processing the JAR file again the next time the Analyzer is started. Also, this allows multiple JARs to be processed (after each one has finished processing) for one Analyzer.

To enable JAR writing, run the command:

AnalyzerManager -debugcmd write_jar on

To disable JAR writing, run the command:

AnalyzerManager -debugcmd write_jar off

To enable JAR reading, run the command:

AnalyzerManager -debugcmd read_jar on

To disable JAR reading, run the command:

AnalyzerManager -debugcmd read_jar off

Correlated Event Threshold Exceeded

The Analyzer issues the following warnings when it detects potential problems correlating events into local transactions.

```
TransactionVision Warning(): Number of correlated events exceeds threshold: event ID='(1245,23)', count=30.
```

The event with the specified ID has been correlated into a message path together with an unusually large number of related events. If you are using custom correlation beans, this warning could indicate that the logic in the custom beans does not work correctly.

If no custom correlation logic is in place, this warning could indicate that the message data in the events used for correlation are not well-suited for the standard correlation, and custom correlation may be necessary. For example, applications may re-use WebSphere MQ message IDs.

Local Transaction Threshold Exceeded

The Analyzer issues the following warnings when it detects potential problems correlating local transactions into transactions

```
TransactionVision Warning(): Number of local transactions in one business transaction exceeds threshold: business txn ID='32', count=1000.
```

The transaction with the specified ID consists of an unusually large number of local transactions. If you are using custom analysis beans, this warning could indicate that the logic in the custom beans does not work correctly.

If no custom analysis logic is in place, this warning could indicate that the nature of the message data might not be well-suited for the standard analysis algorithms, and custom correlation/transaction logic may be necessary.

DataPower Event Correlation

If the proper steps are not made to generate and propagate the correlation information through the monitored DataPower services, the topology graphs in TransactionVision may not properly show message flow between events. Or, worse, some events may not be properly correlated into their proper business transaction.

There are several things that can be done to help diagnose and fix problems with correlation. In addition to the standard DataPower troubleshooting tools such as the multistep probe and log files, the events received in TransactionVision can be examined for potential correlation issues.

The following six fields in the DataPower events are important for correlation:

ws-correlator-sfid	Main TransactionVision Correlator
ws-dp-socode	DataPower Service Id
ws-client-socode	Client Service Id
ws-server-socode	Server Service Id
ws-client-hopcount	Client Hop Count
ws-server-hopcount	Server Hop Count

The fields are relevant as follows:

- Business Transaction Correlation

  The **ws-correlator-sfid** is used to correlate DataPower events into the same business transaction. If this field is missing, there is no correlation being created for the service.

- Message and Service Call Flow Paths

  While business transaction correlation is the most important part of correlation in TransactionVision, correlating DataPower service to service calls is important to properly show service call flow with the TransactionVision topology graphs.

In addition to correlating events into a business transaction, TransactionVision also correlates events to other events for message or call relationships or paths. This deeper event level correlation is controlled by the Service Id and the Hop Count fields.

The ws-dp-socode contains a unique id for the service. If an event contains a ws-client-socode, it means that another service of the given id has called this service. If an event contains a ws-server-socode, it means that this service called another service of the given id.

The Hop Counts help TransactionVision determine the call order.

For general DataPower Agent troubleshooting information, see "Installing and Configuring the IBM WebSphere DataPower Agent" in the *HP TransactionVision Deployment Guide* PDF.

Batching Multiple Listener Events in a Transaction

In environments where the WebSphere MQ channels have a non-zero BATCHINT (batch interval) parameter, events are batched up by the WebSphere MQ listener process and committed after n events. This causes TransactionVision to group together transactions that would normally be separate.

To work around this issue, a local transaction definition file can be defined that splits up listener messages into separate local transactions or unit of works (UOW), regardless of their actual commit point.

The following is a sample definition file. See the *HP TransactionVision Advanced Customization Guide* PDF for more information about this file. The following rules may be further customized to include specific queues, queue managers or event brokers.

```
<LocalTransactionDefinition>

  <LocalTransactionType dbschema="*" hasMultiTracking= "false" >
    <Match xpath="/Event/StdHeader/TechName" operator= "EQUAL"
value="MQSERIES"/>
    <Match xpath="/Event/StdHeader/ProgramName" operator= "EQUAL"
value="AMQRMPPA"/>

    <LocalTransactionAttributes>

        <Attribute name="LookupKey">
          <Path>/LocalTransaction/LookupKey</Path>

          <ValueRule name="SetLookupKey">

             <Value type="XPath">/Event/EventID/ @programInstID</Value>

             <Value type="Constant">-</Value>
             <Value type="XPath">/Event/EventID/ @sequenceNum</Value>

       </ValueRule>

       </Attribute>

       </LocalTransactionAttributes>

  </LocalTransactionType>

</LocalTransactionDefinition>
```

JDBC Batch Error Due to Incorrect Database Version

The following exception in the Analyzer logs indicates that DB2 8.1 FixPack 11 (also known as DB2 8.2 FixPack 4) is not installed:

```
2006-07-05 15:15:28,562 [MQJava0-July05-11] ERROR AppLog - TransactionVision
Error(DBFailedEventStore): Failed to store events from ('chenhpc.tv1.manager'
'TVISION.EVENT.QUEUE') in database ('tv'): Non-atomic batch failure. The batch was
submitted, but at least one exception occurred on an individual member of the batch.
Use getNextException() to retrieve the exceptions for specific batched elements.

2006-07-05 15:15:28,562 [MQJava0-July05-11] ERROR AppLog - TransactionVision
Error(ProcessEventError): Error while processing event.

2006-07-05 15:15:28,562 [MQJava0-July05-11] ERROR AppLog - TransactionVision
Error(JDBCBatchError): Error while executing JDBC batch operation 'INSERT INTO
JULY05.USER_DATA (proginst_id, sequence_no, data_num, type, ccsid, event_time,
user_data) VALUES (?,?,?,?,?,?,?)' - SQL Error : com.ibm.db2.jcc.b.vd: Non-atomic
batch failure. The batch was submitted, but at least one exception occurred on an
individual member of the batch. Use getNextException() to retrieve the exceptions for
specific batched elements.
```

4

Communication Links

This chapter includes:

Concepts

- Communication Links Overview on page 156
- Message Queues in Communication Links on page 157
- Message Queue Configurations on page 160
- Message Queue Requirements on page 162
- Default Communication Links on page 171

Tasks

- How to Create a Communication Link on page 174
- How to Assign a Communication Link to an Analyzer on page 175
- How to Test a Communication Link on page 175
- How to Modify a Communication Link on page 177
- How to Assign a Data Collection Filter to a Communication Link on page 177
- How to Monitor a Communication Link on page 178
- How to Configure DataPower Communication Links for Multiple Analyzers on page 178
- How to Remove a Communication Link on page 180

Reference

- Communication Links User Interface on page 181

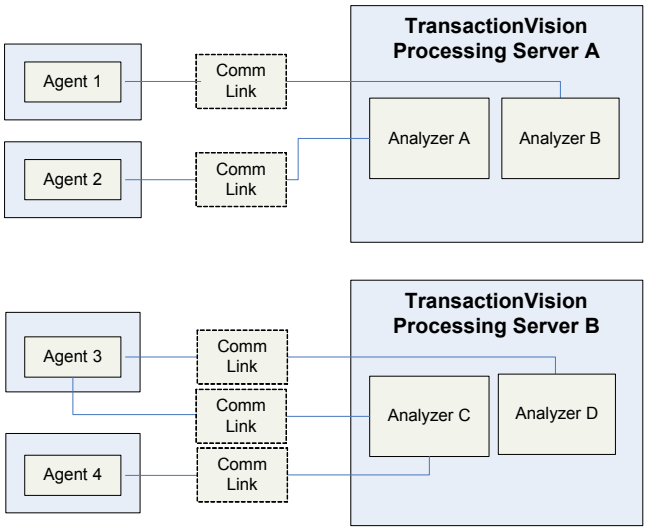
Troubleshooting and Limitations on page 205

Concepts

Communication Links Overview

A communication link defines the message queues used to pass configuration and event messages between an agent and an Analyzer. Communication links are created and managed through TransactionVision. Message queues are created and directly managed by a messaging middleware product. TransactionVision works with any of four messaging middleware products, including SonicMQ which is bundled and installed with the TransactionVision Processing Server. For more information about supported messaging middleware providers, see the *HP TransactionVision Deployment Guide* PDF.

Each agent has a communication link between it and the Analyzer to which it is sending events. For example, the following deployment scenario has 5 communication links:

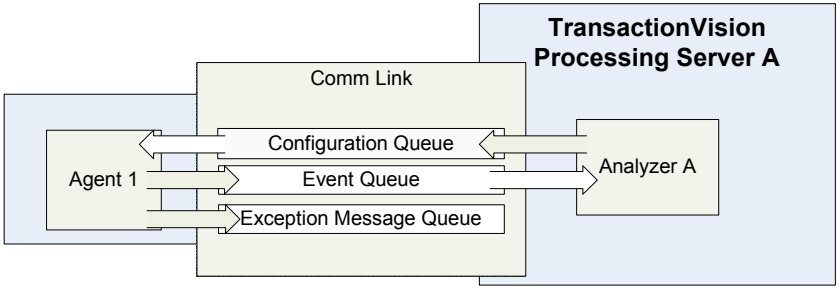


Message Queues in Communication Links

Each communication link uses three message queues:

- **Configuration queue**. Agents monitor this queue for configuration messages from the Analyzer.
- **Event queue**. The Analyzer monitors this queue for event messages from agents.
- **Exception message queue**. Contains events that fail to be processed by the Analyzer. These events are removed from the event queue and put on the exception message queue.

The following diagram shows the way events flow to and from the message queues:



This section describes the following queues:

- "Configuration Queues" on page 158
- "Event Queues" on page 159
- "Exception Message Queues" on page 159

Configuration Queues

The Analyzer sends configuration messages to the configuration queues. This configuration message defines the data collection filter conditions, the name of the event queue that event messages should be sent to, and when event collection should stop.

The Analyzer sends configuration messages at these times:

- A communication link setting is changed, such as specifying a new Time Server location.
- A data collection filter assigned to the communication link is changed, such as new criteria was added or some criteria removed.
- A new data collection filter is assigned to the communication link.
- The Analyzer state changes from INACTIVE to ACTIVE.
- The Analyzer is started.

When the Analyzer stops collecting events, it sends a new configuration message to the configuration queue indicating that no more events should be collected. The Analyzer can stop collecting events for any of the following reasons:

- Event collection is stopped for the Analyzer.
- Events stop being sent by the agent because the end time specified in the data collection filter is reached.

Event Queues

The agent evaluates each event in the monitored application against the data collection filter conditions specified in the configuration message. If the filter conditions are met, the agent creates an event entry message about the API call. The agent then delegates control to the user application to finish the work. When the work finishes, the agent creates an event exit message, and then passes the return on to the calling application. Entry and exit events are grouped into a single message to be put on the specified event queue. There may be more than one configuration message, so event messages may be put on more than one event queue.

After the first successfully monitored event, the agent continues to check the configuration queue periodically for new configuration messages.

Exception Message Queues

If an event fails to be processed by the Analyzer, either because of an invalid event or an unexpected error or exception, the event is removed from the event queue and put on the exception message queue.

Any fatal Java errors (such as **OutOfMemoryError**) causes the event to be put on the exception message queue and events are no longer pulled off the event queue until the Analyzer is restarted. If the exception message queue name is not specified, the failed events are discarded.

You can use the **ManageQueue** utility to transfer messages from the specified exception message queue to an Analyzer for reprocessing. For more information, see "ManageQueue" on page 400.

Message Queue Configurations

This section describes configuration requirements for the following communication link types:

- "WebSphere MQ Configurations" on page 160
- "TIBCO EMS Configurations" on page 161
- "SonicMQ Configurations" on page 161
- "WebLogic JMS Configurations" on page 162
- "HTTP Based Queues" on page 162

WebSphere MQ Configurations

The following are possible configurations for agents that use WebSphere MQ:

- Both the agent and the Analyzer communicate directly with the same queue manager.
- The agent uses one queue manager and the Analyzer uses a different one.
- The agent uses one queue manager, and the Analyzer sends configuration messages to a second queue manager and monitors an event queue on a third queue manager.
- The agent uses one queue manager to monitor for configuration messages and a second queue manager to send event messages to; the Analyzer uses a third queue manager.

Most agent types can use WebSphere MQ, but not all can. For more information, see "Supported Messaging Middleware Providers" in chapter 3 of the *HP TransactionVision Deployment Guide* PDF.

TIBCO EMS Configurations

The following are possible configurations for agents that use TIBCO EMS:

- Both the agent and the Analyzer communicate directly with the TIBCO EMS server.
- The agent uses one TIBCO EMS server, and the Analyzer uses a different one. The configuration queue and/or event queue on one TIBCO EMS server is defined as a route to the TIBCO route home queue of the same name on the other TIBCO EMS server.
- The agent uses one TIBCO EMS server, and the Analyzer sends configuration messages to one TIBCO EMS server but use another TIBCO EMS server to retrieve events.

Only some agent types can use Tibco EMS. For more information, see "Supported Messaging Middleware Providers" in chapter 3 of the *HP TransactionVision Deployment Guide* PDF.

SonicMQ Configurations

The following are possible configurations for agents that use SonicMQ:

- Both the agent and the Analyzer communicate directly with the same SonicMQ broker.
- The agent uses one SonicMQ broker, and the Analyzer uses a different one. The configuration queue and/or event queue on one SonicMQ broker is listed as a routing definition to the queue on the other SonicMQ broker.
- The agent uses one SonicMQ broker, and the Analyzer sends configuration messages to one SonicMQ broker but uses another SonicMQ broker to retrieve events.

Only some agent types can use SonicMQ. For more information, see "Supported Messaging Middleware Providers" in chapter 3 of the *HP TransactionVision Deployment Guide* PDF.

For information about SonicMQ's Dynamic Routing Architecture, see the *Progress SonicMQ Deployment Guide*.

WebLogic JMS Configurations

The configuration queue and event queue must use the same WebLogic JMS server to make a connection.

Only some agent types can use WebLogic JMS. For more information, see "Supported Messaging Middleware Providers" in chapter 3 of the *HP TransactionVision Deployment Guide* PDF.

HTTP Based Queues

If the agents in your deployment environment are not supported by one of the above messaging middleware providers, you can use HTTP-based message queues.

This type of message queue is used by the Tuxedo and NonStop TMF agents. This type of queue can also be used by any custom agent.

For more information about the HTTP acceptor, see the Sonic MQ product's Deployment Guide (*mq_deploy.pdf*).

Only some agent types can use HTTP based queues. For more information, see "Supported Messaging Middleware Providers" in chapter 3 of the *HP TransactionVision Deployment Guide* PDF.

Message Queue Requirements

If you are not using the SonicMQ messaging middleware product bundled with TransactionVision, you need to create and manage queues using the messaging middleware provider's tools.

Note: TIBCO EMS and WebLogic JMS are supported only for release 8.0x agents.

The queues to be used by TransactionVision have requirements as described below for each event transport provider type:

- "WebSphere MQ" on page 163
- "SonicMQ" on page 167
- "TIBCO EMS" on page 168
- "WebLogic JMS" on page 170

WebSphere MQ

The following sections describe the queue requirements for WebSphere MQ:

- "Message Length" on page 163
- "Queue Depth" on page 164
- "Event Queue Manager" on page 164
- "Event Queue Storage" on page 165
- "Event Queue Message Persistency" on page 165
- "Queue Parameters" on page 165
- "Agent Security Permissions" on page 165
- "Analyzer Security Permissions" on page 166

Message Length

The message length requirements for configuration queues are small. A default configuration message uses approximately 500 bytes. However, specifying data collection filter conditions increases the size of configuration messages. A minimum message length of 10,000 bytes is required.

The length of event messages can vary greatly, depending on the parameters passed to the API, the user data passed through the calls, and data collection filter conditions. For event queues, it is recommended to use the default maximum message length of 4194304 bytes; a minimum message length of 10,000 bytes is required. To restrict the message length based on your applications, consider the length of messages sent by your application and allow an additional 4000 bytes for event information.

Queue Depth

For configuration queues, the default queue depth should be sufficient. Analyzers send configuration messages to start data collection, to change data collection filter conditions, to end data collections, and to determine time skews across hosts. The number of configuration messages sent to a configuration queue depends on the number of Analyzers sending messages to the queue.

Event queue depth should be set to a value that matches the queue manager storage space based on the average event message size. It should be adequate to handle the peak volume. It is recommended to use the largest queue depth possible for the event queue and, in the case of remote communication link configurations, all queues in between.

In some cases, agents put event messages on the event queues faster than Analyzers retrieve them. If event delay retry is enabled on the communication link, insufficient queue depth may result in event queues filling up and agents slowing down waiting for the Analyzer to catch up. Increasing the event queue depth helps prevent this situation.

For information about average message size, see the *HP TransactionVision Planning Guide* PDF.

Event Queue Manager

It is best to have the actual event queue hosted on a queue manager other than the production queue manager, so that issues related to event queues (such as running out of disk storage due to event backlog) does not affect normal operations.

Note: The channel on which the TransactionVision event queue is located should have the channel property **CONVERT** set to **NO**. If this property is set to **YES**, event messages are discarded to the exception message queue. If the channel is being shared by queues that have messages that require the property to be set to **YES**, please create a separate channel specifically for the event queue.

Event Queue Storage

Event queue storage should match the event queue depth based on the event message size. It should not exceed the storage capability of the queue manager host system. This is especially important when the event queue and exception message queue are hosted by the production queue manager because running out of storage space may stop the queue manager completely.

Event Queue Message Persistency

The event queue message persistency property should match the event collecting policy.

Queue Parameters

Queue parameters require the following:

- Configuration and event queues must:
  - Have the Default Share Option (**DEFSOPT**) set to **SHARED**.
  - Allow Shared Access (**SHARED**).
  - Allow Get (**GET(ENABLED)**) and Put (**PUT(ENABLED)**) operations.
- TransactionVision configuration queues must allow messages of at least 10,000 bytes in length (**MAXMSGL**).
- TransactionVision event queues must allow messages of at least 10,000 bytes in length (**MAXMSGL**).
- All channels (including client channels) must have **MAXMSGL** of at least 10,000 bytes.

Agent Security Permissions

Userids of programs using the agent require the following:

- **GET** and **BROWSE** authority to the configuration queue.
- **PUT** authority to the event queue.

Analyzer Security Permissions

Users of the Analyzer require the following:

- **PUT**, **GET**, **BROWSE**, and **INQ** authority to all configuration and event queues.
- **Passid** permission to test communication links.

Note that Analyzer users can be given authority to only subsets of the configuration/event queues if you want to limit access to certain agents.

If a user does not have the required access permission on the configuration queue (**TVISION.CONFIGURATION.QUEUE** by default), the following message is found in the agent logs:

```
TransactionVision agent: cannot open event queue
TVISION.CONFIGURATION.QUEUE on queue manager merce.es1.manager: Not
authorized for access.
```

To set access permission of **TVISION.CONFIGURATION.QUEUE** for a user (tester in this example), run the following command:

```
setmqaut -m queuemanager -n TVISION.CONFIGURATION.QUEUE -t queue -p tester.
+get +browse +put +inq
```

It returns a message that the command completed successfully.

To display the access permission of **TVISION.CONFIGURATION.QUEUE** for a user (tester in this example), run the following command:

```
dspmqaut -m queuemanager -n TVISION.CONFIGURATION.QUEUE -t queue -p tester
```

It returns output similar to the following example:

```
Entity tester has the following authorizations for object
TVISION.CONFIGURATION.QUEUE:
  get
  browse
  put
  inq
```

SonicMQ

SonicMQ is included with each TransactionVision Processing Server. The messaging queues are created automatically and you do not need to configure them as described here. See "Default Communication Links" on page 171.

However you can use a standalone version of SonicMQ if necessary for your deployment environment. The following sections describe the queue requirements for SonicMQ:

- "Event Queue SonicMQ Broker" on page 167
- "Event Queue Storage" on page 167
- "Queue Parameters" on page 168
- "Agent Security Permissions" on page 168

Event Queue SonicMQ Broker

It is best to have the actual event queue hosted on a SonicMQ broker other than the production broker, so that issues related to event queues (such as running out of disk storage due to event backlog) do not affect normal operations.

Event Queue Storage

Event queue storage should not exceed the storage capability of the SonicMQ broker host system. This is especially important when the event queue and TransactionVision exception message queue are hosted by the production SonicMQ broker. The behavior of the production SonicMQ broker may be unpredictable if you run out of storage space.

Queue Parameters

Queue parameters require the following:

- TransactionVision configuration and event queues must allow receive, send, and browse operations that are defined at TransactionVision setup, to be granted to the user.
- TransactionVision configuration queues must allow messages of at least 10,000 bytes in length (**maxbytes**).
- TransactionVision event queues must allow messages of at least 10,000 bytes in length (**maxbytes**).

Agent Security Permissions

The user name that is used by the agent must have the following permissions to use the configuration and event queues:

- Receive and browse permission to the TransactionVision configuration queue.
- Send permission to the TransactionVision event queue.

Furthermore, to allow the SonicMQ Agent to resolve route and bridge information correctly, the user name provided in SonicMQ broker setup must have administrative permission.

TIBCO EMS

The following sections describe the queue requirements for TIBCO EMS:

- "Event Queue TIBCO EMS Server" on page 169
- "Event Queue Storage" on page 169
- "Event Queue Message Persistency" on page 169
- "Queue Parameters" on page 169
- "Agent Security Permissions" on page 170
- "Analyzer Security Permissions" on page 170

Event Queue TIBCO EMS Server

It is best to have the actual event queue hosted on a TIBCO EMS server other than the production EMS server, so that issues related to event queues (such as running out of disk storage due to event backlog) do not affect normal operations.

Event Queue Storage

Event queue storage should not exceed the storage capability of the TIBCO EMS server host system. This is especially important when the event queue and TransactionVision exception message queue are hosted by the production TIBCO EMS server because running out of storage space may cause unpredictable behavior of the production EMS server.

Event Queue Message Persistency

The event queue message persistency property should match the event collecting policy.

Queue Parameters

Queue parameters require the following:

- Configuration and event queues must allow receive, send, browse operations are granted to the user that are defined at TransactionVision setup.
- Configuration queues must allow messages of at least 10,000 bytes in length (**maxbytes**).
- Event queues must allow messages of at least 10,000 bytes in length (**maxbytes**).

Agent Security Permissions

The user used by the agent to connect to configuration and event queues must have the following permissions:

- Receive and browse permission to the configuration queue.
- Send permission to the event queue.

Furthermore, to allow the TIBCO EMS Agent to resolve route and bridge information correctly, the user name provided in TIBCO EMS server setup must have administrative permission.

Analyzer Security Permissions

The user used by the Analyzer to connect to configuration and event queues must have the following permissions:

- Send, receive, browse permission to all configuration and event queues.
- View-destination permission on event queues to check the pending unprocessed event count.

WebLogic JMS

Queues

Point-to-point queues should be used for all TransactionVision queues.

JMS Server Configuration

If possible, create a separate JMS Server to host the queues needed for TransactionVision.

Configure the JMS Server to use "File Persistent Store".

Default Communication Links

When a TransactionVision Processing Server is installed, it includes a SonicMQ Broker. The SonicMQ Broker is automatically started and stopped when an Analyzer on that Processing Server is started or stopped.

When a TransactionVision Processing Server is created in the Transaction Management Administration page, four default communication links are automatically created as described below. The name of these communication links is prefixed with the name of the Processing Server.

The following sections describe the default communication links:

- "Default SonicMQ" on page 171
- "Default SonicMQ RUM" on page 171
- "Default SonicMQ DataPower" on page 172
- "Default SonicMQ HTTP" on page 173

Default SonicMQ

Use this communication link when TransactionVision Agents (such as the Java and .NET Agents) use the default messaging queues on the Processing Server.

Important: Do not assign this communication link to more than one Analyzer in your deployment environment.

Default SonicMQ RUM

Use this communication link to allow a RUM Engine to send events to TransactionVision. When this communication link is assigned to an Analyzer and the RUM Engine is configured to send events to the SonicMQ Broker RUM event queue, then TransactionVision can merge data from the two sources into a single transaction.

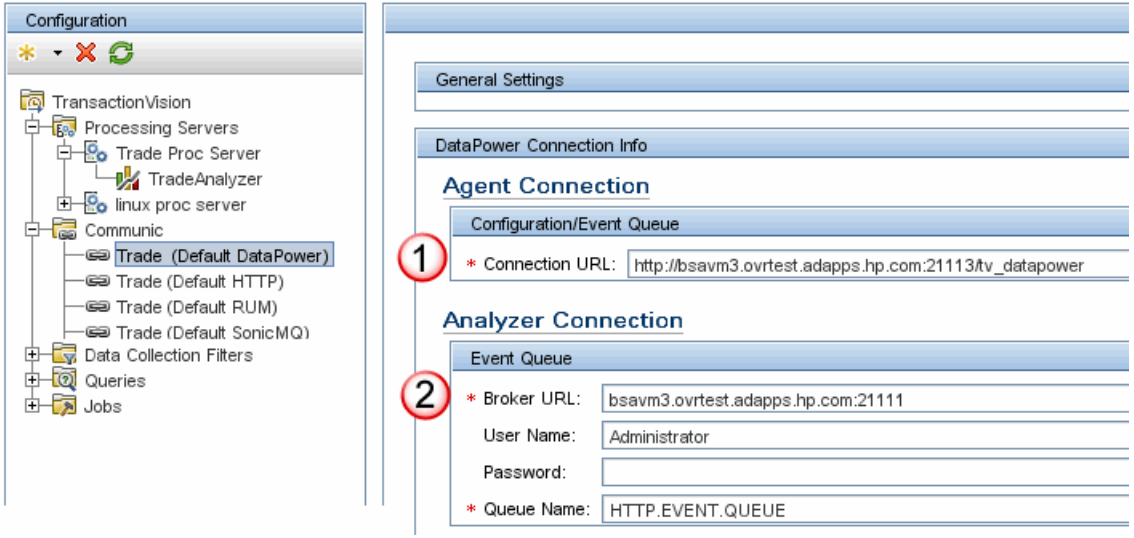
Caution: Do not assign this communication link to more than one Analyzer in your deployment environment. Also no data collection filter can be assigned to this communication link.

See also "Enabling an Analyzer for RUM" in "Key Configuration Concepts for Analyzers" on page 85.

Default SonicMQ DataPower

This communication link is used only for DataPower Agents. When this communication link is assigned to an Analyzer, all DataPower agents send events to the SonicMQ Broker HTTP event queue in that Analyzer's Processing Server.

The communication link is prepopulated as follows:



The following table describes the numbered locations in this page:

Callout	Description
1	The Agent sends event messages to the SonicMQ queue residing in the associated Processing Server through an HTTP acceptor defined by the Connection URL. In this case the associated Processing Server is **Trade**.
2	This Broker URL used by the Analyzer to retrieve event messages sent by the DataPower Agents.

Note: DataPower events are not automatically enabled for correlation. You must enable them. See "Correlation For DataPower Events" on page 96.

Default SonicMQ HTTP

Use this communication for TransactionVision Agents that use the HTTP protocol to send events to the SonicMQ Broker HTTP event queue, such as Tuxedo and NonStop Agents.

The event queue is called **HTTP.EVENT.QUEUE**, and the exception queue is called **HTTP.EXCEPTION.QUEUE**.

Agents post HTTP messages to the HTTP acceptor URL **http://<hostname>:21113/tv_http** and SonicMQ broker receives the messages and forwards them to **HTTP.EVENT.QUEUE** as JMS messages, which are then read by the Analyzer.

To configure the HTTP communication link to use SSL, you would change the URL to be **https://<hostname>:21114/tv_http**.

Tasks

How to Create a Communication Link

This task describes how to create a new communication link.

This task includes the following steps:

- "Prerequisites" on page 174
- "Step Through the Communication Link Wizard" on page 174

1 Prerequisites

You should first determine if one of the built-in communication links meets your requirements. If it does, you do not need to create a new communication link. For information about the default communication links, see "Default Communication Links" on page 171.

The Communication Link Wizard will collect information from you about the agent's environment and the Analyzer's environment. The information that you will need to specify is described in "Message Queue Requirements" on page 162. You will also need to know the names of the event queue, configuration queue, and the exception queue.

Except for SonicMQ, these queues are created by using third-party messaging middleware provider tools.

2 Step Through the Communication Link Wizard

a Select **Admin** > **Transaction Management**.

b (left pane) Select the **Configuration** tab.

c Click the **New** button and select **New Communication Link**.

The Communication Link Wizard appears. For details about this interface, see "Communication Link Wizard" on page 182.

How to Assign a Communication Link to an Analyzer

This task describes how to assign a communication link to an Analyzer.

This task includes the following steps:

- "Prerequisites" on page 175
- "Assign the Communication Link to the Analyzer" on page 175
- "Verify the Communication Link is working" on page 175

1 Prerequisites

Determine which Analyzer should handle the incoming events. For more information, see the *HP TransactionVision Planning Guide PDF*.

2 Assign the Communication Link to the Analyzer

See "How to Assign a Communication Link to an Analyzer" on page 107.

3 Verify the Communication Link is working

To test the communication link; see "How to Test a Communication Link" on page 175.

If the test shows the communication link is working successfully, the communication link is effective immediately. Events should appear on the Event Analysis report.

How to Test a Communication Link

This task describes how to test a communication link.

1. Select **Admin** > **Transaction Management**.
2. (left pane) Select the **Configuration** tab.
3. Select **Communication Links**.
4. (right pane) Select the communication link to test.

5 Choose the Analyzer with which to test from the drop-down list. Typically you would select the Analyzer to which you are going to assign the communication link.

6 Click **Test**.

If the selected communication link uses WebSphere MQ queues, the selected Analyzer sends a request message to the configuration queue requesting a confirmation on arrival (COA). The COA report message is sent to the event queue by the configuration queue that the communication link is configured to monitor. The end result is much like the behavior of typical communication between the Analyzer and agents. The communication link entry in the list of communication links is updated to show the Analyzer and test status.

If the selected communication link uses SonicMQ, WebLogic JMS, or TIBCO EMS, the test only checks whether the selected Analyzer can successfully make a connection to the given configuration queue and event queue.

If the test fails, the error message is displayed in the **Test Status** column. The following are the most common reasons for a communication link test to fail:

- Timeout waiting for a reply message.

  This error is typically caused by channel problems with remote configurations. The request message may not make it to the final queue manager that is to report, or the report message may not make it to the queue manager to which the Communication Link Editor is connected. If the link test fails for this reason, see "Troubleshooting and Limitations" on page 205.

- WebSphere MQ errors.

  WebSphere MQ problems within the communication link are reported with the WebSphere MQ error reason, which usually provides enough information to isolate the problem.

How to Modify a Communication Link

This task describes how to modify a communication link:

1. Select **Admin** > **Transaction Management**.
2. (left pane) Select the **Configuration** tab.
3. Expand **Communication Links**.
4. Select the communication link to modify.
5. (right pane) Modify the desired settings. See "Communication Link Page" on page 204.
6. Retest the communication link if desired.

How to Assign a Data Collection Filter to a Communication Link

Data collection files can be assigned only to communication links that are assigned to an Analyzer.

This task describes how to assign a data collection filter to a communication link.

1. Select **Admin** > **Transaction Management**.
2. (left pane) Select the **Configuration** tab.
3. Expand **Processing Servers**.
4. Navigate to and select the Analyzer that has the target communication link.
5. (right pane) In the Communication Link and Data Collector Filter Assignment area, right-click the target communication link and select **Assign a Data Collection Filter**.
6. Choose the target data collection filter from the Available data collection filters dialog and click **OK**.
7. Click **Apply**.

How to Monitor a Communication Link

The Processing Server Summary page displays the status and activity of each communication link in the deployment environment. See "Processing Servers Summary Page, Status Tab" on page 61.

How to Configure DataPower Communication Links for Multiple Analyzers

In some deployment scenarios, there may be a need to collect DataPower events from two DataPower appliances, and send them to two separate Analyzers on the same Processing Server. For example, you may want to keep DataPower events collected from a development environment separate from those collected from a QA environment.

To achieve this, create an additional DataPower communication link and configure the DataPower Agent to communicate with it.

This procedure is not required if the Analyzers each reside in separate Processing Servers. In those cases, separate communication links for each Processing Server are already in place.

This procedure requires two tasks:

➤ "Task 1: Create a second DataPower communication link" on page 178

➤ "Task 2: Configure the DataPower agent to send events to the second DataPower communication link" on page 179

Task 1: Create a second DataPower communication link

1 Select **Admin** > **Transaction Management**.

2 (left pane) Select the **Configuration** tab.

3 Expand **Communication Links**.

4 Right-click the default DataPower communication link and choose **Copy**.

5 In the newly created DataPower communication link:

   a Set **Connection URL** to a different URI from that in the original DataPower communication link (for example, **http://bsavm3.ovrtest.adapps.hp.com:21113/tv_datapower_dev**).

   b Set **Event Queue** to a different queue name from that in the original DataPower communication link (for example, HTTP.DEV.EVENT.QUEUE).

6 Access the Sonic Management Console on the Processing Server to create the new event queue as named in Step 5 above.

   You drill down to Queues under the message broker, right-click and choose **New Queue...**.

   For information about using the Console, see the *Progress SonicMQ Configuration and Management Guide*.

7 Assign the newly created DataPower Communication Link to the second Analyzer. See "How to Assign a Communication Link to an Analyzer" on page 175.

Task 2: Configure the DataPower agent to send events to the second DataPower communication link

On the DataPower appliance host:

1 Change the <**monitor-url**> value in **local:///properties.xml** in the HPTVMonitoring domain to the new URL defined in the second DataPower communication link (for example, http://bsavm3.ovrtest.adapps.hp.com:21113/tv_datapower_dev).

2 Change the **X-JMS-DestinationQueue Header Injection Parameter** in the **hptv-wsman-subscriber Multi-Protocol Gateway** to the new event queue name defined in the second DataPower communication link.

   For details, see "Step 2: Define TransactionVision Multi-Protocol Gateway HTTP Header Injections" in "Transformations" in the "Installing and Configuring the IBM WebSphere DataPower Agent" chapter of the *HP TransactionVision Deployment Guide* PDF.

How to Remove a Communication Link

1. Select **Admin** > **Transaction Management**.
2. (left pane) Select the **Configuration** tab.
3. Expand **Communication Links**.
4. Right-click the communication link name and select **Delete**.

Reference

Communication Links User Interface

This section describes:

- Communication Link Wizard on page 182
- Communication Links Summary Page on page 202
- Communication Link Page on page 204

Communication Link Wizard

This wizard enables you to create a communication link.

To access	**1** Select **Admin** > **Transaction Management.** **2** (left pane) Click the **Configuration** tab. **3** From the **New** menu, select **New Communication Link.**
Important information	➤ Queues must be created before the communication link can be used, but do not need to exist to define the communication links with this wizard. See "Message Queue Requirements" on page 162 for details. ➤ You can also create a communication link by copying an existing one and modifying its properties. See "How to Modify a Communication Link" on page 177 for details.
Relevant tasks	"How to Create a Communication Link" on page 174
Wizard map	The Communication Link wizard contains: General Page > Agent Connection Page > Analyzer Connection Page > Queue Settings Page > Connection Settings Page > Domain Manager Page> Event Options Page > Time Server Page > Miscellaneous Information Page
See also	➤ "Message Queues in Communication Links" on page 157 ➤ "Message Queue Configurations" on page 160 ➤ "Message Queue Requirements" on page 162

General Page

This wizard page enables you to name the new communication link and specify the type of event transport provider it requires.

Important information	➤ General information about this wizard is available here: "Communication Link Wizard" on page 182. ➤ TransactionVision includes one event transport provider product: SonicMQ. Other event transport providers are supported but must be obtained, installed and configured using the product vendor's tools.
Wizard map	The Communication Link Wizard contains: **General Page** > Agent Connection Page > Analyzer Connection Page > Queue Settings Page > Connection Settings Page > Domain Manager Page> Event Options Page > Time Server Page > Miscellaneous Information Page
See also	➤ "Message Queues in Communication Links" on page 157 ➤ "Message Queue Configurations" on page 160 ➤ "Message Queue Requirements" on page 162

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
Communication link name:	Enter a name for the communication link, which must be unique to the deployment environment.
Select an event transport provider	Specify the event transport provider that the agents and Analyzer for this communication link use. Options are: WebSphere MQ TIBCO EMS Progress SonicMQ BEA WebLogic JMS Other JMS Provider **Default:** IBM WebSphere MQ

Agent Connection Page

This wizard page enables you to specify the queue details for the Agent-specific queues.

Important information	General information about this wizard is available here: "Communication Link Wizard" on page 182
Wizard map	The Communication Link Wizard contains: General Page > **Agent Connection Page** > Analyzer Connection Page > Queue Settings Page > Connection Settings Page > Domain Manager Page> Event Options Page > Time Server Page > Miscellaneous Information Page
See also	➤ "Message Queues in Communication Links" on page 157 ➤ "Message Queue Configurations" on page 160 ➤ "Message Queue Requirements" on page 162

User interface elements are described below (unlabeled elements are shown in angle brackets):

WebSphere MQ

UI Element	Description
Queue manager the Agent will connect to:	The name of the queue manager that the agents connects to. This queue manager must be used by applications to be monitored.
Queue the Agent will receive configuration messages from:	The name of the queue that agents check for configuration messages. This queue must be a local queue on the specified queue manager. The configuration queue used by a WebSphere MQ Agent must be on the same queue manager used by the application being monitored. **Default value:** TVISION.CONFIGURATION.QUEUE
Queue the Agent will send event messages to:	The name of the queue to which agents send event messages. This queue should only be used by TransactionVision. **Default value:** TVISION.CONFIGURATION.QUEUE
Is the queue manager in which the monitored WebSphere MQ applications are connecting to running in a VMware environment?	Select **Yes** if the queue manager is running in a VMware environment. This activates the Time Server options, which configures the agents to use the TransactionVision TimeServer to acquire the time skew instead of the default WebSphere MQ Confirmation-On-Arrival method. The use of the TransactionVision TimeServer provides a more accurate time skew when running in a VMware environment. Select **No** if the queue manager is not running in a VMware environment. **Default value:** No

UI Element	Description
Is event queue in a cluster?	Select **Yes** if the queue is in a cluster. If it is not a local queue, select **No** and specify either the name of the transmission queue that has the same name as the remote queue manager or a queue manager alias that resolves to the remote queue manager. This method of sending messages is described in the *WebSphere MQ Application Programming Guide*. **Default value:** No **Should Agent use the above Queue as Transmission Queue/Queue Manager alias to open event queue?** Select **No** to enter a different transmission queue or queue manager alias name below. **Transmission Queue/Queue Manager alias name.** The transmission queue or queue manager alias name.

TIBCO EMS

UI Element	Description
EMS Server hostname and port the Agent will connect to:	The EMS Server host and port to which the agent connects. Choose the protocol: **tcp://**. For a communication link between an agent and the Analyzer that is unsecured. **ssl://**. For a communication link between an agent and the Analyzer that is to be secured by SSL. SSL must be explicitly enabled on other components to provide a full SSL deployment. See "Security" in the *HP TransactionVision Deployment Guide* PDF.
User Name:	The user name the agent uses to connect to the EMS Server.
Password:	The password associated with the user name above.

UI Element	Description
Queue the Agent will receive configuration messages from:	The name of the queue that agents check for configuration messages.
Queue the Agent will send event messages to:	The name of the queue to which agents send event messages. This queue should only be used by TransactionVision.

Progress SonicMQ

UI Element	Description
Broker hostname and port the Agent will connect to:	The broker host and port to which the agent connects. Choose the protocol: **ttcp://**. For a communication link between an agent and the Analyzer that is unsecured. **ssl://**. For a communication link between an agent and the Analyzer that is to be secured by SSL. SSL must be explicitly enabled on other components to provide a full SSL deployment. See "Security" in the *HP TransactionVision Deployment Guide* PDF.
User Name:	The user name the agent uses to connect to the broker.
Password:	The password associated with the user name above.
Queue the Agent will receive configuration messages from:	The name of the queue that agents check for configuration messages. **Default value:** TVISION.CONFIGURATION.QUEUE
Queue the Agent will send event messages to:	The name of the queue to which agents send event messages. This queue should only be used by TransactionVision. **Default value:** TVISION.EVENT.QUEUE

WebLogic

UI Element	Description
Broker hostname and port the Agent will connect to:	The broker host and port to which the agent connects. Choose the protocol: **tcp://**. For a communication link between an agent and the Analyzer that is unsecured. **ssl://**. For a communication link between an agent and the Analyzer that is to be secured by SSL. SSL must be explicitly enabled on other components to provide a full SSL deployment. See "Security" in the *HP TransactionVision Deployment Guide* PDF. **Default value:** 7001
Queue Connection Factory JNDI Name:	The JNDI name of the queue connection factory.
User Name:	The user name the agent uses to connect to the broker.
Password:	The password associated with the user name above.
Queue the Agent will receive configuration messages from:	The name of the queue that agents check for configuration messages. **Default value:** TVISION.CONFIGURATION.QUEUE
Queue the Agent will send event messages to:	The name of the queue to which agents send event messages. This queue should only be used by TransactionVision. **Default value:** TVISION.EVENT.QUEUE

Other JMS Providers

UI Element	Description
JNDI Provider Context Factory:	The initial context factory of the JMS provider.
JNDI Provider URL Agent will connect to:	The URL that the agent uses to connect to the JNDI naming service.

UI Element	Description
Queue Connection Factory JNDI Name:	The JNDI name of the queue connection factory.
User Name:	The user name the agent uses to connect to the JNDI provider.
Password:	The password associated with the user name above.
Queue the Agent will receive configuration messages from:	The name of the queue that agents check for configuration messages. **Default value:** TVISION.CONFIGURATION.QUEUE
Queue the Agent will send event messages to:	The name of the queue to which agents send event messages. This queue should only be used by TransactionVision. **Default value:** TVISION.EVENT.QUEUE

Analyzer Connection Page

This wizard page enables you to specify which queues the Analyzer connects to.

Important information	General information about this wizard is available here: "Communication Link Wizard" on page 182
Wizard map	The Communication Link Wizard contains: General Page > Agent Connection Page > **Analyzer Connection Page** > Queue Settings Page > Connection Settings Page > Domain Manager Page> Event Options Page > Time Server Page > Miscellaneous Information Page
See also	➤ "Message Queues in Communication Links" on page 157 ➤ "Message Queue Configurations" on page 160 ➤ "Message Queue Requirements" on page 162

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
Should the Analyzer connect to the same queue manager and queues used by the Agent?	Select **Yes** if the Analyzer should connect to the same queues that the Agent connects to. This requires that the Analyzer is running on the same host on which the queue manager, SonicMQ Broker, JMS Server or TIBCO is installed. Select **No** if the Analyzer should connect to different queues than the agent connects to. You will specify the queue details on the next page of the wizard.

Queue Settings Page

This wizard page enables you to specify the queue information for the Analyzer. This page appears only if No was specified on the previous page (Analyzer Connection page).

Important information	General information about this wizard is available here: "Communication Link Wizard" on page 182
Wizard map	The Communication Link Wizard contains: General Page > Agent Connection Page > Analyzer Connection Page > **Queue Settings Page** > Connection Settings Page > Domain Manager Page> Event Options Page > Time Server Page > Miscellaneous Information Page
See also	➤ "Message Queues in Communication Links" on page 157 ➤ "Message Queue Configurations" on page 160 ➤ "Message Queue Requirements" on page 162

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
Is the Analyzer going to use the same queue manager for configuration and event messages?	Select **Yes** if the Analyzer should use the same queue manager for both configuration and event messages. Select **No** if the Analyzer should use different queue managers for configuration and event messages.
Queue Manager to which the Analyzer will connect for sending configuration messages	The name of the queue manager the Analyzer uses.
Broker host name and port Analyzer will use for sending configuration messages	(Tibco EMS only) The name and port of the EMS Broker the Analyzer uses. Specify the credentials for the Broker.
Queue the Analyzer will send configuration messages to	The name of the queue that the Analyzer should send configuration messages to. This queue must be a remote queue for the configuration queue on the agent queue manager. **Default value:** TVISION.CONFIGURATION.QUEUE
Queue Manager to which the Analyzer will connect for receiving event messages	The name of the transmission queue or queue manager alias. This option is used only if **No** is selected above.
Broker host name and port Analyzer will use for receiving configuration messages	(Tibco EMS only) The name and port of the EMS Broker the Analyzer uses. Specify the credentials for the Broker.
Queue the Analyzer will receive event messages from:	The name of the queue the Analyzer checks for event messages. **Default value:** TVISION.EVENT.QUEUE

Connection Settings Page

This wizard page enables you to specify the connection information for the Analyzer.

Important information	General information about this wizard is available here: "Communication Link Wizard" on page 182
Wizard map	The Communication Link Wizard contains: General Page > Agent Connection Page > Analyzer Connection Page > Queue Settings Page > **Connection Settings Page** > Domain Manager Page> Event Options Page > Time Server Page > Miscellaneous Information Page
See also	➤ "Message Queues in Communication Links" on page 157 ➤ "Message Queue Configurations" on page 160 ➤ "Message Queue Requirements" on page 162

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
Will the Analyzer use client connection mode to communicate with the Queue Manager?	**Yes** if the Analyzer will connect in client mode. **No** if the Analyzer will connect in server mode. Specify specific options below. **Default value:** No
Channel for <configuration queue manager>	The channel for the configuration queue manager.
Host for <configuration queue manager>	The host name or IP address for the configuration queue manager.
Port for <configuration queue manager>	The port for the configuration queue manager.
CCSID for <configuration queue manager>	Choose the CCSID for the event queue manager. Required only if the queue manager uses double-byte.
Channel for <event queue manager>	The channel for the event queue manager.
Host for <event queue manager>	The host name or IP address for the event queue manager.
Port for <event queue manager>	The port for the event queue manager.
CCSID for <event queue manager>	Choose the CCSID for the event queue manager. Required only if the queue manager uses double-byte.

Domain Manager Page

This wizard page enables you to specify SonicMQ domain information if you are not using the default domain.

Important information	General information about this wizard is available here: "Communication Link Wizard" on page 182
Wizard map	The Communication Link Wizard contains: General Page > Agent Connection Page > Analyzer Connection Page > Queue Settings Page > Connection Settings Page > **Domain Manager Page**> Event Options Page > Time Server Page > Miscellaneous Information Page
See also	➤ "Message Queues in Communication Links" on page 157 ➤ "Message Queue Configurations" on page 160 ➤ "Message Queue Requirements" on page 162

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
Domain Name:	The SonicMQ Domain Name to which the brokers belong. **Default value:** Domain1.
Domain Manager hostname and port Analyzer will connect to	The host and port of the SonicMQ Domain Manager listener to which the Analyzer connects. This allows you to query for the event queue depth. Specify in the form of hostname:port. **Default value(port):** 21111
User Name:	A user name to connect to the SonicMQ Domain Manager. **Default value:** none
Password:	The password associated with the user name above. **Default value:** none
Name of broker with event queue to be queried via the Domain Manager:	The name of the broker to which the Analyzer connects to collect events. This name is used to query the event queue depth through the Domain Manager. If not specified, the unprocessed event count that normally appears on the Analyzer Status page is not available.

Event Options Page

This wizard page enables you to configure the event queue maximum size and the behavior to follow when an event message cannot be delivered.

Important information	General information about this wizard is available here: "Communication Link Wizard" on page 182
Wizard map	The Communication Link Wizard contains: General Page > Agent Connection Page > Analyzer Connection Page > Queue Settings Page > Connection Settings Page > Domain Manager Page> **Event Options Page** > Time Server Page > Miscellaneous Information Page
See also	➤ "Message Queues in Communication Links" on page 157 ➤ "Message Queue Configurations" on page 160 ➤ "Message Queue Requirements" on page 162

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
Maximum Message Size	**Event Queue Maximum Message Length (10000 - 2147483647)**. The maximum event message length in bytes. A minimum message length of 10,000 bytes is required; to calculate the size needed for the monitored application see "Message Queue Requirements" on page 162. **Default value**: 4194304
Event Retry on Failure	(Does not apply to WebSphere Agents) If the agent cannot connect to the Queue Manager, EMS Server, JMS Server or SonicMQ broker, specifies how often reconnection is attempted: **Retry forever every <interval> at every Transaction Management event.** Continues to retry at the specified interval as long as event collection is in progress. Note that this option can cause every application API call to slow down in the event of a connection failure, reducing application performance significantly. This option is recommended for an audit environment where all events must be logged. **Retry every <interval> across Transaction Management events**. Continues to retry at the specified interval as long as event collection is in progress. However, the reconnection time delay is not incurred for each API call, but at a given frequency. **Retry once per Transaction Management event**. Retries once for each event. This option reduces the impact on application performance if a connection is lost, but Transaction Management events are lost if the retry attempt is not successful. **Retry every <interval> for <duration> at every TransactionVision event.** Continues to retry at the specified interval until the timeout is reached. **Default value**: Retry every 10 seconds across Transaction Management events.

UI Element	Description
Enable retry event upon failure	Retry if the agent fails to put an event on the event queue. Not recommended for a production environment. The Event Delivery Retry on Failure elements are enabled if this option is specified: **Retry timeout.** The number of seconds, minutes, or hours to continue to retry. **Default**: retry forever. **Retry interval.** The number of milliseconds or seconds that the Agent waits between failed retry attempts. **Default value:** retry forever every 10 seconds.

Time Server Page

This wizard page enables you to specify the time server to calculate the time skew information for the Analyzer and agents.

Important information	General information about this wizard is available here: "Communication Link Wizard" on page 182
Wizard map	The Communication Link Wizard contains: General Page > Agent Connection Page > Analyzer Connection Page > Queue Settings Page > Connection Settings Page > Domain Manager Page> Event Options Page > **Time Server Page** > Miscellaneous Information Page
See also	"Key Configuration Concepts for Analyzers" on page 85

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
Use the Time Server Running in the Processing Server	Agent uses the time server within the Processing Server hosting the Analyzer to calculate all time skews. Otherwise, specify the name of the host running the Time Server and its port number in Time Server Host and Time Server Port. **Default value**: Checked
Time Server Host:	The host name on which the time server is running. This field can also be used to specify an alternate IP address for the time server running in the Processing Server. Agents running in NAT environments typically require a NAT IP to connect to the time server.
Time Server Port:	The port number for the time server.

Miscellaneous Information Page

This wizard page enables you to specify miscellaneous information for the Analyzer and agents.

Important information	General information about this wizard is available here: "Communication Link Wizard" on page 182
Wizard map	The Communication Link Wizard contains: General Page > Agent Connection Page > Analyzer Connection Page > Queue Settings Page > Connection Settings Page > Domain Manager Page> Event Options Page > Time Server Page > **Miscellaneous Information Page**
See also	"Message Queue Requirements" on page 162

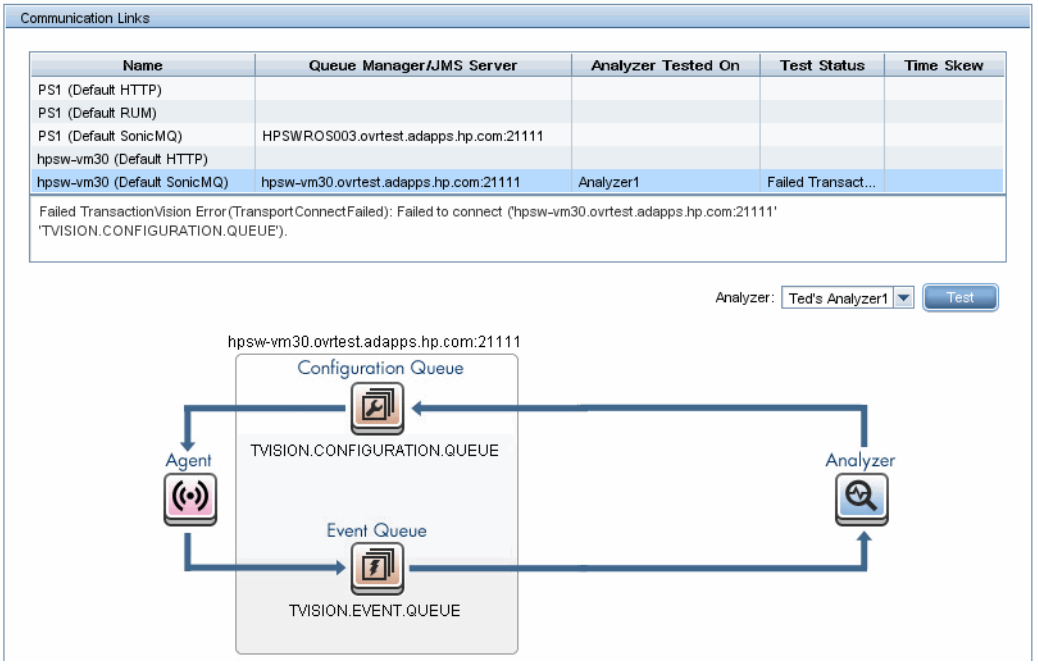
User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
Exception message queue name	The name of the exception message queue. This queue is always managed by the same queue manager, broker, or Server that manages the event queue. **Default value**: TVISION.EXCEPTION.QUEUE.
Number of event collection threads (1-100)	The number of event collection threads, from 1 to 100. This value should match the results from the DBMS insert test: DB2Test, OracleTest, SQLServerTest. For information about these utilities, see the *HP TransactionVision Deployment Guide* PDF. **Default value:** 1
Enable Agent trace logging	Enables verbose logging information of TransactionVision agent operation. This option is intended for troubleshooting and should not be turned on in production environments. For information about the agent log file locations, see the *HP TransactionVision Deployment Guide* PDF. **Default**: Off
Enable event persistence	Enables event persistence. **Default**: Off Event persistence guarantees that an event message is physically stored on a queue. For instance, the event messages are no lost if the queue manager/broker/server is restarted. Event persistence is a feature of the messaging middleware provider.
Enable support for processing user defined events	Enables user-defined events processing. User events are events created by user applications beyond those originating from the standard TransactionVision agents. For information about implementing user events in applications, see the *HP TransactionVision Advanced Customization Guide* PDF. **Default value**: Off. The communication link is configured to process standard agent events only.

UI Element	Description
Store user data:	Causes the Analyzer to store the user data portion of the event message in the database and the data is therefore available in the Event Detail report. Storing user data increases the Analyzer size requirements and may degrade Analyzer performance. **Default value**: On
Store event document:	Causes the Analyzer to store the event XML data in the database and the data is therefore available in the Event Detail page. Storing event XML data increases the Analyzer size requirements and may degrade Analyzer performance. If set to Off, the Analyzer is not able to recover event analysis data if the database. **Default value**: On
Store raw events:	Causes the Analyzer to store a copy of the raw events in the database. Raw events are those pulled directly from the event queue and not yet processed by the Analyzer. This option is intended for troubleshooting and should not be turned on in production environments. For information about using raw events, see "Storing and Using Raw Events" on page 146. **Default value**: Off

Communication Links Summary Page

This page enables you to view the status of all communication links in the deployment environment. The following is an example of the Communication Links Summary page.



To access	**1** Select **Admin** > **Transaction Management.** **2** (left pane) Click the **Configuration** tab. **3** Select **Communication Links.**
Important information	The diagram represents the queue names and configuration for the selected communication link.
See also	➤ "How to Test a Communication Link" on page 175 ➤ "How to Create a Communication Link" on page 174 ➤ "How to Remove a Communication Link" on page 180

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
Name	The communication link name.
Queue Manager/JMS Server	The queue manager or JMS server that the Agent connects to.
Analyzer Tested On	The time at which the communication link was last tested.
Time Skew	At the time the communication link was last tested, the time skew between the Analyzer host and the host on which the configuration queue's queue manager/JMS server/SonicMQ broker host is running.
Test Status	Test results. If the communication link test fails, the error message should indicate which queue, queue manager or event broker failed — make sure the specified queue, queue manager or event broker has been created.
Analyzer	The Analyzer with which to test the communication link.
	Tests the selected communication link.

Communication Link Page

This page enables you to edit the communication link settings.

To access	**1** Select **Admin** > **Transaction Management.** **2** (left pane) Click the **Configuration** tab. **3** Expand the **Communication Links** folder. **4** Select <communication link>.
Important information	Depending on your changes, the following message may be displayed: This significant change to the communication link could leave unprocessed events intended for the previous communication link definition on the event queue. It is recommended to only make this change while the event queue is empty. Proceed with this change? [Yes] [No] Then click **Yes** to confirm your changes. Due to event packaging, the agent may have events buffered but not yet sent to the event queue. To force buffered events to be sent to the event queue restart the Analyzer.
See also	➤ "Communication Links Overview" on page 156 ➤ "Message Queues in Communication Links" on page 157 ➤ "Message Queue Configurations" on page 160 ➤ "Message Queue Requirements" on page 162

General Settings Area

The settings in this area are described as part of the Communication Link Wizard, where they are originally set. See "General Page" on page 183, "Time Server Page" on page 198 and "Miscellaneous Information Page" on page 199.

<event transport provider> Connection Area

The settings in this area are described as part of the Communication Link Wizard, where they are originally set. See "Connection Settings Page" on page 192.

Event Retry on Failure Area

The settings in this area are described as part of the Communication Link Wizard, where they are originally set. See "Event Options Page" on page 196.

Troubleshooting and Limitations

If the communication link does not appear to be collecting the expected events, there may be a problem with the communication link.

To diagnose the problem, perform the following steps:

1. Ensure that the communication link is assigned to at least one Analyzer and the results of testing the communication link are successful.
2. Ensure that data collection filters are not filtering out the events you expect to see.
3. If you are viewing the collected events based on a query, ensure that the query is getting the right event data from the database.
4. Ensure that an agent is intercepting WebSphere MQ API calls from the monitored application. On UNIX and Windows platforms, set the **TVISION_BANNER** environment variable, then start the application. A banner indicating that the application is loading the agent should appear. To disable this behavior, unset **TVISION_BANNER**. This environment variable can be set to any value. On Windows, it must be set to a value other than an empty string. On i5/OS, **TVISION_BANNER** does not display the library path as it does on UNIX.

   If not, set the environment variable for your platform to cause the application to load the agent library instead of the standard WebSphere MQ library.

The following table shows the appropriate environment variables:

Platform	Environment Variable
Windows	PATH
Sun Solaris	LD_LIBRARY_PATH
HP-UX	SHLIB_PATH
Linux	LD_LIBRARY_PATH
IBM AIX	LIBPATH

On z/OS, the CICS region prints a message when it connects to the queue manager or event broker, indicating that the WebSphere MQ crossing exit was found and used. If the WebSphere MQ crossing exit was not found or loaded, a message indicating the error appears.

5 If the agent library is being loaded by the application, check the monitored application host's system log for any error messages.

6 If there are no error messages in the system log, enable trace logging for the active communication links.

Look at the trace messages in the application host's UNIX system log, Windows event log, z/OS operator console log, or i5/OS user's job log for any WebSphere MQ-related errors or messages that indicate that the data collection criteria such as collection time or collection entities were too restrictive.

On UNIX platforms, you can specify the log facility by setting the **TVISION_SYSLOG** environment variable to one of the following values: **user**, **local0**, **local1**, **local2**, **local3**, **local4**, **local5**, **local6**, or **local7**. If **TVISION_SYSLOG** is not set or is set to a value other than those listed, TransactionVision uses **local0**. The target log file must already exist.

To view the job log on i5/OS, use the **DSPJOBLOG** command.

5

Data Collection Filters

This chapter includes:

Concepts

- Data Collection Filters Overview on page 208
- Categories of Filter Criteria on page 209
- About Building the Data Collection Filter on page 210
- Guidelines for Good Data Collection Filtering on page 211

Tasks

- How to Create a Data Collection Filter on page 213
- How to Assign a Data Collection Filter to a Communication Link on page 215
- How to Modify a Data Collection Filter on page 215
- How to Remove a Data Collection Filter on page 216

Reference

- Data Collection Filters User Interface on page 218

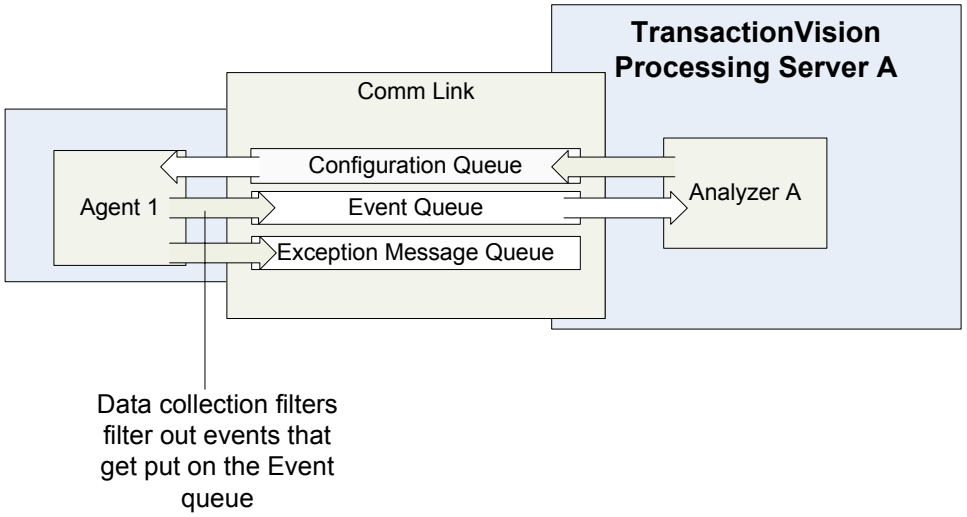
Concepts

Data Collection Filters Overview

Collecting all possible event information about all API calls can cause the TransactionVision database to grow quite large. It also adds to your communication link overhead — the more events that are collected, the more event messages are sent back to the Analyzer for processing.

You typically do not need to collect all information about every event that occurs in the monitored application to successfully obtain business transaction information. For example, you may only need to collect events that occur on a specific host, events for a specific API, events with a specific MQI reason code, or events with a specific value in the user data. Additionally, you may require only a subset of the user data for each event.

Data collection filters limit the number of events and the amount of data for each event that agents collect and forward to the Analyzer. The Analyzer communicates data collection filter criteria to agents through configuration messages. The agent only collects events that match the filter criteria specified in the configuration message:



A data collection filter is assigned to a particular communication link. A communication link can have more than one data collection filter assigned to it, and a specific data collection filter can be assigned to more than one communication links.

Categories of Filter Criteria

A data collection filter has many criteria that can be specified as part of the filter. The criteria are grouped into the following categories:

- Common Options

  Criteria that can be specified for any type of agent such as the hose on which the event occurred, or the program name that generated the event.

- Technology-specific options

  CICS: criteria for the CICS Agent
  EJB: criteria for EJB-related messages of the Java Agent
  JDBC: criteria for JDBC-related messages of the Java Agent
  JMS: criteria for JMS-related messages of the Java Agent
  Servlet: criteria for servlet-related messages of the Java Agent
  WebSphere MQ: criteria for WebSphere MQ Agent
  User Event: criteria for the Tuxedo, .NET, and NonStop TMF Agents

All filter criteria in the Common Options category can be overridden in the technology-specific category. For example, you can specify **All Hosts** as a common option that applies to all technologies. Then for the JMS specific technology, you can specify host **Host1**. Then only JMS events originating on **Host1** match the filter criteria.

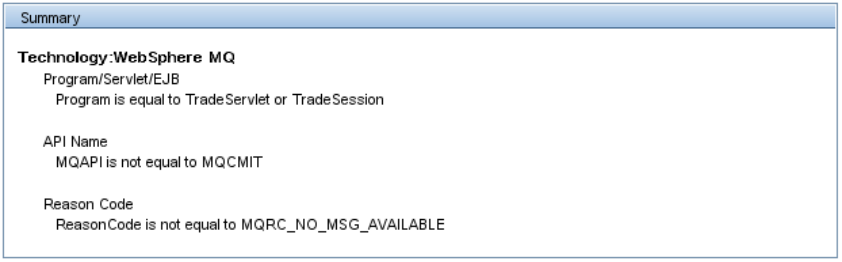
About Building the Data Collection Filter

When creating a data collection filter, you typically supply multiple criteria. The criteria work together as follows:

- Selections within the same criteria and data collection filter are ORed together.
- Include/Exclude is applied to the overall criteria. For example: !(a OR b OR c)). An event only meets filter criteria if it matches all criteria specified for the filter. The AND operation is used within each filter.
- Different criteria selections within the same data collection filter are ANDed together.
- You may need to define more than one filter to meet your specific requirements.
- Multiple Data Collection Filters are ORed. For example: (Filter1) OR (Filter2)). The agent collects events that meet the criteria for any assigned filter. The OR operation is used between filters.

For example, suppose you want to collect all events from applications TradeServlet and TradeSession, except MQCMIT events. You also don't want to collect any event with the MQRC_NO_MSQ_AVAILABLE reason code.

You would use a single data collection filter with the three criterion settings. The Summary pane in the Data Collection Filter page displays the filter as follows:



Only events that meet all three criteria are collected; the criteria are ANDed.

If you have another application on the monitored system for which you do want to collect MQCMIT events, you would create a second data collection filter for this and apply both data collection filters to the communication link.

The Default Filter

If no data collection filter is specified, then the default filter is used. This filter can be customized but has the following initial settings:

- For WebSphere MQ:
  - Only MQGET, MQPUT, and MQPUT1 API names are included.
  - Do not send browsing MQGET is set.
- For JMS: send, receive, receiveNoWait, publish, and onMessage methods are included.
- For Servlet: only the first 1024 bytes of HTTP request and response data are included.
- The JDBC technology is off.
- For all technologies: event packaging is enabled and the number of events to package is set to 10 events.

Guidelines for Good Data Collection Filtering

The primary goal of a data collection filter is to avoid unneeded events. Some common examples of things to exclude:

- Programs and/or Queues that are not part of “in scope” transactions (customer specific)
- Steps that are not of any interest (and aren’t needed for correlation)
- MQ APIs others than MQGET, MQPUT, MQPUT1
- amqrmppa channel program
- MQ explorer or MQ monitoring tools

The secondary goal of the data collection filter is to limit the data collected per event. This is typically done by specifying the Data Range criterion to avoid unlimited data ranges when necessary. Typically, the following guidelines are good starting points:

- Limit MQ message sizes collected to the first 1-10K
- Limit HTTP request/response data collected to the first 1-5K

Tasks

How to Create a Data Collection Filter

This task describes how to create a new data collection filter.

1 Prerequisites

Determine the technology that applies to the application being monitored. Then, determine which events you want to include or exclude.

Review the list of technologies on "Categories of Filter Criteria" on page 209. Review the list of event categories on "Data Collection Filter Page, Filter Criteria Tab" on page 221.

2 Select **Admin** > **Transaction Management**.

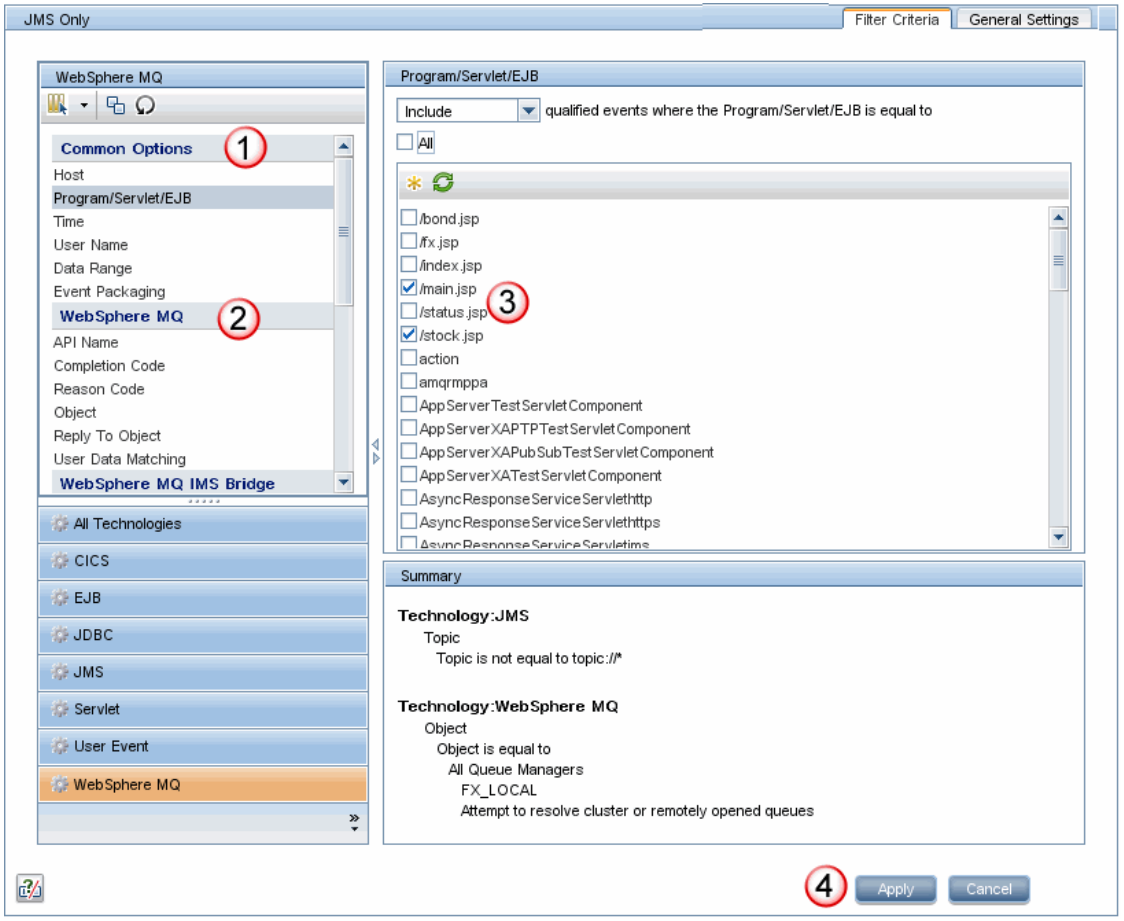
3 (left pane) Click the **Configuration** tab.

4 Right-click **Data Collection Filters** select **New Data Collection Filter**.

The New Data Collection Filter dialog appears.

5 Enter a name and click **OK**.

6 (right pane) Use the **Filter Criteria** tab to manage and set the criteria for the data collection filter. On this tab, you perform these general steps:



The following table describes the locations in this page:

Callout	Description
1	Choose the category of criteria that you want to set.
2	Choose the criterion that you want to set.
3	Set the criteria.
4	Review the settings in the Summary panel and then click Apply.

For more details about this interface, see "Data Collection Filter Page, Filter Criteria Tab" on page 221.

How to Assign a Data Collection Filter to a Communication Link

You need to assign the data collection filter to a communication link before it can have any effect. See "How to Assign a Data Collection Filter to a Communication Link" on page 177.

Note: Multiple data collection filters are not supported by the Tuxedo Agent. The Tuxedo Agent applies the first data collection filter it finds that matches the Technology and Program/Servlet/EJB criteria. Therefore you should enable only one data collection filter for this type of agent on each Analyzer.

How to Modify a Data Collection Filter

This task describes how to modify a data collection filter.

1 Prerequisites

Determine which events you want to include or exclude.

2 Select **Admin** > **Transaction Management**.

3 (left pane) Click the **Configuration** tab.

4 Select **Data Collection Filters** > <data collection filter>.

5 (right pane) Use the **Filter Criteria** page to change the criteria for the data collection filter. For details about this interface, see "Data Collection Filter Page, Filter Criteria Tab" on page 221.

After you click Apply, the filter becomes effective. Any associated agents are notified about the filter change through a new configuration message. For more information about configuration messages, see "Message Queues in Communication Links" on page 157.

6 Use the **General Settings** tab to rename the data collection filter if desired. For details about this interface, see "Data Collection Filter Page, General Settings Tab" on page 257.

Note: Renaming a data collection filter unassigns it from any communication links that it was assigned to before renaming.

How to Remove a Data Collection Filter

This task describes how to remove a data collection filter.

1 Prerequisites

Make sure no communication links are using the data collection filter.

To see the communication link and filter assignments for an Analyzer:

a Select **Admin** > **Transaction Management** > **Configuration** tab in the left pane.

b Select the Analyzer under **Processing Servers** then in the right pane, select the **Configuration** tab.

For user interface details, see "Analyzer Page, Configuration Tab" on page 124.

2 Select **Admin** > **Transaction Management**.

3 Click the **Configuration** tab in the left pane.

4 Select **Data Collection Filters** > <data collection filter>.

5 Right-click and select **Delete**.

Reference

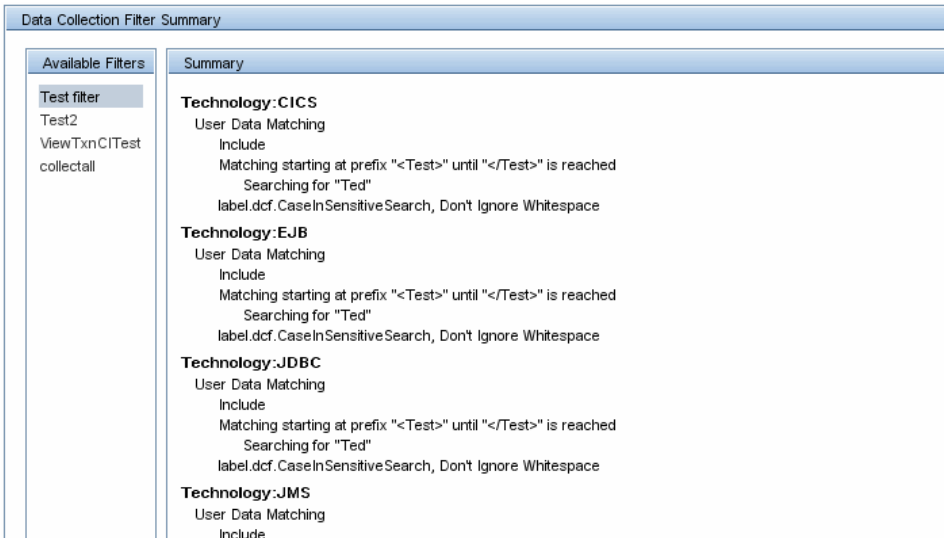
Data Collection Filters User Interface

This section describes:

- Data Collection Filters Summary Page on page 219
- Data Collection Filter Page, Filter Criteria Tab on page 221
- Data Collection Filter Page, General Settings Tab on page 257

Data Collection Filters Summary Page

This page enables you to view all data collection filters and the technologies to which a selected filter apply.



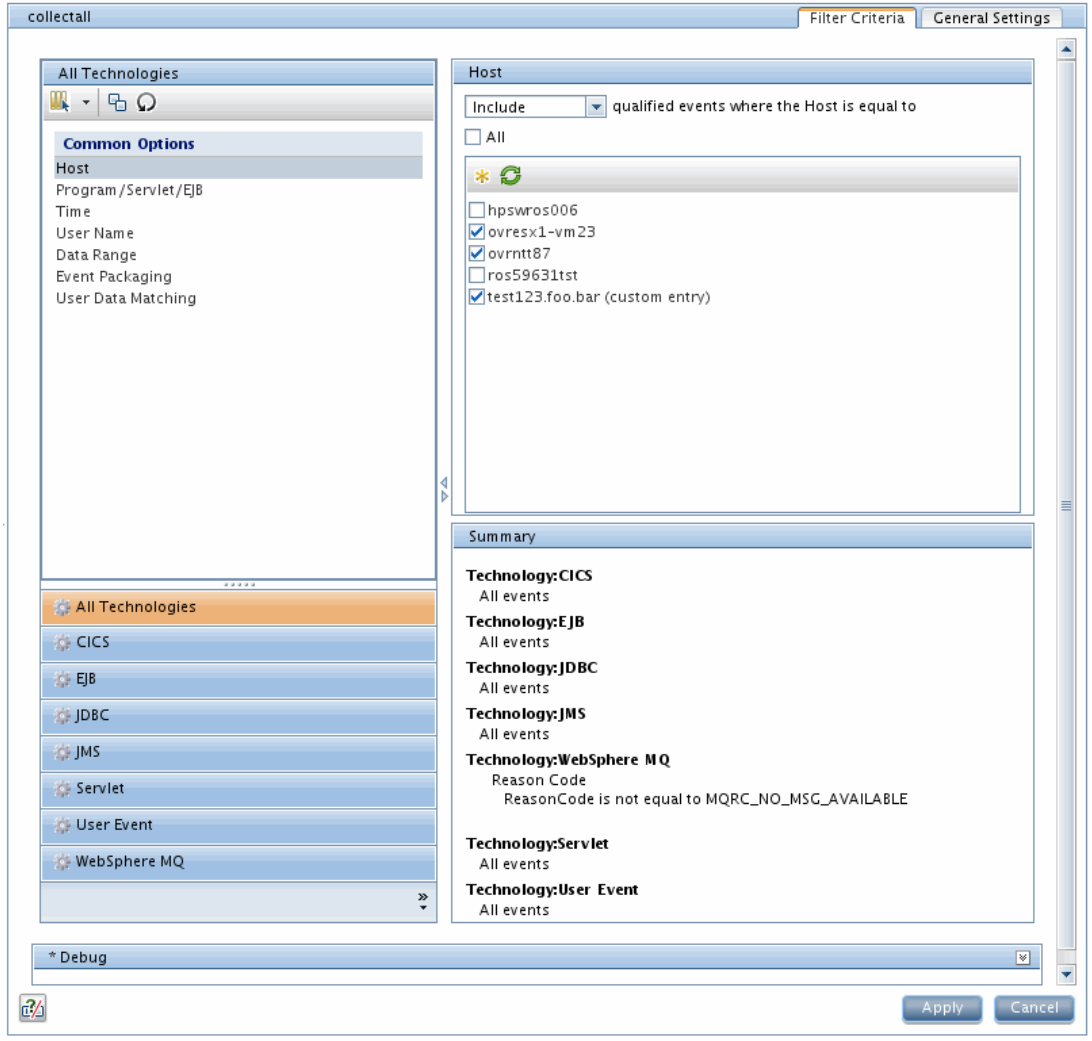
To access	**1** Select **Admin** > **Transaction Management.** **2** (left pane) Click the **Configuration** tab. **3** Select **Data Collection Filters**.
Important information	You can click on an entry in the Summary pane to go to the criterion setting area for that data collection filter.
Relevant tasks	➤ "How to Create a Data Collection Filter" on page 213 ➤ "How to Modify a Data Collection Filter" on page 215

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element (A-Z)	Description
Available Filters	The data collection filters currently defined in the deployment environment.
Summary	For a selected data collection filter, the non-default settings are shown.

Data Collection Filter Page, Filter Criteria Tab

This page enables you to specify data collection filter criteria.



To access	For an existing data collection filter: **1** Select **Admin** > **Transaction Management.** **2** (left pane) Click the **Configuration** tab. **3** Select **Data Collection Filters** > <filter name>. **4** (right pane) Select the **Filter Criteria** tab.
Important information	➤ View the Summary pane in the lower right-hand side of the page to see the combined effect of all criteria set for the filter. The **Apply** button updates the Summary pane. ➤ Any Common Options settings specified at the All Technologies level are inherited, and can be overridden at a specific technology level. ➤ For the WebSphere MQ and CICS Agents, a blank field is treated as a match when a data collection filter criteria is specified. For example, if you create a filter that collects events with the user name J_SMITH, events with an empty user name are also collected by the WebSphere MQ and CICS Agents. ➤ User-created entries for a filter are specific to that filter. For instance, if you add a new host named "Server1" to a particular data collection filter, that entry does not appear in the Host list of any other data collection filter.
Relevant tasks	➤ "How to Create a Data Collection Filter" on page 213 ➤ "How to Modify a Data Collection Filter" on page 215

Common Options Criteria

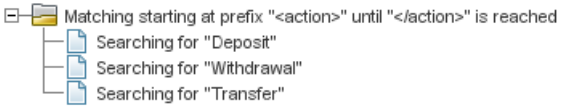
User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
User Data Range	(WebSphere MQ Agents only.) Include qualified events in the filter criteria. For CICS and WebSphere MQ technologies, limit the data by choosing one of the following from the drop-down list: ➤ **API Names + Call Arguments + User Buffer.** All possible event details are returned. ➤ **API Names + Call Arguments.** The user buffer of the event is not returned. ➤ **API Names Only.** The user buffer and call arguments of the event are not returned. Queries against the event data cannot include API parameter values. For EJB, JDBC, JMS, Servlet, User Event technologies, limit the data by choosing one of the following from the drop-down list: ➤ **Collect User Data.** User buffer data is included; you can select a range to specify how many bytes of the user data buffer to collect. Check the box next to All Data (no data ranges applied) to collect the entire user data buffer. To manually add a new range, click and then enter the starting byte number (0 based index) in the Start Value field and the ending byte number in the End Value field. Specify a * in the End Value field to collect to the end of the user data buffer. Agents collect data in all selected data range definitions, so remove any data ranges you do not want to collect. To remove an existing range, clear the data range check box. **Example.** If you know that the event data you are interested in is at the beginning of the event payload, for example in a meta data header, you can collect the first 1KB of the payload by specify a Start Value of 0 and an End Value of 1023. ➤ **Do not Collect User Data.** All buffer data is included. For the Tuxedo, NonStop TMF, JMS, Servlet, EJB, and JMS Agents this criterion is not evaluated.

UI Element	Description
Event Packaging	**Use Event Packaging.** Enables agents to buffer events until the event packaging criteria (below) is met or other conditions within the agent force it to flush the package. At that point, the packaged events are sent to the Event queue. Event Packaging directly improves the performance of the agent. However, if the monitored application crashes, event packaging may result in some collected events not being sent to the Analyzer. Specify the following options if Event Packaging is enabled: **Maximum Package Size.** The limit on the amount of memory an agent can set aside for storing events to be packaged. When the agent reaches or exceeds this limit while packaging events together, it sends the package to the event queue. Leave this field blank or specify a value of zero to use the **Number of Events for the agent to package together** field to determine when the package is sent. Note that if the Maximum Package Size is smaller than the event size, each event is sent immediately. In this case, Event Packaging is effectively turned off. The minimum event size is approximately 4000 bytes. **Number of Events for the agent to package together.** The number of events stored in the agent before sending a package to the Event Queue. As soon as the agent reaches this number of events while packaging events together, it sends the package to the Event queue. The number of events should be set to a value suitable to the environment, usually 10 events per message. Leave this field blank or specify a value of zero to use the Maximum Package Size field to determine when the package is sent. If you specify a value for both the number of events and the maximum package size, agents combine messages into packages based on which limit is reached first. If both are zero or blank, the default behavior of no event packaging is used. For the NonStop TMF Agent this criterion is not evaluated.

UI Element	Description
Host	**Include** or **Exclude** selected hosts in the filter criteria. The host list is generated from events collected and stored in the database; you can add custom entries to the list. To include all hosts in the filter criteria, check **All**. To include a specific host in the filter criteria, select the host name check box. To remove a host name from the filter criteria, clear the host name check box. To manually add a host to the list, click and then enter the host name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of host names, click .
Program/Servlet/EJB	**Include** or **Exclude** selected programs, servlets, and/or EJBs in the filter criteria. The program list is generated from events collected and stored in the database. To include all programs, servlets, and EJBs in the filter criteria, check **All**. To include a specific program, servlet, or EJB in the filter criteria, select the program, servlet, or EJB name check box. To remove a program, servlet, or EJB name from the filter criteria, clear the program, servlet, or EJB name check box. To manually add a program, servlet, or EJB to the list, click and then enter the program, servlet, or EJB name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of program, servlet, or EJB names, click . For the Tuxedo Agent, the program, servlet, or EJB name must be the executable process that made the Tuxedo call. This can be done by either including the name explicitly or by specify **All**. For the NonStop TMF Agent, this criterion is not evaluated.

UI Element	Description
Time	Specify the time mode: **Any Time.** Events are collected with no time restrictions. **From a set start to end time.** Specify a time range (based on the clock of the host running the agent) to collect events. To select the date from a calendar, click the drop-down button in the field and then click the desired date. The events are collected from the specified start time until the specified end time. **From a set start time.** Specify a time range (based on the clock of the host running the Agent) to collect events. To select the date from a calendar, click the drop-down button in the field and then click the desired date. The events are collected from a specific start time until event collection is stopped. Even if you specify a start time, you must initialize event recording before that start time. Initializing event recording causes the Analyzer to send a configuration message with filter information to all assigned communication links. For the .NET Agent, Tuxedo Agent, NonStop TMF Agent, and Java User Event technology this criterion is not evaluated.
User Name	**Include** or **Exclude** selected users in the filter criteria. The user name list is generated from events collected and stored in the database; you can add custom entries to the list. To include all user names in the filter criteria, check **All**. To include a specific user name in the filter criteria, select the user name check box. To remove a user name from the filter criteria, clear the user name check box. To manually add a user name to the list, click and then enter the user name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of user names, click . For the .NET Agent, Tuxedo Agent, and NonStop TMF Agent, and Java User Event technology this criterion is not evaluated.

UI Element	Description
User Data Matching	**Include** or **Exclude** selected strings of data in the filter criteria. To add a new string to the list, click and then enter the prefix characters, suffix characters, and the matching value. An asterisk can be used at the beginning or the end as a wildcard. The suffix is optional. If it is not specified, the end of the user data buffer acts as the end of the string. Text within binary user data can be searched as null characters are ignored. A prefix/suffix pair can be applied to multiple matching values. Right click an existing entry and select **Add New Item** to build the list:  Check **Ignore white space** to ignore any white-space characters in the prefix, suffix, and matching value. Check **Case sensitive search** to consider case in the matching value. **Examples.** Assume the user data payload contains account and transactional information as follows: <BankTransaction> <Name>Stewie Gatesman</Name> <Account>123456789</Account> <Action>Deposit</Action> <Amount>1000000.00</Amount> </BankTransaction> Perhaps the user is classifying transactions based on the Action and they want to route all Deposits to a dedicated Analyzer. The following User Data Matching criteria for the XML element value could be specified: Prefix: <Action> Suffix: </Action> Matching String(s): Deposit To match on a particular element tag, regardless of value, the following criteria could be specified: Prefix: <Deposit> Suffix: Matching String(s): * For the CICS, .NET Agent, Tuxedo Agent, NonStop TMF Agent, and Java User Event technology this criterion is not evaluated.

CICS Agent Criteria

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element (A-Z)	Description
API Name	**Include** or **Exclude** events for selected APIs. To include all APIs in the filter criteria, check **All**. To specifically include an API, select the API name check box. The APIs are grouped by API type. To remove an API from the list, clear the API check box.
File Name	**Include** or **Exclude** events for selected file names. The file name list is generated from events collected and stored in the database; you can add custom entries to the list. To include all file names in the filter criteria, check **All**. To include a specific file name in the filter criteria, select the file name check box. To remove a host name from the filter criteria, clear the file name check box. To manually add a file name to the list, click and then enter the file name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of file names, click .
Response Code	**Include** or **Exclude** events for selected response codes. To include all response codes in the filter criteria, check **All**. To specifically include a response code, select the response code name check box. To remove a response code from the list, clear the response code check box.

UI Element (A-Z)	Description
SYSID	**Include** or **Exclude** events for selected CICS SYSIDs. The SYSID list is generated from events collected and stored in the database; you can add custom entries to the list. To include all SYSIDs in the filter criteria, check **All**. To include a specific SYSID in the filter criteria, select the SYSID check box. To remove a SYSID from the filter criteria, clear the SYSID check box. To manually add a SYSID to the list, click and then enter the file name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of SYSIDs, click .
TD Queue Name	**Include** or **Exclude** events for selected TD (transient data) queue names. The TD queue name list is generated from events collected and stored in the database; you can add custom entries to the list. To include all TD queue names in the filter criteria, check **All**. To include a specific TD queue names in the filter criteria, select the TD queue names check box. To remove a TD queue names from the filter criteria, clear the TD queue names check box. To manually add a TD queue names to the list, click and then enter the TD queue names. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of TD queue names, click .
Terminal ID	**Include** or **Exclude** events for selected terminal IDs. The terminal ID list is generated from events collected and stored in the database; you can add custom entries to the list. To include all terminal IDs in the filter criteria, check **All**. To include a specific terminal ID in the filter criteria, select the terminal ID check box. To remove a terminal ID from the filter criteria, clear the terminal ID check box. To manually add a terminal ID to the list, click and then enter the terminal ID. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of terminal IDs, click .

UI Element (A-Z)	Description
Transaction ID	**Include** or **Exclude** events for selected CICS transaction IDs. The transaction ID list is generated from events collected and stored in the database; you can add custom entries to the list. To include all transaction IDs in the filter criteria, check **All**. To include a specific transaction ID in the filter criteria, select the transaction ID check box. To remove a transaction ID from the filter criteria, clear the transaction ID check box. To manually add a transaction ID to the list, click and then enter the transaction ID. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of transaction IDs, click .
TS Queue ID	**Include** or **Exclude** events for selected TS (transient data) queue IDs. The TS queue ID list is generated from events collected and stored in the database; you can add custom entries to the list. To include all TS queue IDs in the filter criteria, check **All**. To include a specific TS queue IDs in the filter criteria, select the TS queue ID check box. To remove a TS queue name from the filter criteria, clear the TS queue ID check box. To manually add a TS queue ID to the list, click and then enter the TS queue ID. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of TS queue IDs, click .

UI Element (A-Z)	Description
User ID	**Include** or **Exclude** events for selected user IDs. The user ID list is generated from events collected and stored in the database; you can add custom entries to the list. To include all user IDs in the filter criteria, check **All**. To include a specific user ID in the filter criteria, select the user ID check box. To remove a user ID from the filter criteria, clear the user ID check box. To manually add a user ID to the list, click and then enter the user ID. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of user IDs, click .

EJB Agent Criteria

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element (A-Z)	Description
Application Server	**Include** or **Exclude** events for selected application servers. The application server list is generated from events collected and stored in the database; you can add custom entries to the list. To include all application servers in the filter criteria, check **All**. To include a specific application server in the filter criteria, select the application server name check box. To remove a application server name from the filter criteria, clear the application server name check box. To manually add a application server to the list, click and then enter the application server name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of application server names, click .
EJB Name	**Include** or **Exclude** events for selected EJB names. The EJB name list is generated from events collected and stored in the database; you can add custom entries to the list. To include all EJB names in the filter criteria, check **All**. To include a specific EJB name in the filter criteria, select the EJB name check box. To remove an EJB name from the filter criteria, clear the EJB name check box. To manually add an EJB name to the list, click and then enter the EJB name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of EJB names, click .
Exception Setting	Include both normal and exception events, normal events only, or exception events only.

UI Element (A-Z)	Description
Method	**Include** or **Exclude** events for selected EJB methods. The EJB method list is generated from events collected and stored in the database; you can add custom entries to the list. To include all EJB methods in the filter criteria, check **All**. To include a specific EJB method in the filter criteria, select the EJB method name check box. To remove a EJB method name from the filter criteria, clear the EJB method name check box. To manually add a EJB method to the list, click and then enter the EJB method name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of EJB method names, click .
Web Application	**Include** or **Exclude** events for selected Web application names. The Web application list is generated from events collected and stored in the database; you can add custom entries to the list. To include all Web applications in the filter criteria, check **All**. To include a specific Web application in the filter criteria, select the Web application name check box. To remove a Web application name from the filter criteria, clear the Web application name check box. To manually add a Web application to the list, click and then enter the Web application name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of Web application names, click .

JDBC Agent Criteria

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element (A-Z)	Description
Class	**Include** or **Exclude** events for selected JDBC classes. The list of JDBC class names is generated from events collected and stored in the database. To include all JDBC class names in the filter criteria, check **All**. To specifically include a JDBC class names, select the JDBC class name check box. To remove JDBC class name from the list, clear the JDBC class name check box.
Database	**Include** or **Exclude** events for selected JDBC databases. The database list is generated from events collected and stored in the database; you can add custom entries to the list. To include all databases in the filter criteria, check **All**. To include a specific database in the filter criteria, select the database name check box. To remove a database name from the filter criteria, clear the database name check box. To manually add a database to the list, click and then enter the database name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of database names, click .

UI Element (A-Z)	Description
Database Object	**Include** or **Exclude** events that make use of certain database objects. The object list is generated from events collected and stored in the database; you can add custom entries to the list. To include all objects of all schemas in the filter criteria, check **All**. To include all objects in a specific schema in the filter criteria, select the schema name check box in the Database Schemas pane. To include specific objects in a specific schema in the filter criteria, clear the schema name check box but keep the database schema name selected. Then select the objects to add to the filter criteria by setting their check boxes. For an example of this, see the Object entry in the WebSphere MQ Agent Criteria section below. To remove a schema from the filter criteria, clear the schema name check box. To manually add a new schema and object to the list, click and then enter the name schema and object names. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of schema and object names, click . To edit the filter for a selected schema name, click .
Method	**Include** or **Exclude** events for selected JDBC methods. To include all JDBC method names in the filter criteria, check **All**. To specifically include a JDBC method names, select the JDBC method name check box. To remove a JDBC method name from the list, clear the JDBC method name check box.
Result Code	**Include** or **Exclude** events that have a particular result code. The list of result codes is generated from events collected and stored in the database. To include all result codes in the filter criteria, check **All**. To specifically include a result code, select the result code name check box. To remove a result code from the filter criteria, clear the result code check box.

UI Element (A-Z)	Description
SQL Code	**Include** or **Exclude** events that have a particular SQL Code. The SQL code list is generated from events collected and stored in the database; you can add custom entries to the list. To include all SQL codes in the filter criteria, check **All**. To include a specific SQL code in the filter criteria, select the SQL code check box. To remove an SQL code from the filter criteria, clear the SQL code check box. To manually add an SQL code to the list, click and then enter the SQL code name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of SQL code names, click . SQL codes used by the JDBC application vary based on the underlying database vendor.
SQL State	**Include** or **Exclude** events that have a particular SQL state. The SQL state list is generated from events collected and stored in the database; you can add custom entries to the list. To include all SQL states in the filter criteria, check **All**. To include a specific SQL state in the filter criteria, select the SQL state check box. To remove an SQL state from the filter criteria, clear the SQL state check box. To manually add an SQL code to the list, click and then enter the SQL stated name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of SQL states, click . Similar to SQL Code, SQL state values used by the JDBC application vary based on the underlying database vendor.

UI Element (A-Z)	Description
SQL Statement	**Include** or **Exclude** events for a particular SQL statement type. To include all SQL statement types in the filter criteria, check **All**. To specifically include an SQL statement type, select the SQL statement type check box. To remove an SQL statement type from the list, clear the SQL statement type name check box.
Web Application	**Include** or **Exclude** events for selected Web applications. The Web application list is generated from events collected and stored in the database; you can add custom entries to the list. To include all Web applications in the filter criteria, check **All**. To include a specific Web application in the filter criteria, select the Web application name check box. To remove a Web application name from the filter criteria, clear the Web application name check box. To manually add a Web application to the list, click and then enter the Web application name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of Web application names, click .

JMS Agent Criteria

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element (A-Z)	Description
Class	**Include** or **Exclude** events for selected JMS classes. To include all JMS class names in the filter criteria, check **All**. To specifically include a JMS class, select the JMS class name check box. To remove a JMS class from the filter criteria, clear the JMS class name check box.
Connection Name	**Include** or **Exclude** events for selected connection names. The connection name list is generated from events collected and stored in the database; you can add custom entries to the list. To include all connection names in the filter criteria, check **All**. To include a specific connection name in the filter criteria, select the connection name check box. To remove a connection name from the filter criteria, clear the connection name check box. To manually add a connection name to the list, click and then enter the connection name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of connection names, click .
Exception Setting	Select whether to include both normal and exception events, normal events only, or exception events only.
Method	**Include** or **Exclude** events for selected JMS methods.The method list is generated from events collected and stored in the database; you can add custom entries to the list. To include all methods in the filter criteria, check **All**. To include a specific method in the filter criteria, select the method name check box. To remove a method name from the filter criteria, clear the method name check box. Check **Do not collect events when no message is available** to avoid collecting JMS Consumer events with no message.

UI Element (A-Z)	Description
Queue	**Include** or **Exclude** events for selected queues. To include all JMS queue names in the filter criteria, check **All**. To specifically include a JMS queue, select the JMS queue name check box. To remove a JMS queue from the filter criteria, clear the JMS queue name check To manually add a connection name to the list, click and then enter the connection name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of queue names, click .
Topic	**Include** or **Exclude** events for selected topics. The topic list is generated from events collected and stored in the database; you can add custom entries to the list. To include all topics in the filter criteria, check **All**. To include a specific topic in the filter criteria, select the topic name check box. To remove a topic name from the filter criteria, clear the topic name check box. To manually add a topic to the list, click and then enter the topic name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of topic names, click .
Web Application	**Include** or **Exclude** events for selected Web applications. The Web application list is generated from events collected and stored in the database; you can add custom entries to the list. To include all Web application names in the filter criteria, check **All**. To include a specific Web application in the filter criteria, select the Web application name check box. To remove a Web application name from the filter criteria, clear the Web application name check box. To manually add a Web application to the list, click and then enter the Web application name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of Web application names, click .

Servlet Agent Criteria

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element (A-Z)	Description
Application Server	**Include** or **Exclude** events for selected application servers. The application server list is generated from events collected and stored in the database; you can add custom entries to the list. To include all application servers in the filter criteria, check **All**. To include a specific application server in the filter criteria, select the application server name check box. To remove a application server name from the filter criteria, clear the application server name check box. To manually add a application server to the list, click and then enter the application server name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of application server names, click .
Client Host/IP	**Include** or **Exclude** events for selected client host/IPs. The host/IP list is generated from events collected and stored in the database; you can add custom entries to the list. To include all host/IPs in the filter criteria, check **All**. To include a specific host/IP in the filter criteria, select the host/IP name check box. To remove a host/IP name from the filter criteria, clear the host/IP name check box. To manually add a host/IP to the list, click and then enter the host/IP name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of host/IP names, click .
Exception Setting	Determine whether to include both normal and exception events, normal events only, or exception events only.

UI Element (A-Z)	Description
Servlet Method	**Include** or **Exclude** events for selected servlet methods. To include all servlet method names in the filter criteria, check **All**. To specifically include a servlet method, select the servlet method name check box. To remove a servlet method from the list, clear the servlet method name check
Status Code	**Include** or **Exclude** events for selected status codes. To include all status code names in the filter criteria, check **All**. To specifically include a status code, select the status code name check box. To remove a status code from the list, clear the status code name check.
URI	**Include** or **Exclude** events for selected URIs. The URI list is generated from events collected and stored in the database; you can add custom entries to the list. To include all URIs in the filter criteria, check **All**. To include a specific URI in the filter criteria, select the URI name check box. To remove a URI name from the filter criteria, clear the URI name check box. To manually add a URI to the list, click and then enter the URI name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of URI names, click .

UI Element (A-Z)	Description
Web Application	**Include** or **Exclude** events for selected Web applications. The Web application list is generated from events collected and stored in the database; you can add custom entries to the list. To include all Web applications in the filter criteria, check **All**. To include a specific Web application in the filter criteria, select the Web application name check box. To remove a Web application name from the filter criteria, clear the Web application name check box. To manually add a Web application to the list, click and then enter the Web application name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of Web application names, click .

User Event Agent Criteria

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element (A-Z)	Description
Class	**Include** or **Exclude** events for selected classes. The class list is generated from events collected and stored in the database; you can add custom entries to the list. To include all classes in the filter criteria, check **All**. To include a specific class in the filter criteria, select the class name check box. To remove a class name from the filter criteria, clear the class name check box. To manually add a class to the list, click and then enter the class name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of class names, click .
Completion Code	**Include** or **Exclude** events for selected User Event method completion codes. To include all completion code in the filter criteria, check **All**. To specifically include a completion code, select the completion code check box. To remove a completion code from the list, clear the completion code name check box. Not available for Tuxedo, .NET, NonStop TMF, or Java User Event Agents.

UI Element (A-Z)	Description
Method	**Include** or **Exclude** events for selected methods. The method list is generated from events collected and stored in the database; you can add custom entries to the list. To include all methods in the filter criteria, check **All**. To include a specific method in the filter criteria, select the method name check box. To remove a method name from the filter criteria, clear the method name check box. To manually add a method to the list, click and then enter the method name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of method names, click .
Queue	**Include** or **Exclude** events for selected User Event queue strings. The queue list is generated from events collected and stored in the database; you can add custom entries to the list. To include all queues in the filter criteria, check **All**. To include a specific queue in the filter criteria, select the queue name check box. To remove a queue name from the filter criteria, clear the queue name check box. To manually add a queue to the list, click and then enter the queue name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of queue names, click .

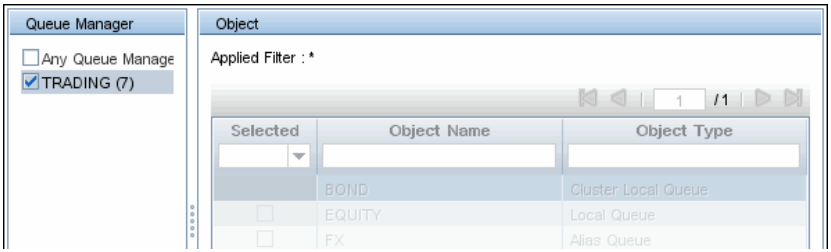
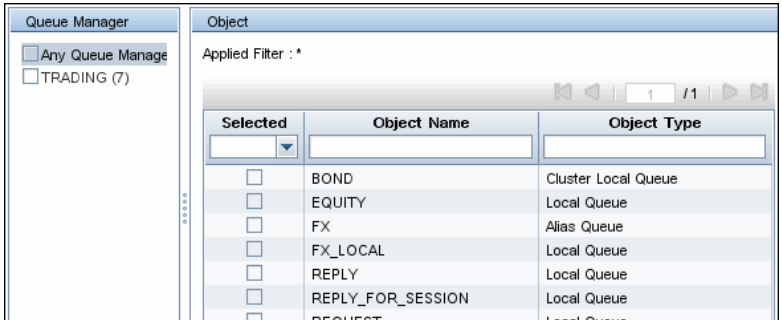
UI Element (A-Z)	Description
Queue Space	**Include** or **Exclude** events for selected User Event queue space strings. The string list is generated from events collected and stored in the database; you can add custom entries to the list. To include all queue space strings in the filter criteria, check **All**. To include a specific queue space string in the filter criteria, select the queue space string name check box. To remove a queue space string from the filter criteria, clear the queue space string check box. To manually add a queue space string to the list, click and then enter the string. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of queue space string names, click . For the Tuxedo Agent, Queue Space must match the queue space for which the tpenqueue/tpdequeue operation is intended. Not available for .NET, NonStop TMF, or Java User Event Agents.
Service	**Include** or **Exclude** events for selected User Event service strings. The service strings list is generated from events collected and stored in the database; you can add custom entries to the list. To include all service strings in the filter criteria, check **All**. To include a specific service string in the filter criteria, select the service string name check box. To remove a service string name from the filter criteria, clear the service string name check box. To manually add a service string to the list, click and then enter the service string name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of service string names, click .

UI Element (A-Z)	Description
Status	**Include** or **Exclude** events for selected User Event status strings. The status list is generated from events collected and stored in the database; you can add custom entries to the list. To include all status strings in the filter criteria, check **All**. To include a specific status in the filter criteria, select the status string name check box. To remove a status name from the filter criteria, clear the status name check box. To manually add a status to the list, click and then enter the status name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of status names, click . Not available for .NET, Tuxedo, or NonStop TMF Agents.
Technology	The type of User Event Agent. Enter: .NET, Tuxedo, NonStop-TMF, or Java.

WebSphere MQ Agent Criteria

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element (A-Z)	Description
API Name	**Include** or **Exclude** selected WebSphere MQ APIs in the filter criteria. The list of class names is generated from events collected and stored in the database. Click **Refresh** to obtain the current set of class names. To include all MQ API names in the filter criteria, check **All**. To specifically include an MQ API, select the MQ API name check box. To remove an MQ API from the filter criteria, clear the MQ API name check box. Check **Discard MQCMIT in empty transactions** to avoid collecting empty transactions. Empty transactions occur in applications that pool a queue by making MQGET calls in a loop. When there are no inputs, the MQGET calls fail with a MQRC_NO_MSG_AVAILABLE and are typically filtered out by customers in TransactionVision. However, the MQGET may be followed by an MQCMIT. This option provides an easy way to filter out these MQCMIT events. Check **Do not send MQDISC exit event** to report only the entry information of each MQDISC call. This improves performance and reduces overhead for the agent. Otherwise, the agent must reconnect and reopen the event queue to send MQDISC exit information, and this information is typically not necessary. Check **Do not send browsing MQGET** if you do not want to collect MQGET calls used to browse messages on the queue.
Completion Code	**Include** or **Exclude** events for selected WebSphere MQ API completion codes. To include all MQ API completion codes in the filter criteria, check **All**. To specifically include an MQ API completion code, select the MQ API completion code check box. To remove an MQ API completion code from the filter criteria, clear the MQ API completion code name check box.

UI Element (A-Z)	Description
Object	**Include** or **Exclude** queue managers and objects. The list is generated from events collected and stored in the database; you can add custom entries to the list. To specify all objects of all queue managers, check **All**. To specify all objects in a specific queue manager, select the queue manager name in the Queue Managers pane. For example, here all 13 objects in the **TRADING** queue manager are selected to be excluded or included:  Note that a queue manager entry can be selected as well as having its check box set or cleared. If the entry is selected and its check box is cleared, you can specify individual objects of that queue manager. If the entry is selected and its check box is set, all objects of that queue manager are implicitly selected (and the object list is disabled as shown above). To specify a specific object across all queue managers in the filter criteria, select the **All Queue Managers** entry and then select the object name in the list and set its check box. For example, here all **SYSTEM** objects are selected to be excluded or included: 

UI Element (A-Z)	Description
Object (Continued)	If the object you want to specify is not in the **Object Name** list, manually add it as follows: **1** Click . **2** From the **Queue Managers** drop-down list, choose **All Queue Managers** to add the object to all queue managers. Or if you know the specific queue manager on which the object resides, specify that name. **3** Enter the object name. An asterisk can be used at the beginning or the end as a wildcard. **4** Click **OK**. To remove a queue manger and all of its objects from the filter criteria, clear the queue manager name check box. To refresh the list of queue manager and object names, click . To edit the filter for a selected queue manager name, click . Note that with the clustered queues the same object name is present on multiple queue managers. To prevent the WMQ Agent from attempting to resolve remote cluster queues or queues which are opened remotely through a transmission queue, set the **Do not attempt to resolve cluster or remotely opened queues** check box. By default, this option is not specified. This option is relevant only if MQ Event Messaging is enabled in your environment, and need to reduce LOCALEV messages being sent to the SYSTEM.ADMIN.QMGR.EVENT queue because of failed MQOPEN operations during WMQ Agent queue resolution. All active data collection filters must have this option checked for this resolution to be stopped in the WMQ Agent. By stopping this resolution, cluster queues cannot be resolved.
Reason Code	**Include** or **Exclude** events for selected WebSphere MQ API reason codes. To include all reason codes in the filter criteria, check **All**. To specifically include a reason code, select the reason code check box. To remove a reason code from the filter criteria, clear the reason code check box. **Note**: MQGET calls with a MQRC_NO_MSG_AVAILABLE reason code are filtered out by default. When monitoring WebSphere MQ Brokers, make sure to filter out MQGET calls to WebSphere MQ Broker system queues with a reason code of MQRC_NO_MSG_AVAILABLE.

UI Element (A-Z)	Description
ReplyTo Object	**Include** or **Exclude** queue managers and objects. The list is generated from events collected and stored in the database; you can add custom entries to the list. To include all objects of all queue managers in the filter criteria, check **All**. To include all objects in a specific queue manager in the filter criteria, select the queue manager name check box in the Queue Managers pane. To include specific objects in a specific schema in the filter criteria, clear the queue manager name check box but keep the queue manager name selected. Then identify the objects to add to the filter criteria by using the Object pane as described for Database Objects in the JDBC Criteria section. To remove a queue manger and all of its objects from the filter criteria, clear the queue manager name check box. To manually add a new queue manager and object to the list, click and then enter the names. An asterisk can be used at the beginning or the end as a wildcard. An object filter can be applied. To refresh the list of queue manager and object names, click . To edit the filter for a selected queue manager name, click . Selecting objects not listed under a queue manager is useful if the same object name is present on multiple queue managers (for example, as in clustered queues). The object list is generated from events collected and stored in the database. To collect events that do not have the MQMD ReplyTo fields, select **Collect MQPUT, MQPUT1, and MQGET events if both "ReplyTo" fields are not set**.

WebSphere MQ IMS Bridge Agent Criteria

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element (A-Z)	Description
MQIMS Bridge API	**Include** or **Exclude** events for selected MQIMS bridge APIs. To include all MQIMS bridge APIs in the filter criteria, check **All**. To specifically include a MQIMS bridge API, select the MQIMS bridge API check box. For more information about the MQSeries-IMS Bridge Agent, see the *HP TransactionVision Deployment Guide* PDF.

WebSphere MQ z/OS Agent Criteria

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element (A-Z)	Description
SYSID	**Include** or **Exclude** events for selected CICS SYSIDs. The SYSID list is generated from events collected and stored in the database; you can add custom entries to the list. To include all SYSIDs in the filter criteria, check **All**. To include a specific SYSID in the filter criteria, select the SYSID check box. To remove a SYSID name from the filter criteria, clear the SYSID name check box. To manually add a SYSID to the list, click and then enter the SYSID name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of host names, click . Only events generated on the z/OS CICS platform have a CICS SYSID. To collect events that do not have a CICS SYSID (non-z/OS CICS events), choose **Collect events regardless of whether the field "CICS SYSID" is available**. Choose **Collect events if and only if the field "CICS SYSID" is available** to collect events that only have CICS SYSIDs.

UI Element (A-Z)	Description
CICS Task	**Include** or **Exclude** events for selected CICS tasks. The task list is generated from events collected and stored in the database; you can add custom entries to the list. To include all tasks in the filter criteria, check **All**. To include a specific task in the filter criteria, select the task check box. To remove a task name from the filter criteria, clear the task name check box. To manually add a task to the list, click and then enter the task name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of task names, click . Only events generated on the z/OS CICS platform have a CICS task. To collect events that do not have a CICS task (non-z/OS CICS events), choose **Collect events regardless of whether the field "CICS Task" is available**. Choose Collect events if and only if the field "CICS Task" is available to collect events that only have CICS tasks.
CICS Transaction	**Include** or **Exclude** events for selected z/OS CICS transactions. The transaction list is generated from events collected and stored in the database; you can add custom entries to the list. To include all transactions in the filter criteria, check **All**. To include a specific transaction in the filter criteria, select the transaction check box. To remove a transaction name from the filter criteria, clear the transaction name check box. To manually add a transaction to the list, click and then enter the transaction name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of transaction names, click . Only events generated on the z/OS CICS platform have a CICS transaction. To collect events that do not have a CICS transaction (non-z/OS CICS events), choose **Collect events regardless of whether the field "CICS Transaction" is available**. Choose **Collect events if and only if the field "CICS Transaction" is available** to collect events that only have CICS transactions.

UI Element (A-Z)	Description
IMS Identifier	**Include** or **Exclude** events for selected IMS identifiers. The identifier list is generated from events collected and stored in the database; you can add custom entries to the list. To include all identifiers in the filter criteria, check **All**. To include a specific identifier in the filter criteria, select the identifier check box. To remove an identifier name from the filter criteria, clear the identifier name check box. To manually add an identifier to the list, click and then enter the identifier name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of identifier names, click . Only events generated on the z/OS IMS platform have an IMS identifier. To collect events that do not have an IMS identifier (non-z/OS IMS events), choose **Collect events regardless of whether the field "IMS Identifier" is available**. Choose **Collect events if and only if the field "IMS Identifier" is available** to collect events that only have IMS identifiers.
IMS PSB	**Include** or **Exclude** events for selected IMS PSBs. The PSB list is generated from events collected and stored in the database; you can add custom entries to the list. To include all PSBs in the filter criteria, check **All**. To include a specific PSB in the filter criteria, select the PSB name check box. To remove a PSB from the filter criteria, clear the PSB name check box. To manually add a PSB to the list, click and then enter the PSB name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of PSB names, click . Only events generated on the z/OS IMS platform have an IMS PSB. To collect events that do not have an IMS PSB (non-z/OS IMS events), choose **Collect events regardless of whether the field "IMS PSB" is available**. Choose **Collect events if and only if the field "IMS PSB" is available** to collect events that only have IMS PSBs.

UI Element (A-Z)	Description
IMS Region Identifier	**Include** or **Exclude** events for selected IMS region identifiers. The IMS region identifier list is generated from events collected and stored in the database; you can add custom entries to the list. To include all IMS region identifiers in the filter criteria, check **All**. To include a specific IMS region identifier in the filter criteria, select the IMS region identifier name check box. To remove a IMS region identifier from the filter criteria, clear the IMS region identifier name check box. To manually add an IMS region identifier to the list, click and then enter the IMS region identifier name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of IMS region identifiers, click . Only events generated on the z/OS IMS platform have an IMS region identifier. To collect events that do not have an IMS region identifier (non-z/OS IMS events), choose **Collect events regardless of whether the field "IMS Region Identifier" is available**. Choose **Collect events if and only if the field "IMS Region Identifier" is available** to collect events that only have IMS region identifiers.

UI Element (A-Z)	Description
IMS Region Type	**Include** or **Exclude** events for selected IMS region types. The IMS region type list is generated from events collected and stored in the database; you can add custom entries to the list. To include all IMS region types in the filter criteria, check **All**. To include a specific IMS region type in the filter criteria, select the IMS region type name check box. To remove a IMS region type name from the filter criteria, clear the IMS region type name check box. To manually add a IMS region type to the list, click and then enter the IMS region type name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of IMS region type names, click . Only events generated on the z/OS IMS platform have an IMS region type. To collect events that do not have an IMS region type (non-z/OS IMS events), choose **Collect events regardless of whether the field "IMS Region Type" is available**. Choose **Collect events if and only if the field "IMS Region Type" is available** to collect events that only have IMS region types.

UI Element (A-Z)	Description
IMS Transaction	**Include** or **Exclude** events for selected IMS transactions. The IMS transaction list is generated from events collected and stored in the database; you can add custom entries to the list. To include all transactions in the filter criteria, check **All**. To include a specific transaction in the filter criteria, select the transaction name check box. To remove a transaction from the filter criteria, clear the transaction name check box. To manually add a transaction to the list, click and then enter the transaction name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of transactions, click . Only events generated on the z/OS IMS platform have an IMS transaction. To collect events that do not have an IMS transaction (non-z/OS IMS events), choose **Collect events regardless of whether the field "IMS transaction" is available**. Choose **Collect events if and only if the field "IMS Transaction" is available** to collect events that only have IMS transactions.
Jobs and Steps	**Include** or **Exclude** selected z/OS job steps in the filter criteria. The job step list is generated from events collected and stored in the database; you can add custom entries to the list. To include all job steps in the filter criteria, check **All**. To include a specific job and step in the filter criteria, select the job step name check box. To remove a job step from the filter criteria, clear the job step name check box. To manually add a job step to the list, click and then enter the job and step name. An asterisk can be used at the beginning or the end as a wildcard. Only events generated on the z/OS platform have job names and job steps. To collect events that do not have a CICS job and step (non-z/OS events), choose **Collect events regardless of whether the field "Jobs and Steps" is available**. Choose **Collect events if and only if the field "Jobs and Steps" is available** to collect events that only have the CICS job names and steps.

WebSphere MQ i5/OS Agent Criteria

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element (A-Z)	Description
Job Name	**Include** or **Exclude** events for selected job names. The job name list is generated from events collected and stored in the database; you can add custom entries to the list. To include all job names in the filter criteria, check **All**. To include a specific job name in the filter criteria, select the job name check box. To remove a job from the filter criteria, clear the job name check box. To manually add a job to the list, click and then enter the job name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of job names, click .

Data Collection Filter Page, General Settings Tab

This page enables you to specify data collection filter names and descriptions.

To access	**1** Select **Admin** > **Transaction Management.** **2** (left pane) Click the **Configuration** tab. **3** Select **Data Collection Filters** > <filter name>. **4** (right pane) Select the **General Settings** tab.
Important information	Click the criteria category name on the left side for the condition that you want to change and make desired changes on the right side of the page.
Relevant tasks	➤ "How to Create a Data Collection Filter" on page 213 ➤ "How to Modify a Data Collection Filter" on page 215

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element (A-Z)	Description
Filter Description	Enter text that helps administrators know the purpose of the filter.
Filter Name	Enter a name for the data collection filter. The name must be unique to the TransactionVision deployment environment.

6

Queries

This chapter includes:

Concepts

- Queries Overview on page 260
- Optimizing Query Performance on page 260

Tasks

- How to Add a Query on page 262
- How to Modify a Query on page 264
- How to Test a Query on page 264

Reference

- Queries User Interface on page 265

Concepts

Queries Overview

Queries control which events in the database are displayed in the Event Analysis report or used for analysis in the Component Topology. You can specify to use a built-in query or a custom query. The built-in queries are:

- All Events: Includes all events in all schemas in the database.
- Last 24 hours: Includes only the events collected in the last 24 hours.

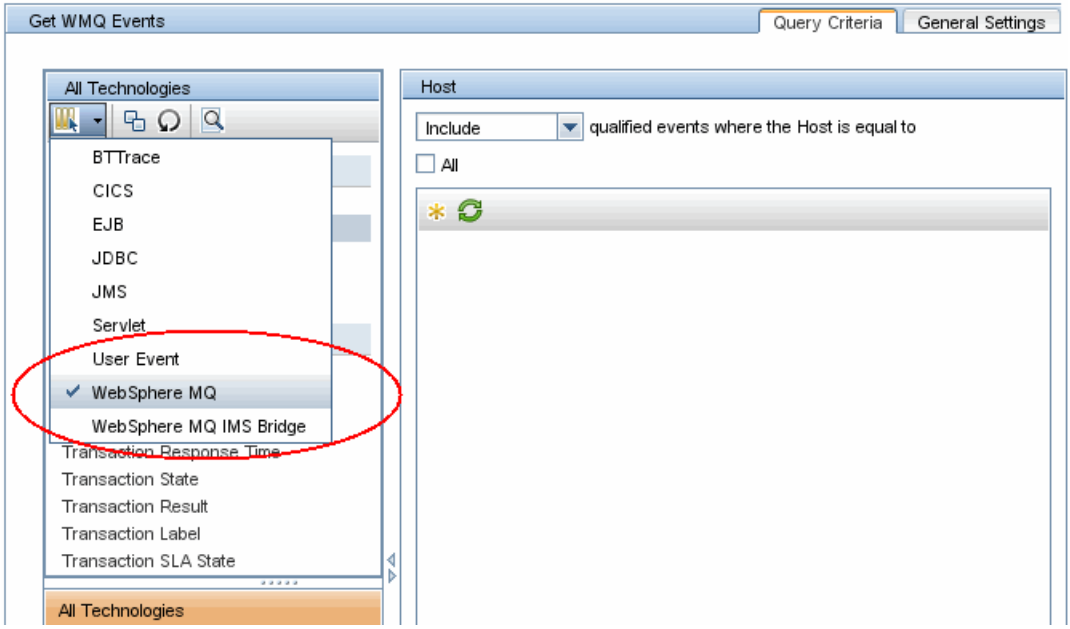
A custom query includes events based on any number of criteria. For instance, specific hosts, Queue Managers, programs, time frames and APIs can all be used to exclude or include events. This allows you to choose only the event data you need for analysis or to resolve a specific problem.

Optimizing Query Performance

To improve the time it takes to render the Event Analysis report or the Component Topology, define queries that meet the following guidelines:

- Include only those technologies that you need. By default, all technologies are specified.

For example, if your query need only include WebSphere MQ events, disable the remaining technologies on the Technology drop-down list so that only WebSphere MQ remains selected:



➤ Specify User Buffer Data criteria in its own query.

For most criteria, TransactionVision uses a database lookup table, which stores a set of predefined key values along with event IDs. However, it uses a linear search for the User Buffer Data criteria. Although linear searching enhances query capabilities, it reduces query performance. If you include these criteria in your query, query performance is degraded. To minimize the impact on query performance, avoid specifying these criteria together in the same query when possible.

Another drawback of linear searches is that the resulting reports have limited page navigation capabilities since the total number of events and result pages is unknown. For example, after running a query that contains a query condition for event User Buffer data, the Event Analysis report would show page forward and backward options, but would not provide the number of events, the number of event pages, or controls to navigate to a particular page.

Tasks

How to Add a Query

This task describes how to create a new query.

1 Prerequisites

Determine the technology that applies to the events you want to extract from the database. Then, determine which events you want to include or exclude.

Review the list of technologies in "Categories of Filter Criteria" on page 209. The technologies used for data collection filters are the same that apply to queries with one exception: queries have the additional Transaction Attributes category in Common Options.

Review the list of event categories on "Query Page, Query Criteria Tab" on page 266.

2 Select **Admin** > **Transaction Management**.

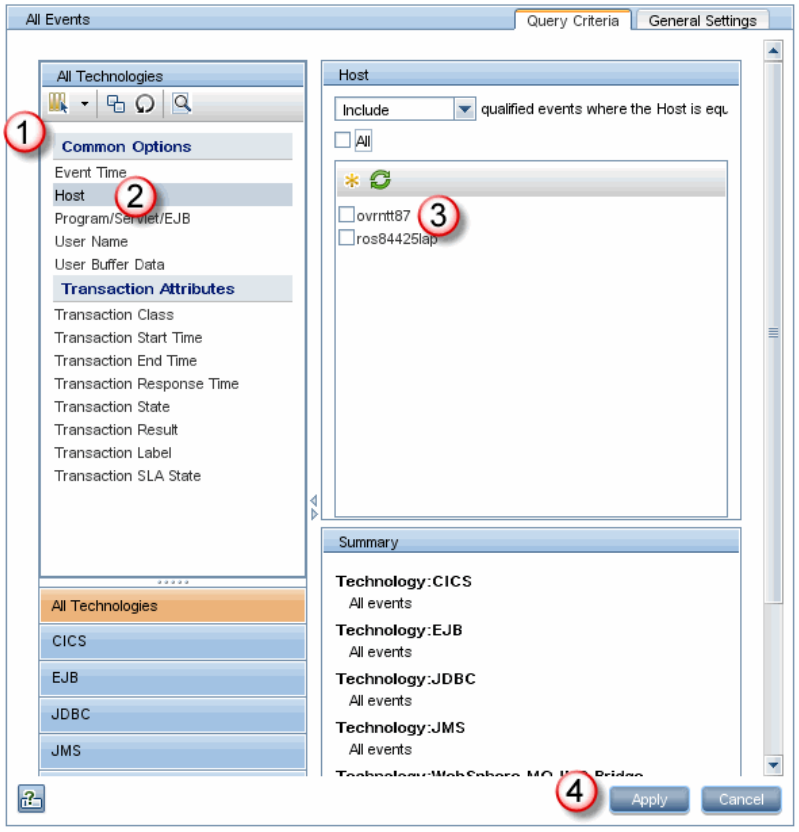
3 (left pane) Click the **Configuration** tab.

4 Right-click **Queries** and select **New Query**.

The New Query dialog appears.

5 Enter a name and click **OK**.

6 In the right pane, use the **Query Criteria** tab to manage and set the criteria for the query. On this tab, you perform these general steps:



The following table describes the locations in this page:

Callout	Description
1	Choose the category of criteria that you want to set.
2	Choose the criterion that you want to set.
3	Set the criteria.
4	Review the settings in the Summary panel and then click Apply.

For more details about this interface, see "Query Page, Query Criteria Tab" on page 266.

How to Modify a Query

To modify a query, change the criteria settings as follows:

1. Select **Admin** > **Transaction Management.**
2. (left pane) Click the **Configuration** tab.
3. Select **Queries** > <query name>.
4. (right pane) On the Query Criteria tab, update the query criteria.
5. Click **Apply**.

How to Test a Query

After you define a query, use the **Preview** button to test it.

You should see the expected events on the Event Analysis report. See "Event Analysis Report" on page 691.

Reference

Queries User Interface

This section describes:

- Queries Summary Page on page 265
- Query Page, Query Criteria Tab on page 266
- Query Page, General Settings Tab on page 314

Queries Summary Page

The Queries Summary page enables you to view and manage all queries in the deployment environment.

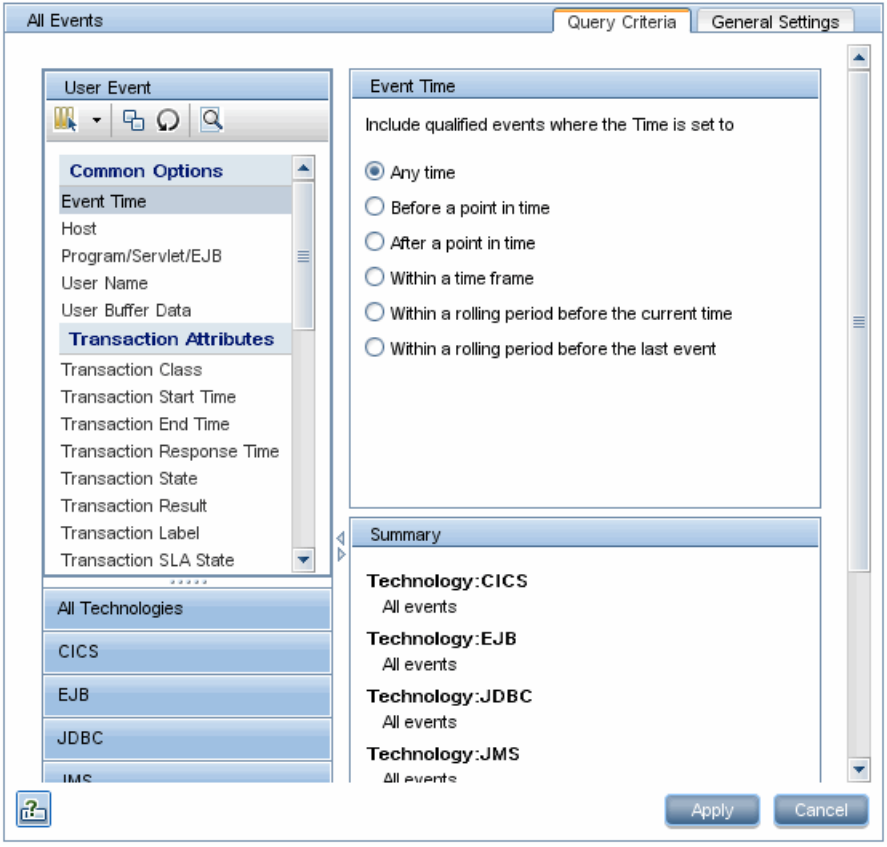
To access	**1** Select **Admin> Transaction Management > Configuration**. **2** (left pane) Select **Queries**. **3** (right pane) From the **Available Queries** list, choose the query for which you want to view a summary.

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
Available Queries	Lists all queries in the deployment environment. Select a criteria to view its summary information.
Summary	Lists the query criteria, grouped by technology. Click the technology name to go to that area in the query criteria.

Query Page, Query Criteria Tab

This page enables you to specify query criteria.



To access	**1** Select **Admin**> **Transaction Management** > **Configuration**. **2** (left pane) Select **Queries** > <query name>. **3** (right pane) Select the **Query Criteria** tab.
Important information	➤ View the Summary pane in the lower right-hand side of the page to see the combined effect of all criteria set for the query. The **Apply** button updates the Summary pane. ➤ Any Common Options settings specified at the All Technologies level are inherited, and can be overridden at a specific technology level. ➤ For the WebSphere MQ and CICS Agents, a blank field is treated as a match when a query criteria is specified. For example, if you create a query that collects events with the user name J_SMITH, events with an empty user name are also collected by the WebSphere MQ and CICS Agents. ➤ User-created entries for a query are specific to that query. For instance, if you add a new host named "Server1" to a particular query, that entry does not appear in the Host list of any other query.
Relevant tasks	➤ "How to Add a Query" on page 262 ➤ "How to Modify a Query" on page 264

Query Criteria Tab, Common Options

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element (A-Z)	Description
Event Time	When you create a new query, the default event time is events from the last 24 hours, relative to the last collected event. For example, if the time and date of last event in the schema is 07:56 on 7-22-2005, the query includes events collected since 07:56 on 7-21-2005. ➤ Select the time mode. ➤ **Any Time**: Events are collected with no time restrictions. ➤ **From a set start to end time**: Specify a time range (based on the clock of the host running the agent) to collect events. To select the date from a calendar, click the drop-down button in the field and then click the desired date. The events are collected from the specified start time until the specified end time. ➤ **From a set start time**: Specify a time range (based on the clock of the host running the Agent) to collect events. To select the date from a calendar, click the drop-down button in the field and then click the desired date. The events are collected from a specific start time until event collection is stopped. Even if you specify a start time, you must initialize event recording before that start time. Initializing event recording causes the Analyzer to send a configuration message with query information to all assigned communication links. For the .NET Agent, Tuxedo Agent, NonStop TMF Agent, and Java User Event technology this criterion is not evaluated.

UI Element (A-Z)	Description
Host	**Include** or **Exclude** selected hosts in the query criteria. The host list is generated from events collected and stored in the database; you can add custom entries to the list. To include all hosts in the query criteria, check **All**. To include a specific host in the query criteria, select the host name check box. To remove a host name from the query criteria, clear the host name check box. To manually add a host to the list, click and then enter the host name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of host names, click .
Program/Servlet/EJB	**Include** or **Exclude** selected programs, servlets, and/or EJBs in the query criteria. The program list is generated from events collected and stored in the database. To include all programs, servlets, and EJBs in the query criteria, check **All**. To include a specific program, servlet, or EJB in the query criteria, select the program, servlet, or EJB name check box. To remove a program, servlet, or EJB name from the query criteria, clear the program, servlet, or EJB name check box. To manually add a program, servlet, or EJB to the list, click and then enter the program, servlet, or EJB name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of program, servlet, or EJB names, click . For the Tuxedo Agent, the program, servlet, or EJB name must be the executable process that made the Tuxedo call. This can be done by either including the name explicitly or by specifying **All**. For the NonStop TMF Agent, this criterion is not evaluated.

UI Element (A-Z)	Description
User Name	**Include** or **Exclude** selected users in the query criteria. The user name list is generated from events collected and stored in the database; you can add custom entries to the list. To include all user names in the query criteria, check **All**. To include a specific user name in the query criteria, select the user name check box. To remove a user name from the query criteria, clear the user name check box. To manually add a user name to the list, click and then enter the user name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of user names, click . For the .NET Agent, Tuxedo Agent, and NonStop TMF Agent, and Java User Event technology this criterion is not evaluated.
User Buffer Data	**Note**: The TransactionVision Analyzer can be configured to use an XML Event Compression bean, which can reduce the size of the resulting event database. However, if this bean is in use by the Analyzer, it is not possible to query on the user buffer data. Specify where to search. If you have non-XML based user data in events, select Search in user data. If you have XML user data and have enabled the DefaultModifierBean in the Analyzer, select Search in event XML to search for strings in that XML data. Click the appropriate radio button to indicate whether the specified user data is in binary string or normal string format and enter the buffer data. For normal string format, select the correct code page. To query on a double-byte character set (DBCS) string, you may either type in the DBCS text string or enter the binary representation of the text. Select the appropriate **Code Page** for the desired character coding. TransactionVision uses linear searching for this criteria. Although linear searching enhances query capabilities, it reduces query performance. If you include this criteria in your query, query performance is degraded.

Query Criteria Tab, Transaction Attributes

The transaction related query criteria follow the same basic rules as other query criteria, allowing you to add criteria based on the transaction in which the event was a part.

For example, you can combine these two criteria:

- Ask for all events that have a certain program name.
- Ask only for events that are part of transactions that have been marked as failed.

User interface elements are described below:

UI Element	Description
Transaction Class	The Business Transaction CI name.
Transaction Start Time	The start time of the transaction.
Transaction End Time	The end time of the transaction.
Transaction Response Time	The end-to-end response time of the transaction. See "Key Information Reported For a Transaction" on page 641.
Transaction State	The completion state of the transaction.
Transaction Result	The result state of the transaction.
Transaction Label	The transaction label.
Transaction SLA State	The SLA state of the transaction.

These elements are populated based on transaction data rules. See "Business Transaction Tracing" on page 513.

Query Criteria Tab, CICS Agent Criteria

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element (A-Z)	Description
API Name	Select whether to **Include** or **Exclude** events for selected APIs. To include all APIs in the query criteria, check **All**. To specifically include an API, select the API name check box. The APIs are grouped by API type. To remove an API from the list, clear the API check box.
API Type	Select whether to **Include** or **Exclude** events for selected API types. To include all APIs in the query criteria, check **All**. To specifically include an API, select the API name check box. The APIs are grouped by API type.
CICS Task	Include or Exclude events for selected CICS tasks. The task list is generated from events collected and stored in the database; you can add custom entries to the list. To include all tasks in the query criteria, check **All**. To include a specific task in the query criteria, select the task check box. To remove a task name from the query criteria, clear the task name check box. To manually add a task to the list, click and then enter the task name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of tasks, click . To edit the filter for a selected MQ queue manager name, click . To search for specific tasks that are listed in the CICS Task pane (in the case where there are thousands of them), do any of the following: ➤ Check the check boxes of tasks that you want to see, and select **True** in the **Selected** dropdown list to see all tasks that are checked. You can also select **False** to see all tasks that are unchecked. ➤ Enter all or part of the task name (it is not case-sensitive) in the **CICS Task** text box and all tasks that include that text are displayed.

UI Element (A-Z)	Description
File Name	Select whether to **Include** or **Exclude** events for selected file names. The file name list is generated from events collected and stored in the database; you can add custom entries to the list. To include all file names in the query criteria, check **All**. To include a specific file name in the query criteria, select the file name check box. To remove a host name from the query criteria, clear the file name check box. To manually add a file name to the list, click and then enter the file name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of file names, click .
Response Code	Select whether to **Include** or **Exclude** events for selected response codes. To include all response codes in the query criteria, check **All**. To specifically include a response code, select the response code name check box. To remove a response code from the list, clear the response code check box.
SYSID	**Include** or **Exclude** events for selected CICS SYSIDs. The SYSID list is generated from events collected and stored in the database; you can add custom entries to the list. To include all SYSIDs in the filter criteria, check **All**. To include a specific SYSID in the filter criteria, select the SYSID check box. To remove a SYSID from the filter criteria, clear the SYSID check box. To manually add a SYSID to the list, click and then enter the file name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of SYSIDs, click .

UI Element (A-Z)	Description
TD Queue Name	**Include** or **Exclude** events for selected TD (transient data) queue names. The TD queue name list is generated from events collected and stored in the database; you can add custom entries to the list. To include all TD queue names in the query criteria, check **All**. To include a specific TD queue names in the query criteria, select the TD queue names check box. To remove a TD queue names from the query criteria, clear the TD queue names check box. To manually add a TD queue names to the list, click and then enter the TD queue names. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of TD queue names, click .
Terminal ID	**Include** or **Exclude** events for selected terminal IDs. The terminal ID list is generated from events collected and stored in the database; you can add custom entries to the list. To include all terminal IDs in the query criteria, check **All**. To include a specific terminal ID in the query criteria, select the terminal ID check box. To remove a terminal ID from the query criteria, clear the terminal ID check box. To manually add a terminal ID to the list, click and then enter the terminal ID. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of terminal IDs, click .

UI Element (A-Z)	Description
Transaction ID	**Include** or **Exclude** events for selected CICS transaction IDs. The transaction ID list is generated from events collected and stored in the database; you can add custom entries to the list. To include all transaction IDs in the query criteria, check **All**. To include a specific transaction ID in the query criteria, select the transaction ID check box. To remove a transaction ID from the query criteria, clear the transaction ID check box. To manually add a transaction ID to the list, click and then enter the transaction ID. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of transaction IDs, click .
TS Queue ID	**Include** or **Exclude** events for selected TS (transient data) queue IDs. The TS queue ID list is generated from events collected and stored in the database; you can add custom entries to the list. To include all TS queue IDs in the query criteria, check **All**. To include a specific TS queue IDs in the query criteria, select the TS queue ID check box. To remove a TS queue name from the query criteria, clear the TS queue ID check box. To manually add a TS queue ID to the list, click and then enter the TS queue ID. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of TS queue IDs, click .

Query Criteria Tab, EJB Agent Criteria

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element (A-Z)	Description
Application Server	**Include** or **Exclude** events for selected application servers. The application server list is generated from events collected and stored in the database; you can add custom entries to the list. To include all application servers in the query criteria, check **All**. To include a specific application server in the query criteria, select the application server name check box. To remove an application server name from the query criteria, clear the application server name check box. To manually add an application server to the list, click and then enter the application server name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of application server names, click .
EJB Application	Select whether to **Include** or **Exclude** events for selected EJB names. The EJB name list is generated from events collected and stored in the database; you can add custom entries to the list. To include all EJB names in the query criteria, check **All**. To include a specific EJB name in the query criteria, select the EJB name check box. To remove an EJB name from the query criteria, clear the EJB name check box. To manually add an EJB name to the list, click and then enter the EJB name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of EJB names, click .

UI Element (A-Z)	Description
EJB Method	**Include** or **Exclude** events for selected EJB methods. The EJB method list is generated from events collected and stored in the database; you can add custom entries to the list. To include all EJB methods in the query criteria, check **All**. To include a specific EJB method in the query criteria, select the EJB method name check box. To remove a EJB method name from the query criteria, clear the EJB method name check box. To manually add a EJB method to the list, click and then enter the EJB method name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of EJB method names, click .
EJB Status	Select whether to **Include** or **Exclude** events for the selected EJB status. Check Failure to include only failed EJB events, or check Success to include only successful EJB events. Check Select All EJB Status to include successful and failed EJB events in the query criteria.

Query Criteria Tab, JDBC Agent Criteria

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element (A-Z)	Description
Application Server	Select whether to **Include** or **Exclude** events for selected application servers. The application server list is generated from events collected and stored in the database; you can add custom entries to the list. To include all application servers in the query criteria, check **All**. To include a specific application server in the query criteria, select the application server name check box. To remove a application server name from the query criteria, clear the application server name check box. To manually add a application server to the list, click and then enter the application server name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of application server names, click .

UI Element (A-Z)	Description
Class	Select whether to **Include** or **Exclude** events for selected JDBC classes. The list of JDBC class names is generated from events collected and stored in the database. To include all JDBC class names in the query criteria, check **All**. To specifically include a JDBC class names, select the JDBC class name check box. To remove JDBC class name from the list, clear the JDBC class name check box.
Database	Select whether to **Include** or **Exclude** events for selected JDBC databases. The database list is generated from events collected and stored in the database; you can add custom entries to the list. To include all databases in the query criteria, check **All**. To include a specific database in the query criteria, select the database name check box. To remove a database name from the query criteria, clear the database name check box. To manually add a database to the list, click and then enter the database name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of database names, click .

UI Element (A-Z)	Description
Database Object	Select whether to **Include** or **Exclude** events that make use of certain database objects. The object list is generated from events collected and stored in the database; you can add custom entries to the list. To include all objects of all schemas in the query criteria, check **All**. To include all objects in a specific schema in the query criteria, select the schema name check box in the Database Schema pane. To include specific objects in a specific schema in the query criteria, clear the schema name check box but keep the database schema name selected. Then select the objects to add to the filter criteria by setting their check boxes. For an example of this, see the Object entry in the WebSphere MQ Agent Criteria section below. To remove a schema from the filter criteria, clear the schema name check box. To manually add a new schema and object to the list, click the **Add new JDBC Object entry** icon and then enter the name schema and object names. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of schema and object names, click . To edit the filter for a selected schema name, click . **Find events that access any of the selected objects.** Matches any SQL statements that contain a reference to one or more of the selected database objects. **Find events that access all selected objects**. Matches any SQL statements that access all of the selected database objects within the same statement. Note that selecting an entire databse in conjunction with this selection would result in matches only if the statement contained all tables within that database.
Method	Select whether to **Include** or **Exclude** events for selected JDBC methods. To include all JDBC method names in the query criteria, check **All**. To specifically include a JDBC method names, select the JDBC method name check box. To remove a JDBC method name from the list, clear the JDBC method name check box.

UI Element (A-Z)	Description
Result Code	Select whether to **Include** or **Exclude** events that have a particular result code. The list of result codes is generated from events collected and stored in the database. To include all result codes in the query criteria, check **All**. To specifically include a result code, select the result code name check box. To remove a result code from the query criteria, clear the result code check box.
SQL Code	Select whether to **Include** or **Exclude** events that have a particular SQL Code. The SQL code list is generated from events collected and stored in the database; you can add custom entries to the list. To include all SQL codes in the query criteria, check **All**. To include a specific SQL code in the query criteria, select the SQL code check box. To remove an SQL code from the query criteria, clear the SQL code check box. To manually add an SQL code to the list, click and then enter the SQL code name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of SQL code names, click . SQL codes used by the JDBC application vary based on the underlying database vendor.

UI Element (A-Z)	Description
SQL State	Select whether to **Include** or **Exclude** events that have a particular SQL state. The SQL state list is generated from events collected and stored in the database; you can add custom entries to the list. To include all SQL states in the query criteria, check **All**. To include a specific SQL state in the query criteria, select the SQL state check box. To remove an SQL state from the query criteria, clear the SQL state check box. To manually add an SQL code to the list, click and then enter the SQL stated name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of SQL states, click . Similar to SQL Code, SQL state values used by the JDBC application vary based on the underlying database vendor.
SQL Statement	Select whether to **Include** or **Exclude** events for a particular SQL statement type. To include all SQL statement types in the query criteria, check **All**. To specifically include an SQL statement type, select the SQL statement type check box. To remove an SQL statement type from the list, clear the SQL statement type name check box.
SQL Statement Type	Select whether to **Include** or **Exclude** events for a particular SQL statement type. To include all SQL statement types in the query criteria, check **All**. Check the box next to the statement type (such as Select, Merge, Insert, and Update) you want to include specifically in the query criteria, or check the box next to Select All SQL Statement Type to include all statement types in the query criteria.

UI Element (A-Z)	Description
Web Application	Select whether to **Include** or **Exclude** events for selected Web applications. The Web application list is generated from events collected and stored in the database; you can add custom entries to the list. To include all Web applications in the query criteria, check **All**. To include a specific Web application in the query criteria, select the Web application name check box. To remove a Web application name from the query criteria, clear the Web application name check box. To manually add a Web application to the list, click and then enter the Web application name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of Web application names, click .

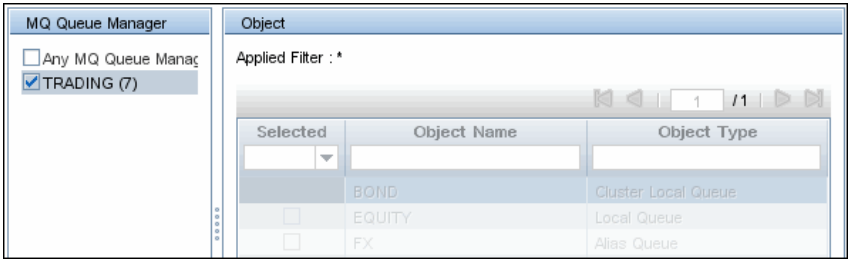
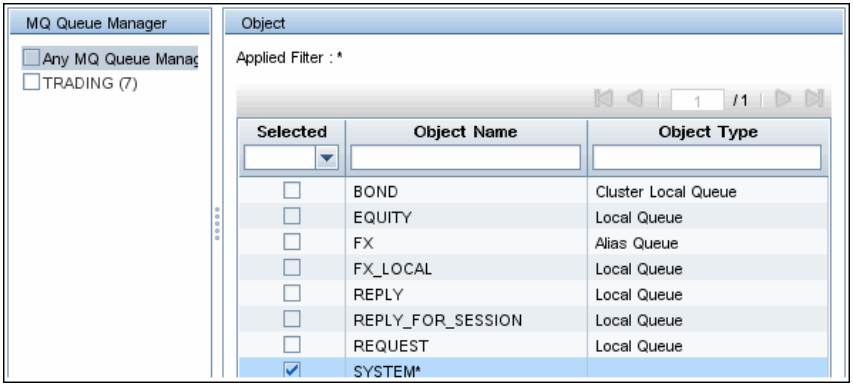
Query Criteria Tab, JMS Agent Criteria

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element (A-Z)	Description
Class	Select whether to **Include** or **Exclude** events for selected JMS classes. To include all JMS class names in the query criteria, check **All**. To specifically include a JMS class, select the JMS class name check box. To remove a JMS class from the query criteria, clear the JMS class name check box.
Connection Name	Select whether to **Include** or **Exclude** events for selected connection names. The connection name list is generated from events collected and stored in the database; you can add custom entries to the list. To include all connection names in the query criteria, check **All**. To include a specific connection name in the query criteria, select the connection name check box. To remove a connection name from the query criteria, clear the connection name check box. To manually add a connection name to the list, click and then enter the connection name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of connection names, click .
Data Size	Select whether to **Include** or **Exclude** events for the selected data size. Select the matching criteria for the selected data size. Choices include less than, less than or equal to, equal to, not equal to, greater than or equal to, or greater than. Enter the desired data size in bytes.
Exception Code	Select whether to **Include** or **Exclude** events for selected exception codes. Check the box next to each exception code you want to include specifically in the query criteria, or check the box next to Select All Exception Code to include all exception codes in the query criteria. The list of exception codes is generated from events collected in the project database.

UI Element (A-Z)	Description
Exception Class Name	Select whether to **Include** or **Exclude** events for selected exception class names. Check the box next to each exception class name you want to include specifically in the query criteria, or check the box next to Select All Exception Class Name to include all exception class names in the query criteria. The list of exception class names is generated from events collected in the project database.
Method	Select whether to **Include** or **Exclude** events for selected JMS methods.The method list is generated from events collected and stored in the database; you can add custom entries to the list. To include all methods in the query criteria, check **All**. To include a specific method in the query criteria, select the method name check box. To remove a method name from the query criteria, clear the method name check box. Check **Do not collect events when no message is available** to avoid collecting JMS Consumer events with no message.
JMS Queue	Select whether to **Include** or **Exclude** events for selected JMS queues. To include all JMS queue names in the query criteria, check **All**. To specifically include a JMS queue, select the JMS queue name check box. To remove a JMS queue from the query criteria, clear the JMS queue name check To manually add a connection name to the list, click and then enter the connection name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of queue names, click .

UI Element (A-Z)	Description
JMS Topic	Select whether to **Include** or **Exclude** events for selected topics. The JMS Topic list is generated from events collected and stored in the database; you can add custom entries to the list. To include all topics in the query criteria, check **All**. To include a specific topic in the query criteria, select the topic name check box. To remove a topic name from the query criteria, clear the topic name check box. To manually add a topic to the list, click and then enter the topic name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of topic names, click .
Web Application	Select whether to **Include** or **Exclude** events for selected Web applications. The Web application list is generated from events collected and stored in the database; you can add custom entries to the list. To include all Web application names in the query criteria, check **All**. To include a specific Web application in the query criteria, select the Web application name check box. To remove a Web application name from the query criteria, clear the Web application name check box. To manually add a Web application to the list, click and then enter the Web application name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of Web application names, click .

UI Element (A-Z)	Description
WebSphere Queue Manager and Queue	Select whether to **Include** or **Exclude** WebSphere queue managers and queues. The list is generated from events collected and stored in the database. You can add custom entries to the list, see instructions below. To specify all queues of all WebSphere queue managers, check **All**. To specify all queues in a specific WebSphere queue manager, select the queue manager name in the MQ Queue Manager pane. Any queues on the MQ Queue Manager is selected—whether it is listed as a known object or not. For example, here all objects in the **TRADING** MQ queue manager are selected to be excluded or included:  Note that an MQ queue manager entry can be selected as well as having its check box set or cleared. If the entry is selected and its check box is cleared, you can specify individual objects of that MQ queue manager. If the entry is selected and its check box is set, all objects of that MQ queue manager are implicitly selected (and the object list is disabled as shown above). To specify a specific object across all MQ queue managers in the filter criteria, select the **Any MQ Queue Managers** entry and then select the object name in the list and set its check box. For example, here all **SYSTEM*** objects are selected to be excluded or included: 

UI Element (A-Z)	Description
Queue Manager and Object (continued)	To search for specific objects that are listed in the Object pane (in the case where there are thousands of them), do any of the following: ➤ Check the check boxes of objects you want to see, and select **True** in the **Selected** dropdown list to see all objects that are checked. You can also select **False** to see all objects that are unchecked. ➤ Enter all or part of the object name (it is not case-sensitive) in the **Object Name** text box and all objects that include that text are displayed. For example, if you enter **reply**, all object names that include REPLY are listed. To remove an MQ queue manger and all of its objects from the filter criteria, clear the MQ queue manager name check box. To refresh the list of MQ queue manager and object names, click . To edit the filter for a selected MQ queue manager name, click . If the object you want to specify is not in the **Object Name** list, manually add it as follows: **1** Click . **2** From the **MQ Queue Managers** drop-down list, choose **Any MQ Queue Manager** to add the object to all queue managers. Or if you know the specific MQ queue manager on which the object resides, specify that name. **3** Enter the object name. An asterisk can be used at the beginning or the end as a wildcard. **4** Click **OK**.

Query Criteria Tab, Servlet Agent Criteria

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element (A-Z)	Description
Application Server	Select whether to **Include** or **Exclude** events for selected application servers. The application server list is generated from events collected and stored in the database; you can add custom entries to the list. To include all application servers in the query criteria, check **All**. To include a specific application server in the query criteria, select the application server name check box. To remove a application server name from the query criteria, clear the application server name check box. To manually add a application server to the list, click and then enter the application server name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of application server names, click .
Data Size	Select whether to **Include** or **Exclude** events for the selected data size. Select the matching criteria for the selected data size. Choices include less than, less than or equal to, equal to, not equal to, greater than or equal to, or greater than. Enter the desired data size in bytes.
Method	Select whether to **Include** or **Exclude** events for selected servlet methods. To include all servlet method names in the query criteria, check **All**. To specifically include a servlet method, select the servlet method name check box. To remove a servlet method from the list, clear the servlet method name check box.
Status Code	Select whether to **Include** or **Exclude** events for selected status codes. To include all status code names in the query criteria, check **All**. To specifically include a status code, select the status code name check box. To remove a status code from the list, clear the status code name check box.

UI Element (A-Z)	Description
URI	Select whether to **Include** or **Exclude** events for selected URIs. The URI list is generated from events collected and stored in the database; you can add custom entries to the list. To include all URIs in the query criteria, check **All**. To include a specific URI in the query criteria, select the URI name check box. To remove a URI name from the query criteria, clear the URI name check box. To manually add a URI to the list, click and then enter the URI name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of URI names, click .
Web Application	Select whether to **Include** or **Exclude** events for selected Web applications. The Web application list is generated from events collected and stored in the database; you can add custom entries to the list. To include all Web applications in the query criteria, check **All**. To include a specific Web application in the query criteria, select the Web application name check box. To remove a Web application name from the query criteria, clear the Web application name check box. To manually add a Web application to the list, click and then enter the Web application name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of Web application names, click .

Query Criteria Tab, User Event Agent Criteria

The User Event Agent criteria includes the following criteria categories: User Event, Generic Event, DataPower, and Tuxedo.

User interface elements are described below:

UI Element	Description
User Event Criteria	
Class	Select whether to **Include** or **Exclude** events for selected classes. The class list is generated from events collected and stored in the database; you can add custom entries to the list. To include all classes in the query criteria, check **All**. To include a specific class in the query criteria, select the class name check box. To remove a class name from the query criteria, clear the class name check box. To manually add a class to the list, click and then enter the class name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of class names, click .
Completion Code	Select whether to **Include** or **Exclude** events for selected User Event method completion codes. To include all completion code in the query criteria, check **All**. To specifically include a completion code, select the completion code check box. To remove a completion code from the list, clear the completion code name check box. Not available for Tuxedo, .NET, NonStop TMF, or Java User Event Agents.
Data Size	Select whether to **Include** or **Exclude** events for the selected data size. Select the matching criteria for the selected data size. Choices include less than, less than or equal to, equal to, not equal to, greater than or equal to, or greater than. Enter the desired data size in bytes.

UI Element	Description
Method	Select whether to **Include** or **Exclude** events for selected methods. The method list is generated from events collected and stored in the database; you can add custom entries to the list. To include all methods in the query criteria, check **All**. To include a specific method in the query criteria, select the method name check box. To remove a method name from the query criteria, clear the method name check box. To manually add a method to the list, click and then enter the method name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of method names, click .
Queue	Select whether to **Include** or **Exclude** events for selected User Event queue strings. The queue list is generated from events collected and stored in the database; you can add custom entries to the list. To include all queues in the query criteria, check **All**. To include a specific queue in the query criteria, select the queue name check box. To remove a queue name from the query criteria, clear the queue name check box. To manually add a queue to the list, click and then enter the queue name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of queue names, click .

UI Element	Description
Queue Space	Select whether to **Include** or **Exclude** events for selected User Event queue space strings. The string list is generated from events collected and stored in the database; you can add custom entries to the list. To include all queue space strings in the query criteria, check **All**. To include a specific queue space string in the query criteria, select the queue space string name check box. To remove a queue space string from the query criteria, clear the queue space string check box. To manually add a queue space string to the list, click and then enter the string. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of queue space string names, click . Not available for .NET, NonStop TMF, or Java User Event Agents.
Service	Select whether to **Include** or **Exclude** events for selected User Event service strings. The service strings list is generated from events collected and stored in the database; you can add custom entries to the list. To include all service strings in the query criteria, check **All**. To include a specific service string in the query criteria, select the service string name check box. To remove a service string name from the query criteria, clear the service string name check box. To manually add a service string to the list, click and then enter the service string name. An asterisk can be used at the beginning or the end as a wildcard. To reload the list of service string names, click .

UI Element	Description
Status	Select whether to **Include** or **Exclude** events for selected User Event status strings. The status list is generated from events collected and stored in the database; you can add custom entries to the list. To include all status strings in the query criteria, check **All**. To include a specific status in the query criteria, select the status string name check box. To remove a status name from the query criteria, clear the status name check box. To manually add a status to the list, click and then enter the status name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of status names, click . Not available for .NET, Tuxedo, or NonStop TMF Agents.
Technology	Select whether to **Include** or **Exclude** events for selected User Event technology strings. The technology list is generated from events collected and stored in the database; you can add custom entries to the list. To include all technology strings in the query criteria, check **All**. To include a specific technology in the query criteria, select the technology string name check box. To remove a technology name from the query criteria, clear the technology name check box. To manually add a User Event technology to the list, click and then enter the type of User Event Agent: .NET, Tuxedo, NonStop-TMF, or Java. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of status names, click .

UI Element	Description
Generic Event	
Custom Object	Select whether to **Include** or **Exclude** events for selected Generic Event custom objects. To include all custom objects in the query criteria, check **All**. To specifically include a custom object, select the completion code check box. To remove a custom object from the query criteria, clear the custom object check box. To manually add a custom object to the list, click and then enter the custom object name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of custom objects, click . Not available for Tuxedo, .NET, NonStop TMF, or Java User Event Agents.
Event Action	Select whether to **Include** or **Exclude** events for selected Generic Event actions. The event action list is generated from events collected and stored in the database; you can add custom entries to the list. To include all classes in the query criteria, check **All**. To include a specific class in the query criteria, select the class name check box. To remove a class name from the query criteria, clear the class name check box. To manually add a technology type to the list, click and then enter the class name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of class names, click .

UI Element	Description
Event Status	Select whether to **Include** or **Exclude** events for selected Generic Event status strings. The status list is generated from events collected and stored in the database; you can add custom entries to the list. To include all status strings in the query criteria, check **All**. To include a specific status in the query criteria, select the status string name check box. To remove a status name from the query criteria, clear the status name check box. To manually add a status to the list, click and then enter the status name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of status names, click .
Technology Type	Select whether to **Include** or **Exclude** events for selected Generic Events technology types. The technology type list is generated from events collected and stored in the database; you can add custom entries to the list. To include all technology types in the query criteria, check **All**. To manually add a technology type to the list, click and then enter the technology type name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of technoogy types, click .
DataPower Criteria	
DataPower Frontend URL	Select whether to **Include** or **Exclude** events that have DataPower Frontend URLs. The DataPower Frontend URL list is generated from events collected and stored in the database; you can add custom entries to the list. To include all events that have DataPower Frontend URLs in the query criteria, check **All**. To manually add a new DataPower Frontend URL to the list, click and then enter the URL. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of DataPower Frontend URLs, click .

UI Element	Description
DataPower Backend URL	Select whether to **Include** or **Exclude** events that have DataPower Backend URLs. The DataPower Backend URL list is generated from events collected and stored in the database; you can add custom entries to the list. To include all events that have DataPower Backend URLs in the query criteria, check **All**. To manually add a new DataPower Backend URL to the list, click and then enter the URL. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of DataPower Backend URLs, click .
Tuxedo Criteria	
Queues	Select whether to **Include** or **Exclude** events for selected Tuxedo queue strings. The queue list is generated from events collected and stored in the database; you can add custom entries to the list. To include all queues in the query criteria, check **All**. To include a specific queue in the query criteria, select the queue name check box. To remove a queue name from the query criteria, clear the queue name check box. To manually add a queue to the list, click and then enter the queue name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of queue names, click .

UI Element	Description
QueueSpace	Select whether to **Include** or **Exclude** events for selected Tuxedo queue space strings. The string list is generated from events collected and stored in the database; you can add custom entries to the list. To include all queue space strings in the query criteria, check **All**. To include a specific queue space string in the query criteria, select the queue space string name check box. To remove a queue space string from the query criteria, clear the queue space string check box. To manually add a queue space string to the list, click and then enter the string. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of queue space string names, click . For the Tuxedo Agent, QueueSpace must match the queue space for which the **tpenqueue/tpdequeue** operation is intended. Not available for .NET, NonStop TMF, or Java User Event Agents.

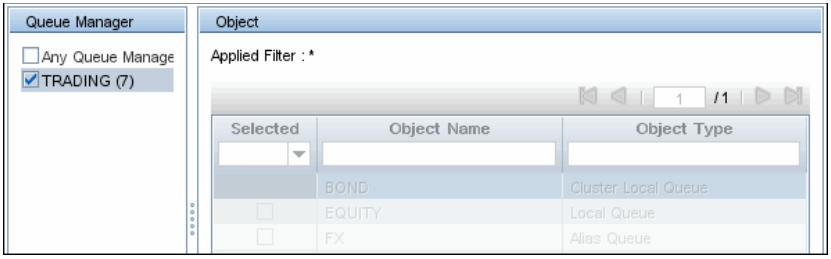
Query Criteria Tab, WebSphere MQ Agent Criteria

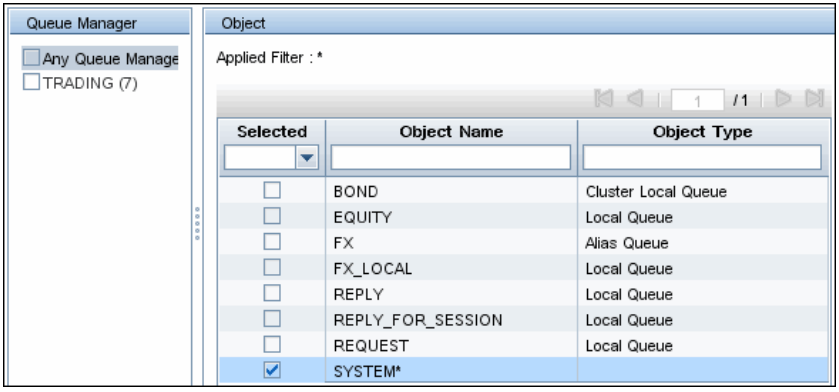
The WebSphere MQ Agent criteria includes the following criteria categories: WebSphere MQ, z/OS, and i5/OS.

User interface elements are described below:

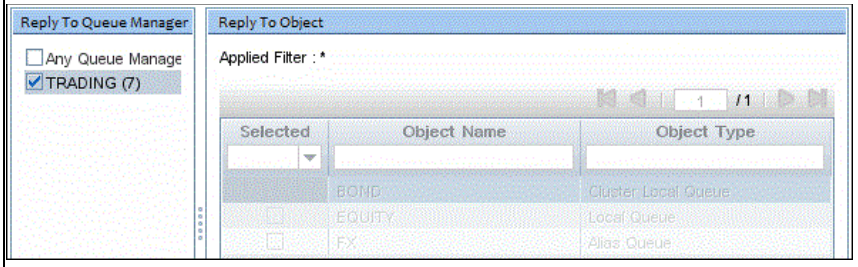
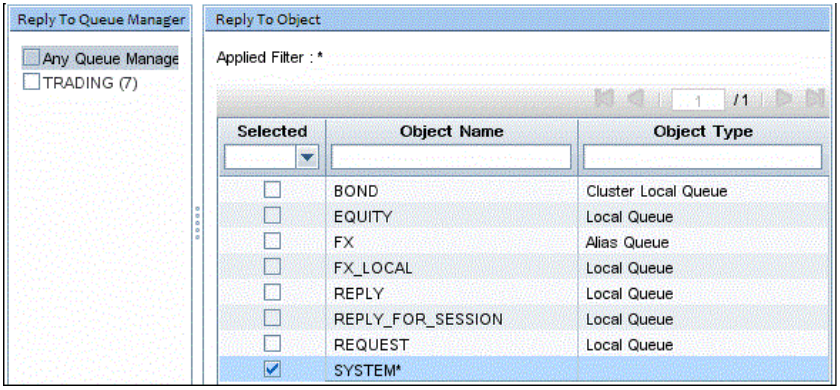
UI Element (A-Z)	Description
WebSphere MQ Criteria	
API Name	Select whether to **Include** or **Exclude** selected WebSphere MQ APIs in the query criteria. The list of class names is generated from events collected and stored in the database. Click **Refresh** to obtain the current set of class names. To include all MQ API names in the query criteria, check **All**. To specifically include an MQ API, select the MQ API name check box. To remove an MQ API from the query criteria, clear the MQ API name check box. Check **Discard MQCMIT in empty transactions** to avoid collecting empty transactions. Empty transactions occur in applications that pool a queue by making MQGET calls in a loop. When there are no inputs, the MQGET calls fail with a MQRC_NO_MSG_AVAILABLE and are typically filtered out by customers in TransactionVision. However, the MQGET may be followed by an MQCMIT. This option provides an easy way to query out these MQCMIT events. Check **Do not send MQDISC exit event** to report only the entry information of each MQDISC call. This improves performance and reduces overhead for the agent. Otherwise, the agent must reconnect and reopen the event queue to send MQDISC exit information, and this information is typically not necessary. Check **Do not send browsing MQGET** if you do not want to collect MQGET calls used to browse messages on the queue.

UI Element (A-Z)	Description
Completion Code	Select whether to **Include** or **Exclude** events for selected WebSphere MQ API completion codes. To include all MQ API completion codes in the query criteria, check **All**. To specifically include an MQ API completion code, select the MQ API completion code check box. To remove an MQ API completion code from the query criteria, clear the MQ API completion code name check box.
Connection Name	The connection name and queue manager are typically the same, except in the case where a queue on one queue manager is opened using a connection (specified with the **hConn** parameter) to a different queue manager. For example, suppose a program opens queue QM1.Q using a connection to queue manager QM1, then later opens QM1.Q using a connection to queue manager QM2. This event shows QM1.Q as the object name, QM1 as the queue manager, and QM2 as the connection name. Select whether to **Include** or **Exclude** events for the selected WebSphere MQ connections. The list of connection names is generated from events collected in the project database. To include all connections in the query criteria, check **All**. To include a specific connection, check the box next to each connection you want to include in the query criteria. To remove a connection from the query criteria, clear the connection check box. To manually add a connection to the list, click and then enter the Connection Name. An asterisk can be used at the beginning or the end as a wildcard. In a customized query, the * character represents zero or more characters, and the double quote (") character represents an escape. To refresh the list of connections and remove all customized connections, click .
Correlation ID	Select whether the Correlation ID contains or does not contain either a binary string or normal string. Click the appropriate radio button to indicate whether the specified correlation ID is in string or binary format and enter the correlation ID. For string format, select the correct **Code Page**.

UI Element (A-Z)	Description
Data Size	Select whether to **Include** or **Exclude** events for the specified data size. Select the matching criteria for the specified data size. Choices include less than, less than or equal to, equal to, not equal to, greater than or equal to, or greater than. Enter the desired data size in bytes.
Message ID	Select whether to **Include** or **Exclude** events for the specified Message ID that contains either a binary string or normal string. Click the appropriate radio button to indicate whether the specified message ID is in string or binary format and enter the message ID. For string format, select the correct Code Page.
Queue Manager and Object	Select whether to **Include** or **Exclude** queue managers and objects. The list is generated from events collected and stored in the database; you can add custom entries to the list. To specify all objects of all queue managers, check **All**. To specify all objects in a specific queue manager, select the queue manager name in the Queue Manager pane. Any object on the Queue Manager is selected—whether it is listed as a known object or not. For example, here all objects in the **TRADING** queue manager are selected to be excluded or included:  Note that a queue manager entry can be selected as well as having its check box set or cleared. If the entry is selected and its check box is cleared, you can specify individual objects of that queue manager. If the entry is selected and its check box is set, all objects of that queue manager are implicitly selected (and the object list is disabled as shown above).

UI Element (A-Z)	Description
Queue Manager and Object (continued)	To specify a specific object across all queue managers in the filter criteria, select the **Any Queue Managers** entry and then select the object name in the list and set its check box. For example, here all **SYSTEM*** objects are selected to be excluded or included:  To search for specific objects that are listed in the Object pane (in the case where there are thousands of them), do any of the following: ➤ Check the check boxes of objects you want to see, and select **True** in the **Selected** dropdown list to see all objects that are checked. You can also select **False** to see all objects that are unchecked. ➤ Enter all or part of the object name (it is not case-sensitive) in the **Object Name** text box and all objects that include that text are displayed. For example, if you enter **reply**, all object names that include REPLY are listed. To remove a queue manger and all of its objects from the filter criteria, clear the queue manager name check box. To refresh the list of queue manager and object names, click . To edit the filter for a selected queue manager name, click .

UI Element (A-Z)	Description
Queue Manager and Object (continued)	If the object you want to specify is not in the **Object Name** list, manually add it as follows: **1** Click . **2** From the **Queue Managers** drop-down list, choose **Any Queue Manager** to add the object to all queue managers. Or if you know the specific queue manager on which the object resides, specify that name. **3** Enter the object name. An asterisk can be used at the beginning or the end as a wildcard. Click **OK**.
Reason Code	Select whether to **Include** or **Exclude** events for selected WebSphere MQ API reason codes. To include all reason codes in the query criteria, check **All**. To specifically include a reason code, select the reason code check box. To remove a reason code from the query criteria, clear the reason code check box. Note: MQGET calls with a MQRC_NO_MSG_AVAILABLE reason code are filtered out by default. When monitoring WebSphere MQ Brokers, make sure to filter out MQGET calls to WebSphere MQ Broker system queues with a reason code of MQRC_NO_MSG_AVAILABLE.
ReplyTo Queue Manager and Object	elect whether to **Include** or **Exclude** reply to queue managers and objects. The list is generated from events collected and stored in the database. You can add custom entries to the list, see instructions below. To specify all objects of all reply to queue managers, check **All**. To specify all objects in a specific reply to queue manager, select the reply to queue manager name in the Reply To Queue Manager pane. Any object on the Reply To Queue Manager is selected—whether it is listed as a known object or not.

UI Element (A-Z)	Description
ReplyTo Queue Manager and Object (continued)	For example, here all objects in the **TRADING** reply to queue manager are selected to be excluded or included:  Note that a reply to queue manager entry can be selected as well as having its check box set or cleared. If the entry is selected and its check box is cleared, you can specify individual objects of that reply to queue manager. If the entry is selected and its check box is set, all objects of that reply to queue manager are implicitly selected (and the reply to object list is disabled as shown above). To specify a specific object across all reply to queue managers in the filter criteria, select the **Any Reply To Queue Managers** entry and then select the object name in the list and set its check box. For example, here all **SYSTEM*** objects are selected to be excluded or included: 

UI Element (A-Z)	Description
ReplyTo Queue Manager and Object **(continued)**	To search for specific objects that are listed in the Object pane (in the case where there are thousands of them), do any of the following: ➤ Check the check boxes of objects you want to see, and select **True** in the **Selected** dropdown list to see all objects that are checked. You can also select **False** to see all objects that are unchecked. ➤ Enter all or part of the object name (it is not case-sensitive) in the **Object Name** text box and all objects that include that text are displayed. For example, if you enter **reply**, all object names that include REPLY are listed. To remove a reply to queue manager and all of its objects from the filter criteria, clear the reply to queue manager name check box. To refresh the list of reply to queue manager and object names, click . To edit the filter for a selected reply to queue manager name, click . If the object you want to specify is not in the **Object Name** list, manually add it as follows: **1** Click . **2** From the **Reply To Queue Managers** drop-down list, choose **Any Reply To Queue Manager** to add the object to all reply to queue managers. Or if you know the specific reply to queue manager on which the object resides, specify that name. **3** Enter the object name. An asterisk can be used at the beginning or the end as a wildcard. **4** Click **OK**.
z/OS	
CICS SYSID	Include or Exclude events for selected CICS SYSIDs. The SYSID list is generated from events collected and stored in the database; you can add custom entries to the list. To include all SYSIDs in the query criteria, check **All**. To include a specific SYSID in the query criteria, select the SYSID check box. To remove a SYSID name from the query criteria, clear the SYSID name check box.

UI Element (A-Z)	Description
CICS SYSID (continued)	To manually add a SYSID to the list, click and then enter the SYSID name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of SYSIDs, click .
CICS Task	Include or Exclude events for selected CICS tasks. The task list is generated from events collected and stored in the database; you can add custom entries to the list. To include all tasks in the query criteria, check **All**. To include a specific task in the query criteria, select the task check box. To remove a task name from the query criteria, clear the task name check box. To manually add a task to the list, click and then enter the task name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of tasks, click .
CICS Transaction	Include or Exclude events for selected CICS transactions. The transaction list is generated from events collected and stored in the database; you can add custom entries to the list. To include all transactions in the query criteria, check **All**. To include a specific transaction in the query criteria, select the transaction check box. To remove a transaction name from the query criteria, clear the transaction name check box. To manually add a transaction to the list, click and then enter the transaction name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of transactions, click .

UI Element (A-Z)	Description
IMS Identifier	Include or Exclude events for selected IMS identifiers. The identifier list is generated from events collected and stored in the database; you can add custom entries to the list. To include all identifiers in the query criteria, check **All**. To include a specific identifier in the query criteria, select the identifier check box. To remove an identifier name from the query criteria, clear the identifier name check box. To manually add an identifier to the list, click and then enter the identifier name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of identifiers, click .
IMS PSB	Include or Exclude events for selected IMS PSB. The PSB list is generated from events collected and stored in the database; you can add custom entries to the list. To include all PSBs in the query criteria, check **All**. To include a specific PSB in the query criteria, select the PSB check box. To remove a PSB name from the query criteria, clear the PSB name check box. To manually add a PSB to the list, click and then enter the PSB name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of PSBs, click .

UI Element (A-Z)	Description
IMS Region Identifier	Include or Exclude events for selected IMS region identifiers. The region identifier list is generated from events collected and stored in the database; you can add custom entries to the list. To include all region identifiers in the query criteria, check **All**. To include a specific region identifier in the query criteria, select the region identifier check box. To remove a region identifier name from the query criteria, clear the region identifier name check box. To manually add a region identifier to the list, click and then enter the region identifier name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of region identifiers, click .
IMS Region Type	Include or Exclude events for selected IMS region types. The region type list is generated from events collected and stored in the database; you can add custom entries to the list. To include all region types in the query criteria, check **All**. To include a specific region type in the query criteria, select the region type check box. To remove a region v name from the query criteria, clear the region type name check box. To manually add a region type to the list, click and then enter the region type name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of region types, click .

UI Element (A-Z)	Description
IMS Transaction	Include or Exclude events for selected IMS transactions. The IMS transaction list is generated from events collected and stored in the database; you can add custom entries to the list. To include all transactions in the query criteria, check **All**. To include a specific transaction in the query criteria, select the transaction check box. To remove a transaction name from the query criteria, clear the transaction name check box. To manually add a transaction to the list, click and then enter the transaction name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of transactions, click .
Job Name	Include or Exclude events for selected z/OS job names. The job name list is generated from events collected and stored in the database; you can add custom entries to the list. To include all job names in the query criteria, check **All**. To include a specific job name in the query criteria, select the job name check box. To remove a job name from the query criteria, clear the job name check box. To manually add a job name to the list, click and then enter the job name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of job names, click .

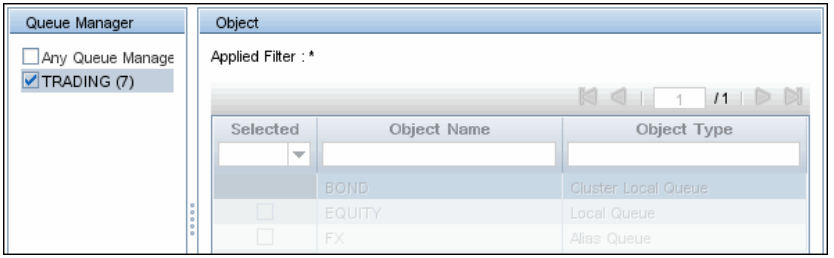
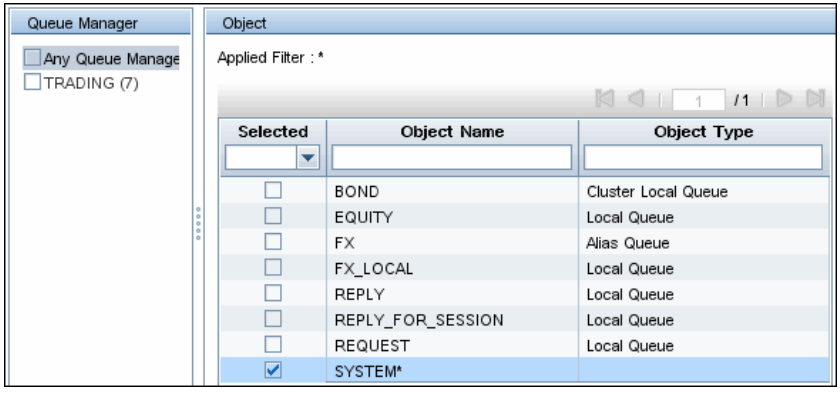
UI Element (A-Z)	Description
Job Steps	Include or Exclude events for selected job steps. The job step list is generated from events collected and stored in the database; you can add custom entries to the list. To include all job steps in the query criteria, check **All**. To include a specific job step in the query criteria, select the job step check box. To remove a job step from the query criteria, clear the job step check box. To manually add a job step to the list, click and then enter the job step name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of job steps, click .
i5/OS	
Job Name	Include or Exclude events for selected i5/OS job names. The job name list is generated from events collected and stored in the database; you can add custom entries to the list. To include all job names in the query criteria, check **All**. To include a specific job name in the query criteria, select the job name check box. To remove a job name from the query criteria, clear the job name check box. To manually add a job name to the list, click and then enter the job name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of job names, click .

Query Criteria Tab, WebSphere MQ IMS Bridge Agent Criteria

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element (A-Z)	Description
Bridge API	**Include** or **Exclude** events for selected MQIMS Bridge APIs. To include all MQIMS Bridge APIs in the query criteria, check **All**. To specifically include a MQIMS Bridge API, select the MQIMS bridge API check box. For more information about the MQSeries-IMS Bridge Agent, see the *HP TransactionVision Deployment Guide* PDF.
Bridge connection name	**Include** or **Exclude** events for selected MQIMS Bridge connection names. The connection name list is generated from events collected and stored in the database; you can add custom entries to the list. To include all connection names in the query criteria, check **All**. To include a specific connection name in the query criteria, select the connection name check box. To remove a connection name from the query criteria, clear the connection name check box. To manually add a connection name to the list, click and then enter the connection name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of connection names, click .
Bridge correlation ID	Select whether the MQIMS Bridge correlation ID contains or does not contain either a binary string or normal string. Click the appropriate radio button to indicate whether the specified correlation ID is in **Normal Strin**g or **Binary String** format and enter the correlation ID. For string format, select the correct **Code Page**.

UI Element (A-Z)	Description
Bridge IMS Transaction	**Include** or **Exclude** events for selected z/OS CICS transactions. The transaction list is generated from events collected and stored in the database; you can add custom entries to the list. To include all transactions in the query criteria, check **All**. To include a specific transaction in the query criteria, select the transaction check box. To remove a transaction name from the query criteria, clear the transaction name check box. To manually add a transaction to the list, click and then enter the transaction name. An asterisk can be used at the beginning or the end as a wildcard. To refresh the list of transaction names, click .
Bridge Message ID	Select whether the MQIMS Bridge Message ID contains or does not contain either a binary string or normal string. Click the appropriate radio button to indicate whether the specified message ID is in **Normal String** or **Binary String** format and enter the message ID. For string format, select the correct **Code Page**.
Bridge MQ Object	Select whether to **Include** or **Exclude** Brdige MQ queue managers and objects. The list is generated from events collected and stored in the database; you can add custom entries to the list. To specify all MQ objects of all queue managers, check **All**. To specify all MQ objects in a specific queue manager, select the queue manager name in the Queue Manager pane. Any object on the Queue Manager is selected—whether it is listed as a known object or not.

UI Element (A-Z)	Description
Bridge MQ Object (continued)	For example, here all objects in the **TRADING** queue manager are selected to be excluded or included:  Note that a queue manager entry can be selected as well as having its check box set or cleared. If the entry is selected and its check box is cleared, you can specify individual objects of that queue manager. If the entry is selected and its check box is set, all objects of that queue manager are implicitly selected (and the object list is disabled as shown above). Note that a queue manager entry can be selected as well as having its check box set or cleared. If the entry is selected and its check box is cleared, you can specify individual objects of that queue manager. If the entry is selected and its check box is set, all objects of that queue manager are implicitly selected (and the object list is disabled as shown above). To specify a specific object across all queue managers in the filter criteria, select the **Any Queue Managers** entry and then select the object name in the list and set its check box. For example, here all **SYSTEM*** objects are selected to be excluded or included: 

UI Element (A-Z)	Description
Bridge MQ Object (continued)	To search for specific objects that are listed in the Object pane (in the case where there are thousands of them), do any of the following: ➤ Check the check boxes of objects you want to see, and select **True** in the **Selected** dropdown list to see all objects that are checked. You can also select **False** to see all objects that are unchecked. ➤ Enter all or part of the object name (it is not case-sensitive) in the **Object Name** text box and all objects that include that text are displayed. For example, if you enter **reply**, all object names that include REPLY are listed. To remove a queue manger and all of its objects from the filter criteria, clear the queue manager name check box. To refresh the list of queue manager and object names, click . To edit the filter for a selected queue manager name, click . If the object you want to specify is not in the **Object Name** list, manually add it as follows: **1** Click . **2** From the **Queue Managers** drop-down list, choose **Any Queue Manager** to add the object to all queue managers. Or if you know the specific queue manager on which the object resides, specify that name. **3** Enter the object name. An asterisk can be used at the beginning or the end as a wildcard. **4** Click **OK**.

Query Page, General Settings Tab

The General Settings tab enables you to view the query name and description.

To access	**1** Select **Admin> Transaction Management > Configuration.** **2** (left pane) Select **Queries** > <query name> **3** (right pane) Select the **General Settings** tab.
Relevant tasks	"How to Add a Query" on page 262

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
Query Description	Enter text that helps administrators know the purpose of the query.
Query Name	Enter a name for the query. The name must be unique to the TransactionVision deployment environment.

7

Jobs

This chapter includes:

Concepts

- Built-in Jobs on page 316
- Custom Jobs on page 317

Tasks

- How to Modify the Built-in Jobs on page 318
- How to Add a Custom Job on page 318

Reference

- Jobs User Interface on page 320

Concepts

Built-in Jobs

Transaction Management uses these built-in jobs to manage data synchronization between various data repositories. It is not recommend to change the settings in these jobs, but they can be used for debugging purposes.

TransactionVision provides the following built-in jobs:

Job Name	Purpose
RTSM Population	Publishes the following to the Run-Time Service Model (RTSM): ➤ New business transaction CIs ➤ CIs corresponding to the infrastructure elements discovered by TransactionVision agents ➤ Any dependency relationship between business transactions CIs and corresponding infrastructure elements
RTSM Update	Updates infrastructure CIs corresponding to current business transactions, to prevent the aging out of these CIs in the RTSM.
BSM Aggregate TV Data Sample	Creates and delivers aggregate business transaction data to the BSM Profile database. By default, the sample is resent every 5 minutes. The sample has a predefined format. See *Reports*.
TV Alert Data Sample	Generates alerts on incoming business transactions and sends them to BSM Service Health.

Custom Jobs

Transaction Management can also run custom jobs. Custom jobs are created by writing Java classes that use the JobBean basic class and implement the IJob interface.

Tasks

How to Modify the Built-in Jobs

To modify the BSM Aggregate TV Data Sample Job:

1 Select **Admin** > **Transaction Management.**

2 Select **Configuration** > **Transaction Vision** > **Jobs** in the left pane.

3 Right-click **BSM Aggregate TV Data Sample** and select **Edit Job**.

4 The **Edit Job** dialog appears. For details about this interface, see "Edit Job Dialog" on page 326.

To modify the RTSM Population or Update Job:

1 Select **Admin** > **Transaction Management.**

2 Select **Configuration** > **Transaction Vision** > **Jobs** in the left pane.

3 Right-click **RTSM Population or RTSM Update and** select **Edit Job**.

4 The **Edit Job** dialog appears. For details about this interface, see "Edit Job Dialog" on page 326.

How to Add a Custom Job

This task describes how to create a new custom job for the TransactionVision deployment environment.

This task includes the following steps:

➤ "Prerequisite" on page 319

➤ "Fill out the Add Job dialog" on page 319

➤ "Start the Job if necessary" on page 319

1 Prerequisite

Write and test the Java class that represents the task you want the job to do. For an example of a custom job Java class, see the *HP TransactionVision Advanced Customization Guide* PDF.

2 Fill out the Add Job dialog

Note: If class files are part of a package, they must be placed in that corresponding directory. For example, a Java class named com.mycompany.TestClass would need to be placed in the com/mycompany directory within WEB-INF/classes.

---

a Select **Admin** > **Transaction Management**.

b Select **Configuration** in the left pane.

c Click New and then select **New Job**.

d The **Add Job** dialog appears. For details about this interface, see "Add Job Dialog" on page 325.

3 Start the Job if necessary

The job name and description appear on the Job Status page.

Reference

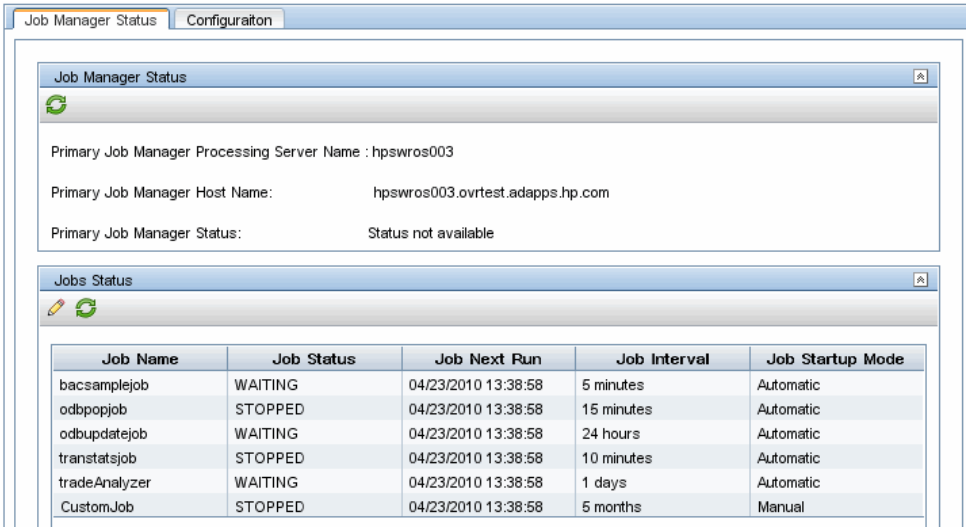
Jobs User Interface

This section describes:

- Job Summary Page, Job Manager Status tab on page 321
- Job Status Page on page 323
- Add Job Dialog on page 325
- Edit Job Dialog on page 326

Job Summary Page, Job Manager Status tab

The Job Manager Status page enables you to view the status of built-in and custom jobs. The following is an example of the Job Manager Status tab.



To access	**1** **Admin** > **Transaction Management** > **Configuration** **2** (left pane) Click the **Jobs** folder. **3** (right pane) Click the **Job Manager Status** tab.

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
Job Manager Status	**Primary Job Manager Processing Server Name.** The Processing Server on which the Job Manager is running. **Primary Job Manager Host Name.** The host name associated with the Processing Server specified above. **Primary Job Manager Status.** The status of the primary Job Manager: Running. Not running. Click **Refresh** to refresh the Job Manager status.
Jobs Status	For each job: **Job Name**. Name of the job. **Job Status**. Status of the job: **Waiting**, **Running**, **Disabled**, **Stopped**, **Starting**, **Quiescing**. **Job Next Run**. Next time the job is scheduled to run. **Job Interval**. Frequency with which the job runs. **Job Startup Mode**. How the job is started. Click **Refresh** to refresh the job list. Click **Edit** to change the job configuration for the selected job.

Job Status Page

The Job Status page enables you to view the status of the selected job, as well as start and stop it.

To access	**1** **Admin > Transaction Management > Configuration** **2** (left pane) Expand the **Jobs** folder. **3** (left pane) Click the <job name> tab.
Relevant tasks	"How to Modify the Built-in Jobs" on page 318
See also	"Exporting TransactionVision Data to BSM" on page 363

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
	Start the job. Starting a job initializes and activates it. If it's next scheduled time has already passed, it runs immediately; otherwise, it is placed in a waiting state until its scheduled time arrives.
	Stop the job. If the job is currently running, a cancel command is first issued, followed by commands to stop and shut it down.
	Force stop the job. This button is only enabled if the selected job is in Quiescing mode.
	Cancel the current job run without stopping the job. The job changes to a waiting state until the next scheduled run time.
	Run job now rather than continuing to wait for the scheduled time.
	Refresh job list.
	Edit job. Allows you to edit a job's configuration, see "Edit Job Dialog" on page 326.

UI Element	Description
Job Status	The status and next run time for the job. Possible status values are: **Running**. The job is in the process of executing the task it is programmed to do. **Waiting**. The job is active and waiting for its scheduled run time to arrive. **Quiescing**. The job is in the process of shutting down, performing any specific cleanup or shutdown processes applicable. A job initially transfers to this state when it is stopped, then automatically moves from this state to Stopped when the shutdown or cleanup process is complete. When in Quiescing mode, use the Force Stop button to end a job immediately without completing pending tasks. **Stopped**. The job is not active, is not loaded into memory, and does not perform any tasks until started with the **Start** button,
Job Logs	History information showing when the job has been initialized, run, and completed.
Next Run	The next scheduled run of the job.

Add Job Dialog

The Add Job dialog box enables you to create a new custom job.

To access	**1** Select **Admin** > **Transaction Management** > **Configuration** **2** Right-click **Jobs**, and select **Add Jo**b in the left pane.
Relevant tasks	"How to Add a Custom Job" on page 318.

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
Class name	The name of the Java class. This class must implement the IJob interface. Full classpath: **com.bristol.tvision.job.IJob**
Description	Description for the new job.
Name	Name for the new job.
Interval	Enter the frequency with which the job should be repeated. The interval may be in minutes, hours, days, or months.
Parameters	Any startup parameters needed for this job.
Startup mode	The startup mode for the job. Options are: **Automatic**: the job is automatically initialized and started when the web server starts, or when a user clicks Start All. **Manual**: the job is only started when the job is selected and started **Disabled**: the job cannot be run until its startup mode is changed to Automatic or Manual. Disabling a job is useful if you have an automatically starting job that you want to temporarily turn off so it doesn't start when the system comes up or when starting all jobs.

Edit Job Dialog

The Edit Job dialog box enables you to modify a job's configuration.

To access	**1** Select **Admin** > **Transaction Management** > **Configuration** **2** In the right pane, right-click **TransactionVision** > **Jobs** and select **Edit Jo**b.
Relevant tasks	"How to Modify the Built-in Jobs" on page 318.

The Edit Job dialog includes different fields, depending on which job you are editing

➤ "Edit Job Dialog for the BSM Aggregate TV Data Sample Job" on page 326

➤ "Edit Job Dialog for the RTSM Population and Update Jobs" on page 328

Edit Job Dialog for the BSM Aggregate TV Data Sample Job

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
Name	Name of the job. Do not change this name.
Description	Description of the job.
Class name	The predefined job name.
Parameters	The startup parameters for this job. You cannot change these parameters.

UI Element	Description
Startup Mode	The startup mode for the job. Options are: **Automatic**. The job is automatically initialized and started when the Job Manager starts, or when the Start All button is clicked. **Manual**. The job is only started when the job is selected and started. **Disabled**. The job cannot be run until its startup mode is changed to Automatic or Manual. Disabling a job is useful if you have an automatically starting job that you want to temporarily turn off so it doesn't start when the system comes up or when starting all jobs.
Interval	Enter the frequency with which the job should be repeated. The interval may be in minutes, hours, days, or months.
Time Slice Delta (sec)	Interval at which the sample is collected. **Default value**: 300 For instance, every 300 seconds the sampling period starts and continues for the time specified by the Sample Delta value (default is 360 seconds).
In-Process Aging (sec)	Affects the Time Slice Delta. By default, all of the in-process transactions are processed for the sample. This option, if set, allows only transactions that are still in-process at this adjusted time to be processed for the sample: Endtime is after (Current Time - In Process Aging value)
Sample Aging Period (sec)	Affects the Sample Delta. By default, all completed transactions that end in the sample period are processed. However, if this option is set, the sample period is adjusted as follows: CurrentTime - Sample Aging Period value
Sample Update Aging Period (min)	Determines which sample records are deleted from the database. For example, a value of 1440 means any sample records which are 2 days old are deleted. **Default value**: 300

UI Element	Description
Sample Delta (sec)	Time duration for which samples are gathered. **Default value**: 360
Debug Logging	Enable debug logging for jobs.

Edit Job Dialog for the RTSM Population and Update Jobs

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
Class name	The predefined job name.
Description	Description of the job.
Interval	Enter the frequency with which the job should be repeated. The interval may be in minutes, hours, days, or months.
Name	Name of the job. Do not change this name.

UI Element	Description
Parameters	The startup parameters for this job: **-clearDirtyBit.** Mark all CIs as published even when they have publishing errors. This may be used to stop repetitive errors on some CIs that are missing required information to be published to RTSM. **-debug**. Set this parameter to true to enable debug statements. **-domain** <domain>. Set this parameter to set the corresponding value as the probe_domain attribute for populating hosts in RTSM (default value is **DefaultDomain**). **-force.** Force all CIs to be republished even if they have already been marked as published. **-populate_temp_queues** (for RTSM Population). **-update_temp_queues** (for RTMS Update). Populate/ update temporary queue CIs. By default, temporary WebSphere MQ and JMS queues are not populated to RTSM. **Note**: If the **-domain** parameter is used, it should be set to the same value as the RTSM Update job.
Startup mode	The startup mode for the job. Options are: **Automatic**. The job is automatically initialized and started when the Job Manager starts, or when a user clicks Start All. **Manual**. The job is only started when the job is selected and started **Disabled**. The job cannot be run until its startup mode is changed to Automatic or Manual. Disabling a job is useful if you have an automatically starting job that you want to temporarily turn off so it doesn't start when the system comes up or when starting all jobs.

8

Advanced Configuration

This chapter includes:

Concepts

➤ About Advanced Configuration on page 332

Tasks

➤ How to Modify the XDM Files on page 334

➤ How to Modify the Built-In SonicMQ Broker Settings on page 334

➤ How to Modify the Maximum File Size for User Data Display on page 336

➤ How to Enable Session Tracking on page 336

Reference

➤ Advanced Configuration User Interface on page 340

Concepts

About Advanced Configuration

Advanced configuration refers to modifications you typically do not need to make as an administrator of the TransactionVision deployment environment. Advanced configuration can be divided into two categories:

➤ "Customizing the TransactionVision Database Schema" on page 332

➤ "Customizing System Information" on page 333

Customizing the TransactionVision Database Schema

In some cases information in the message data that is needed for analysis is not captured by default. In these cases, you can extract fields from the message data and map them to database columns where they can be stored. Once they are stored in the database, they are available for analysis and use in the Transaction Management reports they are visible in any reports dealing with instance level detail, for example the Transaction Tracking or Transaction Detail report.

Before these fields can be written to a database column by the Analyzer, they need to be extracted from the message and converted to XML (if not already in the XML format). For information about how to extract data from a message, see chapter 3 in the *HP TransactionVision Advanced Customization Guide* PDF.

The TransactionVision database schema also must be customized to accept the new data. The schema is made extensible through the XML to Database Mapping (XDM) files. As message data specific columns are added to the database, the XDM files can be updated to describe the new schema. Therefore, XML to Database mapping serves multiple purposes:

➤ To describe the layout of the database schema tables. Each Analyzer has its own unique schema in the database, but all schemas must have the same table layout.

- To describe to the Analyzer the fields that are to be extracted from the XML event data and stored in event lookup tables for fast searching and retrieval.
- To describe to the Analyzer the fields that are to be extracted from the transaction XML document and stored in the transaction lookup tables.
- To describe the database schema to the query services for use in Transaction Management reports and topologies.

Customizing System Information

Some properties can require customization. These properties affect the entire deployment.

The properties are in the following groups (for descriptions of each property, see "Configuration Tab, General Tab" on page 341):

- DataCollectionFilter
- DefaultUserColumnData
- FixedXPaths
- PresentationQuery
- Server
- SystemModelDefinition
- UIBeans
- UIProperties

Tasks

How to Modify the XDM Files

This task describes how to modify the default XDM files. Modifying these files changes the table structure used to store data of the corresponding type. Therefore, once you modify a default or existing XDM file, previous versions of the XDM file cannot be used.

When you modify XDM files related to transactions, any preferences stored for the Transaction Tracking Report are deleted.

Caution: Changing the table structure requires that the tables be recreated (this occurs automatically). Existing data in the tables will be lost.

To modify an XDM file:

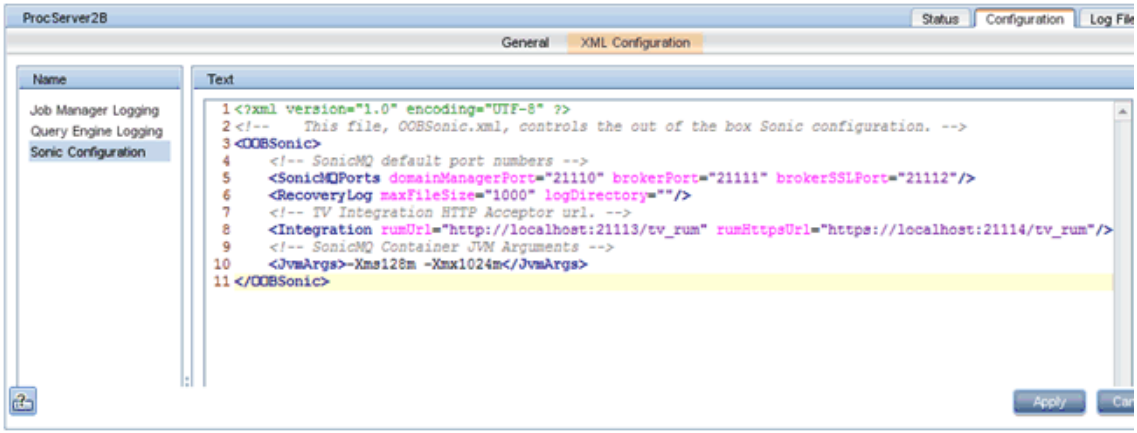
1. Determine which XDM file you need to modify.
2. Select **Admin** > **Transaction Management.**
3. (left pane) Click the **TransactionVision** top-level folder.
4. (right pane) Select **Configuration** > **XDM** tab.
5. Make the needed changes and click **Apply**.

For information about this tab, see "Configuration Tab, XDM Tab" on page 342.

How to Modify the Built-In SonicMQ Broker Settings

This task describes how to modify the configuration of the built-in SonicMQ Broker in the Transaction Management Admin user interface. For example, you might want to change the size of the **RecoveryLog maxFileSize** or the port used by the SonicMQ Broker.

Following is the default TransactionVision XML Sonic configuration:



Notice that the JVM heap size has been set to 1024 by default to make sure that SonicMQ Message Broker has enough memory for a higher load. You can change the JVM heap size here or on the container within the SonicMQ Administration Console. Modifying this Sonic configuration overrides and sets the value within the SonicMQ Administration Console.

To modify the Sonic Configuration:

1. Select **Admin** > **Transaction Management**.
2. Select **Processing Servers** > and the <processing_server> in the left pane for which you want to edit the Sonic Configuration.
3. Select **Configuration** > **XML Configuration** tab (right pane).
4. Choose **Sonic Configuration**.

   Make the needed changes and click **Apply**.

Note: Changes to the recovery log size or location need to also be made to the SonicMQ Broker itself through the SonicMQ Management Console.

---

5 Restart the TransactionVision Processing Servers by using <**TVISION_HOME**>**bin****SupervisorStop** and **SupervisorStart** on each one.

How to Modify the Maximum File Size for User Data Display

This task describes how to change the maximum file size for which user data is displayed in tree format in the Event Details report. Displaying user data in tree format improves readability, but takes more time to build before the report can be displayed.

By default, the maximum file size for which XML data is displayed in tree format is 50000 bytes.

1 Select **Admin** > **Transaction Management.**

2 (left pane) Click the **TransactionVision** top-level folder.

3 (right pane) Select **Configuration** > **General** tab.

4 Choose **uiproperties**.

5 Locate the **eventListXmlPayloadMax** entry.

6 Make the needed change to the setting and click **Apply**.

7 Restart the TransactionVision Processing Servers by using <**TVISION_HOME**>**bin****SupervisorStop** and **SupervisorStart** on each one.

How to Enable Session Tracking

This task describes how to enable analysis and classification of Session ID with a servlet's transaction, to allow classifying and analyzing Session ID data for the Web Session report.

To enable session tracking:

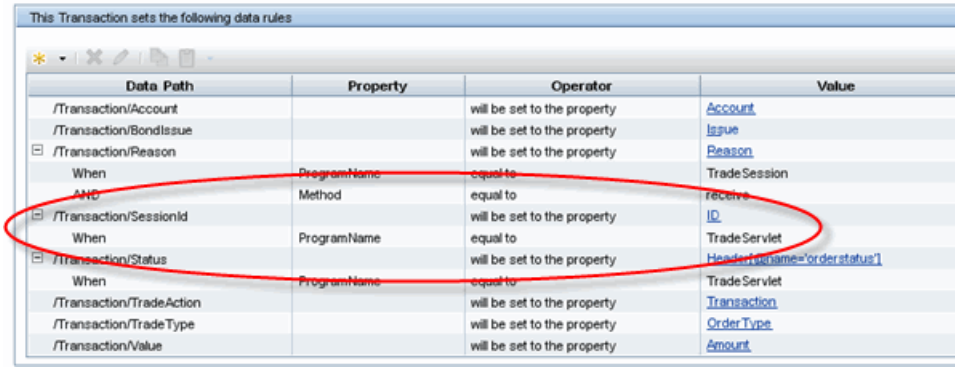
1 Navigate to the **transaction.xdm** file as described in "Configuration Tab, XDM Tab" on page 342.

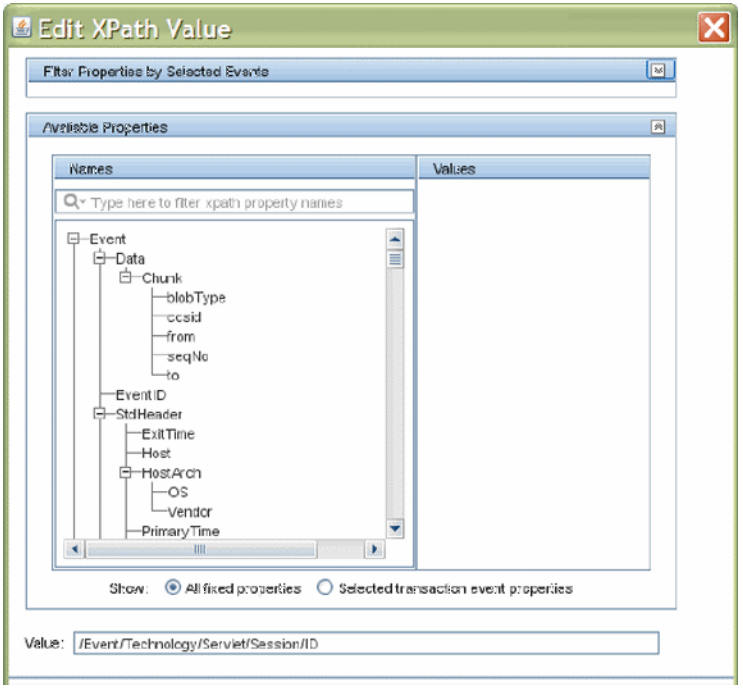
2 To the bottom of the file, add the following lines:

```
<Column name="sessionid" type="VARCHAR" size="32"
description="SessionId">
<Path>/Transaction/SessionId</Path>
</Column>
```

After saving the file changes, a warning displays indicating that the transaction model will be changed. Transaction data in the previous model will no longer be available.

3 Create a data rule in your transaction tracing rule that populates the HTTP Session ID from your event data (/Event/Technology/Servlet/Session/ID).





4 Start collecting events.

The transaction class using this rule should now be setting the Session ID. This can be verified by going to the Transaction Tracking report and seeing transactions listed in the Session ID column or by running the Web Session report.

Reference

Advanced Configuration User Interface

This section describes:

- Configuration Tab, General Tab on page 341
- Configuration Tab, XDM Tab on page 342
- Configuration Tab, BSM Settings Tab on page 343

Configuration Tab, General Tab

This tab enables you to customize various system-wide property settings.

To access	**1** Select **Admin** > **Transaction Management.** **2** (left pane) Click the **Configuration** tab. **3** Click the **TransactionVision** top-level folder. **4** (right pane) Click the **Configuration** tab. **5** Click the **General** tab.
Relevant tasks	"How to Enable Session Tracking" on page 336
See also	"About Advanced Configuration" on page 332

Name

The Name column lists the editable properties files.

User interface elements are described below:

UI Element	Description
DataCollentionFilter	Configuration file template used for creating new Data Collection Filters, and is not meant to be customized.
DefaultuserColumnDatabase	Allows to add custom XDM columns for display in the Event Analysis View. For more details, see 6.2 "Adding Columns to the Event List View" in the *HP TransactionVision Advanced Customization Guide* PDF.
FixedXPaths	Configuration file that lists all XPaths available in the XPath - picker in the user interface (such as when editing match conditions). This file is not meant to be customized.
PresentationQuery	Allows you to add query conditions for custom XDM columns when defining a new query.This **PresentationQuery** default file contains detailed commented descriptions for each option. For more details, see 6.1. A"dding Query Pages" in the *HP TransactionVision Advanced Customization Guide* PDF.

UI Element	Description
Server	Allows you to filter the system model notification, by enabling/disabling notification and setting object type filters in the **Server.properties** file.
SystemModelDefinition	Allows you to define custom system model object types for user events. For more details, see 4.7.1. " User Events" in the *HP TransactionVision Advanced Customization Guide* PDF.
UIBeans	Internal file not to be customized.
uiproperties	Defines several global properties that are relevant to the Transaction Management user interface. This **uiproperties** default file contains detailed commented descriptions for each option.

Text

This window contains the XDM script for each property file.

Configuration Tab, XDM Tab

This tab enables you to customize the XDM files that represent the TransactionVision database tables.

You can add a new table but you cannot remove any of the built-in tables.

To access	**1** Select **Admin** > **Transaction Management.** **2** (left pane) Click the **Configuration** tab. **3** Click the **TransactionVision** top-level folder. **4** (right pane) Click the **Configuration** tab. **5** Click the **XDM** tab.
Important information	Editing the XDM changes the database table and therefore it becomes incompatible with any previous version.

Relevant tasks	"How to Modify the XDM Files" on page 334
See also	"About Advanced Configuration" on page 332 For information about how to modify the built-in XDM files, see the *HP TransactionVision Advanced Customization Guide* PDF.

User interface elements are described below:

UI Element (A-Z)	Description
	Add a new custom XDM entry.
	Delete the selected XDM entry. Only custom XDM entries can be deleted.
Name	The XDM file name. This is the XML to Database Mapping (XDM) files.
Text	The content of the XDM file in XML format.

Configuration Tab, BSM Settings Tab

This tab enables you to configure access to the BSM Gateway Server, the Run-Time Service Model (RTSM) and the RUM engine.

To access	**1** Select **Admin** > **Transaction Management.** **2** (left pane) Click the **Configuration** tab. **3** Click the **TransactionVision** top-level folder. **4** (right pane) Click the **Configuration** tab. **5** Click the **BSM Settings** tab.
Important information	➤ The settings on this tab are initialized when the TransactionVision application is deployed through the **Admin** > **Platform** > **Setup and Maintenance** pages. ➤ To restore the default settings, click **Reset to Default**.

User interface elements are described below:

UI Element (A-Z)	Description
BSM User Name	The user name used by the TransactionVision Processing Server when communicating with the BSM Gateway Server. In some cases, such as in an LDAP environment, the default user name of **admin** may not be valid. In those cases specify a valid user name here. **Default value**: admin
Customer Name	Always **Default Client**.
Enable Basic Authentication	Use basic authentication (in addition to your current LW-SSO configuration) for communication between the BSM Gateway server and all TransactionVision Processing Servers. **Default value**: off If checked, also enter: **Basic Authentication User Name**. The user name to access the BSM Gateway server with basic authentication. **Basic Authentication Password**. The password associated with the user name above.
Host	Enter the fully-qualified domain name of the BSM Gateway Server. This setting is automatically set based on a typical deployment and setup. In special deployments or where you have changed BSM settings you need to specify this field.
Port	Enter the listening port on the BSM Gateway server for all TransactionVision Processing Servers. For details about ports on the BSM Gateway, see "Port Usage" in *Platform Administration*. **Default value**: 80

UI Element (A-Z)	Description
Protocol	Choose the protocol used for communication between the BSM Gateway server and all TransactionVision Processing Servers. Options are: HTTP or HTTPS. **Default value**: HTTP A value of HTTPS requires that components be enabled for SSL communication. See "Using SSL with TransactionVision" in the *HP Business Service Management Hardening Guide* PDF
RTSM User Name	You can specify access credentials through which all RTSM access from TransactionVision occurs. **Default value**: TVISION
RTSM Password	The password associated with the user name above. **Default value**: Automatically set
RUM Publish Port	Enter the RUM publish port. Must correspond to the Port setting in the TransactionVision Connection page of the RUM Web Console. **Default value**: 21113
RUM Publish Protocol	Choose the protocol used for communication between a RUM Engine and a TransactionVision Analyzer. Must correspond to the Protocol setting in the TransactionVision Connection page of the RUM Web Console. **Default value**: HTTP A value of HTTPS requires that all components be enabled for SSL communication. See "Using SSL with TransactionVision" in the *HP Business Service Management Hardening Guide* PDF and "Security" in the *HP TransactionVision Deployment Guide* PDF.

9

Working with Other Applications in BSM

This chapter includes:

Concepts

- Service Health on page 348
- MyBSM on page 353
- Service Level Management on page 353
- End User Management on page 355
- HP Diagnostics on page 357
- Business Process Insight on page 358
- TransactionVision Data in the BSM Profile Database on page 361
- TransactionVision Data in the RTSM on page 363
- Exporting TransactionVision Data to BSM on page 363

Tasks

- How to Verify or Modify Communication Between BSM and TransactionVision on page 364
- How to Enable Communication Between RUM and TransactionVision on page 364
- How to Enable Communication Between BPI and TransactionVision on page 365

Troubleshooting and Limitations on page 365

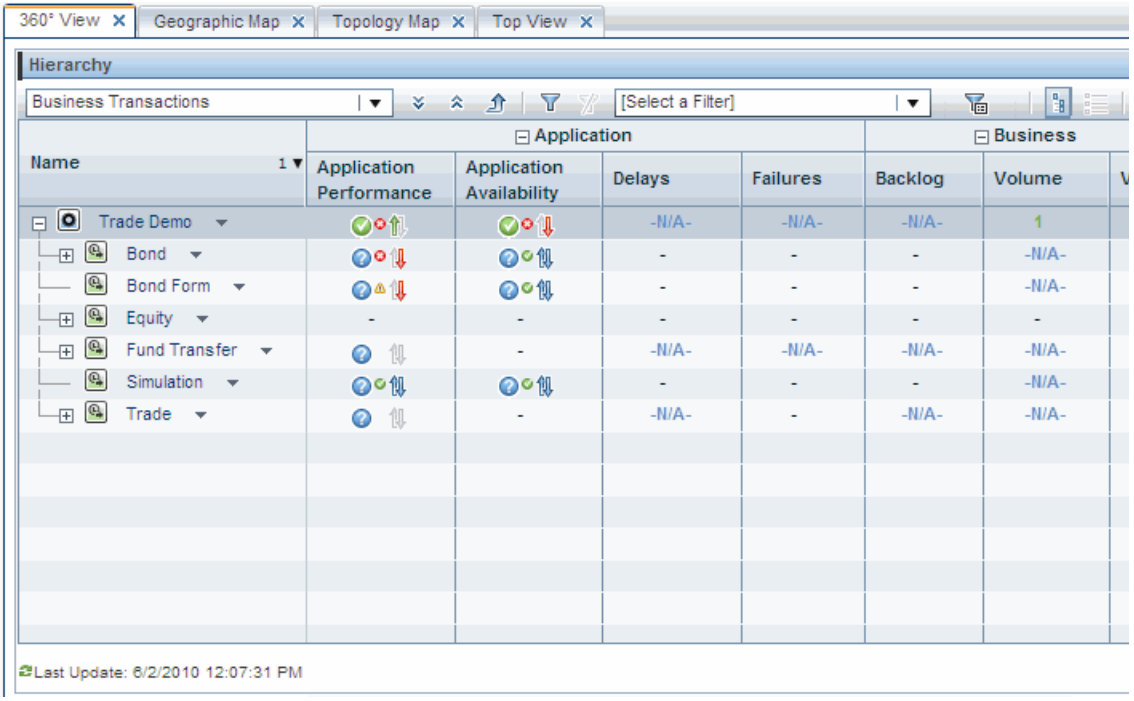
Concepts

Service Health

Data Sent From TransactionVision For Use With Service Health

The TransactionVision Processing Servers automatically send their business transaction CI data to BSM. This data is used to calculate some Health Indicators and KPIs for use with the Service Health application.

For example, the KPIs for a business transaction display on the 360 View:



Each Business Transaction CI has default KPIs and HIs (health indicators) that define what is monitored for the transaction.

The following table shows the relationship among the predefined Business Transaction CI KPIs, Health Indicators and TransactionVision data.

KPI	Contributing Health Indicators	Supporting TransactionVision Data in the Sample
Application Performance	Backend Transaction Average Duration or	The average backend Response Time of all instances of the specified Business Transaction CI type that have: ➤ Completed in the specified time range ➤ Any transaction state See "Response Time" on page 642.
	End User Transaction Average Duration or	The average end user Response Time of all instances of the specified Business Transaction CI type that have: ➤ Completed in the specified time range ➤ Any transaction state See "Response Time" on page 642.
	End-to-end Transaction Average Duration	The average end-to-end Response Time of all instances of the specified Business Transaction CI type that have: ➤ Completed in the specified time range ➤ Any transaction state See "Response Time" on page 642.
Backlog	In-process Transaction Count	The total Transaction Count (or Volume) of all instances of the specified Business Transaction CI type that have: ➤ Been started in the specified time range but not completed ➤ Any transaction state
Delays	In-process Transaction Delayed Value	The total Value of all instances of the specified Business Transaction CI type that have: ➤ Been started in the specified time range but not completed ➤ Transaction state of Delayed See "Value" on page 644.

KPI	Contributing Health Indicators	Supporting TransactionVision Data in the Sample
Exceptions	In-process Transaction Exception Value	The total Value of all instances of the specified Business Transaction CI type that have: ➤ Been started in the specified time range but not completed ➤ Transaction state of Exception See "Value" on page 644.
Failures	In-process Transaction Failure Value	The total Value of all instances of the specified Business Transaction CI type that have: ➤ Been started in the specified time range but not completed ➤ Transaction state of Failure See "Value" on page 644.
Value	In-process Transaction Value and Completed Transaction Value	The total Value of all instances of the specified Business Transaction CI type that have: ➤ Been started in the specified time range but not completed and those that have completed ➤ Any transaction state See "Value" on page 644.
Volume	Backend Transaction Count	The total Value of all instances of the specified Business Transaction CI type that have: ➤ completed in the specified time range ➤ Transaction state of Completed any transaction state See "Value" on page 644.

You can view the Indicators that exist in the BSM deployment environment by selecting **Admin** > **Service Health** > **Repositories** > **Indicators** > **CI Types** > **Business Element** > **Business Transaction**. For more details about the Indicators, see "Indicator Repository" in *Using Service Health*.

You can view the KPIs that exist in the BSM deployment environment by selecting **Admin** > **Service Health** > **Repositories** > **KPIs**. For more details about the KPIs, see "KPI Repository" in *Using Service Health*.

The Transaction States Relevant to Service Health

The transaction state is the current state of the transaction and can be one or more of the following:

State	Description
Successful	The total count of transaction instances, minus the failed count. See Failed below.
Late	Transaction instances with a response time that exceeded the threshold value specified in the transaction tracing properties. Also referred to as delayed.
Failed	Transaction instances that did not meet a success threshold defined by the Failures rule. For example, a stock buy transaction would fail if the account balance was less than $10,000, which is required to do the purchase, or a margin buy would fail if the account ratio is below a minimum balance threshold. For information about the Failure data rule, see "Transaction Tracing Overview" on page 514.
Transaction Exceptions	Transaction instances that did not follow the expected flow path, as defined by the Exception data rule, see "Transaction Tracing Overview" on page 514.

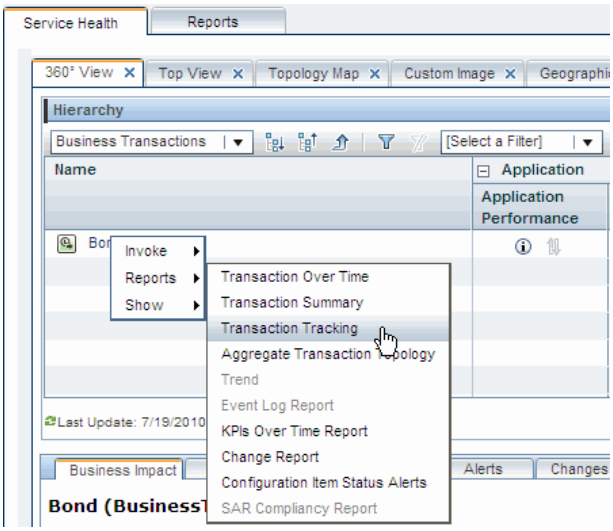
Transactions can be classified in more than one state category, for example, as both late and as successful.

Drilldowns From Service Health to the Transaction Management Reports and Topologies

There are drilldowns from Service Health to the following Transaction Management reports:

- Transaction Over Time Report
- Transaction Summary Report
- Transaction Tracking Report
- Aggregate Topology

These drilldowns are accessed by right-clicking any TransactionVision-defined Business Transaction CI that appears in the Service Health Application. For example:



For more information about the Service Health Application, see *Using Service Health*.

MyBSM

In MyBSM you can create your own workspace based on built-in or custom pages. To those pages, the following components based on Transaction Management reports can be added:

- Transaction Summary report.
- The Volume/Response graph of the Transaction Over Time Report.
- The Volume Analysis graph of the Transaction Over Time Report.
- The Transaction State graph from the Transaction Over Time Report" in Using Transaction Management.
- The Transaction Capacity Planning Graph of the Transaction Over Time Report.
- The Transaction Measurement Correlation Graph of the Transaction Over Time Report.

For details on how to create your workspace and use MyBSM, see "Working with MyBSM" in *Using MyBSM*.

Service Level Management

Similar to the way that TransactionVision data is used in Service Health, it is used in Service Level Management.

See also "KPIs Based On Monetary Value" on page 507 in *Using Service Level Management*.

The following table shows the relationship among the predefined Business Transaction CI KPIs, Health Indicators and TransactionVision data.

KPI	Contributing Health Indicators	Supporting TransactionVision Data in the Sample
Application Performance	End-to-end Transaction Average Duration	The average end-to-end Response Time of all instances of the specified Business Transaction CI type that have: ➤ completed in the specified time range ➤ any transaction state See "Response Time" on page 642.
Backlog	In-process Transaction Average Count	The total Transaction Count (or Volume) of all instances of the specified Business Transaction CI type that have: ➤ been started in the specified time range but not completed ➤ any transaction state
Delays	Percent Delayed Transactions	The percent of all instances of the specified Business Transaction CI type that have: ➤ been started in the specified time range but not completed ➤ a transaction state of Delayed See "Value" on page 644.
Exceptions	Percent Exception Transactions	The percent of all instances of the specified Business Transaction CI type that have: ➤ been started in the specified time range but not completed ➤ a transaction state of Exception See "Value" on page 644.
Failures	Percent Failed Transactions	The percent of all instances of the specified Business Transaction CI type that have: ➤ been started in the specified time range but not completed ➤ a transaction state of Failure See "Value" on page 644.

KPI	Contributing Health Indicators	Supporting TransactionVision Data in the Sample
Throughput	Transaction Throughput	The percent of all instances of the specified Business Transaction CI type that have: ➤ completed in the specified time range ➤ a transaction state of Successful
Value	Transaction Value	The total Value of all instances of the specified Business Transaction CI type that have: ➤ been started in the specified time range but not completed

End User Management

Transaction Management interacts with End User Management in the following ways:

➤ "Data Sent From RUM to TransactionVision" on page 356

➤ "Drilldowns From RUM Reports to the Transaction Management Reports and Topologies" on page 356

➤ "Drilldowns From BPM Reports to the Transaction Management Reports and Topologies" on page 357

Data Sent From RUM to TransactionVision

RUM data can be sent to TransactionVision. This allows the end user perspective data to be added to the business transaction data collected by TransactionVision.

This integration has the following requirements:

- The application must be monitored by both RUM and either TransactionVision or HP Diagnostics.
- Only J2EE Web transactions are eligible.
- The TransactionVision deployment environment must have an Analyzer enabled for RUM. See "How to Enable Communication Between RUM and TransactionVision" on page 364.

Drilldowns From RUM Reports to the Transaction Management Reports and Topologies

The integration between End User Management and TransactionVision helps isolate problems occurring on the server side. In TransactionVision you can identify which parts of a server call are causing long server response times, and see all the events that participate in a server call plus the time taken for each sub-step of the server call.

The following RUM reports provide drilldown links to Transaction Management reports:

Drill From this EUM Report	To this Transaction Management Report:
RUM Action Summary	Transaction Tracking report
Action Over Time	Transaction Tracking report
RUM End User Group by Action	Transaction Tracking report
Application Infrastructure by Action	Transaction Tracking report
Action Raw Data	Transaction Tracking report

Drill From this EUM Report	To this Transaction Management Report:
Triage	Transaction Tracking report
Triage Raw Data	Transaction Tracking report

Drilldowns From BPM Reports to the Transaction Management Reports and Topologies

The following BPM reports provide drilldown links to Transaction Management reports:

Drill From this BPM Report	To this Transaction Management Report:
Triage Raw Data	Transaction Tracking report

The drilldown is based on Business Process Monitor's BPM Profile, BPM Location, and BPM Transaction.

HP Diagnostics

If HP Diagnostics is deployed in the BSM environment, you can drill down to Diagnostics data from some of the Transaction Management reports and topologies.

TransactionVision automatically detects when Diagnostics has data on a component, shows it with a special Diagnostics drilldown icon where relevant.

The drilldowns have the following requirements:

- Applies only to Servlet/JSP components and hosts.
- The Agent monitoring the applications is configured to act as both a Diagnostics probe and a TransactionVision Agent.

➤ Both tracing and monitoring are enabled on the transaction.

For more information about the drilldowns, see "Reports Overview" on page 648.

Business Process Insight

Transaction Management interacts with Business Process Insight in the following ways:

➤ "Data Sent by TransactionVision to Business Process Insight" on page 358

➤ "Drilldowns Between Business Process Insight and the Transaction Management Reports and Topologies" on page 358

➤ "Design-Time Integration with Business Process Insight" on page 359

➤ "Run-Time Integration with Business Process Insight" on page 360

Data Sent by TransactionVision to Business Process Insight

You can define rules that define which business transaction events to make available to Business Process Insight. TransactionVision then collects these events and aggregates the data related to the events.

The events and the aggregated data are placed on a JMS queue and are available for Business Process Insight to import into the Model Repository. See the task "How to Send Events to BPI" on page 534.

Drilldowns Between Business Process Insight and the Transaction Management Reports and Topologies

➤ Progress to Business Process Insight business processes with data received from TransactionVision business transaction events.

➤ Link from details of TransactionVision instances, to the process instance details page within the Business Process Insight Business Process Dashboard.

Design-Time Integration with Business Process Insight

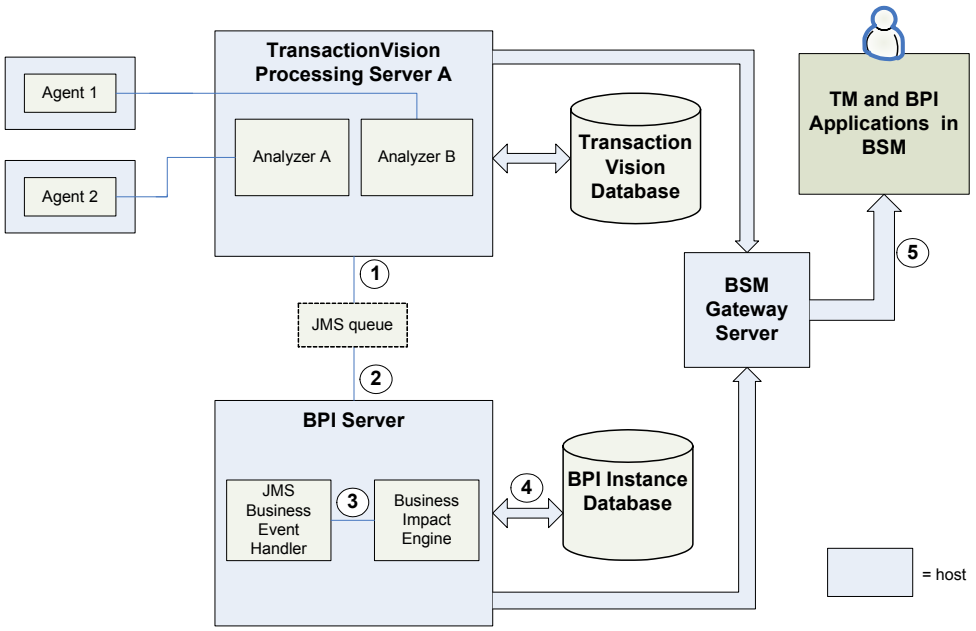
Event and transaction rules are defined to filter the incoming transactions according to your requirements. The required transaction events are placed on the configured Business Process Insight JMS queue.

The event importer component reads the transaction events from the JMS queue and imports them into the Model Repository, where they are available to the Modeler.

You define your processes as usual and create Data definitions, which can then subscribe to the imported Event definitions.

Run-Time Integration with Business Process Insight

When TransactionVision and Business Process Insight are both deployed in the BSM deployment environment, event data flows from each TransactionVision Processing Server to the BPI Server as follows:



The numbered labels in the diagram have the following meaning:

1	Using the BPI rules defined for the business transactions, the TransactionVision Processing Server places the appropriate events on a JMS queue that you have defined. See "Events tab" in "Analyzer Page, Configuration Tab" on page 124. For information about creating BPI rules for business transactions, see "How to Send Events to BPI" on page 534.
2	The JMS Business Event Handler is listening for new events that are arriving on the queue.
3	The JMS Business Event Handler processes the events and sends them to the Business Impact Engine for processing.

4	Details of the process and its progression are stored in the Business Process Insight database as usual.
5	The Business Process Insight application presents the results of the process progression, providing drilldown to the appropriate business transaction events within the Transaction Management reports and topologies.

TransactionVision Data in the BSM Profile Database

The TransactionVision Sample is frequently uploaded to the BSM Profile database where it used for other BSM applications such as Service Health. This data can also be extracted into custom reports by using the BSM API. For more details, see "Use Report Manager to Manage Custom Reports" in *Reports*.

For more details about the fields in the data sample structure than below, see "Data Sample for TransactionVision" on page 605 in *Reference Information*.

The Sample contains information for each Business Transaction CI, including:

- Transaction name.

  The Business Transaction CI name. This name is specified when the CI was created. For information about creating Business Transaction CIs, see "How to Create a Business Transaction CI" on page 525.

- Transaction instance response times for completed transactions.

  Sum of end-to-end response time of transactions instances that completed in the reporting interval and did not have a state of Failed.

  The maximum response time of any transactions instance that completed in the reporting interval (failed transaction instances are not eligible) is also included.

  The minimum response time of any transactions instance that completed in the reporting interval (failed transaction instances are not eligible) is also included.

  For more information about response times, see "Key Information Reported For a Transaction" on page 641.

- Transaction instance end user response times for completed transactions.

  If RUM is also providing data about the monitored application:

  - Total end user response time of transactions instances that completed in the reporting interval and did not have a state of Failed.
  - The maximum end user response time of any transactions instance that completed in the reporting interval (failed transaction instances are not eligible) is also included.
  - The minimum response time of any transactions instance that completed in the reporting interval (failed transaction instances are not eligible) is also included.

- Transaction instance counts for completed transactions.

  Total numbers of transaction instances that completed in the reporting interval, including subtotals for failed, late, and exception transaction instances.

- Transaction instance counts for in-process transactions.

  Total numbers of transaction instances that exist but have not yet completed in the reporting interval, including subtotals for failed, late, and exception transaction instances.

- Transaction instance values for completed transactions.

  Total value of transaction instances that completed in the reporting interval, including subtotals for failed, late, and exception transaction instances.

- Transaction instance values for in-process transactions.

  Total value of transaction instances that exist but have not yet completed in the reporting interval, including subtotals for failed, late, and exception transaction instances.

- Customer name.

  Customer name to which the sample belongs. This is only applicable for HP Software-as-a-Service deployments. For all others it is Default client.

- TV id.

  The ID for the transaction in TransactionVision.

TransactionVision Data in the RTSM

TransactionVision publishes CIs other than Business Transaction CIs.

It publishes information corresponding to the infrastructure elements discovered by TransactionVision agents. These dependencies can then be viewed from the Business Transactions view in System Health.

Exporting TransactionVision Data to BSM

The TransactionVision Data Samples are sent to BSM through the built-in **BSM Aggregate TV Data Sample** job. Along with other built-in jobs, this job is run by the TransactionVision Job Manager.

The job creates and then delivers aggregate Business Transaction CI data to the BSM Profile database. It queries the TransactionVision database, and every 5 minutes (300 seconds) delivers it to BSM via HTTP Post.

The TransactionVision infrastructure CIs are sent to BSM through the **RTSM Population** and **RTSM Update** job. These jobs publish and refresh, respectively, the following information about the infrastructure:

- J2EE applications
- J2EE servers on which the J2EE applications reside
- Any machines on which J2EE servers are hosted

This job also creates and reports each dependency relationship between business transactions and corresponding J2EE applications.

For more information about these job, see "Built-in Jobs" on page 316.

Tasks

How to Verify or Modify Communication Between BSM and TransactionVision

1 Select **Admin** > **Transaction Management.**

2 (left pane) Click the **TransactionVision** top-level folder.

3 (right pane) Select **Configuration** > **BSM Settings** tab.

For information about this tab, see "Configuration Tab, BSM Settings Tab" on page 343.

How to Enable Communication Between RUM and TransactionVision

In the Transaction Management Administration pages:

Designate one (and only one) Analyzer in the TransactionVision deployment environment to process RUM events. This involves setting the **Do you want to enable this Analyzer for RUM events**? check box in the Analyzer wizard. See "Key Configuration Concepts for Analyzers" on page 85.

The built-in RUM Commlink is automatically assigned to the Analyzer. See "Default Communication Links" on page 171.

In the RUM Administration user interface:

Designate one and only one RUM Engine to connect to the TransactionVision Analyzer from step 1 above.

This involves using the Configuration drop-down menu on the RUM Engine Web console to access the TV Connection Settings. See the *Real User Monitor Administration* PDF.

How to Enable Communication Between BPI and TransactionVision

In the Transaction Management Administration pages:

1 Designate a single Analyzer in the TransactionVision deployment environment to process BPI events. This involves setting options in the BPI group in the Analyzer's Configuration Page, Events tab. See "Configuration Tab, Events Tab" on page 130.

2 Designate which events you want to export to BPI. See "How to Send Events to BPI" on page 534.

In the BPI Server Administration user interface:

1 Specify the JMS Service Provider configuration that matches the values specified on the BPI group in the Analyzer's Configuration Page, Events tab.

If you are using the built-in SonicMQ as the messaging middleware product, the default settings in the JMS Service Provider configuration should apply.

This makes the events automatically available within process definitions.

2 In the BPI Modeler, add the events.

For complete details on importing events into Business Process Insight, see *Using Business Process Insight*.

Troubleshooting and Limitations

This section includes the following topics:

➤ "Time Zone On Target Applications Must Be Set Correctly" on page 365

➤ "Samples Not Delivered to BSM" on page 366

Time Zone On Target Applications Must Be Set Correctly

Make sure all components (for example TransactionVision, Real User Monitor, Business Process Monitor) are time synched with BSM, or the data might not be consistent across the different components.

Samples Not Delivered to BSM

When TransactionVision samples are not delivered to BSM check the settings described on the "Configuration Tab, BSM Settings Tab" on page 343.

If you still do not see TransactionVision samples in the BSM applications such as Service Health, you can debug further by turning on tracing. See "How to Modify the Built-in Jobs" on page 318.

10

Administration Utilities

This chapter includes:

Reference

- AnalyzerManager on page 368
- ConfigurationTool on page 371
- CorrelationUtil on page 373
- CreateSqlScript on page 384
- DataUtil on page 388
- DeleteEvents on page 391
- FlushStatusUtil on page 399
- ManageQueue on page 400
- MigrateDB on page 408
- nanny on page 408
- PartitionUtil on page 411
- PassGen on page 413
- rebind_agent on page 414
- rebind_tux_sensor on page 415
- runSupportSnapshot on page 416
- SetupModule on page 419
- TimeServer on page 420
- ValidateXml on page 421

Reference

AnalyzerManager

Location:

<TVISION_HOME>/bin/AnalyzerManager.[sh|bat]

Purpose:

Manage the TransactionVision Analyzer service.

Syntax:

```
AnalyzerManager [-host HOST] [-index INDEX | -schema SCHEMA] <COMMAND>
```

<COMMAND> can be one of the following:

-status
-status commlink COMMLINKNAME [-csv]
-status commlinkhealth | analyzerhealth | currentanalyzerhealth | dbhealth
-clearmsgs
-reconfig
-exit [keepcollect]
-debug [on|off]
-trace [on|off]
-perf [on|off]
-processinfo
-version

Options:

Option	Description
-status	Reports the current Analyzer status. The following information is displayed: ➤ Collection status of the Analyzer. ➤ Statistics for all communication links. If the option **-csv** is specified, the statistics are printed in CSV format. ➤ Communication link health. The result of message roundtrip tests for all assigned communication links. ➤ Analyzer health. Includes any Analyzer processing errors and status of all internal Analyzer threads. ➤ Database connectivity. Checks that database connectivity is functional.
-status commlink	Reports the status and statistics for a specific communication link.
-status commlinkhealth	Performs a message roundtrip test for all communication links. The result is **OK** or **failed**.
-status analyzerhealth	Checks if all internal threads are alive, and if any errors occurred during event processing. The result is **OK** or **failed**.
-status currentanalyzerhealth	Similar to **analyzerhealth**, but returns the *current* status of the Analyzer. Failures in the past are ignored.
-status dbhealth	Checks if the Analyzer can connect to the database. The result is **OK** or **failed**.
-clearmsgs	Clears all current Analyzer and communication link messages.
-processinfo	Reports the process status of all Analyzers on this host. This also returns the Analyzer index which is needed for other options of this utility and for the nanny utility.
-reconfig	Causes the Analyzer to reload its configuration settings without stopping event collection.
-version	Returns Analyzer version information.

Option	Description
-exit	Causes the Analyzer to stop collecting event data and the process to exit.
-exit keepcollect	Specifies that stop messages are not sent to agents and event collection continues.
-debug	Enables or disables the Analyzer in debug mode.
-trace	Enables or disables Analyzer trace logging.
- perf	Enables or disables Analyzer performance logging.
- host	Name of the host on which the Analyzer runs. Can be either name or IP address. Local host is used if not specified.
-index	Index number of the Analyzer to perform the operation against. If no index is specified, then the Analyzer with index 1 is used by default.
-schema	Schema name of the Analyzer to perform the operation against. The Analyzer with index 1 is used by default if no schema or index is specified.

Examples:

To exit the Analyzer without sending a stop collection message to the agent (so that the agent continues to collect events), use the following command:

```
AnalyzerManager -exit -keepcollect
```

Note that the agent only continues to collect events until the last configuration message has expired, since there will not be any new configuration messages sent by the Analyzer. You can change the configuration message expiry on the Analyzer configuration.

Example:

```
AnalyzerManager –stop –host HOST
```

ConfigurationTool

Location:

<TVISION_HOME>/bin/ConfigurationTool.[sh|bat]

Purpose:

Use this utility to manually send Stop configuration messages to TransactionVision agents. This tool can be used as an emergency means of stopping TransactionVision agents from sending events.

This utility works with WebSphere MQ and SonicMQ event transport providers only.

The database name, database host IP, database schema id, and communication link id are specified to construct the Stop Configuration message. ManageQueue -show_event_ids can be used to examine the events on the event queue to determine what to use for these values.

Syntax:

```
ConfigurationTool [Queue Connection Options] [Configuration Options] <Action>
[Input/Output Options] [Miscellaneous Options]
```

Options:

Option	Description
<Action>	**-stop** Sends a Stop configuration message to the specified queue.
[Queue Connection Options]	
-qm QUEUEMANAGER	WebSphere MQ queue manager to connect to.
-sonicmq (-s)	SonicMQ server to connect to.
-host (-h) HOST	Host name for client connection to queue manager.
-port (-p) PORT	Listener port for client connection to queue manager.
-channel (-c) CHANNEL	Client connection channel to queue manager.
-username USERNAME	Username to connect with.

Option	Description
-password PASSWORD	Password to connect with.
[Input/Output Options]	
-outputqueue (-oq) QUEUE	Output queue.
[Configuration Options]	
-schema ID|NAME	Specifies the schema ID or name to use in the configuration message.
-db_name NAME	Specifies the database name to use in the configuration message. If not specified, the currently defined database name is used.
-db_host IP|HOST	Specifies the database host IP or name to use in the configuration message. If not specified, the currently defined database host IP is used.
-commlink ID|NAME	Specifies the communication link ID or name to use in the configuration message.
[Miscellaneous Options]	
-verbose (-v)	Print verbose output.
-trace (-t)	Enable agent trace logging.

Usage Notes:

In some cases, events continue to be collected from the WMQ Agent despite the fact that collection has been stopped from the TransactionVision user interface. This may be due to orphaned configuration messages on the config queues that had infinite timeout, which would cause them to stay enabled indefinitely. Shutting down the queue managers would resolve the problem, but this is not possible in a production environment.

To clear out the orphaned messages from the configuration queue, and thereby stop event collection use the ConfigurationTool utility to issue an emergency stop message.

CorrelationUtil

Location:

Run from the following command line on the Processing Server host:

```
RunClass.bat com.bristol.tvision.admin.CorrelationUtil
```

Purpose:

Use this utility to manually manage correlation after a correlation problem has been identified. For instance, collected events from multiple transactions are being correlated into one single transaction.

This utility works with all technologies.

To troubleshoot and identify the cause of correlation issues, all events in this transaction have to be analyzed in an iterative fashion. Local transaction containment, standard and custom event relations, and tracking ids must all be taken into account.The **CorrelationUtil** utility automates this process, by performing SQL queries, XML data retrieval, event comparison, and so forth.

The input to the utility is a transaction in the database, a list of one or more events that are known to be valid in the transaction, and a list of one or more events that are known to be correlated into the transaction by error. The output is the path that explains why those events are correlated together.

Syntax:

```
CorrelationUtil
  -schema(-s) schemaname
    -find(-f) [pii,seqNo | btxnId] pii,seqNo |
    -txninfo(-ti) [pii,seqNo | btxnId] |
    -correlinfo(-ci) pii,seqNo |
    -alltxninfo(-ati)
  { -timeout seconds }
  { -dbparams(-db) [db2|oracle|sqlserver|derby] host dbname port user passwd }
```

Options:

Option	Description
-schema(-s) schemaname	Database schema for where events are stored.
-find	Find a correlation path between two events. See "Usage Notes and Examples for the -find Option" below.
-txninfo	Detailed correlation information for a specific transaction.
-correlinfo	Detailed correlation info for a specific event. See "Usage Notes and Examples for the -correlationinfo Option" below.
-alltxninfo	General information for all transactions in the schema. See "Usage Notes and Examples for the -find Option" below.
-timeout	Timeout for path search, default: 60 seconds.
-dbparams	By default the tool uses the database configuration for the local Processing Server. The database connection can also be specified explicitly, which allows you to use this tool with other databases or TransactionVision installations.

Usage Notes and Examples for the -find Option

The **-find** option searches for a correlation path from one event in a transaction to another. The starting event can be specified via event id (pii,seqno), or via business transaction id (the first event of the transaction is chosen). The end event is specified via event id (pii, seqno).

Use this option to determine why specific events are correlated together incorrectly. A typical use case would be a transaction that contains one or more events that definitely should not belong in this transaction. Choose an event that correctly belongs to the transaction as the starting event, and chose one of the incorrect events as the end event. The utility determines the correlation path between both events. You can identify which correlation step needs to be broken up to exclude the incorrect event.

Example:

RunClass.bat com.bristol.tvision.admin.CorrelationUtil -s trade -find 232,2320 292,83

```
==========================================
Path (232,2320) -> (292,83) in btxn 20:
==========================================

(232,2320) Servlet HTTP_POST
=> (78,2326) JMS publish
  TrackingId
SQ000000000SQ000000000GATESMAN8400:119cfe77829:fc00fc0.7715527662769313175
=> (266,2337) JMS onMessage
  Relation
535130303030303030303029:fc00fc0.7715527662769313:654976778:1210393699429:61
=> (258,2332) JMS send
  TrackingId
SQ000000000SQ000000000GATESMAN8400:119cfe77829:fc00fc0.7715527662769313178
=> (312,2381) JMS onMessage
  Relation
535130303030303030303020202020203361244820 01D71232393132343634323831323130 33
39333639393432390000
=> (308,2376) JMS send
  TrackingId
SQ000000000SQ000000000GATESMAN8400:119cfe77829:fc00fc0.771552766276931317b
=> (288,1) MQSERIES MQGET
  Relation
535130303030303030303020202020203361244820 01D51032393132343634323831323130 33
39333639393432390000
=> (292,82) CICS LINK
  TrackingId CICSBTCD78003419360605516C
=> (292,83) CICS WRITEQ TD

  LocalTxn TVCICS-00 5E 00 F2 00 9B 00 40 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00
```

- HTTP POST (232,2320) is correlated to JMS Publish (78,2326) via tracking id
- JMS Publish (78,2326) is correlated to JMS onMessage (266,2337) via event relationship
- JMS onMessage (266,2337) is correlated to JMS Send (258,2332) via tracking id
-[…]
-CICS LINK (292,82) is correlated to CICS WRITEQ TD (292,83) via local transaction

Usage Notes and Examples for the -alltxninfo Option

The **-alltxninfo** option displays general information about all business transactions in the given database schema.

Example:

RunClass.bat com.bristol.tvision.admin.CorrelationUtil -s trade -alltxninfo

```
=======================================
All txns in schema trade :
=======================================

Txn id      Starttime          Txn Class          ResponseTime  EventCount

20          2011-01-13 13:58:40  -Unclassified-      4982 ms      53
3           2011-01-13 14:01:23  -Unclassified-      5173 ms      54
15          2011-01-13 14:04:16  -Unclassified-      5253 ms      54

Total txns: 3

Txns with a large number of events:

TxnId: 3          Event Count: 54
TxnId: 15         Event Count: 54
TxnId: 20         Event Count: 53
```

Usage Notes and Examples for the -correlationinfo Option

The **correlationinfo** option displays detailed correlation information for a specific event.

Example:

RunClass.bat com.bristol.tvision.admin.CorrelationUtil -s trade -correlationinfo 78,2326

```
============================================
Correlation info for (78,2326) :
============================================

Btxn id: 20

Ltxn key:
SQ000000000SQ000000000TVTXID:GATESMAN8400:119cfe77829:fc00fc0.77155276627693
13:7e5

Events with same ltxn key:
(78,2326) JMS publish

Tracking IDs:
SQ000000000SQ000000000GATESMAN8400:119cfe77829:fc00fc0.7715527662769313175

Events with same tracking ID:
(78,2368) JMS receive
(212,2389) Servlet HTTP_POST
(232,2320) Servlet HTTP_POST
(78,2387) EJB processTrade
(228,2388) Servlet HTTP_POST

Relation keys:
 1
535130303030303030303020202020203361244820 01D30B32393132343634323831323130 33
39333639393432390000
 4 535130303030303030303029:fc00fc0.7715527662769313:654976778:1210393699429:61
 6 535130303030303030303029:fc00fc0.7715527662769313:654976778:1210393699429:61

Correlated events:
(286,1679) MQSERIES MQGET
(266,2337) JMS onMessage
```

Usage Notes and Examples for the -txinfo Option

The **-txninfo** option displays detailed correlation information for a specific transaction. The transaction can be specified by business transaction id, or by specifying the event id of any event in the transaction. Note: if the event time of an event in the event list is preceded with an exclamation mark ('!'), the utility detected that the message put event occurred before the corresponding message get event, and is an indication of a timeskew problem.

Example:

RunClass.bat com.bristol.tvision.admin.CorrelationUtil -s trade -txninfo 20

```
==========================================
Transaction info for btxn 20 :
==========================================

Events total: 53

5 tracking IDs
26 local txn keys
16 relation lookup keys

Some events are out of order!

Detailed info has been written to txninfo.txt

2011-01-13 13:58:40.482000          (232,2320) Servlet    HTTP_POST
2011-01-13 13:58:44.984000          (286,1679) MQSERIES   MQGET
SYSTEM.BROKER.DEFAULT.STREAM
2011-01-13 13:58:44.984000          (286,1681) MQSERIES   MQPUT
SYSTEM.JMS.D.CC.SUBSCRIBER.QUEUE
2011-01-13 13:58:44.984000          (286,1682) MQSERIES   MQPUT
SYSTEM.JMS.REPORT.QUEUE
2011-01-13 13:58:44.992000          (286,1683) MQSERIES   MQCMIT
2011-01-13 13:58:44.995000           (78,2387) EJB        processTrade
2011-01-13 13:58:45.016000!          (78,2326) JMS        publish          topic://REQUEST
2011-01-13 13:58:45.031000          (258,2336) EJB        onMessage
2011-01-13 13:58:45.031000          (266,2337) JMS        onMessage        topic://REQUEST
2011-01-13 13:58:45.109000          (258,2332) JMS        send             queue:///FX
2011-01-13 13:58:45.125000          (308,2380) EJB        onMessage
2011-01-13 13:58:45.125000          (312,2381) JMS        onMessage        queue:///FX_LOCAL
2011-01-13 13:58:45.203000          (268,2338) EJB        qualify
2011-01-13 13:58:45.203000          (272,2339) EJB        setId
2011-01-13 13:58:45.203000          (272,2340) EJB        setType
```

```
2011-01-13 13:58:45.203000          (272,2341) EJB       setData
2011-01-13 13:58:45.203000          (272,2342) EJB       getId
2011-01-13 13:58:45.203000          (272,2343) EJB       ejbCreate
2011-01-13 13:58:45.203000          (272,2344) EJB       getId
2011-01-13 13:58:45.203000          (272,2345) EJB       getId
2011-01-13 13:58:45.203000          (272,2346) EJB       getType
2011-01-13 13:58:45.203000          (272,2347) EJB       getData
2011-01-13 13:58:45.203000          (272,2348) JDBC      executeUpdate
2011-01-13 13:58:45.203000          (272,2349) EJB       ejbPostCreate
2011-01-13 13:58:45.231000          (286,1685) MQSERIES   MQGET
SYSTEM.BROKER.DEFAULT.STREAM
2011-01-13 13:58:45.231000          (286,1686) MQSERIES   MQPUT
SYSTEM.JMS.D.CC.SUBSCRIBER.QUEUE
2011-01-13 13:58:45.231000          (286,1687) MQSERIES   MQPUT
SYSTEM.JMS.REPORT.QUEUE
2011-01-13 13:58:45.239000          (286,1688) MQSERIES   MQCMIT
2011-01-13 13:58:45.247000          (272,2350) EJB       ejbStore
2011-01-13 13:58:45.247000          (272,2351) JDBC      executeQuery
2011-01-13 13:58:45.247000          (272,2352) JDBC      collectStatistics
2011-01-13 13:58:45.247000          (272,2353) EJB       ejbLoad
2011-01-13 13:58:45.267000          (272,2354) EJB       process
2011-01-13 13:58:45.267000          (272,2355) EJB       ejbStore
2011-01-13 13:58:45.267000!         (308,2359) JMS       publish          topic://REPLY
2011-01-13 13:58:45.267000          (278,2374) EJB       onMessage
2011-01-13 13:58:45.267000          (284,2375) JMS       onMessage        topic://REPLY
2011-01-13 13:58:45.286000          (278,2367) JMS       send             queue:///
REPLY_FOR_SESSION
2011-01-13 13:58:45.286000           (78,2368) JMS       receive          queue:///
REPLY_FOR_SESSION
2011-01-13 13:58:45.308000          (308,2376) JMS       send             queue:///
CICSREQUEST
2011-01-13 13:58:45.340010           (292,79) CICS       START
2011-01-13 13:58:45.340974           (292,80) CICS       TASK START
2011-01-13 13:58:45.341058           (292,81) CICS       PROGRAM START
2011-01-13 13:58:45.372253            (288,0) MQSERIES   MQOPEN       CICS.INIT.Q
2011-01-13 13:58:45.385832            (288,1) MQSERIES   MQGET        CICS.INIT.Q
2011-01-13 13:58:45.431027           (292,82) CICS       LINK
2011-01-13 13:58:45.431265           (292,83) CICS       WRITEQ TD
2011-01-13 13:58:45.431405           (292,84) CICS       WRITEQ TD
2011-01-13 13:58:45.431658           (292,85) CICS       WRITEQ TD
2011-01-13 13:58:45.431747           (292,86) CICS       LINK
2011-01-13 13:58:45.435097           (292,87) CICS       TASK END
2011-01-13 13:58:45.446000          (228,2388) Servlet    HTTP_POST
2011-01-13 13:58:45.464000          (212,2389) Servlet    HTTP_POST
```

Tracking IDs of transaction:

(22)
SQ000000000SQ000000000GATESMAN8400:119cfe77829:fc00fc0.771552766276931317b
(3) SQ000000000SQ000000000GATESMAN8400:119cfe77829:fc00fc0.771552766276931317e
(10) CICSBTCD78003419360605516C
(6) SQ000000000SQ000000000GATESMAN8400:119cfe77829:fc00fc0.7715527662769313175
(3) SQ000000000SQ000000000GATESMAN8400:119cfe77829:fc00fc0.7715527662769313178

Tracking ID with most events:
SQ000000000SQ000000000GATESMAN8400:119cfe77829:fc00fc0.771552766276931317b
(22)
Events:
(308,2380) onMessage
(312,2381) onMessage
(268,2338) qualify
(272,2339) setId
(272,2340) setType
(272,2341) setData
(272,2342) getId
(272,2343) ejbCreate
(272,2344) getId
(272,2345) getId
(272,2346) getType
(272,2347) getData
(272,2348) executeUpdate
(272,2349) ejbPostCreate
(272,2350) ejbStore
(272,2351) executeQuery
(272,2352) collectStatistics
(272,2353) ejbLoad
(272,2354) process
(272,2355) ejbStore
(308,2359) publish
(308,2376) send

Local txn keys of transaction:
(1)
SQ000000000SQ000000000TVTXID:GATESMAN8400:119cfe77829:fc00fc0.7715527662769313:7f4
(1)
SQ000000000SQ000000000TVTXID:GATESMAN8400:119cfe77829:fc00fc0.7715527662769313:7e4
(1) SQ000000000SQ000000000694561126:119d116254f

(1)
SQ000000000SQ000000000TVTXID:GATESMAN8400:119cfe77829:fc00fc0.77155276627693
13:7f2
(1)
SQ000000000SQ00000000000000119d116252000000010000005e2a168d83facf4c005c522e
8516e0dff3fe42c4b4
(12)
SQ000000000SQ00000000000000119d116252000000010000005f2a168d83facf4c005c522e8
516e0dff3fe42c4b4
(5)
SQ000000000SQ00000000000000119d116253f00000001000000602a168d83facf4c005c522e8
516e0dff3fe42c4b4
(1)
SQ000000000SQ000000000TVTXID:GATESMAN8400:119cfe77829:fc00fc0.77155276627693
13:7f1
(1)
SQ000000000SQ000000000TVTXID:GATESMAN8400:119cfe77829:fc00fc0.77155276627693
13:807
(1)
SQ000000000SQ000000000TVTXID:GATESMAN8400:119cfe77829:fc00fc0.77155276627693
13:806
(1)
SQ000000000SQ000000000TVTXID:GATESMAN8400:119cfe77829:fc00fc0.77155276627693
13:7e2
(1)
SQ000000000SQ000000000TVTXID:GATESMAN8400:119cfe77829:fc00fc0.77155276627693
13:7f8
(1) SQ000000000SQ0000000001126318882:119d11624d2
(1)
SQ000000000SQ000000000TVTXID:GATESMAN8400:119cfe77829:fc00fc0.77155276627693
13:7eb
(1) SQ000000000SQ0000000001203783616:119d1162455
(4) TVCICS-00 5E 00 F2 00 C0 00 40 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00
(4) TV286-1E01000020080510042819415SQ0
(4) TV286-1C01000020080510042702909SQ0
(1) TVCICS-00 5E 00 F2 00 40 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00
(2) TVCICS-00 5E 00 F2 00 E6 00 40 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00
(3) TVCICS-00 5E 00 F2 00 9B 00 40 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00
(1) TVCICS-00 5E 00 A8 00 FA 00 C0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00

(1)
SQ000000000SQ000000000TVTXID:GATESMAN8400:119cfe77829:fc00fc0.77155276627693
13:7e6
(1) SQ000000000SQ0000000001229211972:119d1162474
(1)
SQ000000000SQ000000000TVTXID:GATESMAN8400:119cfe77829:fc00fc0.77155276627693
13:7e8
(1)
SQ000000000SQ000000000TVTXID:GATESMAN8400:119cfe77829:fc00fc0.77155276627693
13:7e5

Local txn with most events: 65 (12)
Events:
(272,2339) setId
(272,2340) setType
(272,2341) setData
(272,2342) getId
(272,2343) ejbCreate
(272,2344) getId
(272,2345) getId
(272,2346) getType
(272,2347) getData
(272,2348) executeUpdate
(272,2349) ejbPostCreate
(272,2350) ejbStore

Relation lookup keys of transaction:
(2)
535130303030303030303020202020203336124482001D30B323931323436343238313231303
033393336393934323900000

(1)
535130303030303030303020202020203336124482001C91F414D512054524144494E472020
202020336124482001D30B

(1) CICSTDQ3X'C25E81F23BF32380'

(2)
535130303030303030303029:fc00fc0.7715527662769313:654976778:1210393699429:61

(2)
535130303030303030303020202020203336124482001D60B323931323436343238313231303
033393336393934323900000

(1)
535130303030303030303020202020203361244820 01C91E32393132343634323831323130333933336393934323900 00

(2)
535130303030303030303029:fc00fc0.7715527662769313:1552571530:1210393699663:62

(1)
535130303030303030303020202020203361244820 01C91E414D512054524144494E472020202020CC65234820002607

(1)
535130303030303030303020202020203361244820 01C920414D512054524144494E472020202020CC65234820002307

(1)
535130303030303030303020202020203361244820 01C92032393132343634323831323130333933336393934323900 00

(2)
535130303030303030303020202020203361244820 01D71232393132343634323831323130333933336393934323900 00

(1)
535130303030303030303020202020203361244820 01C921414D512054524144494E472020202020203361244820 01D60B

(2)
535130303030303030303020202020203361244820 01D51032393132343634323831323130333933336393934323900 00

(1) CICSTDQ2X'C25E81F23BE35300'

(2) CICSX'C25E81F2268D2640'

(2)
535130303030303030303020202020203361244820 01D71332393132343634323831323130333933336393934323900 00

Relation lookup key with most events:
535130303030303030303020202020203361244820 01D30B32393132343634323831323130333933336393934323900 00 (2)

Events:
(286,1679) MQGET
(78,2326) publish
Event with most related events: (78,2326) (2)
Events:
(286,1679) MQGET
(266,2337) onMessage

CreateSqlScript

Location:

<TVISION_HOME>/bin/CreateSqlScript.[sh|bat]

Purpose:

To create and optionally run an SQL script to create, drop, import or export a table in the TransactionVision database.

Syntax:

```
CreateSqlScript
{-create(-c) | -drop(-d) | -import(-i) | -export(-ex) | -grant(-g) | -dbstats(-ds)}
    {-system(-sys) | -schema(-s) SCHEMA | -table(-t) TABLE SCHEMA}
    [-noscript(-n)] [-execute(-e)] [-noprompt(-np)] [-noinsert(-ni)]
    [-tablespace(-ts) TABLESPACE] [-dbMove(-m)] [-fileType(-f) IXF|DEL]
    [-lobPath(-lp) PATH] [-partCount COUNT] [-partStartDate DATE]
    [-partLocalTime] [-partLength LENGTH] [-partInterval INTERVAL]
```

Options:

Option	Description
-create (-c)	Create tables.
-drop (-d)	Drop tables.
-import (-i)	Generates a database import script. To run database scripts, use the command **db2 -n -t -f <sql script filename>**. For a DB2 database, you can combine this option with the -fileType and -lobPath options to customize the data format and the location of LOB. You may NOT combine this option with the -noscript or -execute options.

Option	Description
-export (-ex)	Generates a database export script. To run database scripts, use the command **db2 -n -t -f <sql script filename>**. For a DB2 database, you can combine this option with the -fileType and -lobPath options to customize the data format and the location of LOB. You may NOT combine this option with the -noscript or -execute options.
-grant (-g)	Generates a script for granting all database privileges required for accessing the tables of the schema to the database user specified in Database.properties. (Oracle and DB2 only)
-dbstats (-ds)	Generates a script for generating the database statistics used by the database optimizer.
-system (-sys)	Create script for system tables.
-schema (-s) SCHEMA	Create/drop project tables in schema SCHEMA.
-table (-t) TABLE SCHEMA	Create/drop table TABLE in schema SCHEMA.
-dbMove (-m)	Use db2move to import/export data.
-execute (-e)	Execute script.
-noscript (-n)	No script generation.
-noinsert (-ni)	Do not insert initial table rows.
-tablespace (-ts) TABLESPACE	Use tablespace TABLESPACE. For information about using this option with an Oracle database, see "Usage Notes: " on page 386.
-fileType (-f) IXF|DEL	Specify the DB2 data output file format used for importing/exporting data. For the IXF file type, the import/export script uses the LOBFILE option for rows that contain greater than 32K data. For the DEL type, the import/export script exports LOBFILEs into a single file (requires FixPack 8). **Default value**: IXF

Option	Description
-lobPath (-lp) PATH	Specifies the directory for DB2 LOBFILEs. **Default value:** the current directory.
-noprompt (-np)	Do not prompt for confirmation when dropping tables.
The following options are only relevant when using partitioning	
-partCount	Initial partition count.
-partStartDate	The start date of the first partition. Format: mm/dd/yyyy hh:mm [am|pm]
-partLocalTime	Use local time instead of GMT for start date.
-partLength	The partition length.
-partInterval	The length unit (Hours | Days | Months).

Usage Notes:

Note: The TransactionVision Analyzers and Processing Server must be stopped before performing any database imports or exports. This is to avoid causing the database import/exports to fail because of database locks held by any Analyzers. If a schema is dropped, make sure that there are no active Analyzers or users logged in that are using that schema.

When the **-dbstats** option is specified, the generated SQL script updates statistics about the physical characteristics of a table and the associated indexes. These characteristics include metrics like number of records, number of pages, average record length, and so forth. The optimizer uses these statistics when determining access paths to the data.

This SQL script should be run when a table has had many updates, such as when data is continuously collected into the database by the TransactionVision Analyzer.

It could result in large performance gains in queries made by TransactionVision views and reports, as well as queries made internally by the TransactionVision Analyzer to correlate events.

Alternatively, you can use the database vendor tools (Oracle Enterprise Manager, DB2 Control Center) to regenerate the database statistics.

Note: While the database statistics are being generated the TransactionVision Analyzer processing slows down.

SQL Server does not have the concept of "table spaces," but the "file group" is a similar concept. Though project tables can be distributed to different file groups by using custom DDL scripts, the creation of project tables in specific file groups is not supported.

Examples:

Create system tables with schema as TVISION and execute the procedure without generating SQL script:

```
CreateSqlScript –e –n –c –sys
```

Generate SQL script for creating tables with schema as ANALYZER1 without executing the procedure:

```
CreateSqlScript –c –s ANALYZER1
```

Drop table EVENT in schema ANALYZER1 and execute the procedure without generating SQL script:

```
CreateSqlScript –e –n –d –t EVENT ANALYZER1
```

DataUtil

Location:
<TVISION_HOME>/bin/DataUtil.[sh|bat]

Purpose:
Offers various functionality to display and manipulate TransactionVision data stored in the database, and is mainly used for diagnostics purposes.

Syntax:

```
DataUtil -schema(-s) schema [options]
```

Options:

Option	Description
-schema	The database schema.
Options:	
-dbparams(-db) [db2|oracle|sqlserver|derby] HOST DBNAME PORT USER PASSWD	The database type and connection parameters for the Analyzer database.
-localtxn_list(-ltl) pii,seq | btxnid	Prints data for all local transactions of a business transaction.
-localtxn_ids(-ltids) pii,seq | btxnid	Prints the ids of all local transactions for a business transaction.
-businesstxn_ids(-btids) pii,seq | ltxnid	Prints the business transaction id for a local transaction.
-localtxn_events(-ltev) pii,seq | ltxnid	Prints data for all events of a local transaction.
-businesstxn_events(-btev) pii,seq | btxnid	Prints data for all events of a business transaction.
-localtxn_data(-ltd) pii,seq | ltxnid	Prints data for a local transaction.
-businesstxn_data(-btd) pii,seq | btxnid	Prints data for a business transaction.
-correlated_events(-ce) pii,seq | seqid	Prints out all correlated events for an event.

Option	Description
-eventdetail(-ed) pii,seq | seqid {-conv}	Prints out the event XML document for an event (-conv translates object ids into object names).
-userdata(-ud) pii,seq | seqid	Prints out the user data for an event.
-sysmodel_object(-smo) objid	Prints the data for a system model object.
-eventid(-eid) seqid	Prints proginst_id and sequence_no for a given seq_id.
-seqid(-sid) pii,seq	Prints seq_id for a given proginst_id and sequence_no.
-counts(-c)	Prints transaction counts.
-row_counts(-rc)	Prints the row count for all project tables.
-list_schemas(-ls)	Prints out all schema names registered within TransactionVision.
-show_ids(-si)	Prints schema and commlink IDs.
-sql QUERYFILE	Prints out the generated SQL for an XML query document.
-query QUERYFILE [rows per page]	Executes the query specified by the XML query document and returns the resulting <EventList> document.
-sqlquery SQL [max_rows]	Run the specified query SQL on the database. **max_rows** can be used to limit the number of rows returned.
-sqlupdate SQL	Run the specified update SQL on the database.

Option	Description
-import_classrules [INFILE]	Import classification rules into TransactionVision. If no **INFILE** is specified, the rules are read from **ClassificationRules.xml**. This option can also be used to import rules from a version 8.x **TransactionDefinition.xml** file.
-import_classrules [INFILE]	Imports classification rules from the xml file **INFILE** (if omitted **TransactionDefinition.xml** is used). If the xml file contains transaction class definitions and Analyzer/rule assignments, those are automatically created as well.
-export_classrules [OUTFILE] [-rulesonly]	Exports all classification rules, existing transaction class definitions, and Analyzer/rule assignments into **OUTFILE** (if omitted **TransactionDefinition.xml** is used). If **-rulesonly** is specified then only rule definitions are exported, but no class definitions/assignments.
-pii	Proginst_id a of an event.
-seq	Sequence_no (primary key) of an event.
-ltxnid	Local_trans_id of a local transaction.
-btxnid	Business_trans_id of a business transaction.

DeleteEvents

Location:

<TVISION_HOME>/bin/DeleteEvents.[sh|bat]

Purpose:

Allow the user to delete events in EVENT tables and clean up contents in other tables related to deleted events.

The utility (by default) does only delete the event data. To clean up other related table data-ike system model objects and transaction data, use the options -deletePiis and -deleteTxns described below. If you have deleted event data without these options, you can clean up the remaining data by running the **DeleteEvents** utility again with the option **-cleanup**.

Syntax:

```
DeleteEvents
-from(-f) TIME -to(-t) TIME | -older(-o) TIME | -newer(-n) TIME
   [-timezone(-tz) TZID] [-businesstxns(-bt) [-state STATE] [-result RESULT]]
   [-keepPiis(-kp)] [-keepTxns(-kt)] [-keepLocalTxns(-klt)]
   [-deleteStats(-ds) [topology(t)] [business(b)]]
   [-threadcount(-tc) THREADS] [-nosplit] [-exclude TABLE_LIST]
   [-cleanupQueues]

-eventCountLimit(-ecl) MAXCOUNT
   [-businesstxns(-bt) [-state STATE] [-result RESULT]]
   [-keepPiis(-kp)] [-keepTxns(-kt)] [-keepLocalTxns(-klt)]
   [-deleteStats(-ds) [topology(t)] [business(b)]]
   [-threadcount(-tc) THREADS] [-nosplit]

-query(-q) QUERY
   [-keepPiis(-kp)] [-keepTxns(-kt)]

-cleanup(-cl)
   [-sysmodel(-sys)] [-transactions(-txns)]

-event PII SEQNO
-btxn BTXNID

common options:
   -schema(-s) SCHEMA
   [-commit(-c) COUNT] [-timeout(-ti) MINUTES] [-force(-fc)]
   [-loglevel(-log) LOGLEVEL]
```

Options:

Option	Description
-keepLocalTxns(-klt)	Do not delete local transactions from the transaction tables. (DEFAULT: keep only business transactions, only applicable if using -keepTxns).
-event PII SEQNO	Delete a single event.
-btxn TXNID	Delete a single business transaction.
-query (-q) QUERY	A TransactionVision query name.

Option	Description
-commit(-c) COUNT	Commit database transactions after the specified COUNT. The default interval is 1000.
-from(-f) STARTTIME	Specifies the start time (Time format = mm/dd/yyyy hh:mm:ss.SSS, event time in the time zone you set (your local time zone by default) or, for relative time, n[hr|min]).
-to(-t) ENDTIME	Specifies the end time (Time format = mm/dd/yyyy hh:mm:ss.SSS, event time (your local time zone by default) or, for relative time, n[hr|min]).
-older (-o) TIME	Deletes events older than the specified time.
-newer (-n) TIME	Deletes events newer than the specified time.
-eventCountLimit (-ecl) MAXCOUNT	Deletes all but the latest MAXCOUNT events.
-minage (-m) HOURS	By default, the **`DeleteEvents`** utility does not allow deleting events from the last 48 hours. Use this option to override the default. No events from the last number of hours specified by the HOURS value is to be deleted.
-schema(-s) SCHEMA	Database schema for where events are stored.
-businesstxns (-bt) [-state STATE] [-result RESULT]	Deletes only events of transactions with the specified transaction state or result. State values: -1=UNKNOWN 0=PROCESSING 1=COMPLETED Result values: -1=UNKNOWN 0=FAILED 1=SUCCESS
-deleteStats (-ds) [topology(t)] [business(b)]	Deletes statistics data from the topology or business transaction statistics table. This is optional. The time period specifies the time slices for which statistics are deleted.

Option	Description
-timezone (-z) TZID	Timezone ID (for example, GMT, EST)
-help(-h)	Displays the usage message.
-keepPiis (-kp)	Do not delete program instance or CICS task objects from the system model table. **Default**: off; objects are deleted
-keepTxns (-kt)	Do not delete transactions from the transaction tables. **Default**: off; transactions are deleted
-cleanup (-cl) -sysmodel (-sys) -transactions (-txns)	Cleans up the system model tables (removes all program instance and queue objects which are no longer referenced) or transaction tables (removes all transaction objects which are no longer referenced). You must specify either system model tables or transactions.
-iterationcount (-it) EVENTS	Maximum number of events to process before cleaning out the system model and transaction tables. **Default**: 1000.
-timeout (-ti) MINUTES	Force deletion to end after the specified time. The default is 60 minutes.
-force (-fc)	Force deletion to stop immediately after a cancel or timeout. This option is always is effect in timestamp deletion mode (for example, with the -older option).
-loglevel (-log) LOGLEVEL	Log output level. **Default:** 0
-threadcount (-tc) THREADS	Number of deletion threads (only applicable if using -older). **Default:** 4
-nosplit	Disables deletion of the same table by multiple concurrent threads.

Option	Description
-exclude TABLE_LIST	Excludes specific custom lookup tables from deletion. TABLE_LIST is a comma separated list of full table names or regular expressions.
-cleanupQueues	Cleans up queue system model objects that are no longer referenced.
-schema(-s) SCHEMA	Database schema where events are stored.
-commit(-c) COUNT	Number of database operations to execute before committing. **Default:** 1000

Usage Notes:

This utility operates in three modes: deleting by query mode, deleting by time mode, and cleaning up tables. Therefore, the options are grouped into three sets.

- Deleting by time options include: -from, -to, -older, -newer, -timezone, -keepPiis, -keepTxns, -deleteStats, -businesstxns, -state, -result (and -threadcount, -nosplit if using -older). The -force option is always in effect when deleting by time.
- Deleting by query options include: -query, -keepPiis, -keepTxns
- The cleanup mode options include: -cleanup, -sysmodel, -transactions

Options that can be used in all three modes include: -c, -ti, -fc, -log. Otherwise, these sets of options may not be mixed. For example, the following commands are INVALID and result in errors:

```
DeleteEvents -project MYPROJECT -query MYQUERY -business
DeleteEvents -project MYPROJECT -query MYQUERY -older 05/15/2003
03:00:00.000
```

Running the **DeleteEvents** utility or setting the **Purging** options on the Analyzer configuration for processing events is only allowed for the **-older** and **-eventCountLimit** modes in which the Analyzer runs specific code to safeguard against accidental deletion of data in process. But due to the negative impact of deletion on the Analyzer performance it is still not recommended to delete data while the Analyzer is processing. However, sometimes it may not be feasible to shut down the Analyzer while **DeleteEvents** is running. In that case, make sure that you do not delete transactions for which events are likely to arrive.

On many systems, deleting events from 2 days to a week in the past can ensure this scenario does not occur. However, you need to consider the duration of your business transactions to determine a period beyond which events can be deleted. For example, if your business transactions can span a day, do not delete events in the last day, if your business transactions can span a week, do not delete events in the last week while the Analyzer is running. The utility (by default) does not delete any events that are younger than 48 hours. If you are sure that no transaction duration ever exceeds a certain value, you can override this behavior by specifying the option -minage n, where n is the time buffer to exclude in hours.

Deleting by time mode provides better performance results than deleting by query. The performance of the deletion options can be ordered in the following way:

1 Deletion by query (low performance)

2 Deletion by time with options -from/to or -newer (medium performance)

3 Deletion based on event count (medium-high performance)

4 Deletion by time with option -older (high performance)

The deletion by time with option **-older** is able to take advantage of timestamp columns present in the tables and achieves much higher performance than the other methods. Also, it can execute the deletion process with multiple threads (option -threadcount), where each thread deletes different portions of the tables at the same time. The default number of threads is 4. You should adjust this option to the number of CPUs (or higher) in your system.

The deletion based on event count (**-eventCountLimit**) has similar performance characteristics as the deletion by time. Internally, the deletion module determines the date from which to delete based on the event count parameter, and then performs a **-older** deletion.

The following conditions may also affect the performance of the DeleteEvents utility:

- Make sure your database log space is sufficient, and adjust the number of uncommitted database operations (option -commit) accordingly; otherwise, the deletion process fails running out of logspace. Usually a larger commit count yields better deletion performance.
- On DB2, you could encounter deadlocks if the deletion is using multiple threads and your database lock storage is low. In that case, please adjust the value of LOCKLIST and MAXLOCKS accordingly. On Oracle this is of no concern because Oracle never escalates row locks to table locks. Alternately, use the option -nosplit to DeleteEvents. This disables the concurrent deletion of a single table by multiple threads.

Time formats may be absolute format or relative format. Absolute times are specified as mm/dd/yyyy hh:mm:ss.SSS. Relative times are specified as n[hr|min]. Relative time is offset to the time you run the script. For example "12 hr" means "12 hours before the current time." You can specify the time in any of the following ways. The descriptions assume that the script was run at 9:00 a.m.

Example	Description
-f 2 hr -t 1 hr	Deletes events from 7:00 a.m. to 8:00 a.m.
-o 1 hr	Deletes events that came into the system before 8:00 a.m.
-n 1 hr	Deletes events within the last hour.

Examples:

1 Delete everything (event and related data) from project PROJECT which is older than 72 hours using 2 threads and commit database transactions after 10000 operations:

```
DeleteEvents -project PROJECT -older 72 hr -kp -kt
-ds b t -tc 2 -c 10000
```

2 Delete events from project PROJECT using query QUERY and commit the database transaction after deletion of every 10 events:

```
DeleteEvents –project PROJECT –query QUERY –c 10
```

3 Delete events from GMT time 11:00AM, July 10, 2003 to GMT time 8:00PM, July 11, 2003:

```
DeleteEvents –from 07/10/2003 11:00:00.000 –to 07/11/2003 20:00:00.000 –z GMT
```

4 Delete events collected before 10 minutes together with statistic data from topology table:

```
DeleteEvents –older 1 hr -minage 1 –statistics t
```

5 Delete all but the latest 500,000 events in the project:

```
DeleteEvents -project PROJECT -eventCountLimit 500000
```

FlushStatusUtil

Location:

<TVISION_HOME>/bin/FlushStatusUtil.[sh|bat]

Purpose:

Shows and cleans the flush status of transaction analysis data and cached statistics for the Static Component Topology. The status is stored in the database and determines whether recovery is needed when the Analyzer starts. This utility can be used to force Analyzer startup without performing recovery by cleaning the flush status before starting the Analyzer.

Syntax:

```
FlushStatusUtil –schema SCHEMA [-clean]
```

Options:

Option	Description
-schema	The name of the database schema to show or clean the flush status for.
-clean	Clean the current flush status of the analysis data and statistics (default action is 'show'). The Analyzer starts up in a "clean" state without recovery, even though data might be missing in the database. Use this option with extreme caution.

Usage Notes:

Shows and cleans the flush status of transaction analysis data and cached statistics for the Static Component Topology. The status is stored in the database and determines whether recovery is needed when the Analyzer starts.

For cached statistics, the status contains the IDs of the events for which statistics and analysis data are missing in the database.

This utility obtains database connection information from the TransactionVision Database.properties file.

Example Output:

The following output indicates that IDs 7916-8084 and the set of additional IDs are missing statistics and analysis data and need to be recovered at the next startup time:

```
Last written sequential_id in EVENT_LOOKUP: 8084

DBCACHE flush status:
Last flushed sequential_id: 7915
Additional ids not flushed: 7912,7911,7901,7892,7870,7872

STATISTICS flush status:
Last flushed sequential_id:7915
Additional ids not flushed: 7912,7911,7901,7892,7870,7872
The following output indicates that all statistics and analysis data has been
flushed; no recovery is needed:
Last written sequential_id in EVENT_LOOKUP: 9640

DBCACHE flush status:
Last flushed sequential_id: 9640
STATISTICS flush status:
Last flushed sequential_id: 9640
```

ManageQueue

Location:

<TVISION_HOME>/bin/ManageQueue.[sh|bat]

Purpose:

This utility works with SonicMQ and WebsphereMQ only.

Use this multipurpose tool to manipulate and maintain TransactionVision event queues. This utility allows you to copy, move, and delete messages (usually TransactionVision related, but not limited to) from a queue. The messages can be transferred from the queue to another queue or archived into a jar file that can later be extracted and copied into a queue.

This utility also provides a means to query Analyzer (schema) related database IDs and summarize the contents of an event queue (or other queues).

Syntax:

```
ManageQueue <Action Options> [Queue Connection Options] [Input/Output Options]
[Message Selection Options] [Miscellaneous Options]
```

Options:

Options	Description
Action Options	
-show_db_ids	Display all schema and communication link IDs from the database defined in Database.properties
-show_event_ids	Display all TransactionVision event IDs from a given Input.
-copy [SCHEMA COMMLINK]	Copy messages from Input to Output using Message Selection Options. Optionally targets the messages to a new SCHEMA and COMMLINK. If no SCHEMA and COMMLINK are specified, all messages are copied using their original message IDs.
-move [SCHEMA COMMLINK]	Move messages from Input to Output using Message Selection Options. Optionally targets the messages to a new SCHEMA and COMMLINK. If no SCHEMA and COMMLINK are specified, all messages are moved using their original message IDs.
-delete	Deletes messages from Input using Message Selection Options.
Queue Connection Options	
-qm QUEUEMANAGER	Queue manager to connect to.
-host (-h) HOST	Host name for client connection to queue manager.
-port (-p) PORT	Listener port for client connection to queue manager.
-channel (-c) CHANNEL	Client connection channel to queue manager.
-username USERNAME	Username to connect with.
-password PASSWORD	Password to connect with.

Options	Description
Input/Output Options	
-input_file (-if) FILE1 [FILE2…]	Input files.
-input_queue (-iq) QUEUE1 [QUEUE2…]	Input queues.
-output_file (-of) FILE	Output file.
-output_queue (-oq) QUEUE	Output queue.
Message Selection Options	
-commlink ID|NAME	All TransactionVision events targeted to a given link ID or name.
-schema ID|NAME	All TransactionVision events targeted to a given schema ID or name.
-db_name NAME	All TransactionVision events targeted to a given database name.
-db_host IP|HOST	All TransactionVision events targeted to a given database host (IP or name).
-erroneous	All messages not recognized as TransactionVision events.
-userevents	All messages recognized as TransactionVision User events.
-alltv	All messages recognized as TransactionVision events.
-otherdbs	All TransactionVision events targeted to databases other than the one defined in Database.properties.
-orphaned	All TransactionVision events targeted to schema and/or communication link IDs which no longer exist.
-valid	All TransactionVision events targeted to valid schema and communication link IDs.

Options	Description
Miscellaneous Option	
-count COUNT	Process a specified limited number of messages.
-verbose (-v)	Print verbose output.
-help	Print usage.

Usage Notes:

The utility takes five different types of options, most of which are optional with the exception to the Action. In addition to the Actions, there are Queue Connection Options which provide information needed to connect to a given queue manager or broker if needed.

Input and Output options are used to copy, move, or delete from a queue or jar file. An output option is only needed if there is a copy or move action performed.

Input Message Selection Options are special options only used in conjunction with the -input_queue option. This provides a means for processing the Action on only a subset of the messages on the queue.

Miscellaneous Options provide additional functionality. The -verbose option shows additional information during message processing to show progress status, and displays a summary of what was processed at the end of execution. The –count option allows you to process only the first given number of events from the Input.

With the use of –show_event_ids, you can display a summary listing of all TransactionVision events on a given Input Queue. ManageQueue summarizes them by showing a count of each message and for which database, schema, and communication link they are targeted.

The –show_event_ids option also determines whether the messages are TransactionVision events with a valid destination (valid target schema and communication link on the defined database) and reports them as "Valid." Events which are targeted to a schema and/or communication link which no longer exist are shown as "Orphaned." And events targeted to another database which may or may not be valid are shown as "Other DB."

The –show_event_ids option also shows a count of events not recognized as TransactionVision events as "Erroneous Messages."

Input and Output can be combinations of queues and/or jar files. The following limitations apply to Input Files:

- You cannot use –show_event_ids with Input Files.
- You cannot use –move with Input Files. Use –copy instead.
- You cannot use –delete with Input Files.

When using an Input Queue and/or Output Queue, you must provide Queue Connection Options. For WebSphere MQ, if only a queue manager is specified (-qm), server binding is used. For client connections, use –host, -port, and -channel. The –port option is optional as the default port is 1414 when using the –host and –channel options. For SonicMQ, specify the -sonicmq option along with -host and -port to the SonicMQ broker.

ManageQueue only supports connection to one queue manager or broker. If there is a need you transfer messages from a queue on one queue manager or broker to a queue on a different queue manager, use a remote queue or cluster for the destination queue or broker, or transfer the messages to an Output File. Then transfer them from the file to the destination queue.

When using an Input Queue, you can specify Message Selection Options to process only a subset of the messages on the specified Input Queue. There are two types of message selection:

- Discrete Selectors

  -commlink
  -schema
  -db_name
  -db_host

➤ Validity Selectors

-erroneous
-alltv
-otherdbs
-orphaned
-valid

If you use any Discrete Selectors, all Validity Selectors are ignored, with the exception of **–erroneous** because it does not apply to TransactionVision events. Discrete Selectors are ANDed so all Discrete Selectors specified must match for the message to be processed.

If no Discrete Selectors are used, the specified Validity Selectors are ORed. All messages matching any of the specified Validity Selectors are processed.

If no selectors are used, all messages are processed.

Any option that requires the user to specify a schema and/or communication link can take either the database ID or name. Surround names containing spaces with quotation marks.

When copying or moving messages to a queue, it may be necessary to specify new targeted schema and communication link IDs or names. This is especially useful when the messages have been orphaned (the targeted schema and/or communication link does not exist anymore). If no schema and communication link is specified, each message's original event information is used.

When copying or moving messages to a jar file, new schema and communication link IDs or names are not required. If they are specified, they are ignored by ManageQueue.

Examples:

➤ Remove all messages on the event queue which are not being processed. This includes non-TransactionVision (erroneous) messages and events that are targeted to a schema and/or communication link which no longer exists (orphaned):

```
ManageQueue.sh –qm tv.qmgr –input_queue TVISION.EVENT.QUEUE –
delete –erroneous –orphaned –verbose
```

- Show all messages on the event queue:

  ManageQueue.sh –qm tv.qmgr –input_queue TVISION.EVENT.QUEUE –show_event_ids

  This command results in output similar to the following:

```
Count  Database Name    Database Host IP  CommLink ID  Schema ID  Validity
-----  -------------    ----------------  -----------  ---------  --------
  284  TVISION          172.22.22.46            2          1      Other DB
   22  TVDB             172.22.22.106           6          1      Orphaned
    7  TVDB             172.22.22.106           1          1      Valid
   17  TVDB             172.22.22.106           3          2      Valid
```

- Show all Database IDs:

  ManageQueue.sh –show_db_ids

  This command results in output similar to the following:

```
Schemas:
  1: TVISION
  2: TEST
Communication Links:
  1: TVision - tv.qmgr
  3: Test - TransactionVision.qmgr
```

- Connection to WebSphere MQ:

  ManageQueue.sh -qm tv.qmgr -input_queue TVISION.EVENT.QUEUE -delete -erroneous -orphaned -verbose

- Connection to TransactionVision SonicMQ:

  ManageQueue.sh -sonicmq -host win2003 -port 21111 -input_queue TVISION.EVENT.QUEUE -delete -erroneous -orphaned -verbose

- Move all TransactionVision events from the TransactionVision Exception Queue to the event queue. Target them to the TVISION schema and the TVision – TransactionVision.qmgr communication link.

  ManageQueue.sh –qm tv.qmgr –input_queue TVISION.EXCEPTION.QUEUE –output_queue TVSISION.EVENT.QUEUE –move “TVISION” “TVISION – tv.qmgr” -alltv

  or

  ManageQueue.sh –qm tv.qmgr –input_queue TVISION.EXCEPTION.QUEUE –output_queue TVISION.EVENT.QUEUE –move 1 1 –alltv

- Retarget all events not being collected because they are currently targeted to a schema and/or communication link which no longer exists (orphaned):

  ManageQueue.sh –qm tv.qmgr –input_queue TVISION.EVENT.QUEUE –output_queue TVISION.EVENT.QUEUE –move 1 1 -verbose

  or

  ManageQueue.sh –qm tv.qmgr –input_queue TVISION.EVENT.QUEUE –output_queue TVSION.EVENT.QUEUE –move “TVISION” “TVision – tv.qmgr” –verbose

- Move all TransactionVision events from the TransactionVision Exception Queue to their respective event queue. This example assumes that the schema TVISION uses the communication link “TVision – tv.qmgr” which in turn uses TVISION.EVENT.QUEUE, and schema TVISION2 uses communication link “Test – tv.qmgr” which in turn uses TVISION.TEST.QUEUE.

  ManageQueue.sh –qm tv.qmgr –input_queue TVISION.EXCEPTION.QUEUE –output_queue TVISION.EVENT.QUEUE -move –schema “TVISION” –commlink “TVision – tv.qmgr”

  ManageQueue.sh –qm tv.qmgr –input_queue TVISION.EXCEPTION.QUEUE –output_queue TVISION.TEST.QUEUE –move –schema “TEST” –commlink “Test – tv.qmgr”

- Archive all messages on the event queue into a jar file while leaving them on the queue:

  ManageQueue.sh –qm tv.qmgr –input_queue TVISION.EVENT.QUEUE –output_file events.jar –copy –verbose

- Replay all events stored in a jar file for the TVISION schema to the **TVision – tv.qmgr** communication link:

  ManageQueue.sh –qm tv.qmgr –input_file events.jar -output_queue TVISION.EVENT.QUEUE –copy “TVISION’ “TVision – tv.qmgr” -verbose

  or

  ManageQueue.sh –qm tv.qmgr –input_file events.jar –output_queue TVISION.EVENT.QUEUE –copy 1 1 -verbose

MigrateDB

Location:

<TVISION_HOME>/bin/MigrateDB.[sh|bat]

Purpose:

Migrates project database files from an older version of TransactionVision to the current version. This script has to be run after an upgrade installation process. It must be run in a configured TransactionVision environment; the **Database.properties** must be set correctly for communication with the database.

Syntax:

```
MigrateDB
```

nanny

Location:

<TVISION_HOME>/bin/nanny.[sh|bat]

Purpose:

Manages the services controlled by the Nanny Manager. TransactionVision components such as SonicMQ Domain Manager, SonicMQ Broker, and the Processing Server (Analyzers, query engine, and job manager) are services that are managed (started/stopped/monitored) by the Nanny Manager.

The Nanny Manager is a service (named HP Business Service Management) on Windows and a process on UNIX (nannyManager). This **nanny** utility can be used to find out which components are being managed by the Nanny Manager.

Syntax:

```
nanny  -s <hostname> -p <port> <cmd>
```

Options:

Option	Description
-s <hostname>	Hostname of the host where the Analyzer is installed. It defaults to localhost.
-p <port>	Port of the nanny server. **Default value**: 11020

Option	Description
<cmd>	One of the following commands: ➤ disableService <servicename> ➤ enableService <servicename> ➤ getServiceInfo <servicename> ➤ isServiceRunning <servicename> ➤ listAllDisabledServicesNames ➤ listAllServicesNames ➤ listDeadServicesNames ➤ listLiveServicesNames ➤ listServiceInfo ➤ listStartingServicesNames ➤ restartService <servicename> ➤ retrieveNannyManagerHTMLAdapterPort ➤ showStackTrace <servicename> ➤ startAllServices ➤ startService <servicename> ➤ stopAllServices ➤ stopService <servicename> where <servicename> is one of the following: ➤ tv_processmanager - The main Processing Server process ➤ tv_analyzer1 - the process for analyzer 1 […] ➤ tv_analyzer5 - the process for analyzer 5 ➤ tv_jobmanager - the job manager process ➤ tv_queryengine - the query engine process ➤ tv_message_broker - TransactionVision SonicMQ Message Broker ➤ tv_domain_manager - TransactionVision SonicMQ Domain Managerv_analyzer - TransactionVision Analyzer

Examples:

To list all services being managed by the Nanny Manager, enter:

```
nanny.bat listAllServiceNames
```

The command generates the following output:

```
Executing: listAllServicesNames Service:  on (localhost,11020)
Result = [tv_as, tv_message_broker, tv_domain_manager, tv_analyzer]
```

The **nanny** utility can enable or disable a component. Once a component is disabled, it is not managed by the Nanny Manager until it is enabled.

To disable the Analyzer, run the following command:

```
nanny.bat disableService tv_analyzer
```

To enable the Analyzer, run the following command:

```
nanny.bat enableService tv_analyzer
```

The **nanny** utility can also be used to start a component after it has been enabled.

To start the Analyzer component, run the following command:

```
nanny.bat startService tv_analyzer
```

PartitionUtil

Location:

<TVISION_HOME>/bin/PartitionUtil.[sh|bat]

Purpose:

Creates an SQL script for adding and dropping table partitions of TransactionVision tables database tables.

This utility only works on tables that were created with the table partitioning options of CreateSqlScript.

Syntax:

```
PartitionUtil -db DBTYPE [-add OPTIONS | -drop OPTIONS]

    add OPTIONS (date format: mm/dd/yyyy h:mm [am|pm]):
        -startNo PARTITION_NO -count COUNT -startDate DATE [-localTime]
        -length LENGTH -interval {hours | days | months} -schema SCHEMA

    drop OPTIONS:
        -startNo PARTITION_NO -count COUNT -schema SCHEMA
```

Options:

Option	Description
-db	The database type: 'oracle' or 'db2'.
-startNo	The start number of the partition to add or drop. Partitions are named PARTn, n = 1,2,… when created via CreateSqlScript.
-count	The number of partitions to add or drop.
-startDate	The start date for the first partition to add. The format for this date is: mm/dd/yyy h:mm [am|pm]. This time has to be in GMT, unless the option -localTime is used.
-localTime	Allows to specify the start date in local time.
-length	The length for each newly added partition.
-interval	The length unit for each added partition, valid values are 'hours', 'days', 'months'.
-schema	The database schema.

Example:

Generate SQL script for dropping the first two table partitions of project schema TRADE:

```
PartitionUtil.bat -db oracle -drop -s TRADE -startNo 1 -count 2
```

Generate SQL script for adding two new partitions spanning one day in tablespace TS1 to project schema TRADE. Assumption: five partitions (PART1 - PART5) have already been defined with CreateSqlScript, and the last partition reaches up to 04/01/2010 4:00 pm:

```
PartitionUtil.bat -db oracle -add -s TRADE -startNo 6 -count 2 -length 1 -interval
days -startDate "04/01/2010 4:00 pm" -ts TS1
```

PassGen

Location:

<TVISION_HOME>/bin/PassGen

Purpose:

A password can be obfuscated by using the PassGen utility.

Syntax:

```
PassGen /system <password>
```

Options:

Option	Description
-password	A string of alpha-numeric characters with a maximum length of 128 characters.

Example:

```
cd <installdir>/bin
$ PassGen /system TheLazyFoxJumpedHigh

password:
OBF:3q6r3xxz3y3r3xjs3wx03yc63n0r3lbr3vc03wd745893wre44u0413j3kn93zw
y40vi432i44fr3m453m894493439040pc40303kjd419r44na3wx0451h3wir3v6m3
lfr3mwj3yi03wre3xpi3xxz3y3r3q23
```

rebind_agent

Location:

<TVISION_HOME>/bin/rebind_agent.sh

Purpose:

This script rebinds the TransactionVision WebSphere MQ Agent on the AIX platform.

In WebSphere MQ support pacs, internal symbols exported from the TransactionVision WebSphere MQ Agent on the AIX platform may change. When an internal symbol that has been exported from the agent library is no longer available in the WebSphere MQ library, the application cannot start and fails with various symbol resolution errors.

Hence, the rebind_sensor script needs to be run whenever a WebSphere MQ support or fix pac is installed.

It modifies the TransactionVision agent libraries in **<TVISION_HOME>/lib**.

On WebSphere MQ 6.0 and above, this utility needs to be run twice, to instrument the 32-bit libraries and the 64-bit libraries, as follows:

```
$TVISION_HOME/bin/rebind_sensor.sh
$TVISION_HOME/bin/rebind_sensor.sh -64
```

Syntax:

```
rebind_sensor.sh [-v|-s|-h][-64]
```

Options:

Option	Description
-v	Writes errors to the console. The default behavior is to write errors to **<TVISION_HOME>/logs/mqsensorbind.log**.
-s	Uses silent mode, which does not prompt before executing.
-h	Displays usage message.
-64	Rebinds the 64-bit agent library. The default, when this option is not present, is to rebind the 32-bit library.

rebind_tux_sensor

Location:

<TVISION_HOME>/bin/rebind_tux_sensor.sh

Purpose:

This script rebinds the TransactionVision Tuxedo Agent on the AIX, Solaris and HP-UX platforms.

On HP-UX PA-RISC systems using a 32-bit version of Tuxedo, all Tuxedo applications should be shut down before running the **rebind_tux_sensor.sh** script. This will ensure the script does not encounter any conflicts due to another process using the Tuxedo library.

Syntax:

```
rebind_tux_sensor.sh [-v|-s|-h]
```

Options:

Option	Description
-v	Writes errors to the console. The default behavior is to write errors to **<TVISION_HOME>/logs/tuxsensorbind.log**.
-s	Uses silent mode, which does not prompt before executing.
-h	Displays usage message.

Usage Notes:

The Tuxedo Agent library must be linked to the Tuxedo library using its full path. If this is not done, an Tuxedo application monitored by the TransactionVision agent may fail with errors indicated unresolved library dependencies or unresolved symbols.

Hence, the rebind_tux_sensor.sh script needs to be run after TransactionVision agent installation and whenever Tuxedo is upgraded or its installation location is changed.

The **TVISION_HOME** and **TUXDIR** environment variables must be set before running this script. **TUXDIR** should be set to the full absolute path of the Tuxedo Agent installation directory. The script will modify the TransactionVision agent libraries in **<TVISION_HOME>/lib** or **<TVISION_HOME>/lib64** as necessary.

runSupportSnapshot

Location:

<Diagnostics_probe_install_dir>/contrib/JASMutilities/snapins/runSupportSnapshot.[sh|cmd]

Purpose:

The **runSupportSnapshot** utility creates a .zip file containing the entire set of files relevant to troubleshooting one or more instances of the Java agent in a Diagnostics or TransactionVision deployment environment. This .zip file is intended for Customer Support.

The .zip file contains the following:

- Files from the **<Diagnostics_probe_install_dir>\etc** directory
- Files from the **<Diagnostics_probe_install_dir>\log** directory
- Files from the **<TransactionVision_sensor_install_dir>\config** directory
- Files from the **<TransactionVision_sensor_install_dir>\logs** directory
- Property Scanner report, which compares a modified agent directory with the release version, and reports differences between the property files, points files, and for TransactionVision agents only, XML files.
- Probe or agent instance information, including property settings. For agents running in 1.5 JVMs, environment variables, stack dumps, and class loader information is also included.

Note: If a Diagnostics probe is configured to support SSL, this utility cannot collect the instance information.

Basic Syntax:

This utility uses defaults that apply to most agent environments. Therefore the typical syntax is simply:

```
runSupportSnapshot.[sh|cmd] -console
```

Full Syntax:

This utility also allows you specify options to override the defaults, allowing it to work in non-typical agent environments. This optional syntax is:

```
runSupportSnapshot.[sh|cmd] -console [ -Zipfile ] [ -AddFiles ]
[ -JAUser ] [ -JAPass ] [ -JAPort ] [ -JACount ]
[ -FTPSite ] [ -User ] [ -Pass ] [ -RemoteFile ]
[ -Source1] [ -Source2 ] [ -DiffOnly ] [ -SubDirs ] [ -Sort ]
```

Optional Syntax Options:

Option	Description
-ZipFile	The name of the .zip file that is to be created. **Default value**: **HPCustomerSupport.zip**
-AddFiles	Any additional directories to include in the .zip. To include all subdirectories of a directory, append an "+" to the directory name.
-JAUser	The user name to be used to access the host on which the Agent is running. **Default value**: admin
-JAPass	The corresponding password. **Default value**: admin
-JAPort	The port on which to start polling for Agent instances. Default: 35000
-JACount	The maximum number of Java Agent instances on this host for which data will be included in the .zip file. **Default value**: 20
-FTPSite	The name of the FTP site.
-User	The user name to be used to access the FTP site.
-Pass	The corresponding password.
-Remote File	The remote file name. Prefix the filename with directory path if desired. **Example: customer/diag/AgentSnapshot.zip**.
-Source1	The **etc** directory of the current Agent installation. **Default value**: **<Java_probe_install_dir>\etc**
-Source2	An archived copy of the out-of-the box **etc** directory in the directory. **Default value**: Internal directory to the agent installation
-DiffOnly	Only display properties with differences (yes or no). **Default value**: Yes

Option	Description
-SubDirs	Specify whether subdirectories are also be compared (yes or no). **Default value**: No
-Sort	Specify whether the output is sorted by property name (yes or no). **Default value**: No

Example:

```
> cd C:\MercuryDiagnostics\JavaAgent\DiagnosticsAgent\contrib\JASMUtilties
\Snapins
> java  -Dcom.hp.javaagent.diagnostics.home="..\..\.." -jar "..\..\..\lib\setupModule.jar"
-launchClass com.mercury.opal.javaprobe.setupModule.SetupModule  -launchMethod
launchSetupModule -importJarList
probe.jar,org.mortbay.jetty-jdk1.2.jar,javax.servlet.jar,mail.jar,activation.jar
-importJarsFrom "..\..\..\lib,lib" -customerSnapshot -console
INFORMATION-> [Looking for Java Agent Instances]: Looking for Java Agent
Instances...please wait...
INFORMATION-> [Java Agent found at http://localhost:35000/inst/
customerSupportSnapshot]: Java Agent Discovery
INFORMATION-> [Creating Probe Information file]: Java Agent Information
INFORMATION-> [Creating Configuration Information file]: Java Agent Configuration
Information
INFORMATION-> [Zip File Created]: Zip file
C:\HPCode\diag_head\javaprobe\build\contrib\JASMUtilities\Snapins\HPCustomerSup
portFile.zip Created with length 159215
```

If desired, review the .zip file for any sensitive data before sending it to Customer Support.

SetupModule

Location:

<java_agent_install_dir>\DiagnosticsAgent\lib

where <java_agent_install_dir> refers to the path of your Java Agent installation directory. The default path is **C:\MercuryDiagnostics\JavaAgent** on Windows and **\opt\MercuryDiagnostics\JavaAgent** on UNIX.

Purpose:
This script starts the Java Agent Setup Module.

Syntax:

```
SetupModule -recordFile name.rec -installFile name.rec -console
```

Options:

Option	Description
-recordFile name.rec	Records.
-installFile name.rec	Plays back recording or automates.
-console	Launches in console mode.

TimeServer

Location:
<TVISION_HOME>/bin/TimeServer.[sh|bat]

Purpose:
Run the time server stand-alone.

Syntax:

```
TimeServer[.sh|.bat] [PORT]
```

Options:

Option	Description
PORT	Optionally, specify the listening port for the time server to use. The default is 21104.

Usage Notes:

In TIBCO and generic JMS environments, TransactionVision is unable to determine time skew across a communication link using a COA as it can with WebSphere MQ JMS. Instead, a time server reports the current time on the host where it is running. Any TransactionVision component can connect to it via TCP/IP and get the time skew from the current host to the host running the time server.

The Analyzer itself can act as a time server, or it can be run on a different host. To run the time server within the Analyzer, modify the Analyzer.properties file as follows:

- Set the time_server property to on
- Optionally, set time_server_port to the listening port you would like it to use. The default port is 21104.

Use the TimeServer utility to run the time server stand-alone.

The time server is configured per communication link found on the Time Server page in the Communication Link Wizard and under the Time Server Definition in the Edit Communication Link page. This definition tells the Analyzer and agents where the tie server is to calculate time skew.

ValidateXml

Location:

<TVISION_HOME>/bin/ValidateXml.[sh|bat]

Purpose:

Validates TransactionVision XML configuration files. Currently only the files TransactionDefinition.xml, EventCorrelationDefinition.xml, and LocalTransactionDefinition.xml can be validated. It is necessary to edit the XML file and remove the comments around the DTD declaration.

Syntax:

```
ValidateXML XML_filename
```

11

Agent Error Messages

This chapter includes:

Reference

➤ Error Messages on page 424

Reference

Error Messages

By default, the agents log informational, error, warning, and trace messages to the local0 facility in the UNIX system log (syslogd), the Windows event log, the z/OS system log (SYSLOG), or the i5/OS user job log.

In the messages below, %s and %1-n represent text strings that are filled in with appropriate information at run time. The message identifiers shown before each message, for example SLDS101E, apply only for agents running under z/OS and do not appear on other platforms.

Identifier	Description
SLDS101E: TransactionVision Agent: failed to allocate space for %s	Out of memory error.
SLDS102E: TransactionVision Agent: failed to reallocate space for %s	Out of memory error.
SLDS103E: TransactionVision Agent: failed to create mutex lock	
SLDS104E: TransactionVision Agent: failed to lock mutex	
SLDS105E: TransactionVision Agent: failed to unlock mutex	
SLDS106E: TransactionVision Agent: failed to destroy mutex lock	
SLDS107E: TransactionVision Agent: error processing %s	
SLDS108E: TransactionVision Agent: Cannot connect to configuration queue manager %s: %s	The specified WebSphere MQ error occurred. See WebSphere MQ documentation for explanation.

Identifier	Description
SLDS109E: TransactionVision Agent: Cannot open configuration queue %s on queue manager %s: %s	The specified WebSphere MQ error occurred. See WebSphere MQ documentation for explanation.
SLDS10AE: TransactionVision Agent: Cannot reconnect to queue manager %s after user's MQDISC call: %s	The specified WebSphere MQ error occurred. See WebSphere MQ documentation for explanation.
SLDS10BE: TransactionVision Agent: Cannot reopen configuration queue %s on queue manager %s after user's MQDISC call: %s	The specified WebSphere MQ error occurred. See WebSphere MQ documentation for explanation.
SLDS10CE: TransactionVision Agent: Cannot connect event queue manager %s: %s	The specified WebSphere MQ error occurred. See WebSphere MQ documentation for explanation.
SLDS10DE: TransactionVision Agent: Cannot open event queue %s on queue manager %s: %s	The specified WebSphere MQ error occurred. See WebSphere MQ documentation for explanation
SLDS10EE: TransactionVision Agent: %d unsuccessful attempts were made to send an event message to queue ('%s' '%s'): %s Continuing to retry every %d ms...	Agent was unable to put an event message to the given event queue. Often, this is a result that occurs if the event queue becomes full if the agent is producing event messages faster than the Analyzer can pull them off the queue. The agent will continue to try to put to the event queue while the Analyzer is catches up. For information about configuring retry attempts, see "Event Options Page" on page 196.
SLDS10FE: TransactionVision Agent: Timed out (%d s) attempting %u times to put event message to queue ('%s' '%s')	Agent was unable to put an event message to the given queue after the stated number of attempts. Often the is a result that occurs if the event queue is full. For information about configuring retry attempts, see "Event Options Page" on page 196.

Identifier	Description
SLDS110I: TransactionVision Agent: successfully put an event message on queue ('%s' '%s') after %d unsuccessful attempts	Informational message stating that the agent has successfully put an event message on the event queue after it had to retry multiple times. For information about configuring retry attempts, see "Event Options Page" on page 196.
SLDS111E: TransactionVision Agent: tried to use invalid handle for putting to event queue ('%s' '%s')	An invalid handle was used to try to put to the event queue. This can occur if an application corrupts or passes a bad Hconn value to subsequent WebSphere MQ calls. The result is no event is sent to the Analyzer.
SLDS112E: TransactionVision Agent: Event message too big for queue ('%s' '%s') max message length. This error is reported only once per event queue and could be occurring more than once. Please increase max message length setting on the event queue.	The event queue max message length size is too small for an event message.
SLDS113W: TransactionVision Agent: attempt to disable crossing exit agent failed	An internal error has occurred.
SLDS114W: TransactionVision Agent: crossing exit agent has been disabled since both are active	If the standard TransactionVision WebSphere MQ Agent library is active, the Crossing Exit Agent is automatically disabled. Only one type of WebSphere MQ Agent can be active at any time.
SLDS115W: TransactionVision Agent: unexpected internal error - Agent turned off.	An internal error has occurred. The WebSphere MQ application will continue to run, but events will no longer be reported.
SLDS116W: TransactionVision Agent: failed to close %s queue '%s' '%s'	

Identifier	Description
SLDS116W: TransactionVision Agent: failed to close %s queue '%s' '%s'	The specified WebSphere MQ error occurred. See WebSphere MQ documentation for explanation.
SLDS117W: TransactionVision Agent: failed to get %s	The specified WebSphere MQ error occurred. See WebSphere MQ documentation for explanation.
SLDS118W: TransactionVision Agent: error removing configuration message from queue ('%s' '%s'): %s	Failed to destructively get an expired or invalid configuration message from the configuration queue because of the specified WebSphere MQ error.
SLDS119W: TransactionVision Agent: cannot extract %s from configuration message	Failed to extract a given field from a configuration message. Possibly an invalid configuration message.
SLDS11AE: TransactionVision Agent: error in communication infrastructure %s: %s	An error occurred with the TransactionVision communication infrastructure. A WebSphere MQ error is included with the error.
SLDS11BI: TransactionVision Agent: errors on queue manager ('%s') will be suppressed until successful communication resumes for this connection.	Informational message indicating that the agent will discontinue logging errors on the specified queue manager until it is able to communicate with the queue manager once again.
SLDS11CI: TransactionVision Agent: Successful communication has been made to the queue manager ('%s'). Logging will resume.	Informational trace message indicating that the agent is able to communication with the specified queue manager. Error logging had been suppressed for the queue manager while the agent was unable to communicate with it, but will now resume.
SLDS11DI: TransactionVision Agent: errors on configuration queue ('%s' '%s') will be suppressed until successful communication resumes.	Informational message indicating that the agent will discontinue logging errors on the specified configuration queue until it is able to communicate with the queue once again.

Identifier	Description
SLDS11EI: TransactionVision Agent: Successful communication has been made to the configuration queue ('%s' '%s'). Logging will resume.	Informational trace message indicating that the agent is able to communication with the specified configuration queue. Error logging had been suppressed for the queue while the agent was unable to communicate with it, but will now resume.
SLDS11FW: TransactionVision Agent: failed to unmarshall configuration message.	The agent failed to unmarshall a configuration message. Possibly an invalid configuration message.
SLDS120W: TransactionVision Agent: invalid %s version.	An invalid version of a given WebSphere MQ structure has been encountered when constructing the event message.
SLDS121W: TransactionVision Agent: invalid message on configuration queue ('%s' '%s') (not minimum configuration message size) (putting to dead letter queue).	An invalid configuration message has been encountered on the configuration queue. The agent will put the invalid message on the exception message queue.
SLMS122W:TransactionVision Agent: invalid message on configuration queue ('%s' '%s') (configuration length does not match message length) (putting to dead letter queue).	An invalid configuration message has been encountered on the configuration queue. The agent will put the invalid message on the exception message queue.
SLDS123W:TransactionVision Agent: invalid message on configuration queue ('%s' '%s') (product name incorrect in configuration message) (putting to dead letter queue)	An invalid configuration message has been encountered on the configuration queue. The agent will put the invalid message on the exception message queue.
SLDS124W: TransactionVision Agent: unsupported configuration message version found on ('%s' '%s') (putting to dead letter queue)	An invalid configuration message has been encountered on the configuration queue. The agent will put the invalid message on the exception message queue.
SLDS125W:TransactionVision Agent: failed to remove configuration message from configuration queue ('%s' '%s') and put to dead letter queue ('%s' '%s'): %s	The agent was unable to remove a configuration message from the configuration queue and place it on the exception message queue. A WebSphere MQ error accompanies this message.

Identifier	Description
SLDS126W: TransactionVision Agent: invalid data range value in configuration message	
SLDS127W: TransactionVision Agent: missing expected token separator in configuration message when parsing %s	
SLDS128W: TransactionVision Agent: missing expected space separator in configuration message when parsing %s	
SLDS129W: TransactionVision Agent: missing expected subsection separator in configuration message when parsing %s	
SLDS12AW: TransactionVision Agent: missing expected section separator in configuration message when parsing %s	
SLDS12BW: TransactionVision Agent: missing expected data range dash in configuration message	
SLDS12CE: TransactionVision Agent: bad checksum for %s	
SLDS12DI: TransactionVision Agent: opening configuration queue '%s' '%s'	Informational trace message.
SLDS12EI: TransactionVision Agent: opening event queue '%s' '%s'	Informational trace message.
SLDS12FI: TransactionVision Agent: got configuration message from '%s' '%s'	Informational trace message.
SLDS130I: TransactionVision Agent: invalidating all previous configuration messages on '%s' '%s' from source analyzer	Informational trace message. Invalidating and removing all configurations from a given Analyzer previous to the configuration message currently being processed.

Identifier	Description
SLDS131I: TransactionVision Agent: invalidating specified configuration message on '%s' '%s' from source analyzer	Informational trace message. Invalidating and removing a given configuration message from a given Analyzer specified by the configuration message currently being processed.
SLDS132I: TransactionVision Agent: removing invalidated configuration message from '%s' '%s' MsgId: %s CorrelId: %s	Informational trace message.
SLDS133I: TransactionVision Agent: removing expired configuration message from '%s' '%s' current GMT: %s expire time: %s MsgId: %s CorrelId: %s	Informational trace message.
SLDS134I: TransactionVision Agent: already have this configuration message from '%s' '%s'	Informational trace message. A duplicate configuration message has been found on the configuration queue.
SLDS135I: TransactionVision Agent: event message (%u bytes) too large for event queue. Removing user data to make event message smaller.	The event will be included in the project, but the user data buffer will not be available.
SLDS136I: TransactionVision Agent: this host ('%s') does not appear in filter rule	The host the agent is running on does not match the collection filter criteria for hosts.
SLDS137I: TransactionVision Agent: this user name ('%s') does not appear in filter rule	The user name associated with the message does not match the collection filter criteria for user names.
SLDS138I: TransactionVision Agent: this program ('%s') does not appear in filter rule	The program name of the program using the agent does not match the collection filter criteria.
SLDS139I: TransactionVision Agent: current time is out of filter start/stop time range: current GMT: %s start time: %s stop time: %s	The current time does not match the collection time specified in the collection filter.

Identifier	Description
SLDS13AI: TransactionVision Agent: this CICS SYSID (‘%s’) does not appear in filter rule	The CICS SYSID does not match the collection filter criteria.
SLDS13BI: TransactionVision Agent: this CICS transaction id (‘%s’) does not appear in filter rule	The transaction id does not match the collection filter criteria.
SLDS13CI: TransactionVision Agent: this job name (‘%s’) and step (‘%s’) pair does not appear in filter rule	The z/OS Batch job name and step combination does not match the collection filter criteria.
SLDS13DI: TransactionVision Agent: the job name (‘%s’) does not appear in filter rule	The i5/OS job name does not match the collection filter criteria.
SLDS13EI: TransactionVision Agent: the IMS identifier (‘%s’) does not appear in filter rule	The IMS identifier does not match the collection filter criteria.
SLDS13FI: TransactionVision Agent: the IMS region type (‘%s’) does not appear in filter rule	The IMS region type does not match the collection filter criteria.
SLDS140I: TransactionVision Agent: the IMS region identifier (‘%s’) does not appear in filter rule	The IMS region identifier does not match the collection filtering criteria.
SLDS141I: TransactionVision Agent: the IMS transaction name (‘%s’) does not appear in filter rule	The IMS transaction name does not match the collection filtering criteria.
SLDS142I: TransactionVision Agent: the IMS PSB name (‘%s’) does not appear in filter rule	The IMS PSB name does not match the collection filtering criteria.
SLDS143I: TransactionVision Agent: API name (‘%s’) does not appear in filter rule	The API name does not match the collection filter criteria.
SLDS144I: TransactionVision Agent: completion code (%u) does not appear in filter rule	The WebSphere MQ API completion code does not match the collection filter criteria.
SLDS145I: TransactionVision Agent: reason code (%u) does not appear in filter rule	The WebSphere MQ API reason code does not match the collection filter criteria.

Identifier	Description
SLDS146I: TransactionVision Agent: technology (‘%s’) in configuration does not match this technology (‘%s’)	The event technology does not match the collection filter criteria.
SLDS147I: TransactionVision Agent: queue manager does not appear in filter rule: ‘%s	The queue manager does not match the data collection filter criteria.
SLDS148I: TransactionVision Agent: queue manager and/or object name does not appear in filter rule: (Conn: ‘%s’) ‘%s’ ‘%s’	The queue manager or WebSphere MQ object name does not match the collection filter criteria.
SLDS149I: TransactionVision Agent: ReplyTo queue manager and/or queue name does not appear in filter rule: ‘%s’ ‘%s’	The ReplyTo queue manager or queue name does not match the collection filter criteria.
SLDS14BI: TransactionVision Agent: collection mask does not match %s for %s	The specified entity does not match the collection filter criteria.
SLDS14EI: TransactionVision Agent: MQCMIT has been filtered out because all other events in the unit of work have been filtered out”	The MQCMIT was filtered out as specified in the collection filter criteria.
SLDS14FI: TransactionVision Agent: Exit part of MQDISC has been filtered out”	The MQDISC exit was filtered out as specified in the collection filter criteria.
SLDS150I: TransactionVision Agent: browsing MQGET has been filtered out	The “Do not send browsing MQGET” option is selected in the collection filter criteria.
SLDS151I: TransactionVision Agent: ignoring unmatched configuration message from ‘%s’ ‘%s’	Ignoring a configuration message because a standard collection filter criteria did not match with the running agent.
SLDS152I: TransactionVision Agent: creating filter rule from matched configuration message from ‘%s’ ‘%s’ expire time: %s MsgID: %s CorrelId: %s	The agent matches the standard collection filter criteria of a given configuration message.
SLDS153I: TransactionVision Agent: removing expired filter rule from memory current GMT: %s expire time: %s	A configuration message that has already been retrieved from the configuration queue has expired.

Identifier	Description
SLDS154I: putting event message on event queue ('%s' '%s') MsgId: %s CorrelId: %s	Informational trace message.
SLDS155I: Management Initiated for MQSeries API Crossing Exit	Informational trace message.
SLDS156I: TransactionVision Agent: Disabling MQSeries API Crossing Exit	Informational trace message.
SLDS157I: TransactionVision Agent: Enabling MQSeries API Crossing Exit	Informational trace message.
SLDS158I: TransactionVision Agent: SLMC Exiting	Informational trace message.
SLDS159I: TransactionVision Agent: The agent is being used with another TransactionVision component. Please remove the agent path from the library search path before starting the Analyzer or Java Agent.	The system library path environment variable has been set to load the agent instead of the real WebSphere MQ library. This library path should be set only for applications being monitored, but not for the Analyzer or the Servlet, JMS, or EJB Agents.
SLDS15AE: TransactionVision Agent: An unexpected error occured in an API exit. The agent will be turned off.	An internal error has occurred.
SLDS15BE: TransactionVision Agent: An unexpected error occurred. The agent will be turned off.	An internal error has occurred.
SLDS15CE: TransactionVision Agent: An unexpected error occurred. The agent will be turned off. Please contact HP Technology with the stack trace that follows. %s	An internal error has occurred. Contact HP Support, providing the stack trace information that follows the error message.
SLDS15DE: TransactionVision Agent: An unexpected error occurred. The agent will be turned off. A failure occurred while trying to produce a stack trace: %s%u	An internal error has occurred.

Identifier	Description
SLDS15EE: TransactionVision Agent: An unexpected error occurred. The agent will be turned off. A failure occurred while trying to produce a stack trace: %s	An internal error has occurred.
SLDS15FE: TransactionVision Agent: invalid pointer passed to API exit ('%s'): %s	An internal error has occurred.
SLDS160E: TransactionVision Agent: NULL parameter passed to API exit"	An internal error has occurred.
SLDS161E: TransactionVision Agent: NULL exit parameters passed to API exit	An internal error has occurred.
SLDS162E: TransactionVision Agent: NULL exit user area passed to API exit	An internal error has occurred.
SLDS163E: TransactionVision Agent: failed to register API exit: %d	An internal error has occurred.
SLDS164E: TransactionVision Agent: failed to allocate space for %s, package size=%d,package count=%d,max size=%d.	The agent was unable to allocate sufficient space for the event package because the package size exceeds the maximum size.
SLDS165E: TransactionVision Agent: Failed to connect to queue manager ('%s') for WebSphere MQ object resolution: (%d) %s	The Agent was unable to connect to the specified queue manager.
SLDS166E: TransactionVision Agent: Failed to disconnect from queue manager ('%s') for WebSphere MQ object resolution: (%d) %s	The agent was unable to disconnect from the specified queue manager.
SLDS272E: TransactionVision Agent: wrong number of parameters.	The agent command contains an incorrect number of parameters. For instructions on starting and stopping the CICS Agent, see the "Configuring the CICS agent," chapter in the *HP TransactionVision Deployment Guide* PDF.
SLDS273E: TransactionVision Agent: input queue manager name (%s) length is bigger than %d (MQ limit).	The specified queue manager name exceeds the allowed length.

Identifier	Description
SLDS274E: FailMemCDC: E: "%s%03XE %s:TransactionVision Agent: Failed to allocate memory for SLDI.	Out of memory error. The CICS Agent will fail to initialize and stop. Try to restart the CICS Agent. If the same problem happens, contact HP Support.
SLDS275E: FailMemCCT: E: "%s%03XE %s:TransactionVision Agent: Failed to allocate memory for config table.	Out of memory error. The CICS Agent will fail to initialize and stop. Try to restart the CICS Agent. If the same problem happens, contact HP Support.
SLDS276E: TransactionVision Agent: Unknown error occurred (type=%d), CICS Agent Driver is shutting down.	
SLDS277I: TransactionVision Agent: The routine that generated abend in CICS Agent Driver is '%.80s', and offset is %08X. Please contact HP Support with the dump information.	Contact HP Support and report the diagnostic information included in the message.
SLDS278E: InitFailed: E: "%s%03XE %s: TransactionVision Agent: CICS %s Agent Driver startup failed. QMGR=%s, CONFIGQ=%s.	CICS Agent driver program failed to initialize due to error(s) previous shown.
SLDS279I: TransactionVision Agent: CICS %s Agent Driver startup completed. QMGR=%s, CONFIGQ=%s.	Informational message indicating the CICS Agent Driver startup completed successfully.
SLDS27AE: WrongEventType: E: "%s%03XE %s: TransactionVision Agent: Unknown event type (type=%d), skip this event.	The event type is not collectable by the CICS Agent, and the event is skipped.
SLDS27BI: TransactionVision Agent: CICS %s Agent Driver is ending, return code = %d.	Informational message indicating the CICS Agent Driver is ending.
SLDS27CI: TransactionVision Agent: this terminal id ('%s') does not appear in filter rule	The terminal ID does not match the collection filter criteria.

Identifier	Description
SLDS27DI: TransactionVision Agent: this API type ('%d') does not appear in filter rule	The API type does not match the collection filter criteria.
SLDS27EI: TransactionVision Agent: this API ('%d') does not appear in filter rule	The API does not match the collection filter criteria.
SLDS27FI: TransactionVision Agent: this file ('%s') does not appear in filter rule	The file does not match the collection filter criteria.
SLDS280I: TransactionVision Agent: this TD queue ('%s') does not appear in filter rule	The TD queue does not match the collection filter criteria.
SLDS281I: TransactionVision Agent: this TS queue ('%s') does not appear in filter rule	The TS queue does not match the collection filter criteria.
SLDS282I: TransactionVision Agent: this EIBRESP ('%d') does not appear in filter rule	The response does not match the collection filter criteria.
SLDS400I: TVISION [TransactionVision Manager | CICS *sysid* TransactionVision agent] startup in progress.	Informational message issued in response to a start TransactionVision Manager command or a start agent command, where *sysid* is the SYSID of the CICS region to monitor.
SLDS401I: TVISION [TransactionVision Manager *tvid* | CICS *sysid* TransactionVision agent] startup complete.	Informational message issued when TransactionVision Manager startup or agent startup is complete, where tvid is the TVID of the TransactionVision Manager and sysid is the SYSID of the CICS region to monitor.
SLDS402I: *tvid* STOP command received.	Informational message issued in response to a stop TransactionVision Manager command, where tvid is the TVID of the TransactionVision Manager.

Identifier	Description
SLDS404I: TVISION [TransactionVision Manager *tvid* | CICS *sysid* TransactionVision agent] termination in progress.	Informational message issued in response to a stop TransactionVision Manager command or a stop Agent command, where tvid is the TVID of the TransactionVision Manager and sysid is the SYSID of the monitored CICS region.
SLDS405I: TVISION [TransactionVision Manager *tvid* | CICS *sysid* TransactionVision agent] termination complete.	Informational message issued when TransactionVision Manager shutdown or Agent shutdown is complete, where tvid is the TVID of the TransactionVision Manager and sysid is the SYSID of the monitored CICS region.
SLDS406E: No parameters specified on the START command. The TVID parameter is required.	A start TransactionVision Manager command was issued without the required TVID parameter.
SLDS407E: The TVID parameter is missing from the START command.	A start TransactionVision Manager command was issued without the required TVID parameter.
SLDS408E: The TVID specified on the START command contains more than the maximum of 4 characters.	A start TransactionVision Manager command specified an invalid TVID parameter.
SLDS409E: The SYSID parameter is missing from the START command.	An incorrect START command has been issued and the TransactionVision CICS A start agent command was issued without the required SYSID parameter.
SLDS410E: The SYSID specified on the START command contains more than the maximum of 4 characters	A start agent command specified an invalid SYSID parameter.
SLDS411E: TVISION *tvid* TransactionVision Manager already started.	A start TransactionVision Manager command specified a TVID value that matches a running TransactionVision Manager, where tvid is the TVID specified.

Identifier	Description
SLDS412E: TVISION CICS *sysid* TransactionVision agent already started.	A start agent command specified a SYSID value that matches a running agent under control of the same TransactionVision Manager, where sysid is the SYSID specified.
SLDS413E: TVISION CICS sysid TransactionVision agent already started by another TransactionVision Manager.	A start Agent command specified a SYSID value that matches a running agent under control of a different TransactionVision Manager, where sysid is the SYSID specified.
SLDS415E: TVISION The parameter value specified on the command is less than the required minimum of min_value.	A start Agent command specified a parameter value less than the minimum required, where parameter is the parameter keyword and min_value is the minimum value allowed.
SLDS416E: TVISION The parameter value specified on the command is greater than the maximum allowed of max_value.	A start Agent command specified a parameter value more than the maximum required, where parameter is the parameter keyword and max_value is the maximum value allowed.
SLDS418E: TVISION The parameter value specified on the command contains non-numeric characters.	A start Agent command specified non-numeric characters for a parameter value that is required to be numeric, where parameter is the parameter keyword.
SLDS420E: TVISION Command name, command_name, invalid. Command ignored.	An invalid command was issued, where command_name is the command.
SLDS421E: TVISION Command operand, operand_name, invalid for command_name command. Command ignored.	An invalid command operand was specified, where operand_name is operand specified and command_name is the command.

Identifier	Description
SLDS422E: TVISION Operand value specified on command name or operand that takes no value specification. Command ignored.	A name(value) form was specified on a command when no value was expected for the command or operand name specified.
SLDS423E: TVISION Syntax error - misplaced "(". Command ignored.	Syntax error.
SLDS424E: TVISION Syntax error - misplaced ")". Command ignored.	Syntax error.
SLDS425E: TVISION Command input is all spaces. Command ignored.	Syntax error.
SLDS427E: TVISION The agent driver startup procedure name contains more than the maximum of 8 characters.	The value of the DRVRPROC parameter on a start Agent command is invalid.
SLDS428E: TVISION The required operand value for the keyword operand, operand_name, is missing.	A value was not specified for an operand that required a value, where operand_name is the operand name.
SLDS429E: TVISION The required agent type operand is missing.	The Agent type, CICS, was omitted from the command. In this release, CICS is the only Agent type supported but subsequent releases will add other types.
SLDS430E: TVISION Invalid agent type specified.	The only valid agent type in this release is CICS.
SLDS431E: TVISION Unsupported agent type specified.	The only valid agent type in this release is CICS.
SLDS432E: TVISION The required command_name command operand, operand_name, is missing.	A required operand was not specified on the command, where command_name is the command and operand_name is the omitted operand.
SLDS440E: TVISION The agent specified on the STOP AGENT command is not started.	A stop Agent command was issued for an agent that is not started.

Identifier	Description
SLDS441S: TVISION Unable to allocate the minimum size buffer queue. Diagnostic data follow:	This error probably results from an installation limit on the size of data spaces. If not, contact HP Support and report the diagnostic data included in the message.
SLDS442W: TVISION The maximum number of allowed buffer queue blocks specified on the MAXQBLKS startup parameter has been reached. Events may be lost.	The agent required an additional queue block but the maximum allowed was already allocated. When the current queue block is full and the maximum is still allocated, events will be lost.
SLDS443I: TVISION Agent quiescing: *n* events in buffer queue.	A stop agent command has been issued. The agent will wait until all events in the buffer queue have been retrieved before completing the shutdown.
SLDS445I: TVISION Agent quiesce completed.	All events in the buffer queue have been retrieved in response to an agent stop command. The agent will complete the shutdown.
SLDS446E: TVISION The agent manager component is not started.	This error may be caused by starting the agent driver manually. The agent driver is automatically started by the agent manager and should not be started any other way. If not the case, contact HP Support.
SLDS448I: TVISION CICS CIC1 Agent statistics: Events in queue: n Events collected:n Events dispatched: n Events_lost:n	Informational message issued at agent shutdown or in response to an INQUIRE command.
SLDS449W: TVISION agent events have been lost due to insufficient buffer queue storage.	The agent manager issues this message the first time an event is lost.

Identifier	Description
SLDS451S: TVISION CICS *sysid* TransactionVision agent stall condition detected. Diagnostic data follow:	The Agent manager is quiescing and detects that events are not being retrieved from the buffer queue by the agent driver, where *sysid* identifies the agent. The agent will exit quiesce mode and complete shutdown. Events in the buffer queue will be lost. Contact HP Support and report the diagnostic data included in the message.
SLDS460I: TVISION Agent exits manager started.	Informational message issued in response to an SLDS transaction or PLTPI processing.
SLDS461I: TVISION Agent exits manager waiting for agent startup.	The agent exits manager has been started but a corresponding agent manager has not.
SLDS462I: TVISION Agent startup completed. Exits monitoring started.	The agent exits manager has detected the startup of a corresponding agent manager.
SLDS463W: TVISION The agent exits manager is already started. Request ignored.	An SLDS transaction was invoked but the agent exits manager is already started.
SLDS464I: TVISION *exit_name* exit [enabled | disabled].	The agent exits manager has enabled or disabled an exit, where exit_name is the name of the exit.
SLDS465I: TVISION Agent exits manager terminated.	An informational message issued when the agent exits manager completes shutdown as a result of an SLDP transaction, PLTSD processing, or the forced disablement of the program start exit.
SLDS466W: TVISION The agent exits manager is not started. Request ignored.	An SLDP or an SLDC transaction was invoked but the exits manager is not started.
SLDS467W: TVISION The request (*1st parm*) was not En(able) or Dis(able).	An SLDC transaction was invoked but the first parameter was invalid.
SLDS468W: TVISION The request (*2nd parm*) was not: PS, KC, PC, IC, TD, TS, or FC.	An SLDC transaction was invoked but the second parameter was invalid.

Identifier	Description
SLDS469I TVISION The program start exit (PS) cannot be disabled. The exit is stopped.	An SLDC transaction specified disablement of the program start exit, which must remain enabled for the exits manager to run. The exit is stopped instead.
SLDS498S: TVISION [TransactionVision Manager | CICS *sysid* TransactionVision agent] system error: message handler unavailable. Diagnostic data follow:	Contact HP Support and report the diagnostic data included in the message.
SLDS499S: TVISION [TransactionVision Manager | CICS *sysid* TransactionVision agent] system error. Diagnostic data follow:	Contact HP Support and report the diagnostic data included in the message.
SLMS101E: TransactionVision agent: failed to allocate space for %s	Out of memory error.
SLMS102E: TransactionVision agent: failed to reallocate space for %s	Out of memory error.
SLMS103E: TransactionVision agent: failed to create mutex lock	
SLMS104E: TransactionVision agent: failed to lock mutex	
SLMS105E: TransactionVision agent: failed to unlock mutex	
SLMS106E: TransactionVision agent: failed to destroy mutex lock	
SLMS107E: TransactionVision agent: error processing %s	Error processing given field within a configuration message.
SLMS108E: TransactionVision agent: Cannot connect to configuration queue manager %s: %s	The specified WebSphere MQ error occurred. See WebSphere MQ documentation for explanation.
SLMS109E: TransactionVision agent: Cannot open configuration queue %s on queue manager %s: %s	The specified WebSphere MQ error occurred. See WebSphere MQ documentation for explanation.

Identifier	Description
SLMS10AE: TransactionVision agent: Cannot reconnect to queue manager %s after user's MQDISC call: %s	The specified WebSphere MQ error occurred. See WebSphere MQ documentation for explanation.
SLMS10BE: TransactionVision agent: Cannot reopen configuration queue %s on queue manager %s after user's MQDISC call: %s	The specified WebSphere MQ error occurred. See WebSphere MQ documentation for explanation.
SLMS10CE: TransactionVision agent: Cannot connect event queue manager %s: %s	The specified WebSphere MQ error occurred. See WebSphere MQ documentation for explanation.
SLMS10DE: TransactionVision agent: Cannot open event queue %s on queue manager %s: %s	The specified WebSphere MQ error occurred. See WebSphere MQ documentation for explanation.
SLMS10EE: TransactionVision agent: %d unsuccessful attempts were made to send an event message to queue ('%s' '%s'): %s Continuing to retry every %d ms...	Agent was unable to put an event message to the given event queue. Often, this is a result that occurs if the event queue becomes full if the agent is producing event messages faster than the Analyzer can pull them off the queue. The agent will continue to try to put to the event queue while the Analyzer is catches up. For information about configuring retry attempts, see "Event Options Page" on page 196.
SLMS10FE: TransactionVision agent: Timed out (%d s) attempting %u times to put event message to queue ('%s' '%s')"	Agent was unable to put an event message to the given queue after the stated number of attempts. Often this is a result that occurs if the event queue is full. For information about configuring retry attempts, see "Event Options Page" on page 196.

Identifier	Description
SLMS110I: TransactionVision agent: successfully put an event message on queue ('%s' '%s') after %d unsuccessful attempts	Informational message stating that the agent has successfully put an event message on the event queue after it had to retry multiple times. For information about configuring retry attempts, see "Event Options Page" on page 196.
SLMS111E: TransactionVision agent: tried to use invalid handle for putting to event queue ('%s' '%s')	An invalid handle was used to try to put to the event queue. This can occur if an application corrupts or passes a bad Hconn value to subsequent WebSphere MQ calls. The result is no event is sent to the Analyzer.
SLMS112E: TransactionVision agent: Event message too big for queue ('%s' '%s') max message length. This error is reported only once per event queue and could be occurring more than once. Please increase max message length setting on the event queue.	The event queue max message length size is too small for an event message. For information about setting maximum message length, see "Event Options Page" on page 196.
SLMS113W: TransactionVision agent: attempt to disable crossing exit agent failed.	An internal error has occurred.
SLMS114W: TransactionVision agent: API Exit Agent has been disabled because the Library agent is active.	If the standard TransactionVision WebSphere MQ Agent library is active, the Crossing Exit Agent is automatically disabled. Only one type of WebSphere MQ Agent can be active at any time.
SLMS115W: TransactionVision agent: unexpected internal error - agent turned off.	An internal error has occurred. The WebSphere MQ application will continue to run, but events will no longer be reported.
SLMS116W: TransactionVision agent: failed to close %s queue '%s' '%s'	The specified WebSphere MQ error occurred. See WebSphere MQ documentation for explanation.

Identifier	Description
SLMS117W: TransactionVision agent: failed to get %s	The specified WebSphere MQ error occurred. See WebSphere MQ documentation for explanation.
SLMS118W: TransactionVision agent: error removing configuration message from queue ('%s' '%s'): %s	Failed to destructively get an expired or invalid configuration message from the configuration queue because of the specified WebSphere MQ error.
SLMS119W: TransactionVision agent: cannot extract %s from configuration message	Failed to extract a given field from a configuration message. Possibly an invalid configuration message.
SLMS11AE: TransactionVision agent: error in communication infrastructure %s: %s	An error occurred with the TransactionVision communication infrastructure. A WebSphere MQ error is included with the error.
SLMS11BI: TransactionVision agent: errors on queue manager ('%s') will be suppressed until successful communication resumes for this connection.	Informational message indicating that the agent will discontinue logging errors on the specified queue manager until it is able to communicate with the queue manager once again.
SLMS11CI: TransactionVision agent: Successful communication has been made to the queue manager ('%s'). Logging will resume.	Informational trace message indicating that the agent is able to communication with the specified queue manager. Error logging had been suppressed for the queue manager while the agent was unable to communicate with it, but will now resume.
SLMS11DI: TransactionVision agent: errors on configuration queue ('%s' '%s') will be suppressed until successful communication resumes.	Informational message indicating that the agent will discontinue logging errors on the specified configuration queue until it is able to communicate with the queue once again.

Identifier	Description
SLMS11EI: TransactionVision agent: Successful communication has been made to the configuration queue ('%s' '%s'). Logging will resume.	Informational trace message indicating that the agent is able to communication with the specified configuration queue. Error logging had been suppressed for the queue while the agent was unable to communicate with it, but will now resume.
SLMS11FW: TransactionVision agent: failed to unmarshall configuration message.	The agent failed to unmarshall a configuration message. Possibly an invalid configuration message.
SLMS120W: TransactionVision agent: invalid %s version.	An invalid version of a given WebSphere MQ structure has been encountered when constructing the event message.
SLMS121W: TransactionVision agent: invalid message on configuration queue ('%s' '%s') (not minimum configuration message size) (putting to dead letter queue).	An invalid configuration message has been encountered on the configuration queue. The agent will put the invalid message on the exception message queue.
SLMS122W:TransactionVision agent: invalid message on configuration queue ('%s' '%s') (configuration length does not match message length) (putting to dead letter queue).	An invalid configuration message has been encountered on the configuration queue. The agent will put the invalid message on the exception message queue.
SLMS123W:TransactionVision agent: invalid message on configuration queue ('%s' '%s') (product name incorrect in configuration message) (putting to dead letter queue).	An invalid configuration message has been encountered on the configuration queue. The agent will put the invalid message on the exception message queue.
SLMS124W: TransactionVision agent: unsupported configuration message version found on ('%s' '%s') (putting to dead letter queue).	An invalid configuration message has been encountered on the configuration queue. The agent will put the invalid message on the exception message queue.

Identifier	Description
SLMS125W:TransactionVision agent: failed to remove configuration message from configuration queue (‘%s’ ‘%s’) and put to dead letter queue (‘%s’ ‘%s’): %s.	The agent was unable to remove a configuration message from the configuration queue and place it on the exception message queue. A WebSphere MQ error accompanies this message.
SLMS126W: TransactionVision agent: invalid data range value in configuration message.	
SLMS127W: TransactionVision agent: missing expected token separator in configuration message when parsing %s.	
SLMS128W: TransactionVision agent: missing expected space separator in configuration message when parsing %s.	
SLMS129W: TransactionVision agent: missing expected subsection separator in configuration message when parsing %s.	
SLMS12AW: TransactionVision agent: missing expected section separator in configuration message when parsing %s.	
SLMS12BW: TransactionVision agent: missing expected data range dash in configuration message.	
SLMS12CE: TransactionVision agent: bad checksum for %s.	
SLMS12DI: TransactionVision agent: opening configuration queue ‘%s’ ‘%s’.	Informational trace message.
SLMS12EI: TransactionVision agent: opening event queue ‘%s’ ‘%s’.	Informational trace message.
SLMS12FI: TransactionVision agent: got configuration message from ‘%s’ ‘%s’.	Informational trace message.

Identifier	Description
SLMS130I: TransactionVision agent: invalidating all previous configuration messages on '%s' '%s' from source analyzer.	Informational trace message. Invalidating and removing all configurations from a given Analyzer previous to the configuration message currently being processed.
SLMS131I: TransactionVision agent: invalidating specified configuration message on '%s' '%s' from source analyzer.	Informational trace message. Invalidating and removing a given configuration message from a given Analyzer specified by the configuration message currently being processed.
SLMS132I: TransactionVision agent: removing invalidated configuration message from '%s' '%s' MsgId: %s CorrelId: %s.	Informational trace message.
SLMS133I: TransactionVision agent: removing expired configuration message from '%s' '%s' current GMT: %s expire time: %s MsgId: %s CorrelId: %s.	Informational trace message
SLMS134I: TransactionVision agent: already have this configuration message from '%s' '%s'.	Informational trace message. A duplicate configuration message has been found on the configuration queue.
SLMS135I: TransactionVision agent: event message (%u bytes) too large for event queue. Removing user data to make event message smaller.	The event will be included in the project, but the user data buffer will not be available.
SLMS136I: TransactionVision agent: this host ('%s') does not appear in filter rule.	The host the agent is running on does not match the collection filter criteria for hosts.
SLMS137I: TransactionVision agent: this user name ('%s') does not appear in filter rule.	The user name associated with the message does not match the collection filter criteria for user names.

Identifier	Description
SLMS138I: TransactionVision agent: this program (‘%s’) does not appear in filter rule.	The program name of the program using the agent does not match the collection filter criteria.
SLMS139I: TransactionVision agent: current time is out of filter start/stop time range: current GMT: %s start time: %s stop time: %s.	The current time does not match the collection time specified in the collection filter.
SLMS13AI: TransactionVision agent: this CICS SYSID (‘%s’) does not appear in filter rule.	The CICS SYSID does not match the collection filter criteria.
SLMS13BI: TransactionVision agent: this CICS transaction id (‘%s’) does not appear in filter rule.	The transaction id does not match the collection filter criteria.
SLMS13CI: TransactionVision agent: this job name (‘%s’) and step (‘%s’) pair does not appear in filter rule.	The z/OS Batch job name and step combination does not match the collection filter criteria.
SLMS13DI: TransactionVision agent: the job name (‘%s’) does not appear in filter rule.	The i5/OS job name does not match the collection filter criteria.
SLMS13EI: TransactionVision agent: the IMS identifier (‘%s’) does not appear in filter rule.	The IMS identifier does not match the collection filter criteria.
SLMS13FI: TransactionVision agent: the IMS region type (‘%s’) does not appear in filter rule.	The IMS region type does not match the collection filter criteria.
SLMS140I: TransactionVision agent: the IMS region identifier (‘%s’) does not appear in filter rule.	The IMS region identifier does not match the collection filtering criteria.
SLMS141I: TransactionVision agent: the IMS transaction name (‘%s’) does not appear in filter rule.	The IMS transaction name does not match the collection filtering criteria.
SLMS142I: TransactionVision agent: the IMS PSB name (‘%s’) does not appear in filter rule.	The IMS PSB name does not match the collection filtering criteria.

Identifier	Description
SLMS143I: TransactionVision agent: API name ('%s') does not appear in filter rule.	The WebSphere MQ API does not match the collection filter criteria.
SLMS144I: TransactionVision agent: completion code (%u) does not appear in filter rule.	The WebSphere MQ API completion code does not match the collection filter criteria.
SLMS145I: TransactionVision agent: reason code (%u) does not appear in filter rule.	The WebSphere MQ API reason code does not match the collection filter criteria.
SLMS146I: TransactionVision agent: technology ('%s') in configuration does not match this technology ('%s').	The event technology does not match the collection filter criteria.
SLMS147I: TransactionVision agent: queue manager does not appear in filter rule: '%s'.	The queue manager does not match the data collection filter criteria.
SLMS148I: TransactionVision agent: queue manager and/or object name does not appear in filter rule: (Conn: '%s') '%s' '%s'.	The queue manager or WebSphere MQ object name does match the collection filter criteria.
SLMS149I: TransactionVision agent: ReplyTo queue manager and/or queue name does not appear in filter rule: '%s' '%s'.	The ReplyTo queue manager or queue name does not match the collection filter criteria.
SLMS14BI: TransactionVision agent: collection mask does not match %s for %s.	The specified entity does not match the collection filter criteria.
SLMS14EI: TransactionVision agent: MQCMIT has been filtered out because all other events in the unit of work have been filtered out.	The MQCMIT was filtered out as specified in the collection filter criteria.
SLMS14FI: TransactionVision agent: Exit part of MQDISC has been filtered out.	The MQDISC exit was filtered out as specified in the collection filter criteria.
SLMS150I: TransactionVision agent: browsing MQGET has been filtered out.	The "Do not send browsing MQGET" option is selected in the collection filter criteria.
SLMS151I: TransactionVision agent: ignoring unmatched configuration message from '%s' '%s'.	Ignoring a configuration message because a standard collection filter criteria did not match with the running agent.

Identifier	Description
SLMS152I: TransactionVision agent: creating filter rule from matched configuration message from '%s' '%s' expire time: %s MsgID: %s CorrelId: %s.	The agent matches the standard collection filter criteria of a given configuration message.
SLMS153I: TransactionVision agent: removing expired filter rule from memory current GMT: %s expire time: %s.	A configuration message that has already been retrieved from the configuration queue has expired.
SLMS154I: putting event message on event queue ('%s' '%s') MsgId: %s CorrelId: %s.	Informational trace message.
SLMS155I: Management Initiated for MQSeries API Crossing Exit.	Informational trace message.
SLMS156I: TransactionVision agent: Disabling MQSeries API Crossing Exit.	Informational trace message.
SLMS157I: TransactionVision agent: Enabling MQSeries API Crossing Exit.	Informational trace message.
SLMS158I: TransactionVision agent: SLMC Exiting.	Informational trace message.
SLMS159I: TransactionVision agent: The agent is being used with another TransactionVision component. Please remove the agent path from the library search path before starting the analyzer or Java Agent.	The system library path environment variable has been set to load the agent instead of the real WebSphere MQ library. This library path should be set only for applications being monitored, but not for the Analyzer or the Servlet, JMS, or EJB Agents.
SLMS15AE: TransactionVision agent: An unexpected error occurred in an API exit. The agent will be turned off.	An internal error has occurred.
SLMS15BE: TransactionVision agent: An unexpected error occurred. The agent will be turned off.	An internal error has occurred.

Identifier	Description
SLMS15CE: TransactionVision agent: An unexpected error occurred. The agent will be turned off. Please contact HP Support with the stack trace that follows. %s.	An internal error has occurred.
SLMS15DE: TransactionVision agent: An unexpected error occurred. The agent will be turned off. A failure occurred while trying to produce a stack trace: %s%u.	An internal error has occurred.
SLMS15EE: TransactionVision agent: An unexpected error occurred. The agent will be turned off. A failure occurred while trying to produce a stack trace: %s.	An internal error has occurred
SLMS15FE: TransactionVision agent: invalid pointer passed to API exit ('%s'): %s.	An internal error has occurred
SLMS160E: TransactionVision agent: NULL parameter passed to API exit.	An internal error has occurred
SLMS161E: TransactionVision agent: NULL exit parameters passed to API exit.	An internal error has occurred
SLMS162E: TransactionVision agent: NULL exit user area passed to API exit.	An internal error has occurred
SLMS163E: TransactionVision agent: failed to register API exit: %d.	An internal error has occurred
SLMS164E: TransactionVision agent: failed to allocate space for %s, package size=%d,package count=%d,max size=%d.	The agent was unable to allocate sufficient space for the event package because the package size exceeds the maximum size.
SLMS165E: TransactionVision agent: Failed to connect to queue manager ('%s') for WebSphere MQ object resolution: (%d) %s.	The agent was unable to connect to the specified queue manager.
SLMS166E: TransactionVision agent: Failed to disconnect from queue manager ('%s') for WebSphere MQ object resolution: (%d) %s.	The agent was unable to disconnect from the specified queue manager.

Identifier	Description
SLMS201E: TransactionVision agent: cannot initialize MQSeries library.	A general error reporting that the loading and initialization of the real WebSphere MQ library failed. WebSphere MQ may not be properly installed, or the library may not be in the library search path.
SLMS251E: TransactionVision agent: could not connect to queue manager: '%s'.	The agent was unable to connect to the queue manager (a WebSphere MQ message is supplied). For information about setting connection retry attempts, see "Event Options Page" on page 196.
SLMS252E: TransactionVision agent: could not connect to queue manager '%s': %s The agent will attempt to reconnect after %d seconds. Events may not be collected during this time.	The agent was unable to connect to the queue manager (a WebSphere MQ message is supplied). It will attempt to reconnect, but no events will be collected until it succeeds. For information about setting connection retry attempts, see "Event Options Page" on page 196.
SLMS253E: TransactionVision agent: Error marshalling event data: null pointer '%s'.	
SLMS254E: TransactionVision agent: Unknown error occurred (type=%d), Event Dispatcher is shutting down.	
SLMS255I: TransactionVision agent: The routine that generated abend in Event Dispatcher is '%.80s', and offset is %08X.	
SLMS256E: TransactionVision agent: Wrong number of parameters. Please specify Queue Manager name only.	Too many parameters were specified; specify the Queue Manager name only.
SLMS257E: TransactionVision agent: Input queue manager name (%s) length is bigger than %d (MQ limit).	The queue manager name is too long.

Identifier	Description
SLMS258E: TransactionVision agent: Event Dispatcher initialization failed.	
SLMS259I: TransactionVision agent: Event Dispatcher initialization completed.	Informational trace message.
SLMS25AI: TransactionVision agent: Event Dispatcher is ending as requested by TVISIONB.	Informational trace message.
SLMS25BE: Unknown function ID.	
SLMS25CE: TransactionVision agent: Unknown return code from SLMXGNE.	
SLMS271E: WrongNumberOfParameters: E: %s%03XE %s:TransactionVision agent: wrong number of parameters.	The agent command contains an incorrect number of parameters. See the "Configuring the CICS Agent," chapter in the the *HP TransactionVision Deployment Guide* PDF for instructions on starting and stopping the CICS Agent.
SLMS272E: TransactionVision agent: input queue manager name (%s) length is bigger than %d (MQ limit).	The specified queue manager name exceeds the allowed length.
SLMS273E: TransactionVision agent: Failed to allocate memory for SLDI.	Out of memory error. The CICS Agent will fail to initialize and stop. Try to restart the CICS Agent, if the same problem happens, contact HP support.
SLMS274E: TransactionVision agent: Failed to allocate memory for config table.	Out of memory error. The CICS Agent will fail to initialize and stop. Try to restart the CICS Agent, if the same problem happens, contact HP support.
SLMS275E: TransactionVision agent: Unknown error occurred (type=%d), %s agent Driver is shutting down.	

Identifier	Description
SLMS276I: TransactionVision agent: The routine that generated abend is '%.80s', and offset is %08X. Please contact HP Support with the dump information.	Contact HP Support and report the diagnostic information included in the message.
SLMS277E: TransactionVision agent: %s %s agent Driver startup failed. QMGR=%s, CONFIGQ=%s.	CICS Agent driver program failed to initialize due to error(s) previous shown.
SLMS278I: TransactionVision agent: %s %s agent Driver startup completed. QMGR=%s, CONFIGQ=%s.	Informational message indicating the CICS Agent Driver startup completed successfully.
SLMS279E: TransactionVision agent: Unknown event type (type=%d), skip this event.	The event type is not collectable by the CICS Agent, and the event is skipped.
SLMS27AI: TransactionVision agent: %s %s agent Driver is ending, return code = %d.	Informational message indicating the CICS Agent Driver is ending.
SLMS27BI: TransactionVision agent: this terminal id ('%s') does not appear in filter rule.	The terminal ID does not match the collection filter criteria.
SLMS27CI: TransactionVision agent: this API type ('%s') does not appear in filter rule.	The API type does not match the collection filter criteria.
SLMS27DI: TransactionVision agent: this API ('%s') does not appear in filter rule.	The API does not match the collection filter criteria.
SLMS27EI: TransactionVision agent: this file ('%s') does not appear in filter rule.	The CICS file name does not match the collection filter criteria.
SLMS27FI: TransactionVision agent: this TD queue ('%s') does not appear in filter rule.	The TD queue does not match the collection filter criteria.
SLMS280I: TransactionVision agent: this TS queue ('%s') does not appear in filter rule.	The TS queue does not match the collection filter criteria.
SLMS281I: TransactionVision agent: this EIBRESP ('%d') does not appear in filter rule.	The response code does not match the collection filter criteria.

Identifier	Description
SLMS282E: TransactionVision agent: Null pointer detected for the structure '%s'.	
SLMS283E: TransactionVision agent: Sturcture ID '%s' does not match the expected value '%s'.	
SLMS284E: TransactionVision agent: Internal error. This function '%s' should never be called. Please contact HP Support.	
SLMS301I: TransactionVision WMQI Agent: %s.	Informational message regarding the WebSphere MQ Integrator-enabled Agent.
SLMS302I: TransactionVision WMQI Agent: inserting attribute '%s'.	Informational message regarding the WebSphere MQ Integrator-enabled Agent.
SLMS303I: TransactionVision WMQI Agent: inserting %s terminal entry for terminal '%s'.	Informational message regarding the WebSphere Business Integration-enabled Agent.
SLMS304I: TransactionVision WMQI Agent: getting attribute by index %d.	Informational message regarding the WebSphere Business Integration-enabled Agent.
SLMS305I: TransactionVision WMQI Agent: getting attribute '%s'.	Informational message regarding the WebSphere Business Integration-enabled Agent.
SLMS306I: TransactionVision WMQI Agent: setting attribute '%s' to '%s'.	Informational message regarding the WebSphere Business Integration-enabled Agent.
SLMS307E: TransactionVision WMQI Agent: failed to allocate space for %s.	Out of memory error.
SLMS308E: TransactionVision WMQI Agent: failed to initialize %s.	
SLMS309E: TransactionVision WMQI Agent: invalid data type for %s.	

Identifier	Description
SLMS30AE: TransactionVision WMQI Agent: failure terminal does not exist.	
SLMS30BE: TransactionVision WMQI Agent: out terminal does not exist.	
SLMS30CE: TransactionVision WMQI Agent: out terminal is not attached.	
SLMS30DE: TransactionVision WMQI Agent: unknown attribute '%s'.	
SLMS30EE: TransactionVision WMQI Agent: attribute index %d not found.	
SLMS30FE: TransactionVision WMQI Agent: %s terminal '%s' not found.	
SLMS310I: TransactionVision WMQI Agent: loaded TransactionVision agent.	Informational message regarding the WebSphere Business Integration-enabled Agent.
SLMS311E: TransactionVision WMQI Agent: failed to load TransactionVision agent.	Ensure that the agent is properly installed. For installation instructions, see the *HP TransactionVision Deployment Guide* PDF.
SLMS312I: TransactionVision WMQI Agent: TransactionVision agent successfully loaded.	Informational message regarding the WebSphere Business Integration-enabled Agent.
SLMS313E: TransactionVision WMQI Agent: Could not find Path value in the registry. Ensure that the TransactionVision agent is properly installed.	Ensure that the agent is properly installed. For installation and configuration instructions, see the *HP TransactionVision Deployment Guide* PDF.
SLMS314E: TransactionVision WMQI Agent: Could not open TransactionVision agent key in registry. Ensure that the TransactionVision agent is properly installed.	Ensure that the agent is properly installed. For installation instructions, see the *HP TransactionVision Deployment Guide* PDF.

Identifier	Description
SLMS315E: TransactionVision WMQI Agent: could not load TransactionVision agent library ('%s'): %s.	Ensure that the agent is properly installed. For installation instructions, see the *HP TransactionVision Deployment Guide* PDF.
SLMS400I: TVISIONB startup in progress.	Informational message. TVISIONB has been started and startup is in progress. This message should be followed by SLMS401I when startup is complete.
SLMS401I: TVISIONB startup complete.	Informational message. TVISIONB startup is complete and the MQSeries-IMS bridge Agent server component is ready to receive events from the bridge.
SLMS402I: STOP command received.	Informational message. The MQSeries-IMS bridge Agent server component is shutting down. Bridge monitoring is disabled to prevent the sending of further events from the OTMA Input/Output Edit exit routine in the IMS control region(s) and all tasks in the TVISIONB address space are terminated. This message should be followed by SLMS403I when termination is complete. There may be a delay while in-flight events are dispatched.
SLMS403I: TVISIONB termination complete.	Informational message. All tasks in the MQSeries-IMS bridge Agent server component have been terminated. The TVISIONB and TVISIOND address spaces have been terminated.
SLMS4041: TVISIONB termination in progress.	TVISIONB is terminating in response to a STOP command or due to an error condition.

Identifier	Description
SLMS4051: TVISIONB dispatching remaining events in buffer queue.	TVISIONB is in quiesce mode in response to a STOP command. When all events in the buffer queue have been dispatched, TVISIONB termination will complete. The SLMS415I message follows to report the number of events in the buffer queue.
SLMS4061: Immediate TVISIONB termination requested. Events in buffer queue will be lost.	TVISIONB is terminating in response to a STOP IMMED command issued while there are events in the buffer queue. The SLMS415I message follows to report the number of events in the buffer queue.
SLMS410I: TVISIONB MQIMSBDG monitoring disabled.	Response to a DISABLE MQIMSBDG command. MQSeries-IMS bridge monitoring is disabled regardless of Analyzer requests.
SLMS411W: TVISIONB configuration messages have requested MQIMSBDG monitoring, which has been disabled by the system operator. Monitoring will not resume until enabled by the operator.	Warning message issued when configuration messages from an Analyzer request MQSeries-IMS bridge monitoring but monitoring has been disabled by a DISABLE MQIMSBDG command.
SLMS412I: TVISIONB MQIMSBDG monitoring already disabled.	Response to a DISABLE MQIMSBDG command when MQSeries-IMS bridge monitoring has already been disabled by a previous DISABLE MQIMSBDG command.
SLMS413I: TVISIONB MQIMSBDG monitoring enabled.	Response to an ENABLE MQIMSBDG command. MQSeries-IMS bridge monitoring is enabled.
SLMS414I: TVISIONB MQIMSBDG monitoring already enabled.	Response to an ENABLE MQIMSBDG command when MQSeries-IMS bridge monitoring is already enabled.

Identifier	Description
SLMS419E: TVISIONB unable to process command.	An invalid command format or unknown command has been issued or TVISIONB encountered a system error attempting to process a command. If the command is correct, contact HP Support and report the diagnostic data included in the message.
SLMS420W: Possible TVISIONB stall condition detected. Some requested events may be not be collected.	The TVISIONB event recording function is not responding in a timely manner. This condition may be caused by a full TransactionVision event queue because the queue has been defined too small or no Analyzer service is running to relieve the queue. If the event queue is full, expand its size and/or assure that an Analyzer is running. Otherwise, contact HP Support.
SLMS421S: The event dispatcher task has unexpectedly terminated. TVISIONB will terminate.	TVISIOND is terminating due to an error condition.
SLMS430E: TVISIONB already started.	An second instance of TVISIONB has been started. Only one instance is allowed. The second instance will be terminated.
SLMS431E: The MQSeries queue manager name missing. Please specify as follows: S TVISIONB,QMGR=CSQ1, for example.	The WebSphere MQ queue manager, to which TVISIONB connects to access its configuration and event queues, must be identified at startup.
SLMS432E: The MQSeries queue manager name specified on the S TVISIONB command contains more than the maximum of 4 characters.	WebSphere MQ queue manager names on z/OS are limited to four characters.
SLMS433E: The MAXQ size specified on the S TVISIONB command is less than the minimum requirement of 3.	At least 3MB of storage are required for TVISIONB to manage its buffer queue.

Identifier	Description
SLMS434E: The MAXQ size specified on the S TVISIONB command is greater than the maximum allowed of 2046.	The maximum amount of storage that TVISIONB can use to manage its buffer queue is 2046MB (2GB less 1MB).
SLMS435E: The MAXQ size specified on the S TVISIONB command contains non-numeric characters.	The TVISIONB MAXQ parameter must be numeric.
SLMS436E: The MAXQ size specified on the S TVISIONB command contains too many characters.	The TVISIONB MAXQ parameter must contain a maximum of 15 characters.
SLMS438E: The event dispatcher startup procedure name contains more than the maximum of 8 characters.	The EDPROC parameter on the TVISIONB startup command or procedure is incorrect.
SLMS440E: Invalid TVISIONB command entered.	An invalid command format or unknown command has been issued.
SLMS441E: Unknown target of TVISIONB DISABLE command.	A DISABLE command has been issued but the object name specified is unknown. In this version of TransactionVision, the only supported object is MQIMSBDG.
SLMS442E: Unknown target of TVISIONB ENABLE command.	A ENABLE command has been issued but the object name specified is unknown. In this version of TransactionVision, the only supported object is MQIMSBDG.
SLMS450S: TVISIONB ENQ error. Diagnostic data follow:	Contact HP Support and report the diagnostic data included in the message.
SLMS451S: TVISIONB unable to allocate required storage. Diagnostic data follow:	This may be caused by an too small REGION size for TVISIONB, which requires an address space size of the MAXQ specification plus 1MB. If the REGION size is sufficient, contact HP Support and report the diagnostic data included in the message.

Identifier	Description
SLMS452S: TVISIONB unable to release obtained storage. Diagnostic data follow:	Contact HP Support and report the diagnostic data included in the message.
SLMS453S: TVISIONB ATTACH error. Diagnostic data follow:	This may be caused by a required program module missing from STEPLIB or SSLMLOAD libraries specified in the TVISIONB startup procedure. Verify that all the modules listed in the installation instructions are located in the appropriate libraries and those libraries are specified in the TVISIONB startup procedure.
SLMS454S: TVISIONB LINK error. Diagnostic data follow:	This may be caused by a required program module missing from STEPLIB or SSLMLOAD libraries specified in the TVISIONB startup procedure. Verify that all the modules listed in the installation instructions are located in the appropriate libraries and those libraries are specified in the TVISIONB startup procedure. If the setup is correct, contact HP Support and report the diagnostic data included in the message.
SLMS455S: TVISIONB LOAD error. Diagnostic data follow:	This may be caused by a required program module missing from STEPLIB or SSLMLOAD libraries specified in the TVISIONB startup procedure. Verify that all the modules listed in the installation instructions are located in the appropriate libraries and those libraries are specified in the TVISIONB startup procedure. If the setup is correct, contact HP Support and report the diagnostic data included in the message.
SLMS456S: TVISIONB DELETE error. Diagnostic data follow:	Contact HP Support and report the diagnostic data included in the message.

Identifier	Description
SLMS457S: TVISIONB OPEN error. Diagnostic data follow:	Contact HP Support and report the diagnostic data included in the message.
SLMS458S: TVISIONB CLOSE error. Diagnostic data follow:	Contact HP Support and report the diagnostic data included in the message.
SLMS459S: TVISIONB BLDL error. Diagnostic data follow:	Contact HP Support and report the diagnostic data included in the message.
SLMS460S: TVISIONB ASCE error. Diagnostic data follow:	Contact HP Support and report the diagnostic data included in the message.
SLMS461S: TVISIONB ASEXT error. Diagnostic data follow:	Contact HP Support and report the diagnostic data included in the message.
SLMS472S: TVISIONB IEANTCR error. Diagnostic data follow:	Contact HP Support and report the diagnostic data included in the message.
SLMS473S: TVISIONB IEANTRT error. Diagnostic data follow:	Contact HP Support and report the diagnostic data included in the message.
SLMS474S: TVISIONB IEANTDL error. Diagnostic data follow:	Contact HP Support and report the diagnostic data included in the message.
SLMS475S: TVISIONB ESTAEX error. Diagnostic data follow:	Contact HP Support and report the diagnostic data included in the message.
SLMS485S: Insufficient storage available for TVISIONB queue requirements. Diagnostic data follow:	This may be caused by a too small REGION size for TVISIONB, which requires an address space size of the MAXQ specification plus 1MB. This error occurs when TVISIONB attempts to allocate its initial buffer queue space of 2MB during startup. TVISIONB is terminated. If the REGION size is sufficient, please contact HP Support and report the diagnostic data included in the message.

Identifier	Description
SLMS486W: Insufficient storage available for TVISIONB queue expansion requirements. Events may be lost.	This may be caused by a too small REGION size for TVISIONB, which requires an address space size of the MAXQ specification plus 1MB. This error occurs when TVISIONB attempts to allocate 1MB to expand its buffer queue space within the MAXQ limitation specified at TVISIONB startup. When all already-allocated queue space is exhausted, subsequent events will be discarded. This condition may be caused by a full TransactionVision event queue because the queue has been defined too small or no Analyzer service is running to relieve the queue. If the event queue is full, expand its size and/or assure that an Analyzer is running. If the event queue is not full and the REGION size is sufficient, please contact HP Support.
SLMS487W: Insufficient storage available for TVISIONB queue expansion requirements. MAXQ limit reached. Events may be lost.	TVISIONB has used almost all of the buffer queue space allowed by the MAXQ limitation specified at TVISIONB startup. When all allowed queue space is exhausted, subsequent events will be discarded. This condition may be caused by a full TransactionVision event queue because the queue has been defined too small or no Analyzer service is running to relieve the queue. If the event queue is full, expand its size and/or assure that an Analyzer is running. Otherwise, increase the MAXQ specification.
SLMS488S: TVISIONB system stall condition detected. Queued events will be lost.	Contact HP Support and report the diagnostic data included in the message.

Identifier	Description
SLMS498S: TVISIONB system error. Diagnostic data follow:	Contact HP Support and report the diagnostic data included in the message.
SLMS499S: TVISIONB system error. Diagnostic data follow:	Contact HP Support and report the diagnostic data included in the message.

12

APIs Monitored by Agents

This chapter includes:

Reference

- CICS Agent on page 468
- WebSphere MQ Agent on page 468
- WebSphere MQ-IMS Bridge Agent on page 469
- Servlet Technology in the Java Agent on page 469
- JMS Technology in the Java Agent on page 471
- EJB Technology in the Java Agent on page 473
- JDBC Technology in the Java Agent on page 474
- Tuxedo Agent on page 475
- NonStop TMF Agent on page 476
- .Net Agent on page 476
- DataPower Agents on page 477

Reference

CICS Agent

The CICS Agent collects events for five groups of CICS APIs: file control, temporary storage, transient data, interval control, and program control. In addition, the CICS Agent collects Task Start, Task End and Program Start events as well.

- File Control APIs

  APIs: WRITE, REWRITE, READ, STARTBR, RESETBR, READNEXT, READPREV, ENDBR, DELETE, UNLOCK

- Temporary Storage APIs

  WRITEQ TS, READQ TS, DELETEQ TS

- Transient Data APIs

  WRITEQ TD, READQ TD, DELETEQ TD

- Interval Control APIs

  DELAY, START, ATTACH, CANCEL

- Program Control APIs

  LINK, XCTL, RETURN

WebSphere MQ Agent

The WebSphere MQ library, API exit and the z/OS WebSphere MQ Agents collect events from the following MQ APIs.

- MQBACK
- MQBEGIN
- MQCLOSE
- MQCMIT

- MQCONN
- MQCONNX
- MQDISC
- MQGET
- MQINQ
- MQOPEN
- MQPUT
- MQPUT1
- MQSET
- MQSUB
- MQSUBRQ

WebSphere MQ-IMS Bridge Agent

The following events are generated by the WebSphere MQ-IMS Bridge Agent when a MQ message is received by the MQ-IMS bridge and when a reply is generated by the WebSphere MQ-IMS bridge.

- MQIMS_BRIDGE_ENTRY
- MQIMS_BRIDGE_EXIT

Servlet Technology in the Java Agent

The following servlet classes are instrumented:

- javax.servlet.http.HttpServlet
- org.apache.jasper.runtime.HttpJspBase
- com.ibm.servlet.PageListServlet

The servlet technology tracks the following methods:

void doGet(javax.servlet.http.HttpServletRequest,
javax.servlet.http.HttpServletResponse)

void doPost(javax.servlet.http.HttpServletRequest,
javax.servlet.http.HttpServletResponse)

void doPut(javax.servlet.http.HttpServletRequest,
javax.servlet.http.HttpServletResponse)

void service(javax.servlet.http.HttpServletRequest,
javax.servlet.http.HttpServletResponse)

void doDelete(javax.servlet.http.HttpServletRequest,
javax.servlet.http.HttpServletResponse)

void doHead(javax.servlet.http.HttpServletRequest,
javax.servlet.http.HttpServletResponse)

void doOptions(javax.servlet.http.HttpServletRequest,
javax.servlet.http.HttpServletResponse)

void doTrace(javax.servlet.http.HttpServletRequest,
javax.servlet.http.HttpServletResponse)

void init()

void init(javax.servlet.ServletConfig)

void _jspService(javax.servlet.http.HttpServletRequest,
javax.servlet.http.HttpServletResponse)

void jspInit()

Note: The TRACE and OPTIONS requests are different from the other requests because the servlet container has the following default implementation of these two requests:

- For TRACE requests, the servlet container returns the whole request string made by the client.
- For OPTIONS requests, the servlet container returns the list of HTTP requests handled by that particular servlet.

If you do NOT override the doTrace and doOptions methods, TransactionVision is unable to trace these requests, although the client still gets the response.

JMS Technology in the Java Agent

The JMS technology tracks the following methods:

Class:: ConnectionFactory
createConnection
createXAConnection

Class:: TopicConnectionFactory
createTopicConnection
createXATopicConnection

Class:: QueueConnectionFactory
createQueueConnection

Class:: Connection
createConnectionConsumer
createDurableConnectionConsumer
createSession
createXASession
start
stop
setExceptionListener

Class:: QueueConnection
- createQueueSession
- createXAQueueSession

Class:: TopicConnection
- createTopicSession
- createXATopicSession

Class:: Session
- createProducer
- createConsumer

Class:: TopicSession
- createDurableSubscriber
- createPublisher
- createSubscriber
- unsubscribe
- commit
- recover
- rollback
- close

Class:: QueueSession
- createBrowser
- createReceiver
- createSender
- commit
- recover
- rollback
- createTemporaryQueue
- close

Class:: MessageProducer

Class:: MessageConsumer

Class:: QueueSender
- send
- close

Class:: TopicPublisher
publish
close

Class:: QueueBrowser
close
getEnumeration

Class:: QueueEnumeration
nextElement
hasMoreElements

Class:: TopicSubscriber
close
receive
setMessageListener

Class:: QueueReceiver
close
receive
setMessageListener

Class:: Message
acknowledge

EJB Technology in the Java Agent

The following EJB classes are instrumented:

- javax.ejb.EntityBean
- javax.ejb.SessionBean
- javax.ejb.MessageDrivenBean
- javax.ejb.EnterpriseBean (parent interface of all three above)

In addition to all business methods defined in public interfaces, the following methods are instrumented:

- ejbCreate()
- ejbPostCreate()

- ejbRemove() (session/entity/message driven bean)
- ejbLoad() (entity bean)
- ejbStore() (entity bean)
- onMessage() (message driven bean)

JDBC Technology in the Java Agent

The JDBC technology monitors the following interfaces and methods:

- Interface java.sql.CallableStatement
  - all execute methods
  - all set* methods
- Interface java.sql.Connection
  - close
  - commit
  - releaseSavepoint
  - rollback
  - setSavePoint
  - prepareCall
  - prepareStatement
- Interface java.sql.Driver
  - connect
- Interface java.sql.PreparedStatement
  - execute
  - executeQuery
  - executeUpdate
  - all set* methods

- Interface java.sql.ResultSet
  - close
  - deleteRow
  - insertRow
  - refreshRow
  - updateRow
  - all get* methods
  - cursor movement methods (next, first, and so on)
- Interface java.sql.Statement
  - addBatch
  - cancel
  - clearBatch
  - close
  - execute
  - executeBatch
  - executeQuery
  - executeUpdate

Tuxedo Agent

The Tuxedo Agent collects events from the following methods in the ATMI API:

- tpcall
- tpenqueue
- tpdequeue

NonStop TMF Agent

The NonStop TMF Agent monitors the following operations on audited Enscribe files:

- Insert
- Modify
- Delete

.Net Agent

The .NET Agent tracks the following:

- Web Services

  ASP.NET (*.asmx) - Client and Server
  WCF (*.svc) - Client and Server

- Database calls executed using ADO.NET
- NET Remoting - Client and Server
- MSMQ - Send and Receive (asynchronous)
- HTTP

  Client outbound
  ASP.NET inbound/server (POST, GET, PUT) (*.aspx)

For more information about configuring event collection in the .NET Agent, see the *HP TransactionVision Deployment Guide* PDF.

DataPower Agents

The DataPower Agent does not monitor a particular API. It monitors calls to DataPower Agents on the WebSphere DataPower SOA Integration Appliance XI50.

For information about configuring the DataPower Agent, see the *HP TransactionVision Deployment Guide* PDF.

For information about DataPower event correlation, see "Correlation For DataPower Events" on page 96.

13

Troubleshooting

Note: Troubleshooting information is also available in specific chapters including Chapter 2, "Processing Servers," Chapter 3, "Analyzers," Chapter 4, "Communication Links," and "Reports and Topologies."

This chapter includes:

Reference

- Enabling SHLIB_PATH on HP-UX on page 481
- Displaying Perl and Java Program Names on page 481
- Null Pointer Exception After BSM Login on page 482
- Monitoring WebSphere MQ Triggered Programs on page 482
- Monitoring Client Applications on page 483
- Optimizing Performance on page 484
- Missing Events on page 484
- Queue Manager or Event Broker Channel Limitations on page 485
- Servlet and EJB Technology Problems on page 485
- JMS Technology Problems on page 486
- Monitored Applications Troubleshooting on page 486
- Event Queue Cleanup on page 487
- Invalid Schema and Communication Link IDs on page 487
- Problems Importing System Tables into a Database on page 490

- Connection to Database Is Lost on page 494
- Problems Using International Characters on page 494
- Agent Configuration Queue Manager or Event Broker on page 496
- Analyzer Configuration Queue Manager or Event Broker on page 498
- Schemas Missing New Tables on page 499
- Database Deadlocks in IBM DB2 Databases on page 499
- Unable to Communicate with Processing Server after Enabling SSL on page 500

Reference

Enabling SHLIB_PATH on HP-UX

When you run a program on HP-UX, the loader may use the library paths embedded in the program instead of the library paths specified in the **SHLIB_PATH** environment variable. You may use the **chatr** command to check the current settings on a program:

```
chatr executable_name
```

The output of this command may be as follows:

```
shared executable
shared library dynamic path search:
  SHLIB_PATH     disabled  second
  embedded path  enabled   first /opt/mqm/lib
```

To enable run-time location of the WebSphere MQ Library Agent, you may use the following command to turn on the loader's use of SHLIB_PATH as the first paths to search for libraries:

```
chatr +s enable +b enable executable_name
```

Displaying Perl and Java Program Names

By default, the TransactionVision displays Perl or java.exe as the program name for Perl or Java applications. To display the program name, set the **TVISION_REPORT_ARGS** environment variable to any value. to disable this behavior, unset **TVISION_REPORT_ARGS**. This environment variable can be set on any platform except z/OS. On Windows, it must be set to a value other than an empty string.

Null Pointer Exception After BSM Login

If you go to the **Admin** > **Transaction Management** > **Configuration** tab (in the left panel), right after logging in to BSM, you may get a null pointer exception error. If this occurs, click the **Refresh** button on the Configuration tab.

To be sure this will not happen again, use the JRE 1.6 update 24 for all Windows platforms.

To install the JRE 1.6 update 24 version:

1. Uninstall the existing JRE version used in the browser.
2. Download and install the JRE 1.6 update 24 version from:

   http://java.sun.com.
3. Restart the browser and launch BSM.

Monitoring WebSphere MQ Triggered Programs

WebSphere MQ triggered programs on UNIX platforms can be monitored if two conditions are met:

1. The runmqtrm trigger monitor is not setuid/gid, or you are running runmqtrm as the mqm user. (This is because the programs started by the setuid/gid runmqtrm ignore any environment variables modifying the path if the user is not the setuid/gid user - the mqm user in this case.)
2. You actually trigger a script instead of your program. The script sets the appropriate environment variable (based on the OS platform) to point to the agent and then runs the actual program. For example, on Solaris this script might be:

```
#!/bin/sh
LD_LIBARAY_PATH=/opt/HP/TransactionVision/lib
  export LD_LIBRARY_PATH
  /usr/local/bin/trigproc “$*”
```

Where /usr/local/bin/trigproc was the program being triggered.

The following table shows the appropriate environment variable and default installation directory for each platform:

Platform	Environment Variable	Default Directory
Windows	**PATH**	**C:\Program Files\Hewlett-Packard\ TransactionVision Sensor\lib**
Sun Solaris	**LD_LIBRARY_PATH**	**/opt/HP/TransactionVision/lib**
HP-UX	**SHLIB_PATH**	**/opt/HP/TransactionVision/lib**
Linux	**LD_LIBRARY_PATH**	**/opt/HP/TransactionVision/lib**
IBM AIX	**LIBPATH**	**/usr/lpp/HP/TransactionVision/ lib**

Note: The **runmqtrm** trigger monitor program cannot itself be sensed by the agent, because it uses internal WebSphere MQ functions which are not compatible with the agent. To sense the trigger monitor itself, use either the amqstrg0.c trigger monitor or build your own using only public WebSphere MQ APIs.

Monitoring Client Applications

If you are monitoring client applications and notice that the WebSphere MQ listener does not generate TransactionVision events when invoked from **inetd**, check the user id and group id of the mqm user as follows:

id mqm

uid=nnn(mqm) gid=nnn(mqm)

Both the user and the group should be **mqm**. If not, the user **mqm** should be added to the group **mqm**. Alternately, you may change the group id on the listener binary to belong to the group that the user **mqm** belongs to.

Optimizing Performance

To achieve optimum performance when using TransactionVision, follow these guidelines:

- Install the Analyzer on a computer that is fast enough to keep up with the rate of incoming messages from agents. For a slower Analyzer host, increase the maximum queue depth on the event queue used for sending event messages from agents to the Analyzer.
- Use data collection filters to restrict event collection to event data absolutely necessary to resolve a particular problem. Be sure to delete the default **Collect All** data collection filter if you define a more restrictive filter, since events that meet any filter are collected by agents.
- Use the **dbstats** script that can be generated by the **CreateSqlScript** utility to optimize database performance. For details, see "CreateSqlScript" on page 384.

Missing Events

If it appears that events are missing from a schema, first make sure that no data collection filter or query is preventing the events from being collected or displayed. In the data collection filter, if the event retry timeout is set to a value other than Retry Forever and the timeout is being met, events are lost.

If the events are still missing and they are from WebSphere MQ, make sure that the WebSphere MQ API call uses a valid hConn parameter. The agent uses the **hConn** value to determine which queue manager or event broker to connect to for configuration messages; if this value is invalid, the agent cannot check for configuration messages.

Queue Manager or Event Broker Channel Limitations

MaxChannels and **MaxActiveChannels** queue manager or event broker properties have to be set to a reasonable amount for TransactionVision. By default, these properties are not set and there is no limitation. For each active communication link, the TransactionVision Analyzer requires two channels plus one channel for each listening thread defined in the communication link.

If these properties are set too low, the user sees strange behavior such as a partial connection to the communication link. It may at first appear that the Analyzer is connected, but it reports a connection error and puts the communication link in a "Trying to Connect" state. This behavior usually points to a problem with these queue manager or event broker properties not being set properly.

Servlet and EJB Technology Problems

If you are not able to collect servlet or EJB events from your web application, check the following:

1. You have run **SensorSetup.sh**(bat) to install the TransactionVision Java Agent on the application server on which your web application is running. The Application server name is case sensitive, "Default Server" is different from "default server" or "DefaultServer". Run SensorSetup.sh (Unix) or SensorSetup.bat (Windows) again if needed.
2. Check **config/sensor/SensorConfiguration.xml** from your TransactionVision home to make sure the information you provided there is correct and the queue manager or event broker is running. Run SensorSetup.sh (Unix) or SensorSetup.bat (Windows) to modify the setting if needed.
3. Check **sensor.log** in your log file directory to see if there are any errors.
4. Check WebSphere's **system.out** and **system.err** log files to see if there are any errors.

JMS Technology Problems

If you are not able to collect JMS events from stand-alone JMS application, check the following:

1. Make sure that file **java\lib\com.ibm.mqjms.jar** is in your TransactionVision home directory. Run SensorSetup.sh (Unix) or SensorSetup.bat (Windows) again if it is not there.
2. Your **ClassPath** points to **java\lib\com.ibm.mqjms.jar** in TransactionVision home AHEAD of the same file from WebSphere MQ directory.
3. Check **config/sensor/SensorConfiguration.xml** from your TransactionVision home to make sure the information you provided there is correct and the queue manager or event broker is running. Run SensorSetup.sh (Unix) or SensorSetup.bat (Windows) to modify the setting if needed.
4. If JMS functions are called from servlets, make sure you answer **YES** to the question “Do you wish to monitor JMS methods along with Servlet methods?” when running SensorSetup.sh (Unix) or SensorSetup.bat (Windows).

Monitored Applications Troubleshooting

TransactionVision does not require changes to monitored applications. However, applications on the AIX platform may fail after installing a WebSphere MQ support pac. In WebSphere MQ support pacs, internal symbols exported from the TransactionVision WebSphere MQ Agent on the AIX platform may change. When an internal symbol that has been exported from the agent library is no longer available in the WebSphere MQ library, the application cannot start and fails with various symbol resolution errors.

To work around this problem, run the rebind_sensor script whenever a WebSphere MQ support pac that modifies the WebSphere MQ libraries (**libmqm.a**, **libmqic.a**, **libmqm_r.a**, **libmqic_r.a**) are modified. For more information about this script, see "Administration Utilities" on page 367.

Event Queue Cleanup

There may be times when messages that are not being collected by a TransactionVision Analyzer accumulate on the event queue. These events may be targeted for invalid schemas or communication links, or they may be erroneous messages. In either case, you can use the ManageQueue utility to remove or replay these messages.

To view a snapshot of the messages on an event queue (TVISION.EVENT.QUEUE on queue manager QMGR1 in this example), enter the following command:

```
ManageQueue.sh –qm qmgr1 –input_queue TVISION.EVENT.QUEUE
–show_event_ids
```

For complete information about **ManageQueue**, see "ManageQueue" on page 400.

Invalid Schema and Communication Link IDs

Events are targeted for a specific database, schema, and communication link identified within TransactionVision by a unique ID. If a schema is deleted or a communication link is changed, the unique ID no longer exists, and any events still on the event queue that targeted for that schema or communication link are no longer valid.

To display all schema and commlink IDs, enter the following command:

```
EventQueueCleanup -showids
```

To delete events targeted for database schemas or communication links which no longer exist from the queue TVISION.EVENT.QUEUE on the queue manager QMGR1, enter the following command:

```
ManageQueue.sh –show_db_ids
```

You can also use ManageQueue to replay events targeted for invalid schema or communication link IDs to a new schema and communication link so they can be processed by the Analyzer:

```
ManageQueue.sh –qm tv.qmgr –input_queue TVISION.EVENT.QUEUE –output_queue
TVISION.EVENT.QUEUE –move 2 11 -verbose
```

In this example, the schema ID for the new schema is 2 and the communication link ID for the new communication link is 11.

This section includes:

- "Erroneous Messages" on page 488
- "Events Not Pulled from Event Queue" on page 488
- "WebSphere MQ Anomalies" on page 489

Erroneous Messages

Erroneous messages are messages on the event queue that do not appear to be TransactionVision event messages. Erroneous messages could include messages generated from other applications or WebSphere MQ COA report messages used by communication link time skew tests.

To remove erroneous messages from TVISION.EVENT.QUEUE on queue manager TransactionVision.qmgr, enter the following command:

```
ManageQueue.sh –qm tv.qmgr –input_queue TVISION.EVENT.QUEUE –delete –
erroneous –orphaned –verbose
```

Events Not Pulled from Event Queue

Event queues can be shared across Analyzers and communication links. Therefore, events have to be retrieved from the queue based on a specific message ID (correlation ID in the case of Java Agent events).

The following diagram shows the layout of the message ID (correlation ID in the case of Java Agent events) on TransactionVision events.

12 Bytes Database Name	4 Bytes Database Host IP	4 Bytes CommLink ID	4 Bytes Schema ID

Given an assigned communication link, an Analyzer gets events only by a specific message ID/correlation ID. Therefore, it is possible to have events on the event queue which may never be removed from the queue. Different sceneries can cause this.

WebSphere MQ Anomalies

TransactionVision events are not the only kinds of messages that may appear on a event queue.

For client connections to a host other than the one running the agent, TransactionVision uses WebSphere MQ Confirmation of Arrival (COA) messages to determine time skew across communication links. This means that a COA request message is sent to the configuration queue, and the COA report is sent back on the event queue.

If a remote (using multiple queue managers or event brokers) communication link is being used, the COA report has to travel across a WebSphere MQ channel to reach the final event queue destination. If there are problems in the WebSphere MQ environment (for example, the channel is down), the Analyzer may timeout waiting for a COA report message. If the WebSphere MQ problem is later fixed, the COA report message may finally arrive in the event queue. This COA report message is never removed from the event queue, because the Analyzer has given up on it.

There is also the possibility of other WebSphere MQ applications maliciously or accidently putting messages on the TransactionVision event queue. The Analyzer ignores these because they are most likely not to have a matching message ID/correlation ID it is looking for, and therefore remain on the event queue indefinitely.

Problems Importing System Tables into a Database

This section includes the following topics:

➤ Oracle on page 490

➤ DB2 on page 492

Oracle

The referential constraints of TransactionVision system tables prevent Oracle data import utility from loading data into existing tables. The Oracle data import utility returns an error similar to the following example:

```
IMP-00019: row rejected due to ORACLE error 2291
IMP-00003: ORACLE error 2291 encountered
ORA-02291: integrity constraint (TVISION.FK1_FILTER) violated - parent key not found
Column 1 17
Column 2 Collect All
Column 3 16
Column 4 00540001020C00000001000000010000000179F0700008ADC00... 0 rows
imported
```

If you encounter this error, disable the constraints, import the data, and then re-enable the constraints.

For more information, see the following Oracle document:

http://download-west.oracle.com/docs/cd/A91202_01/901_doc/server.901/a90192/ch02.htm#1005094

Disabling Referential Constraints

To disable the referential constraints, use the following SQL commands:

```
alter table TVISION.ANALYZER_PROJ_COMMLINK DISABLE CONSTRAINT
FK2_ANLZ_PROJ_LINK;
alter table TVISION.ANALYZER_SCHEMA DISABLE CONSTRAINT
FK2_ANLZ_SCHEMA;
alter table TVISION.COMMLINK_FILTER DISABLE CONSTRAINT FK1_LINK_FILTER;
alter table TVISION.COMMLINK_FILTER DISABLE CONSTRAINT FK2_LINK_FILTER;
alter table TVISION.FILTER DISABLE CONSTRAINT FK1_FILTER;
alter table TVISION.PROJECT DISABLE CONSTRAINT FK1_PROJECT;
alter table TVISION.PROJECT_COMMLINK DISABLE CONSTRAINT
FK1_PROJ_LINK;
alter table TVISION.QUERY DISABLE CONSTRAINT FK1_QUERY;
```

Re-enabling Referential Constraints

To re-enable the referential constraints after importing the data, use the following SQL commands:

```
alter table TVISION.ANALYZER_PROJ_COMMLINK ENABLE CONSTRAINT
FK2_ANLZ_PROJ_LINK;
alter table TVISION.ANALYZER_SCHEMA ENABLE CONSTRAINT
FK2_ANLZ_SCHEMA;
alter table TVISION.COMMLINK_FILTER ENABLE CONSTRAINT FK1_LINK_FILTER;
alter table TVISION.COMMLINK_FILTER ENABLE CONSTRAINT FK2_LINK_FILTER;
alter table TVISION.FILTER ENABLE CONSTRAINT FK1_FILTER;
alter table TVISION.PROJECT ENABLE CONSTRAINT FK1_PROJECT;
alter table TVISION.PROJECT_COMMLINK ENABLE CONSTRAINT
FK1_PROJ_LINK;
alter table TVISION.QUERY ENABLE CONSTRAINT FK1_QUERY;
```

DB2

When importing system tables into a DB2 database, some tables may be placed in a Check Pending state. To resolve this problem, disable the constraints, import the data, and then re-enable the constraints.

Disabling Referential Constraints

To disable the referential constraints, use the following SQL commands:

```
alter table tvision.analyzer_proj_commlink alter FOREIGN KEY fk2_anlz_proj_link NOT
ENFORCED;
alter table TVISION.ANALYZER_SCHEMA alter FOREIGN KEY FK2_ANLZ_SCHEMA
NOT ENFORCED;
alter table TVISION.ANALYZER_SCHEMA alter FOREIGN KEY FK1_ANLZ_SCHEMA
NOT ENFORCED;
alter table TVISION.COMMLINK_FILTER alter FOREIGN KEY FK1_LINK_FILTER
NOT ENFORCED;
alter table TVISION.COMMLINK_FILTER alter FOREIGN KEY FK2_LINK_FILTER
NOT ENFORCED;
alter table TVISION.FILTER alter FOREIGN KEY FK1_FILTER NOT ENFORCED;
alter table TVISION.PROJECT alter FOREIGN KEY FK1_PROJECT NOT
ENFORCED;
alter table TVISION.PROJECT_COMMLINK alter FOREIGN KEY FK1_PROJ_LINK
NOT ENFORCED;
alter table TVISION.QUERY alter FOREIGN KEY FK1_QUERY NOT ENFORCED;
alter table TVISION.STORAGE alter FOREIGN KEY FK1_STORAGE NOT
ENFORCED;
alter table TVISION.REPORT_PARAMETERS alter FOREIGN KEY FK1_REP_PARMS
NOT ENFORCED;
alter table TVISION.JOB_STATUS alter FOREIGN KEY FK1_JOB_ID NOT
ENFORCED;
alter table TVISION.COMMLINK_FILTER alter FOREIGN KEY FK1_LINK_FILTER
NOT ENFORCED;
alter table TVISION.COMMLINK_FILTER alter FOREIGN KEY FK2_LINK_FILTER
NOT ENFORCED;
alter table TVISION.COMMLINK_FILTER alter FOREIGN KEY FK3_LINK_FILTER
NOT ENFORCED;
alter table TVISION.ANALYZER_PROJ_COMMLINK alter FOREIGN KEY
FK1_ANLZ_PROJ_LINK NOT ENFORCED;
alter table TVISION.ANALYZER_PROJ_COMMLINK alter FOREIGN KEY
FK2_ANLZ_PROJ_LINK NOT ENFORCED;
```

Re-enabling Referential Constraints

To re-enable the referential constraints after importing the data, use the following SQL commands:

```
alter table TVISION.ANALYZER_PROJ_COMMLINK alter FOREIGN KEY
fk2_anlz_proj_link ENFORCED;
alter table TVISION.ANALYZER_SCHEMA alter FOREIGN KEY FK1_ANLZ_SCHEMA
ENFORCED;
alter table TVISION.ANALYZER_SCHEMA alter FOREIGN KEY FK2_ANLZ_SCHEMA
ENFORCED;
alter table TVISION.COMMLINK_FILTER alter FOREIGN KEY FK1_LINK_FILTER
ENFORCED;
alter table TVISION.COMMLINK_FILTER alter FOREIGN KEY FK2_LINK_FILTER
ENFORCED;
alter table TVISION.FILTER alter FOREIGN KEY FK1_FILTER ENFORCED;
alter table TVISION.PROJECT alter FOREIGN KEY FK1_PROJECT ENFORCED;
alter table TVISION.PROJECT_COMMLINK alter FOREIGN KEY FK1_PROJ_LINK
ENFORCED;
alter table TVISION.QUERY alter FOREIGN KEY FK1_QUERY ENFORCED;
alter table TVISION.STORAGE alter FOREIGN KEY FK1_STORAGE ENFORCED;
alter table TVISION.REPORT_PARAMETERS alter FOREIGN KEY FK1_REP_PARMS
ENFORCED;
alter table TVISION.JOB_STATUS alter FOREIGN KEY FK1_JOB_ID ENFORCED;
alter table TVISION.COMMLINK_FILTER alter FOREIGN KEY FK1_LINK_FILTER
ENFORCED;
alter table TVISION.COMMLINK_FILTER alter FOREIGN KEY FK2_LINK_FILTER
ENFORCED;
alter table TVISION.COMMLINK_FILTER alter FOREIGN KEY FK3_LINK_FILTER
ENFORCED;
alter table TVISION.ANALYZER_PROJ_COMMLINK alter FOREIGN KEY
FK1_ANLZ_PROJ_LINK ENFORCED;
alter table TVISION.ANALYZER_PROJ_COMMLINK alter FOREIGN KEY
FK2_ANLZ_PROJ_LINK ENFORCED;
```

Connection to Database Is Lost

If the connection to the TransactionVision database is lost, the following message is displayed:

"**TransactionVision Error: The connection to the TransactionVision database has been lost. Please ensure that the database is accessible, and then re-login to BSM again.**"

In some cases, making sure the database is accessible and re-logging into HP Business Service Management does not resolve the problem. In these cases, the problem can be resolved by first closing the browser window and then logging into BSM again.

Problems Using International Characters

For details about how to configure, store and display unicode data, see Database Advanced Parameters on "Database Advanced Parameters Page" on page 59.

Garbage Characters in Database Column Values

The most likely cause of this problem is inserting Unicode characters into a database that has not been created as a UTF-8 database. If your user data contains any Unicode that gets mapped to database tables, you must create the database with the UTF-8 code set.

Value Too Large For Column Exceptions

Database exceptions during event processing reporting that the inserted value is too large for the column indicates that, for custom XDM columns, a corresponding table column might not have been generated large enough to accommodate the number of characters specified in the XDM definition. To resolve this problem, set the attribute `unicode=true` on the column.

Unable to Make a Client Connection to a Queue Manager or Event Broker Using an International Character Set

The problem indicates that the queue manager or event broker is using a coded character set identifier (CCSID) of mixed mode character set (for example, Shift-JIS or another double-byte character set) and the CCSID has not been set correctly.

Agent Configuration Queue Manager or Event Broker

The CCSID of the queue manager or event broker should be set during SensorSetup. If you are, you may see the following errors in the **sensor.log** file:

```
2006-03-10 10:17:23,218 [Thread-1] FATAL sensorLog - TransactionVision
Error(FailedToCreateTransportSessionForListener): Failed to create a transport session
for Sensor configuration listener.

2006-03-10 10:17:23,218 [Thread-1] FATAL sensorLog - TransactionVision
Error(TransportConnectFailed): Failed to connect ('jpsvr.tv1.manager' 'queue:///
TVISION.CONFIGURATION.QUEUE?priority=-1&persistence=-1&targetClient=1').
(2195) Unexpected error occurred.

2006-03-10 10:17:23,218 [Thread-1] FATAL sensorLog - MQJMS2005: failed to create
MQQueueManager for 'bristol-jpsvr:jpsvr.tv1.manager'
2006-03-10 10:17:23,218 [Thread-1] FATAL sensorLog -
javax.jms.JMSException: MQJMS2005: failed to create MQQueueManager for
'bristol-jpsvr:jpsvr.tv1.manager'
  at com.ibm.mq.jms.services.ConfigEnvironment.
    newException(ConfigEnvironment.java:546)
  at com.ibm.mq.jms.MQConnection.createQM(MQConnection.
    java:1450)
  at com.ibm.mq.jms.MQConnection.createQMNonXA(MQConnection.
    java:960)
  at com.ibm.mq.jms.MQQueueConnection.<init>(MQQueueConnection.
    java:159)
  at com.ibm.mq.jms.MQQueueConnection.<init>(MQQueueConnection.
    java:77)
  at com.ibm.mq.jms.MQQueueConnectionFactory.createQueue
    Connection(MQQueueConnectionFactory.java:142)
  at com.bristol.tvision.transport.MQJMSTransportSession.
    connect(MQJMSTransportSession.java:156)
  at com.bristol.tvision.sensor.configurationmgr.Configuration
   ListenerThread.run(ConfigurationListenerThread.java:196)
```

To resolve this problem:

1 Run the following command to identify the CCSID of the target queue manager or event broker:

```
echo display qmgr ccsid | runmqsc <sensor_config_qmgr>
```

2 Re-run **`SensorSetup`** and enter the correct CCSID.

Analyzer Configuration Queue Manager or Event Broker

The CCSID of the queue manager or event broker should be set in the communication link. The following errors appear in the **AppLog** file if a client connection to the Analyzer configuration queue manager or event broker cannot be made:

```
2006-03-10 10:44:44,718 [ConfigurationThread (id=10,action=timeskew)] FATAL
AppLog - TransactionVision Fatal(MQConnectionFailed): Failed to connect
('jpsvr.tv1.manager'): (2195) Unexpected error occurred.
2006-03-10 10:44:44,718 [ConfigurationThread (id=10,action=timeskew)] FATAL AppLog -
com.ibm.mq.MQException: MQJE001: Completion Code 2, Reason 2195
 at com.ibm.mq.MQManagedConnectionJ11.<init>
  (MQManagedConnectionJ11.java:172)
 at com.ibm.mq.MQClientManagedConnectionFactoryJ11.
  _createManagedConnection(MQClientManagedConnection
  FactoryJ11.java:270)
 at com.ibm.mq.MQClientManagedConnectionFactoryJ11.
  createManagedConnection(MQClientManagedConnection
  FactoryJ11.java:290)
 at com.ibm.mq.StoredManagedConnection.<init>
  (StoredManagedConnection.java:80)
 at com.ibm.mq.MQSimpleConnectionManager.allocate
  Connection (MQSimpleConnectionManager.java:150)
 at com.ibm.mq.MQQueueManager.obtainBaseMQQueueManager
  (MQQueueManager.java:682)
 at com.ibm.mq.MQQueueManager.construct(MQQueueManager.
  java:620)
 at com.ibm.mq.MQQueueManager.<init>(MQQueueManager.
  java:393)
 at com.bristol.tvision.services.collection.
  MQConfigurationThread.createConnection(MQConfiguration
  Thread.java:356)
 at com.bristol.tvision.services.collection.
  MQConfigurationThread.getTimeSkew(MQConfiguration
  Thread.java:218)
```

To resolve this problem:

1 Run the following command to identify the CCSID of the target queue manager or event broker:

```
echo display qmgr ccsid | runmqsc <analyzer_config_qmgr>
```

2 Edit the communication link to set the correct CCSID.

Schemas Missing New Tables

When a new XDM file containing XML mappings to new tables is added, older schemas not containing these new tables fail to get loaded by TransactionVision. To work around this, add the new tables to the older schemas with the CreateSqlScript as follows:

```
CreateSqlScript[.sh|.bat] –c –e –t table-name schema-name
```

Database Deadlocks in IBM DB2 Databases

Note: Multiple threads performing concurrent database operations along with a DB2 behavior called Next Row Locking may lead to database deadlocks. To avoid deadlocks when using multiple event-processing threads, set the DB2 variable DB2_RR_TO_RS to ON with the following command:

```
db2set DB2_RR_TO_RS=ON
```

This setting is only effective after a DB2 restart. It affects all DB2 applications and cannot be used if other non-TransactionVision applications that require “Repeatable Read” semantics (Transaction Isolation Level RR) are using the same DB2 instance. If other applications using the same instance require “Repeatable Read” semantics, you must either create a separate DB2 instance for TransactionVision or set the number of event collection threads to 1.

Unable to Communicate with Processing Server after Enabling SSL

Symptom

After enabling SSL on a Processing Server by checking **Enable SSL** in the **Configuration** > **Advanced** area, the user interface is no longer able to communication with the Processing Server's components (**Status** panes display red error icons). In addition, **<TVISION_HOME>/logs/processmanager.log** on the Processing Server installation contains the following error.

ERROR AppLog - TransactionVision Error(SSLConnectorFailedToRun): The SSL connector failed to run. Make sure certificates and keys have been installed before enabling SSL. Until the issue is resolved, you need to disable SSL in the TransactionManagement UI. See the processing server logs for related errors.

Cause

The SSL certificate consisting of public and private key pair was not imported into the Processing Server's keystore.

Resolution

1. Disable SSL on the Processing Server by un-checking **Enable SSL**. To access, select **Admin** > **Transaction Management** > click the **Configuration** tab (left pane) > select **Processing Servers** > <processing_server> > **Configuration** tab (right panel) > **Advanced** area.
2. Verify communication to the Processing Server's components has been re-established by checking the **Status** panes.

3 Obtain a certificate for the Processing Server using a CA (Certificate Authority) or by using the Java keytool command and import it into the Processing Server's keystore which defaults to **<TVISION_HOME>/jre/lib/security/cacerts**.

The certificate password must be the same as the keystore password. Also, the certificate's public key must be imported into the BSM Gateway's truststore. For detailed instructions, see the "Using SSL with TransactionVision" chapter in the *HP Business Service Management Hardening Guide* PDF.

4 Enable SSL on the Processing Server by checking **Enable SSL**, specifying the Keystore Password, and optionally specifying the KeyStore Location if not using the default keystore.

Part II

Business Transaction Administration

14

Introducing Transaction Management

This chapter includes:

Concepts

➤ Transaction Management Overview on page 506

➤ Business Transactions Overview on page 507

Tasks

➤ How To Choose a Transaction Management Workflow on page 511

Concepts

Transaction Management Overview

Transaction Management is an HP Business Service Management (BSM) application that provides the following:

- Transaction monitoring

  Transaction monitoring provides aggregate statistics about all transaction instances in the application environment. It relies on data collected by HP Real User Monitor, HP Diagnostics, or both. When HP Diagnostics is used, it can provide insight into all tiers of the application environment.

  Some of the Transaction Management reports and topologies include transaction information reported by HP Diagnostics.

  The transaction monitoring data is sent to the BSM data repositories where it is used by other BSM applications.

- Transaction tracing

  Transaction tracing provides statistics for each individual transaction instance across all tiers of the application environment. Payload information is also captured. Relies on data collected by the TransactionVision product.

Transaction Management provides two types of reports: aggregate reports and instance reports. When business transaction tracing is not being used, only the aggregate reports have data. When business transaction tracing is being used, both aggregate and instance reports have data.

The aggregate reports include the Transaction Summary report, Transaction Over Time report, and Aggregate Topology report.

When a Transaction Management report has more than one data source, it combines the data from all of the sources together and presents them in a consistent and meaningful way.

Business Transactions Overview

A business transaction is a real-world transaction executed by a business application, such as ordering a book online or withdrawing money at an ATM.

Within BSM, a business transaction is a logical entity created by the user to represent a logical unit of work executed by a business application. A logical unit of work can refer to a real-world business transaction like an online book order, but it can also refer to something else that are executed by a business application, such as searching books, browsing a catalog, or putting a book into an online shopping cart.

In the Transaction Management application, users can configure a business transaction to be monitored by HP Diagnostics, TransactionVision, or both.

- For configuring a business transaction to be monitored by HP Diagnostics, users need to define the mapping between a business transaction and a set of server requests that are automatically discovered by HP Diagnostics.

  Once such mapping is defined, a business transaction is said to be enabled for business transaction monitoring.

- For configuring a business transaction to be traced by TransactionVision, users have to define and assign a transaction tracing rule to the business transaction. A transaction tracing rule defines the criteria for recognizing certain types of end-to-end transaction instances that are automatically discovered by TransactionVision Analyzers. Analyzers can use such criteria to associate such transaction instances with the specific business transaction.

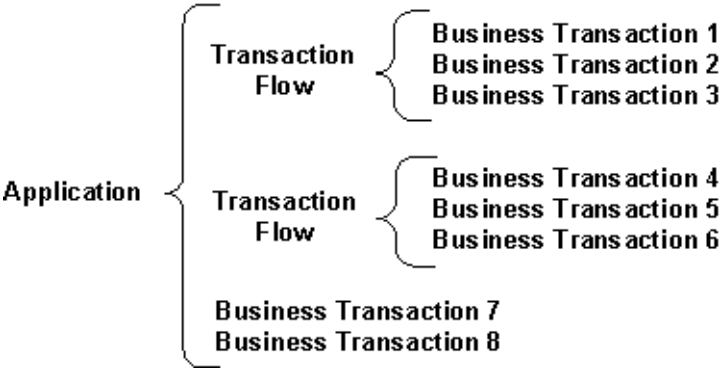
  This process is called transaction classification. After such classification is defined, a business transaction is said to be enabled for business transaction tracing.

  Because the business transactions to be defined and monitored by Transaction Management are closely related to the activities of the business applications in an application environment, a Transaction Management administrator works closely with application specialists to define and configure business transactions.

Business Transactions in Applications or Transaction Flows

A business transaction exists in the context of an application, a group, or a transaction flow. An application is an entity defined in BSM to represent a real-world business application such as an online order application.

Business transactions can also be logically grouped into transaction flows and applications. A business transaction flow can be used to represent a logical grouping of related business transactions, such as a sequence of transaction steps in ordering a book online. Their relationships can be illustrated by the following example:



Note: In End User Management, an application and its transaction flows are significant entities. For best practices in working with transactions that are shared between End User Management and Transaction Management, group all business transactions in a transaction flow that can then be associated with a single business transaction flow.

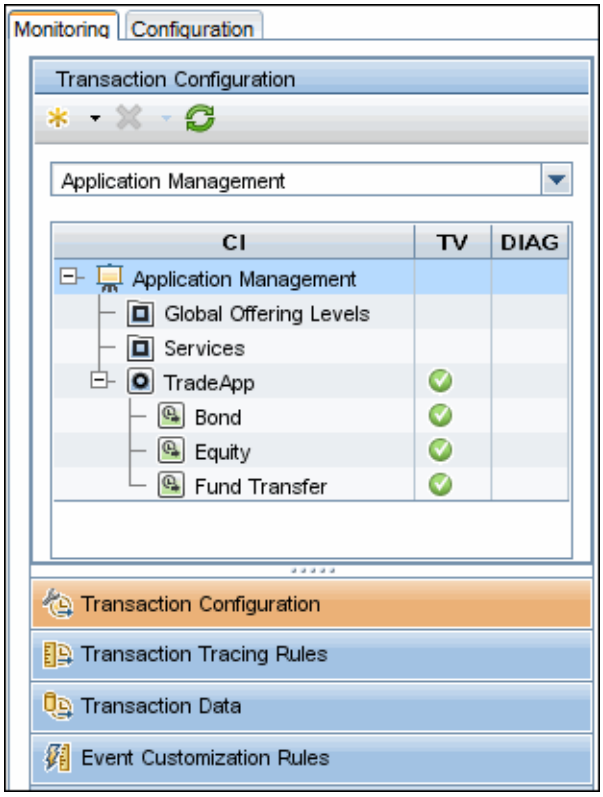
For more information about End User Management, see *Using End User Management*.

Application Management View

In BSM, business transactions, business transaction flows, and applications are defined as entities called configuration items (CIs). Business transaction CIs are one of many CIs that are defined and stored in the BSM RTSM. You use different views to manage particular sets of CIs.

The **Application Management** view is the default view for Transaction Management Administration. The view displays business transaction data on the entire application hierarchy (applications, business transaction flows, CI collections, business transactions). The data is based on the transactions monitored by HP Diagnostics or traced by TransactionVision.

You create and manage these CIs through the Transaction Management **Administration** > **Monitoring** > **Transaction Configuration** tab:



The CIs display in the hierarchy represented by the diagram above.

Business Transaction CIs and other CIs can be configured in Transaction Management Administration. For user interface details, see "Business Transaction Tracing User Interface" on page 544 and "Custom Correlation User Interface" on page 617.

Tasks

How To Choose a Transaction Management Workflow

The application specialist works with the Transaction Management administrator to define the business transactions. This process involves the following tasks:

- Create a Business Transaction CI. See "How to Create a Business Transaction CI" on page 525.
- Determine which product to use to collect the transaction data:
  - Real User Monitor
  - HP Diagnostics
  - TransactionVision

  Transaction Monitoring (HP Diagnostics or Real User Monitor) collects aggregate statistics only. Transaction Tracing (TransactionVision) tracks actual transaction instances which allows for setting the transaction value, state, and other ways to show the business content.

  TransactionVision provides the finer grain information, and is the best way to manage mission-critical transactions. But there is a workload associated with this transaction tracking that may limit the number of transactions on which your system can perform transaction tracing. Additionally, it has a higher overhead on your application environment as it collects more data.

  Transaction monitoring, because it captures less transaction data, has a smaller workload and overhead and can therefore cover more transactions.
- If using TransactionVision, you enable the transaction for tracing and then define the transaction tracing rules. You can use a wizard to define the rules automatically. See Chapter 15, "Business Transaction Tracing."
- If using HP Diagnostics, you enable the transaction for monitoring and then choose the Server Requests that is the entry point of a transaction. See Chapter 16, "Business Transaction Monitoring."

The following tables compares the products that can be used for data collection in Transaction Management.

Attribute	Real User Monitor	HP Diagnostics	TransactionVision
Points to be monitored	Network	Application servers	Application servers, middleware, mainframe
Deployment of agents	No	Yes	Yes
Type of transaction data	Aggregate	Aggregate	Instance
Trace of individual transactions	No	No	Yes
Granularity of monitoring	Course	Fine	Very fine
Payload capture	No	No	Yes
Ease of deployment	Easy	Easy	Requires configuration
Overhead on applications	None	Very low	Low
Scalability	Very high	Very high	High

15

Business Transaction Tracing

This chapter includes:

Concepts

- Transaction Tracing Overview on page 514
- Transaction Tracing Rules on page 515
- Importing/Exporting Transaction Tracing Rules on page 519
- Match Conditions on page 520

Tasks

- How to Set Up a Business Transaction for Tracing on page 521
- How to Create a Business Transaction CI on page 525
- How to Create a Match Condition on page 525
- How to Assign a Rule to the Appropriate Analyzers on page 531
- How to Copy a Rule to Create a New One on page 532
- How to Group/Ungroup Rules on page 532
- How to Stop Tracing on a Business Transaction on page 533
- How to Send Events to BPI on page 534
- How to Create Custom Alerts on page 536

Reference

- Business Transaction Tracing User Interface on page 544

Troubleshooting and Limitations on page 572

Concepts

Transaction Tracing Overview

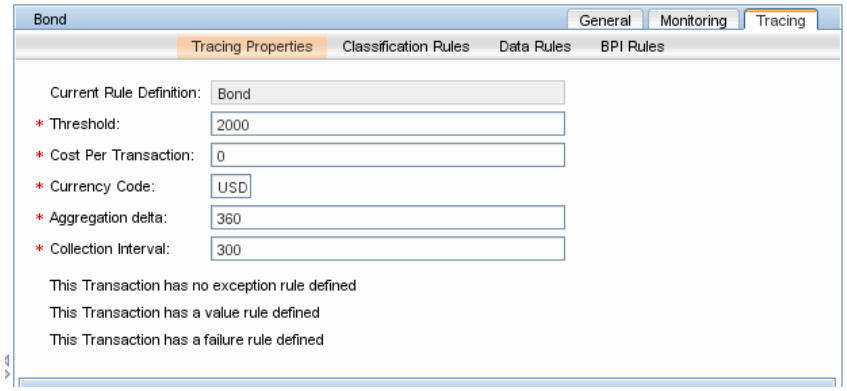
TransactionVision provides a user interface ("Business Transaction Tracing User Interface" on page 544 and "Custom Correlation User Interface" on page 617) to assist managing transaction classes, rules, and their associated attributes. Defining transaction rules maps events to transaction classes, defines how data is mapped between TransactionVision events and its business transactions, and defines any special actions to take when particular criteria are met.

A transaction that is enabled for tracing has two characteristics:

- A transaction tracing rule is assigned to the transaction, see "Transaction Tracing Rules" on page 515.
- Tracing properties are defined for the transaction, described in the following section "Tracing Properties" on page 515.

Tracing Properties

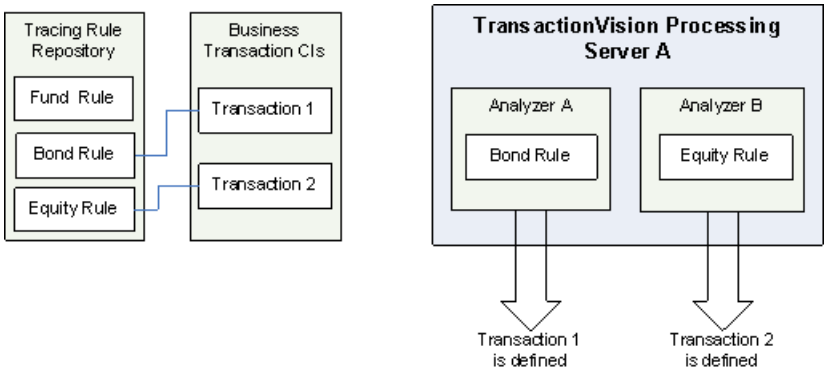
Transaction tracing properties are characteristics of the transaction that you specify so that its collected data is meaningful and accurate. For example, you can see the unit of currency for the Transaction Value and the response time threshold for the transaction, as shown in the following figure.



Transaction Tracing Rules

Transaction tracing rules control the process by which unclassified transactions become business transactions. The rules are executed by the Analyzer as it processes events. If the conditions are met the unclassified transaction becomes a fully-defined and named business transaction.

Transaction tracing rules exist in a repository. You assign one and only one transaction tracing rule from the repository to a transaction. You also assign that tracing rule to the Analyzer collecting events for that transaction—that is, the event data is in that Analyzer's schema. Each Analyzer executes its rules to define the corresponding business transaction when it completes



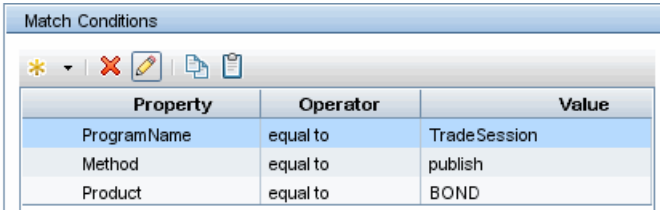
There are four types of transaction tracing rules:

- "Classification Rules" on page 517
- "Data Rules" on page 517
- "BPI Rules" on page 517
- "Alert Rules" on page 518

Classification Rules

Classification rules contain match conditions. If all of the match conditions evaluate to true, the transaction being tested is classified as the business transaction. These rules define the business transactions.

For example, if the event properties match the conditions listed below, the transaction is classified:



Classification rules are typically created automatically by running one of the classification wizards. For user interface details, see "Tracing Tab, Classification Rules Tab" on page 553.

Data Rules

After classification rules define a transaction, key attributes of the transaction are set by executing the data rules. Attributes tell the important and interesting information about the transaction, such as did it complete successfully, and how big of a transaction, in terms of monetary value, it was.

Data rules can be local to a specific transaction tracing rule, or they can be inherited from the global data rules. For user interface details, see "Tracing Tab, Data Rules Tab" on page 554.

BPI Rules

BPI rules define the conditions under which an event is propagated to BPI. For example, if you want to drive an Activity in a Business Process you have modeled in BPI, by an event generated from a TransactionVision business transaction. For user interface details, see "Tracing Tab, BPI Rules Tab" on page 557. For details on using BPI rules, see "How to Send Events to BPI" on page 534.

Alert Rules

The Analyzer sends standard threshold alerting to BSM, but TransactionVision allows you to extend the types of alerts. You can enable the following three types of alerts for TransactionVision business transactions:

- **Response Time Threshold Alert**. Implemented by TransactionVision. When enabled, you set one or more match conditions so that response threshold violation alerts are generated for specific transactions that meet the match condition criteria.
- **Incomplete Transaction Alert**. Implemented by TransactionVision. When enabled and configured, the Analyzer tracks all incomplete transactions and sends the number of incomplete transactions based on whether the transactions failed to complete in the configured time period.
- **Custom Alerts**. You can create custom alerts and provide a matching condition much like a classification rule condition. When the condition is satisfied by the transaction, a custom alert is generated by the Analyzer.

The business transaction alerts appear in the associated TransactionVision health indicators (HIs) in Service Health (to access: **Applications** > **Service Health** > **360o View** > **Hierarchy** > select the **Business Transactions** CI > select the TransactionVision HI). For details, see "360° View" in *Using Service Health*.

Note: BSM alerts cannot be used for TransactionVision transactions.

For more details on alert rules, see:

- "How to Create Custom Alerts" on page 536
- "Tracing Tab, Alert Rules Tab" on page 557

Importing/Exporting Transaction Tracing Rules

You can import and export transaction tracing rules from/to an xml file to create a backup of all the rule assignments and configurations.

- When you export rules, the xml file includes the rule assignments, classification definitions, Analyzer assignments and the related transaction configuration groups and flows. All rules currently on the Transaction Tracing Rules page are exported.

  To export transction tracing rules, from the **Admin** > **Transaction Management** > **Monitoring** tab, select **Transaction Tracing Rules** (bottom-left pane), click the **Export All Transaction Rule Definitions as XML to a file** button, and browse to the location where you want to place the **TransactionDefinition.xml** file (you can change the default xml name).

- When you import exported rules, all rule assignments, classification definitions, and Analyzer assignments remain in tact, and corresponding transaction CIs are created in RTSM. This allows you to resume with your transactions.

Note: Importing rule assignments that refer to a missing transaction class will create the transaction class.

To import transaction tracing rules, from the **Admin** > **Transaction Management** > **Monitoring** tab, select **Transaction Tracing Rules** (bottom-left pane), click the **Import Transaction Rule Definition XML from a file** button, and select the existing **xml** file.

Match Conditions

A match condition is a statement that evaluates to true or false, whether a set of rules should be implemented or not. The request is: when the data in my event matches a particular criteria, then perform the specified actions. A match condition consists of:

- **XPath**: The named location of a piece of data within the event payload that TransactionVision collects from its agents. Each event the agent sends is an XML file, and the XPath indicates to which node in that file you want to refer.

  In the "Create Match Condition Dialog" on page 564, the XPath is called a **Property**. You can toggle between the property name and the complete condition XPath. For example, the complete condition XPath for a property named **ProgramName** is **/Event/StdHeader/ProgramName**. For more detailed information about the XPath and match condition functionality, see the *HP TransactionVision Advanced Customization Guide* PDF.

- **Conditional operator**: A number of conditional operators are available for selection such as equal, greater than, and so on.

- **Value:** What you want to compare against. Values in match criteria may contain one wildcard character, for example, *FlowEngine, DataFlow*, or Data*Engine.

For instructions on creating match conditions, see "How to Create a Match Condition" on page 525. For a description of the Match Condition user interface, see "Create Match Condition Dialog" on page 564.

Tasks

How to Set Up a Business Transaction for Tracing

This task describes how to set up a business transaction for tracing.

This task includes the following steps:

- "Create a new business transaction CI" on page 521
- "Run a Transaction Tracing Rule Wizard if possible" on page 521
- "Configure the tracing rule and properties manually" on page 523
- "Assign the Transaction Tracing Rule to an appropriate Analyzer" on page 524
- "Verify the Business Transaction is Enabled for Tracing" on page 524

1 Create a new business transaction CI

If you are configuring an existing business transaction CI you can skip this step.

For information about creating a new business transaction CI, see "How to Create a Business Transaction CI" on page 525.

2 Run a Transaction Tracing Rule Wizard if possible

Some business transactions can be set up by wizards. The wizard defines the transaction property settings and creates the transaction tracing rule for you, based on your input.

- If the business transaction is based on Servlet technology, you can use the Transaction Tracing Rule wizard for Servlets.
- If the business transaction is based on a business transaction already configured for RUM, you can use the Transaction Tracing Rule wizard for RUM. This wizard will help you create a tracing configuration based on a RUM transaction configuration.

- If the business transaction is based on a business transaction already configured for BPM, you can use the Transaction Tracing Rule wizard for BPM. This wizard helps you create a tracing configuration by finding matching synthetic transaction generated by BPM and collected by TransactionVision. The wizard bases the classification rules on those instances.
- If the business transaction is based on a business transaction already configured for monitoring through HP Diagnostics, you can use the Transaction Monitoring wizard. This wizard helps you create a tracing configuration based on your Diagnostics configuration.

Note: To run the wizards, you must have permission to view User Data. For more information, see "Security" in the *HP TransactionVision Deployment Guide* PDF.

To access the wizards, follow these steps:

a Select **Admin** > **Transaction Management** > **Monitoring.**

b Click the **Transaction Configuration** button in the left pane.

c Right-click the transaction that you want to enable for tracing and choose one of the following:

- **Launch Transaction Tracing Rule Wizard for Servlets**
- **Launch Transaction Tracing Rule Wizard for BPM**
- **Launch Transaction Tracing Rule Wizard for RUM**
- **Launch Transaction Monitoring Wizard**

d Click **Configure Transaction Tracing** in the right pane.

These wizards are fully described within the interface of the wizard itself. The results of running each wizard is that the Transaction Configuration page's Tracing tab is populated. For details about this interface, see "Transaction Configuration Page, Tracing Tab" on page 550.

Note: If the business transaction you need to define is not in the above categories, you need to specify the configuration and transaction tracing rule manually as described in the next step.

---

3 Configure the tracing rule and properties manually

If a wizard cannot be used to create the transaction rule, you must define the rule manually. To do this, follow these steps:

a Select **Admin** > **Transaction Management** > **Monitoring**.

b Click the **Transaction Configuration** button in the bottom left pane.

c Right-click the transaction that you want to enable for tracing and choose **New Transaction Tracing Configuration** > **Create New Rule Definition**.

d Enter a name for the rule and click **OK**. The Transaction Tracing tab appears.

e On the Transaction Tracing tab, enter the transaction tracing properties, classification rules, data rules, BPI rules, and Alert rules as needed. For descriptions of the tabs and fields in the Transaction Tracing tab, see the "Transaction Configuration Page, Tracing Tab" on page 550.

4 Assign the Transaction Tracing Rule to an appropriate Analyzer

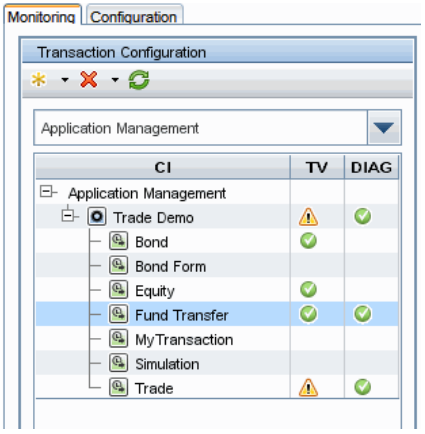
After you have a transaction tracing rule, either defined through a wizard or manually, it is disabled by default shown by the red icon next to the rule name. You must assign it to the appropriate Analyzer — that is, the one that is receiving the events to which the rule applies. When the rule is assigned to an Analyzer or Analyzers, the rule is enabled and the icon turns green. For instructions see "How to Assign a Rule to the Appropriate Analyzers" on page 531.

5 Verify the Business Transaction is Enabled for Tracing

a Select **Admin** > **Transaction Management** > **Monitoring**.

b Click the **Transaction Configuration** button in the left pane.

Each business transaction and its status appear:



A green arrow icon in the **TV** column indicates that the business transaction has been successfully enabled for tracing and is assigned to an Analyzer. A Warning icon indicates that the business transaction has been successfully enabled for tracing but not yet assigned to an Analyzer.

How to Create a Business Transaction CI

This task describes how to create a new business transaction.

1. Select **Admin** > **Transaction Management** > **Monitoring.**
2. Click the **Transaction Configuration** button in the left pane.
3. From the drop-down list, choose the view in which to define the transaction.

   You can use any view you want. The default view is Application Management.
4. Navigate to the parent container that you want to contain the business transaction. The parent container is one of the following: application, business transaction flow, or group. See "Business Transactions Overview" on page 507.
5. Right-click the parent container and choose **New Business Transaction**.
6. The Business Transaction CI wizard appears. For details about this interface, see "Business Transaction CI Creation Dialog Box" on page 545.

   For information about creating other types of CIs, see "Business Transaction Tracing User Interface" on page 544.

How to Create a Match Condition

A match condition evaluates whether a set of rules should be implemented. The match condition can be automatically created by running the Match Condition Servlet Wizard, but most match conditions are created manually.

This task describes how to manually create a match condition for classification rules, BPI rules, alert rules, and custom correlation rules described in Chapter 18, "Custom Correlation."

This task includes the following steps:

➤ "Examine the collected data to determine which fields have the values you want to match" on page 526

- "Locate the rule to which you want to assign match conditions" on page 526
- "Create one or more match conditions" on page 527

1 Examine the collected data to determine which fields have the values you want to match

a Select **Applications** > **Transaction Management** > **Transaction Analysis** > **Event Analysis Report** to view a list of the available TransactionVision events.

b Set the filter for the events you want to see and click the **Run** report icon to view the Event Analysis report events list.

c Select an event to see the event details below the events list.

For details on viewing and using the Event Analysis Report, see "Event Analysis Report" on page 691.

2 Locate the rule to which you want to assign match conditions

a Select **Admin** > **Transaction Management** > **Monitoring**.

b For tracing rules, locate a Matching, Classification, Data, and/or Alert rule from one of the following locations:

- Click the **Transaction Configuration** button in the left pane, select the business transaction that TV is monitoring, click the **Tracing** tab in the right pane, then the appropriate rule: Classification, Data, BPI, or Alert.
- Click the **Transaction Tracing Rules** button in the left pane, select a **Transaction** or **Global** rule, click the **Rule Definitions** tab, and click the appropriate rule tab. Transaction rules can contain Classification, Data, BPI, and/or Alert rules. Global rules can contain Matching and/or Data rules.

c For custom correlation rules, locate a Correlation or Modification rule as follows:

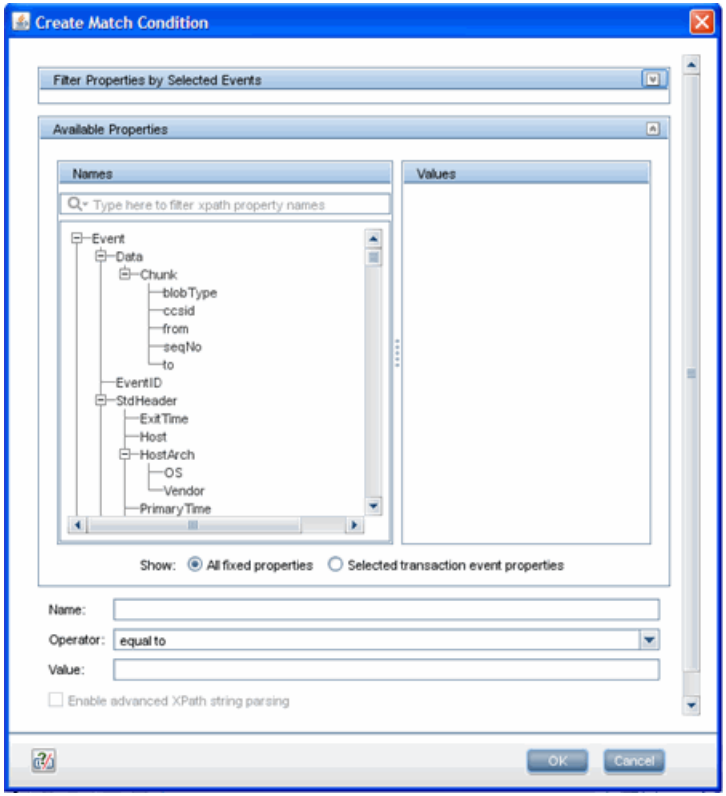
➤ Click the **Event Customization Rules** button in the left pane.

➤ Select a Correlation or Modification rule, then click the **Rule Definitions** tab.

3 Create one or more match conditions

a In the area of the rule's tab where you want to add one or more match conditions, click the New icon and select **New Match Condition**.

The Create Match Condition dialog displays (for user interface details, see "Create Match Condition Dialog" on page 564). By default a set of fixed XPaths is presented, which consists of all the standard locations of TransactionVision event data, as shown in the example below:



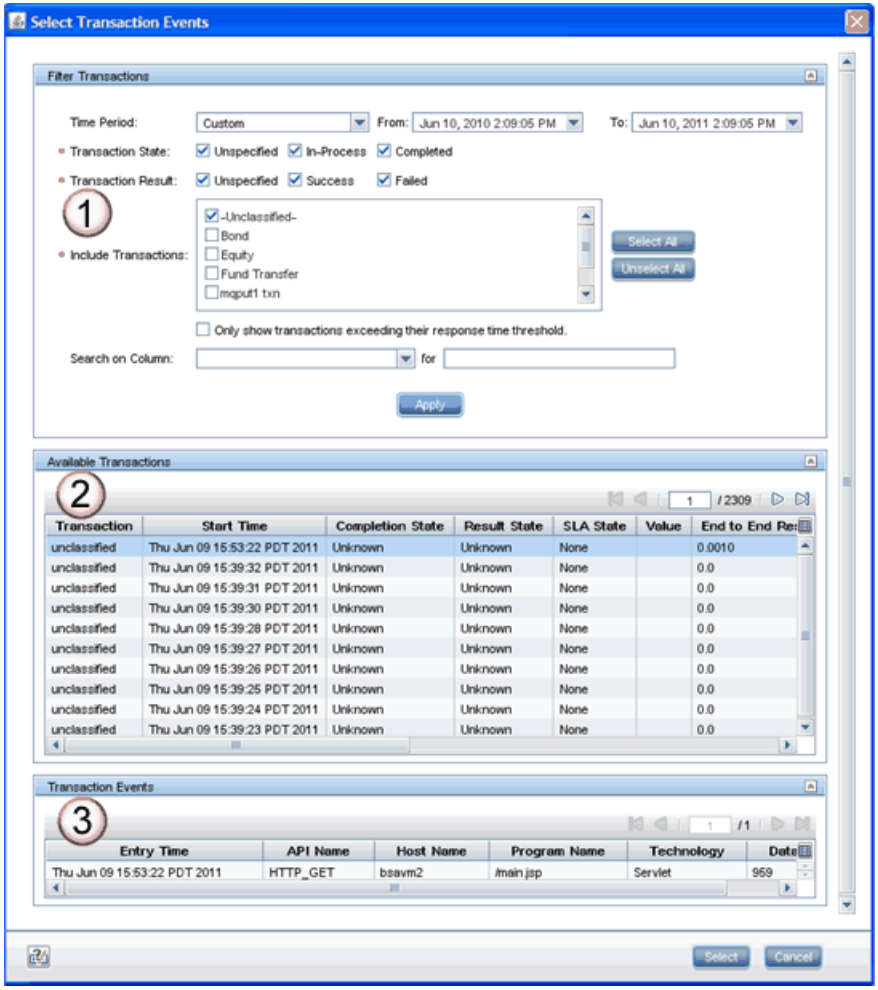
b You can enter the property (XPath) **Name**, **Operator**, and **Value** and click **OK** to create the match condition.

If you do not know the XPath name, you can search for and select the property in the Available Properties window, which populates the Name text box with the full XPath name.

If your event data contains custom fields, such as XML from the payload data or data fields specific to a monitored application, these XPaths do not display in the fixed list of properties. You can search for the transactions that have the event data you want to use to create this match condition described in the next step, Step c.

c Open the top **Filter Properties by Selected Events** collapsible area and click in the window or select the **Load Events** icon to load the transaction events.

The **Select Transaction Events** dialog displays (for user interface details, see "Select Transaction Events Dialog" on page 566). The following table describes the locations in this dialog:



The following table describes the locations in this page:

Callout	Description
1	**Filter Transactions.** Select the appropriate criteria for the type of transaction or transactions with events you want to use to create this match condition and click **Apply**.
2	**Available Transactions.** Displays the available transactions that meet the criteria specified in the Filter Transactions area. Select the appropriate transaction.
3	**Transaction Events.** Displays the event or events associated with the selected transaction. Select the appropriate event.

d Click **Select** to return to the Create Match Condition window.

e Click the **Selected transaction event properties** option to show the XPaths relevant to the selected event or transaction.

Selecting this option also shows the value of each XPath so that you can choose the value directly.

Selected transaction event properties is disabled by default until a transaction is selected.

f Click **OK** to populate the match condition fields on the rule's tab.

Note: When creating classification rules, you can add a new evaluation block (click the New icon and select **New Evaluation Block**), which acts as an OR between a set of match conditions. You can create multple blocks of rules that you OR between. If you do not specify an OR condition, AND is implied by default.

When you edit a match condition, the Edit Match Condition dialog is loaded, which first looks up your specified transaction and parses from all the events within that transaction all XPaths used, including any custom payload data Xpaths, and displays them in the Available Properties tree.

How to Assign a Rule to the Appropriate Analyzers

After creating a rule, you need to assign it to the appropriate Analyzer or Analyzers. That is, assign the rule to the Analyzer or Analyzers receiving the events to which the rule applies.

This task describes how to assign a rule to one or more Analyzers.

1. Select **Admin** > **Transaction Management** > **Monitoring.**
2. Click the **Transaction Tracing Rules** button in the left pane.
3. Select the transaction tracing rule that you want to assign.
4. On the General tab, to assign the rule to a single Analyzer, select the Analyzer in the **Available Analyzer** list and click the single right arrow to move the Analyzer to the **Assigned Analyzers** column in the right pane.

   You can move a single Analyzer back to the **Available Analyzer** list using the single left arrow.
5. To assign the rule to all the available Analyzers, click the double right arrow to move all Analyzers to the **Assigned Analyzers** column.

   You can move all the Analyzers back to the Available Analyzer list using the double left arrow.

   If the Analyzer you want to specify is not in the Available Analyzers list, it is either not active or you do not have permission to access it. See the "Security" chapter in the *HP TransactionVision Deployment Guide* PDF.

   To add Analyzers, see "How to Add an Analyzer to the Deployment Environment" on page 106.
6. Click **Apply**.

How to Copy a Rule to Create a New One

Instead of creating a new rule from scratch, you can copy a rule and then edit it to create a new rule. This task describes how to copy a rule to create a new one.

1. Select **Admin** > **Transaction Management** > **Monitoring**.
2. Click the **Transaction Tracing Rules** button in the left pane.
3. Select a rule.
4. Right-click on the rule and select **Copy**.

   The new rule displays as a copy of the selected rule in the left pane. For example, **Copy of Rule1**.
5. Rename the rule to a unique name on the General tab in the right pane and click **Apply**.
6. Edit the new rule on the Rule Definitions tab and click **Apply** when you are finished.
7. Assign the rule to one or more appropriate Analyzers. For instructions, see "How to Assign a Rule to the Appropriate Analyzers" on page 531.

How to Group/Ungroup Rules

You can group transaction and global rules so that you can see at a glance:

- Each Analyzer and the rules assigned to each Analyzer.
- Unassigned rules.
- Evaluation priority order of the assigned rules within a group. You can increase or decrease the priority of the rule from which the Analyzer retrieves events.

  Transaction rules are evaluated in the order in which the rules appear, from top to bottom. When the analyzer evaluates rules, it stops evaluation and performs a classification on an event on the first set of conditions that completely match the given rules. Keep this in mind when designing your rules.

  At any time, you can ungroup the rules to go back to the original order.

This task describes how to group and ungroup rules and change the evaluation priority order of grouped rules.

To Group Rules Assigned to the Same Analyzers:

1. Select **Admin** > **Transaction Management** > **Monitoring.**
2. Click the **Transaction Tracing Rules** button in the left pane.
3. If the rules were not previously grouped, click the **Group Rules Assigned Analyzer** icon.

   In the left pane, the rules are grouped by Unassigned Tracing and Global rules, and by Analyzer Tracing and Global rules. The assigned rules are grouped in the same order they were in when ungrouped.

To Ungroup Rules:

Click the **Group Rules Assigned Analyzer** icon.

The rules are displayed in their original order.

To Change the Order of Priority of Group Rules:

1. Move a rule in a group up to be evaluated before rules underneath. Select a rule underneath the top rule in the group and click the **Increase the selected rule definition's evaluation priority** icon on the left panel toolbar.
2. Move a rule in a group down to be evaluated after other rules. Select a rule that is above another rule and click the **Decrease the selected rule definition's evaluation priority** icon.

How to Stop Tracing on a Business Transaction

There are a few ways to ways to stop tracing on a business transaction:

Unassign the transaction rule from an Analyzer:

1. Select **Admin** > **Transaction Management** > **Monitoring.**
2. Click the **Transaction Tracing Rules** button in the left pane.
3. Select the tracing rule.

4 In the Assign Analyzers area in the right pane, move the Analyzers from the Assign Analyzers column to the Available Analyzers column.

5 Click **Apply**.

Disable tracing on the transaction:

1 Select **Admin** > **Transaction Management** > **Monitoring.**

2 Click the **Transaction Configuration** button in the bottom-left pane.

3 Select the Business Transaction CI. You may need to first choose the CI view that contains the business transaction CI. By default, business transaction CIs are in the Application Management view.

4 Right-click the Business Transaction CI and choose **Disable Transaction Tracing**.

Delete the Business Transaction CI:

Use the IT Universe Administration pages or RTSM Administration pages to delete the Business Transaction CI (see "CI Lifecycle" in the *RTSM Administration PDF*).

If the business transaction CI that you want to delete is enabled for transaction tracing or monitoring, you may want to first disable these before you delete the CI.

How to Send Events to BPI

As TransactionVision processes events into a business transaction, it can send data to BPI in the form of BPI events. These BPI events can be used to update BPI's view of the progress of a monitored business process instance.

This task describes how to designate which TransactionVision events are delivered to BPI.

1 Determine which transaction has the events you want to send to BPI

Identify the Business Transaction CI that contains the data you want to send to BPI.

The Business Transaction CI must be enabled and configured for tracing.

2 Add BPI rules to the Transaction Tracing rule

Use the BPI Definition Dialog to define the published BPI events. For details about this interface, see "BPI Definition Dialog" on page 571.

3 Handle some data types manually

If you have a Date/Time value computed within TransactionVision, have assigned it to a custom attribute, and want Business Process Insight to be able to read that attribute, the following steps are required:

a When defining the attribute in the XDM column definition in TransactionVision, the field must be declared as a TIMESTAMP column.

b The timestamp column must be populated by doing one of the following:

- Setting the transaction attribute to a long value (using the standard milliseconds since January 1, 1970, 00:00:00 GMT).

  Example of setting a long value:

```
If no dateFormat <Param> is specified, the field must be set using a long value
(milliseconds since Jan 1 1970).

<Column name="order_date" type="TIMESTAMP" description="OrderDate"
conversionType="Date">
    <Path>/Transaction/OrderDate</Path>
</Column>
```

- Parsing a date. In this case you need to specify the date format in the XDM definition

Example of parsing a date:

```
Setting this field in the XDM definition file only allows input in the specified
mm/dd/yyyy hh:mm format.

<Column name="order_date" type="TIMESTAMP" description="OrderDate"
conversionType="Date">
    <Path>/Transaction/OrderDate</Path>
    <Param name="dateFormat" value="mm/dd/yyyy hh:mm"/>
</Column>
```

How to Create Custom Alerts

This task describes how to create custom alerts to be used for business transactions in Service Health. These alerts are only for TransactionVision business transactions. BSM alerts cannot be used. In addition to viewing alerts in Service Health, alerts can be viewed in the Transaction Tracking Report.

Alerts generated by TransactionVision are sent to BSM using a TV Alert Data Sample job. This job runs on a 5-minute interval and sends the count of all alerts that are enabled to BSM for reporting.

This task includes the following steps:

- "Locate the transaction or tracing rule to which you want to create a custom alert" on page 536
- "Create a new custom alert in Transaction Management" on page 537
- "Create a Health Indicator (HI) in Service Health" on page 538
- "Configure the newly created HI with the custom alert name and Synchronize the CI" on page 541
- "Results" on page 543

1 Locate the transaction or tracing rule to which you want to create a custom alert

a Select **Admin** > **Transaction Management** > **Monitoring**.

b To locate a TransactionVision transaction, click the **Transaction Configuration** button in the left pane and select a TransactionVision transaction.

Or

c To locate a tracing rule, click the **Transaction Tracing Rules** button in the left pane and select a Transaction rule.

2 Create a new custom alert in Transaction Management

a Open the Alert Rules page from one of the following areas:

➤ From the TransactionConfiguration page, click the **Tracing** tab in the right pane and click the **Alert Rules** tab.

Or

➤ From the Transaction Tracing Rules page, click the **Rule Definitions** tab in the right pane and click the **Alert Rules** tab.

For details about the Alert Rules tab user interface, see "Tracing Tab, Alert Rules Tab" on page 557.

b In the **Custom Alerts** area at the bottom of the Alert Rules page, click the **New** button and select **Create Custom Alert**.

c Enter the name of the new custom alert and click **OK**.

Note: It is recommended that you note this name, which you need to use in step 4 "Configure the newly created HI with the custom alert name and Synchronize the CI" on page 541.

---

d In the Custom Alerts area, create one or more match conditions for events to be alerted that match the specified criteria. For instructions, see "How to Create a Match Condition" on page 525.

e Assign the tracing rule that contains the custom alert rule to the appropriate Analyzer (or Analyzers) that is receiving the events to which the rule applies. For instructions, see "How to Assign a Rule to the Appropriate Analyzers" on page 531.

At this point, the new custom alert is defined in TransactionVision and will appear in the Transaction Tracking and Transaction Detail reports. Additional configuration is required to see the alerts in Service Health as described in the following steps.

3 Create a Health Indicator (HI) in Service Health

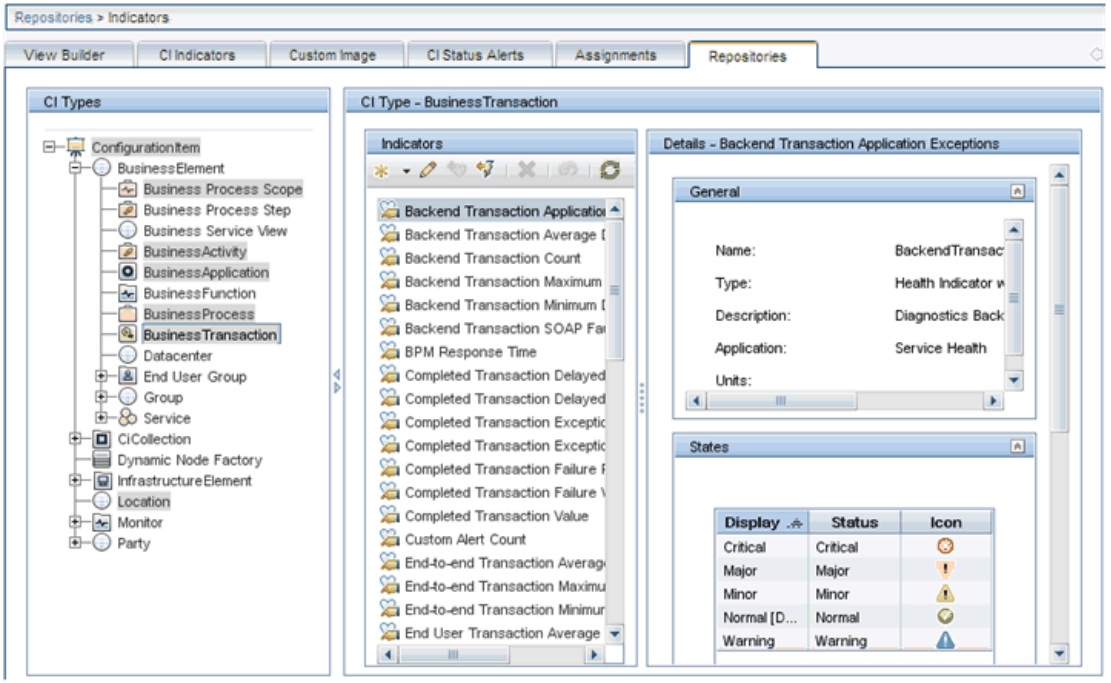
In this step, you create an HI for the newly created custom alert rule to be able to report the custom alert in Service Health.

a Select **Admin** > **Service Health** > **Repositories** > **Indicators**.

The CI Type user interface displays.

b Under CI Types in the left panel, select **Business Element** > **Business Transaction** to list all the HIs for Business Transactions.

Following is an example of the Business Transactions HI list in **Admin** > **Service Health** > **Repositories** > **Indicators**.



c Click the **Health Indicator** button, enter the following required information and click **Save**. For user interface details, see "New Health Indicator/Edit health Indicator Dialog Box" in *Using Service Health*.

➤ **In the General area:**

Display Name: External name of the HI, as displayed in the user interface (for example **Custom Alert Count**).

Name: Internal name of the HI, as defined in the event generating the HI (for example **Custom_Alert_Count**). This is entered automatically when you enter the Display Name.

Description: (optional) Description of the type of alert for the newly created custom alert rule name.

Application: Select **Service Health**.

Units: Type of unit applicable to the rule results displayed in the HI. For alerts, the unit must be **count**.

➤ **In the States area:**

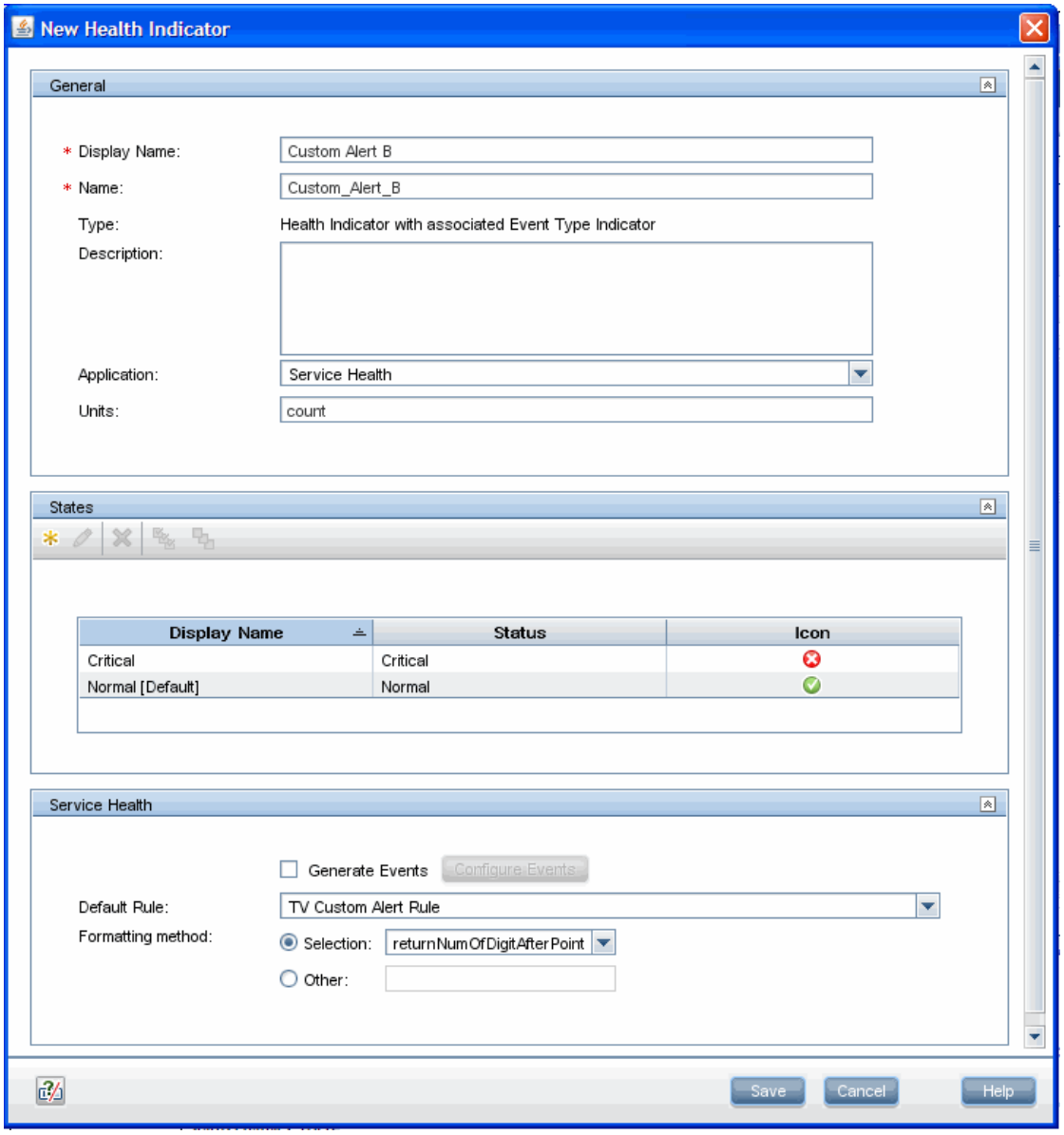
The Normal and Critical states are defined by default. You can Click the **New Indicator State** button to add other states.

➤ **In the Service Health area:**

Default Rule: Select **TV Custom Alert Rule**.

Formatting method: Leave default of **returnNumOfDigitAfterPoint**.

Following is an example of a populated New Health Indicator dialog:



4 Configure the newly created HI with the custom alert name and Synchronize the CI

In this step, you configure the new HI to link with the new custom alert created in step 1, and then synchronize the CI so that the HI will report the custom alerts.

a Select **Admin** > **Service Health** > **Assignments** > **Health Indicator Assignments**.

b Under CI Types in the left panel, select **Business Element** > **Business Transaction** to list all assignments for the Business Transaction CI type.

c Select **TransactionVision HI assignments** and click the **Edit** button.

d Open the **Health Indicator Configurations** area to create a new assignment which will contain all custom alert assignments for TransactionVision. This step needs to be done only once. This assignment can be used to add any number of custom alert HI configurations.

To create a new assignment, click the **Add** button to open the **Add health Indicator To Assignment** dialog. Enter the following information:

➤ **In the Health Indicator area**:

Health Indicator: Select the newly created HI in step 3 "Create a Health Indicator (HI) in Service Health" on page 538 (for example, **Custom Alert Count**).

Business Rule: Select **TV Custom Alert Rule**.

Priority: Keep the default.

➤ **In the Business Rule Parameters area:**

Custom_Alert_Name: Enter the name of the new custom alert rule you created in step 2 "Create a new custom alert in Transaction Management" on page 537. You must enter the exact same name.

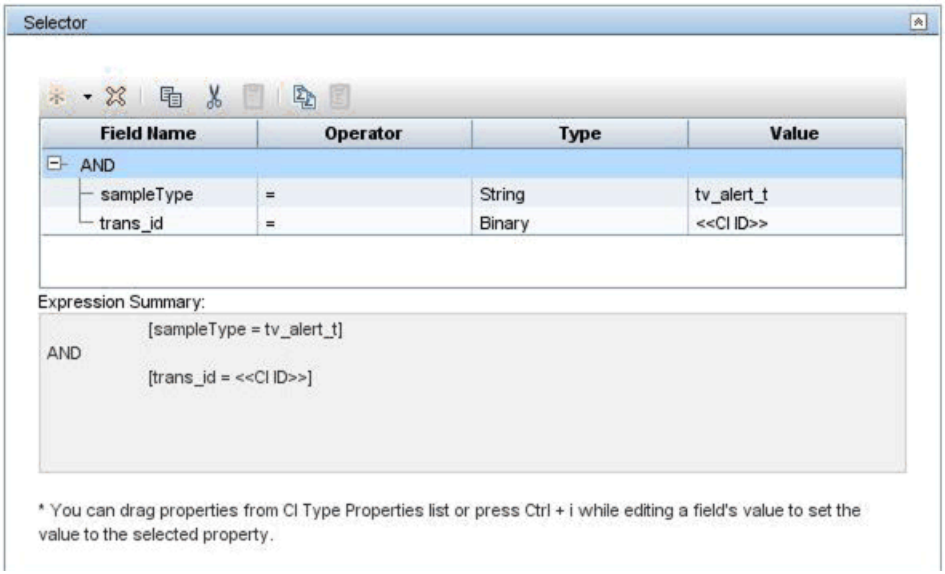
➤ **In the Thresholds area**:

Use the default or enter a setting depending on the severity of the alert.

➤ **In the Selector area:**

Define the selector expression to filter data samples to target only those samples that are relevant for the HI selector that defines which data samples are relevant for the HI.

For example, in the following defined selector, the first entry indicates it is a TransactionVision alert HI (**tv_alert_t**) as compared to a TransactionVision response time HI, and the second entry indicates the CI ID (configuration item), which identifies a unique transaction type in BSM (Universal CMDB). The CI ID is binary data.



e Click **Save** to close the **Add Health Indicator to Assignment** dialog and click Save again to close the **Edit Health Indicator Assignment for CI Type: Business Transaction** dialog.

f Select the **Synchronize CI Type** button to link the new HI with the new custom alert.

5 Results

When a business transaction satisfies all of the match conditions in an alert, the Alert Health Indicator for that business transaction in the 360° view shows the number of alerts in the last 5 minute time interval.

To access the 360° page, select **Application** > **Service Health** > **360 page** > in the CI Type left panel, select **Business Element** > **Business Transaction** and click on the new HI to see the number of alerts.

Reference

Business Transaction Tracing User Interface

This section describes:

- Business Transaction CI Creation Dialog Box on page 545
- Application CI Creation Dialog Box on page 546
- Business Transaction Flow Creation Dialog Box on page 547
- Group CI Creation Dialog Box on page 548
- Transaction Configuration Page on page 549
- Transaction Configuration Page, General Tab on page 549
- Transaction Configuration Page, Tracing Tab on page 550
- Transaction Tracing Rules Page on page 560
- Transaction Tracing Rules Page, General Tab on page 562
- Transaction Tracing Rules Page, Rule Definitions Tab on page 563
- Create Match Condition Dialog on page 564
- Data Rule Configuration Settings Dialog Box on page 568
- BPI Definition Dialog on page 571

Business Transaction CI Creation Dialog Box

This dialog box enables you to create a new business transaction CI.

To access	**1** Select **Admin** > **Transaction Management.** **2** (left pane) Click the **Monitoring** tab. **3** Click the **Transaction Configuration** button. **4** Choose the view in which to create the business transaction CI. **5** Navigate to the parent container that you want to contain the business transaction. If the parent container is not there, create it. See "Business Transactions Overview" on page 507. **6** Right-click the parent container and choose **New Business Transaction**.
Important information	Business Transaction CIs created by this dialog are stored in the RTSM and available to other applications in BSM.

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
Name	The name of the business transaction. The name must be unique to its parent container.
Type	Business Transaction
Description	A description of the business transaction.

Application CI Creation Dialog Box

This dialog box enables you to create a new application CI.

To access	**1** Select **Admin** > **Transaction Management.** **2** (left pane) Click the **Monitoring** tab. **3** Click the **Transaction Configuration** button. **4** Choose the view in which to create the application CI. **5** Click the **New** button.
Important information	Application CIs created by this dialog are stored in the RTSM and available to other applications in BSM.

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
Name	The name of the application. The name must be unique throughout the RTSM.
Type	Application
Description	A description of the application.

Business Transaction Flow Creation Dialog Box

This dialog box enables you to create a new business transaction flow CI.

To access	**1** Select **Admin** > **Transaction Management.** **2** (left pane) Click the **Monitoring** tab. **3** Click the **Transaction Configuration** button. **4** Choose the view in which to create the business transaction flow CI. **5** Navigate to the parent container that you want to contain the business transaction flow. If the parent container is not there, create it first. See "Business Transactions Overview" on page 507. **6** Right-click the parent container and choose **New Business Transaction Flow.**
Important information	Business Transaction Flow CIs created by this dialog are stored in the RTSM and available to other applications in BSM.

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
Name	The name of the business transaction flow. The name must be unique to its parent container.
Type	Business Transaction Flow
Description	A description of the business transaction flow.

Group CI Creation Dialog Box

This dialog box enables you to create a new group CI.

To access	**1** Select **Admin** > **Transaction Management**. **2** Click the **Monitoring** tab in the left pane. **3** Click the **Transaction Configuration** button. **4** Choose the view in which to create the group CI. **5** Navigate to the parent container that you want to contain the group. If the parent container is not there, create it first. See "Business Transactions Overview" on page 507. **6** Right-click the parent container and choose **New Group**.
Important information	Group CIs created by this dialog are stored in the RTSM and available to other applications in BSM.

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
Name	The name of the group. The name must be unique to its parent container.
Type	Group
Description	A description of the group.

Transaction Configuration Page

The Business Transaction page enables you to enable and disable transaction tracing or monitoring, and to view or modify other properties of an existing Business Transaction.

This page includes the following tabs:

- "Transaction Configuration Page, General Tab" on page 549
- "Transaction Configuration Page, Monitoring Tab" on page 578
- "Transaction Configuration Page, Tracing Tab" on page 550

Transaction Configuration Page, General Tab

This tab enables you to view the name and description of an existing business transaction CI. These values were specified when the Business Transaction CI was created. See "Business Transaction CI Creation Dialog Box" on page 545.

Transaction Configuration Page, Tracing Tab

This tab enables you to define transaction tracing properties and rules for a business transaction.

To access	**1** Select **Admin** > **Transaction Management**. **2** Click the **Monitoring** tab in the left pane. **3** Click the **Transaction Configuration** button. **4** Navigate to the business transaction on which you want to view or set transaction tracing properties or rule definitions. **5** Click the **Tracing** tab in the right pane.
Important information	➤ If you ran a wizard to set up the business transaction, this tab is populated based on the values you specified in the wizard. ➤ If you did not run a wizard, the transaction tracing properties and rules do not appear. Instead you are given options to set up these items.
Relevant tasks	"How to Set Up a Business Transaction for Tracing" on page 521

The Tracing tab includes the following tabs:

"Tracing Tab, Tracing Properties Tab" on page 551

"Tracing Tab, Classification Rules Tab" on page 553

"Tracing Tab, Data Rules Tab" on page 554

"Tracing Tab, BPI Rules Tab" on page 557

"Tracing Tab, Alert Rules Tab" on page 557

Tracing Tab, Tracing Properties Tab

User interface elements are described below:

UI Element (A-Z)	Description
Aggregation delta	The time period allocated for the delay while the business transaction data sample is created. For more information about the data sample, see "Exporting TransactionVision Data to BSM" on page 363. **Default value**: 360 seconds (6 minutes)
Assign Analyzers	Choose an Analyzer from the Available Analyzers column to move to the Assign Analyzer column. The current transaction tracing rule is then assigned to this Analyzer.
Collection Interval	The time period, in seconds, covered by each data sample received for this business transaction. **Default value**: 300 seconds (5 minutes)
Cost per Transaction	You can assign a cost to this transaction, for example if there are internal costs associated with completing the transaction. The Cost per Transaction is different from the Transaction Value.
Currency Code	The applicable currency for the Transaction Value. This currency unit is displayed where the Transaction Value is displayed.
Current Rule Definition	The transaction tracing rule currently assigned to this business transaction.
	Disables tracing on this business transaction.
	Loads a tracing rule from the repository.
The Transaction has a/no exception rule defined	Indicates whether an Exception rule is defined for the transaction. If the rule is defined, the Exception-related KPIs are calculated for the business transaction.

UI Element (A-Z)	Description
The Transaction has a/no failure rule defined	Indicates whether a Failure rule is defined for the transaction. If the rule is defined, the Failure and Success related KPI s are calculated for the business transaction.
The Transaction has a/no value rule defined	Indicates whether a Value rule is defined for the transaction. If the rule is defined, the Value-related KPI s are calculated for the business transaction.
Threshold	The response time threshold for the business transaction. Instances of the transaction that take longer than this time are considered **late**. For details, see "The Transaction States Relevant to Service Health" in "Service Health" on page 348.

Tracing Tab, Classification Rules Tab

The Classification Rules tab of the Rules Definition tab enables you to specify the rules that cause the business transaction to be identified as such.

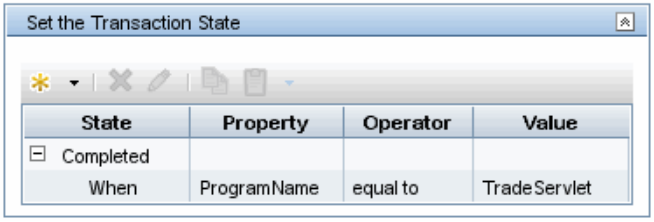
User interface elements are described below:

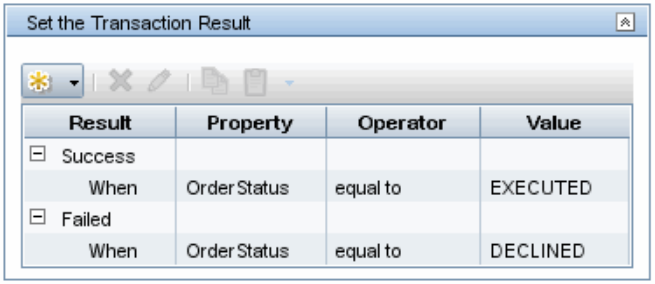
UI Element	Description
	New menu with the following options for creating match conditions (for details see "How to Create a Match Condition" on page 525): ➤ **New Match Condition**. Opens the "Create Match Condition Dialog" on page 564. For instructions, see "How to Create a Match Condition" on page 525. ➤ **New Match Condition using Servlet Wizard**. Launches the Servlet wizard to step you through creating the match condition. ➤ **New Evaluation Block**. Creates an OR branch allowing the match condition to be based on one of multiple sets of conditions. If you do not specify an OR condition, AND is implied by default.
	Delete. Deletes the selected match condition.
	Edit. Opens the Edit Match Condition dialog box, which is similar to the "Create Match Condition Dialog" on page 564. For instructions about using this dialog box, see "How to Create a Match Condition" on page 525.
	Copy button**.** Copies the selected match condition.
	Paste button. Pastes the copied match condition.
Match Conditions	The current set of match conditions for the tracing rule. Match conditions can be created by running one of the tracing rule wizards or by using the "Create Match Condition Dialog" on page 564. See "How to Create a Match Condition" on page 525.

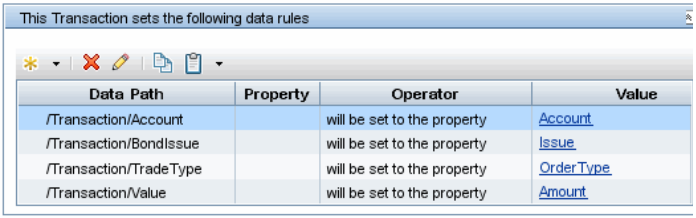
Tracing Tab, Data Rules Tab

The Data Rules tab of the Rules Definition tab enables you to specify the rules that cause the State and Result attributes to be set, as well as any other attribute that you want to have set.

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
(Toolbar elements)	Similar toolbar buttons are used on the Classification Rules tab. See "Tracing Tab, Classification Rules Tab" on page 553. Toolbar buttons that are used differently, are described in the specified area of the Data Rules page.
Set the Transaction State area	The conditions under which the transaction is considered complete or in-process. The table defines the rule or rules for setting the transaction state. Following is an example:  ➤ **Create/New** dropdown. ➤ **Create Processing Rule.** ➤ **Create Completion Rule.** ➤ **New Match Condition.** ➤ **New Match Condition Using Servlet Wizard.** ➤ **Paste** dropdown. ➤ **Paste a data transfer rule.** Pastes the copied rule State attribute. ➤ **Paste a match condition.** Pastes the copied match condition.

UI Element	Description
Set the Transaction Result	The conditions under which the transaction is considered successful or failed. The table defines the rule or rules for setting the transaction result. Following is an example:  **Create/New** dropdown. ➤ **Create Success Rule**. ➤ **Create Failure Rule**. ➤ **New Match Condition**. ➤ **New Match Condition using Servlet Wizard**.

UI Element	Description
The transaction Sets the following data rules	Any data value in the transaction model that needs to be set to a certain value. For example, /Transaction/Value:  The Account, BondIssue, TradeType and Value attributes are defined with corresponding event properties. **Create/New** dropdown. ➤ **Add Data Rules** > ➤ **New Data Rule** ➤ **New Data Rule using Servlet Wizard**. ➤ **Add Match Conditions** > ➤ **New Match Condition**. ➤ **New Match Condition using Servlet Wizard**.
Summary of Global Transaction Rules this Transaction Inherits	All rules that impact this transaction.

Tracing Tab, BPI Rules Tab

The BPI Rules tab of the Rules Definition tab enables you to specify the BPI action that is associated with a specific rule.User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
(Toolbar elements)	The same toolbar buttons are used on the Classification Rules tab. See "Tracing Tab, Classification Rules Tab" on page 553.
Event Group	See "BPI Definition Dialog" on page 571.
Event Name	
Transaction Attributes	
Publish Events to BPI	

Tracing Tab, Alert Rules Tab

The Alert Rules tab of the Rules Definition tab enables you to extend the types of sent to BSM. You can enable the following three types of alerts for TransactionVision business transactions, described in the table below:

- Incomplete Transaction Alert
- Response Time Threshold Alert
- Custom Alerts

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
(Toolbar elements)	The same toolbar buttons are used on the Classification Rules tab. See "Tracing Tab, Classification Rules Tab" on page 553. Toolbar buttons that are used differently, are described in the specified area of the Alert Rules page.
Incomplete Transaction Alert	Collapsible area to enable an incomplete transaction alert. ➤ **Generate Incomplete Transaction Alert.** Check to implement this alert. ➤ **Alert Trigger Response Time (msec):** Enter a value in milliseconds representing how long a transaction can be running before it generates an incomplete alert.

UI Element	Description
Response Time Threshold Alert	Collapsible area to enable the response time alert for business transactions that meet the match condition criteria. An alert is generated based on the threshold setting in the Tracing Properties tab (see "Tracing Tab, Tracing Properties Tab" on page 551). ➤ **Generate Response Time Threshold Alert.** Check to implement this alert. This alert must be checked to be able to add match conditions. ➤ **Match Conditions**. For descriptions of the match condition options, see "Tracing Tab, Classification Rules Tab" on page 553. For instructions about creating a match condition manually, see "How to Create a Match Condition" on page 525.
Custom Alerts	Collapsible area to create custom alerts. For details on creating a custom alert see "How to Create Custom Alerts" on page 536. ➤ **Create Custom Alert**. Opens the Custom Alert Configuration dialog box where you enter the new custom alert name. ➤ Right-click on the custom alert name to see the Match Condition and toolbar options. Most of the options are described on the "Tracing Tab, Classification Rules Tab" on page 513, except for the ones below: ➤ **Paste a Custom Alert.** Pastes the copied custom alert. ➤ **Paste a match condition.** Pastes the copied match condition. ➤ **Show complete condition path**. Toggle to display the complete XPath name in the table or just the property name.

Transaction Tracing Rules Page

The Transaction Tracing Rules page enables you to define transaction tracing rules. The page lists all the rules currently available. You can browse to a specific rule or group them by Analyzer.

To access	**1** Select **Admin** > **Transaction Management** > **Monitoring.** **2** Click the **Transaction Tracing Rules** button in the bottom-left pane.
Important information	The Transaction Tracing Rules left pane includes folders for two types of rules: ➤ **Transaction Rules.** Defined for specific transaction types. ➤ **Global Rules**. Defined for all events. Transaction rules are evaluated in the order in which the rules appear, from top to bottom. When the analyzer evaluates rules, it stops evaluation and performs a classification on an event on the first set of conditions that completely match the given rules. Keep this in mind when designing your tracing rules. Note that to see and change the evaluation order, you need to group rules by Analyzer. For instructions, see "How to Group/Ungroup Rules" on page 532. If an Analyzer contains a global rule that is assigned directly to an Analzyer, the object assigned specifically to the one Analyzer has precedence and is evaluated first.
Relevant tasks	➤ "How to Assign a Rule to the Appropriate Analyzers" on page 531 ➤ "How to Copy a Rule to Create a New One" on page 532 ➤ "How to Group/Ungroup Rules" on page 532
See also	"Transaction Tracing Overview" on page 514

User interface elements are described below:

UI Element	Description
	New menu with the following commands: ➤ **Create Classification Rule**. ➤ **New Classification Rule**. ➤ **Launch Transaction Tracing Rule Wizard for Servlets**. ➤ **Create Common Rule**. ➤ **New Global Rule**. ➤ **Launch Transaction Tracing Rule Wizard for Servlets**.
	Delete the selected item. Deletes the selected unassigned rule. Disabled by default until an assigned rule is selected.
	Refresh.
	Group Rules Assigned Analyzer. Group/ungroup toggle to group/ungroup rules by Unassigned rules and by Analyzers. See "How to Group/Ungroup Rules" on page 532.
	Increase the selected rule definition's evaluation priority.
	Decrease the selected rule definition's evaluation priority.
	Export all Transaction Rule Definitions as XML to a file. For details, see "Importing/Exporting Transaction Tracing Rules" on page 519.
	Import Transaction Rule Definition XML from a file. For details, see "Importing/Exporting Transaction Tracing Rules" on page 519.

This page includes the following tabs:

- "Transaction Tracing Rules Page, General Tab" on page 562
- "Transaction Tracing Rules Page, Rule Definitions Tab" on page 563

Transaction Tracing Rules Page, General Tab

This tab enables you to assign a rule to an Analyzer.

To access	**1** Select **Admin** > **Transaction Management** > **Monitoring.** **2** Click the **Transaction Tracing Rules** button in the bottom-left pane. **3** Navigate to a particular rule. **4** Click the **General** tab in the right pane.

User interface elements are described below:

UI Element	Description
	Move a selected Available Analyzer to the Assigned Analyzer column.
	Move a selected Assigned Analyzer back to the Available Analyzer column.
	Move all Available Analyzers to the Assigned Analyzer column.
	Move all Assigned Analyzers back to the Available Analyzer column.
Available Analyzers	The list of Analyzers in the deployment environment for which you have permission to manage. To add Analyzers, see "How to Add an Analyzer to the Deployment Environment" on page 106.
Assigned Analyzers	The Analyzers to which the rule is assigned. This Analyzer applies the rule continuously to the events it has collected to produce business transactions of the type.
Rule Definition Name	The name of the selected rule.

Transaction Tracing Rules Page, Rule Definitions Tab

The Rule Definitions tab enables you to configure the transaction tracing rules. The Rule Definitions tab changes depending on whether you select a Transaction rule or a Global rule in the left panel.

- Selecting a **Transaction** rule displays the following four tabs, which have the same content as the tabs in the Transaction Configuration page: Classification Rules, Data Rules, BPI Rules, and Alert Rules.

  For descriptions of each tab, see:

  - "Tracing Tab, Classification Rules Tab" on page 553
  - "Tracing Tab, Data Rules Tab" on page 554
  - "Tracing Tab, BPI Rules Tab" on page 557
  - "Tracing Tab, Alert Rules Tab" on page 557

- Selecting a **Global** rule displays the following two tabs: Data Rules (which has the same content as the Data Rules tab in the Transaction Configuration page) and Matching Rules.

  The Matching Rules tab allows you to create match conditions for the global rule. For instructions on creating match conditions, see "How to Create a Match Condition" on page 525, and for user interface details, see "Create Match Condition Dialog" on page 564.

Create Match Condition Dialog

This dialog enables you to create a match condition for a transaction tracing rule. When you edit the match condition, this same dialog displays but is called the Edit Match Condition dialog, with the populated match condition attributes.

To access	For a rule associated with a Business Transaction CI: **1** Select **Admin** > **Transaction Management** > **Administration** > **Monitoring**. **2** Click the **Transaction Configuration** button in the bottom-left pane. **3** Navigate to a particular Business Transaction. **4** Select **Tracing** > **Classification Rules** in the right pane. **5** Click and then select **New Match Condition**. For a rule not associated with a Business Transaction CI: **1** Select **Admin** > **Transaction Management** > **Administration** > **Monitoring**. **2** Click the **Transaction Tracing Rules** button in the bottom-left pane. **3** Navigate to a particular rule. **4** Select **Tracing** > **Classification Rules** in the right pane. **5** Click and then select **New Match Condition**.

User interface elements are described below:

UI Element	Description
Filter Properties by Selected Events	
	Collapsible area in which you can select an existing transaction's events as candidates for property names and values in the Names panel.
	Load events. Opens the **Select Transaction Events** dialog in which you specify criteria for retrieving a set of events. For descriptions of the fields, see "Select Transaction Events Dialog" on page 566.

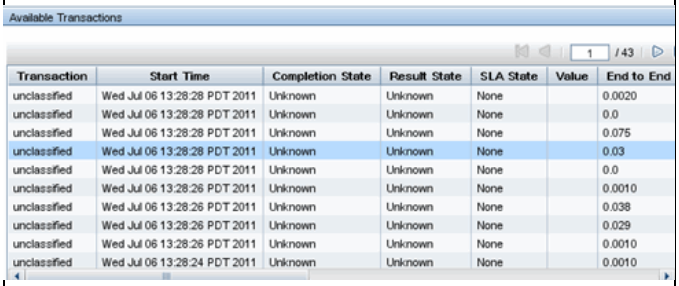
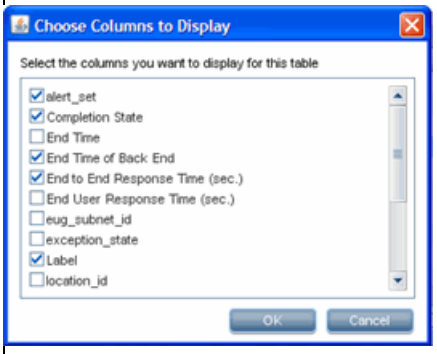
UI Element	Description
	Show transaction events properties and values. When this button is selected, the **Name** and **Values** panes do not show names and values for a single selected transaction event. Instead, the panes display the union of names and values for all events in the transaction. For example, if you have a transaction with one MQSeries and one CICS event, selecting a single event (such as MQSeries) only displays MQSeries under the **Technology** entry in the **Names** pane. And fields such as **StdHeader ExitTime** list the time for this specific event under **Values**. But if you select **Show transaction events properties and values**, the **Technology** entry lists both **MQSeries** and **CICS** and the **ExitTime** displays the time values for both events.
Available Properties	
Names	Shows all available XPath properties. If you selected a transaction and/or event, all XPaths that occur within that event transaction display. For information on populating this tree with a selected transaction's event properties, see "Select Transaction Events Dialog" on page 566.
Values	The value of the selected event property. Only displays when the **Selected transaction event properties** option is enabled and selected.
All fixed properties	Displays all default XPath properties that can occur in TransactionVision events.
Selected transaction event properties	Displays the XPaths relevant to the selected transaction event as well as the standard fixed XPaths. Selecting this option also shows the value of each custom XPath so that you can choose the value directly. **Default**: disabled and inactive. This option is activated and can then be enabled when a transaction is selected.
Name	XPath name of the selected property to be used in the match condition.

UI Element	Description
Operator	Operator to be used in the match condition.
Value	Value to be used in the match condition.
Enable advanced XPath string parsing	The TransactionVision XPath parser does not support all XPath syntax. If you need to use features not supported by TransactionVision, this option enables an external XPath parser (Xalan). Use the TransactionVision XPath parser whenever possible. **Default**: disabled.

Select Transaction Events Dialog

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
Filter Transactions	
Time Period	Select the time period in which the transactions occured.
Transaction State	Select whether the transactions are unspecified, in-process, or completed.
Transaction Result	Select whether the transactions are unspecified, successful, or failed.
Include Transactions	Select one or more transactions to be included.
Search on Column	Select the column in the dropdown list and enter the value in the **for** text field.
Apply	Click to populate the **Available Transactions** area with all transactions that meet the specified criteria.

UI Element	Description
Available Transactions	
<transaction table>	List of transactions that display when **Load transaction events** was selected in the Edit XPath Value dialog.  Select the transaction that includes the events you want, which populates the events table in the **Transaction Events** area.
	Display visible columns. Click to open the dialog to select the columns you want to display. 

UI Element	Description
Transaction Events	
<events table>	Lists the events associated with the selected transaction in the Available Transactions area. Select an event.
Select	Enabled when a transaction is selected. After selecting an event in the Transaction Events area, click to return to the Edit XPath Value dialog where the selected event's properties are populated in the Available Properties area. When you click **Select**, you are returned to the Create/ Edit Match Condition dialog. The **Selected transaction event properties** option is enabled by default so that you can see the values for each selected property.

Data Rule Configuration Settings Dialog Box

This dialog enables you to create a new data rule.

To access	For a rule associated with a Business Transaction CI: **1** Select **Admin** > **Transaction Management** > **Administration** > **Monitoring.** **2** (left pane) Click the **Transaction Configuration** button. **3** Navigate to a particular Business Transaction. **4** (right pane) Select **Tracing** > **Data Rules**. **5** Click and then select **New Data Rule**. For a rule not associated with a Business Transaction CI: **1** Select **Admin** > **Transaction Management** > **Administration** > **Monitoring.** **2** (left pane) Click the **Transaction Tracing Rules** button. **3** Navigate to a particular rule. **4** (right pane) Select **Tracing** > **Data Rules**. **5** Click and then select **New Data Rule**.

User interface elements are described below:

UI Element (A-Z)	Description
	Launches the Create New Data Definition dialog box. See "Create New Data Definition Dialog Box" on page 587
Transaction Path	The path.
Column Type	The column data type.
Column Description	Description of the column.
Advanced Parameters	Enable/disable advanced parameters: ➤ **Final**. By default the <Attribute> rules are only evaluated if the corresponding transaction attribute has no value yet—the definition is considered to be final. Once a final rule has set the value of the transaction attribute, it (and other final rules that refer to the same attribute) are not evaluated again. To allow transaction attributes to be set and updated more than once, the attribute rule can be declared with an attribute **@final** set to **false**. **Default**: on ➤ **Precedence**. Rule attribute **@precedence**, which can be used to control the setting of new values for transaction attributes. This attribute can only be set for rules referencing integer valued transaction attributes. If set to true (on) then an existing attribute value is only overwritten if the new value is greater than the old value. This mainly makes sense for **state** and **result** like attributes where all values can be ordered according to a priority (such as UNKNOWN->PROCESSING->COMPLETE), though it can be applied to any integer valued attribute. All **@precedence** rules are automatically considered to be non-final as well. By default (if the **@precedence** attribute is not specified) the value is false (off). **Default**: off

UI Element (A-Z)	Description
Advanced Parameters (continued)	➤ **Attribute evaluation occurs outside classify section**. Attribute rules are used to set and update values of transaction attributes. They can either be defined inside of a <Classify> section, in which case they are only evaluated once at the time of classification, or they can appear outside of the <Classify> section if they need to be evaluated for all events of the transaction. **Default**: off ➤ **Append or sum attribute updates with existing content. On** appends a new value to an existing value when two transactions are merged together during processing, so that one value does not overwrite the other. **Off** overwrites an existing value with a new value for two transactions that are merged. For more information, see the *HP TransactionVision Advanced Customization Guide*. **Default**: off

BPI Definition Dialog

This dialog enables you to define the mapping of events to BPI.

To access	For a rule associated with a Business Transaction CI: **1** Select **Admin** > **Transaction Management** > **Administration** > **Monitoring.** **2** (left pane) Click the **Transaction Configuration** button. **3** Navigate to a particular Business Transaction. **4** (right pane) Select **Tracing** > **Classification Rules**. **5** Click and then select **New BPI Event**. For a rule not associated with a Business Transaction CI: **1** Select **Admin** > **Transaction Management** > **Administration** > **Monitoring.** **2** (left pane) Click the **Transaction Tracing Rules** button. **3** Navigate to a particular rule. **4** (right pane) Select **Tracing** > **Classification Rules**. **5** Click and then select **New BPI Event**.
Relevant tasks	"How to Send Events to BPI" on page 534
See also	*Using Business Process Insight*

User interface elements are described below:

UI Element	Description
Event Group	The BPI group name. This is defined within the BPI Modeler for the TransactionVision process.
Event Name	The BPI event name. This is defined within the BPI Modeler for the TransactionVision process.

UI Element	Description
Transaction Attributes	From the drop-down list, check the attributes to be exported to BPI. These events are imported to the BPI Modeler. The rule is exported to BPI when it is saved. **Default value**: all attributes are exported
Publish Events to BPI	Events are published at one of these occurrences: ➤ **Only once, when this transaction is classified** ➤ **Any time the specified condition is matched** **Default value**: Only once, when this transaction is classified.

Troubleshooting and Limitations

This section includes the following topics:

Assign Transaction Tracing Rules to only one Analyzer

Correlation cannot occur across Analyzers. All the events for one transaction instance should be coming into the same Analyzer.

Incorrect Sample Counts - Classify Rule Problems

If the classify rules for a transaction class are too general such that different parts of a transaction can be classified while the transaction is still being processed, it is possible that incorrect sample counts are sent in the BSM Aggregate TV Data sample. To avoid this, make the classify rules for a transaction class specific enough so that they do not match multiple events for a transaction.

16

Business Transaction Monitoring

This chapter includes:

Concepts

➤ Business Transaction Monitoring Overview on page 574

Tasks

➤ How to Set Up a Business Transaction for Monitoring on page 576

Reference

➤ Transaction Configuration Page, Monitoring Tab on page 578

Concepts

Business Transaction Monitoring Overview

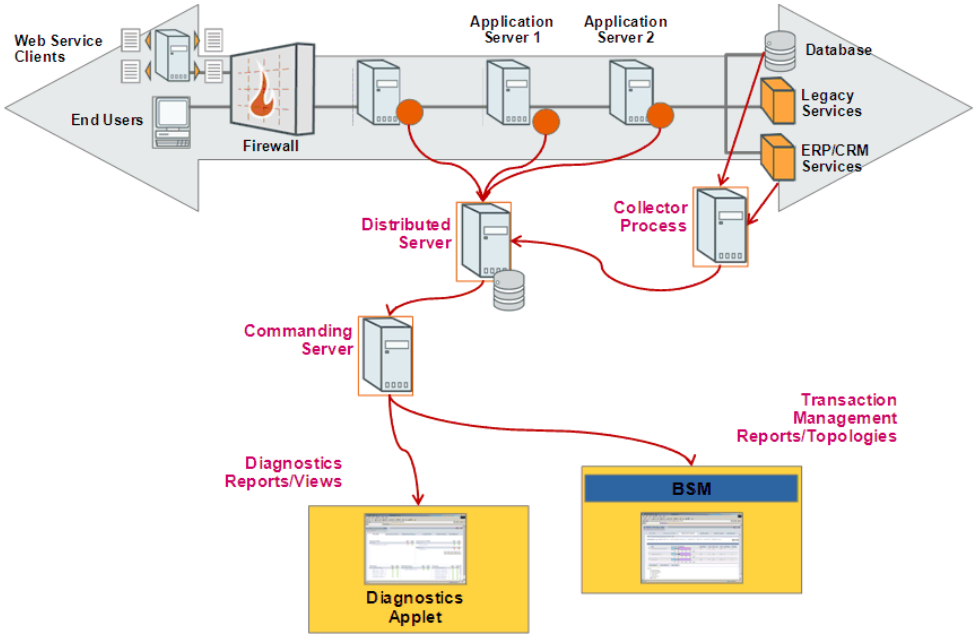
Business transaction monitoring uses data collected and analyzed by various HP Diagnostics components:

- The agents are deployed on application servers, collecting and aggregating statistics data from all transaction instances.
- The Distributed Servers and Commanding Server further aggregate and correlate statistics data from all agents, and organize them based on top-level (starting) server requests. A top-level server request is one which does not have any inbound calls.

A business transaction that is enabled for monitoring is one that is configured by selecting a set of top-level server requests as entry points into the transaction. The probes and server automatically identify all the other server requests that are reachable by this top-level server request.

In the deployment environment, the Transaction Management application receives its data from the HP Diagnostics Commanding Server.

Following is a diagram of the deployment environment.



Tasks

How to Set Up a Business Transaction for Monitoring

This task describes how to configure a business transaction for monitoring.

This task includes the following steps:

- ➤ "Prerequisites" on page 576
- ➤ "Create a new business transaction if necessary" on page 576
- ➤ "Enable the business transaction for monitoring" on page 576
- ➤ "Identify the server requests to comprise the business transaction" on page 577
- ➤ "Verify the business transaction" on page 577

1 Prerequisites

HP Diagnostics must be deployed to the same BSM deployment environment.

2 Create a new business transaction if necessary

If you are configuring an existing business transaction you can skip this step.

For information about creating a new business transaction, see "How to Create a Business Transaction CI" on page 525.

3 Enable the business transaction for monitoring

a Select **Admin** > **Transaction Management** > **Monitoring.**

b (left pane) Click the **Transaction Configuration** button.

c Navigate to and click the business transaction for which you want to enable monitoring.

d (right pane) Click the **Monitoring** tab.

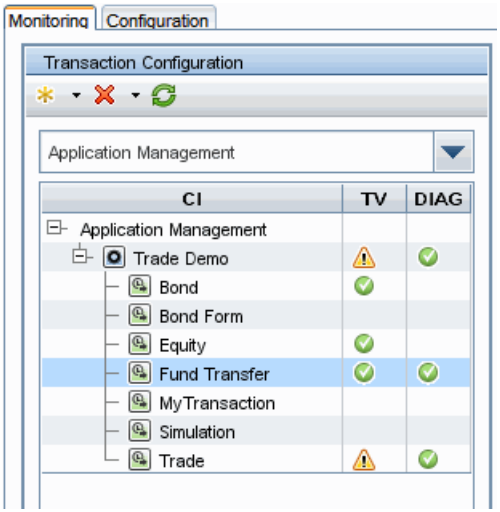
4 Identify the server requests to comprise the business transaction

On the Monitoring tab, you choose from the current set of server requests to identify which ones comprise this business transaction. For details about this interface, see "Transaction Configuration Page, Monitoring Tab" on page 578.

5 Verify the business transaction

In the left pane, click the **Refresh** button.

A green arrow icon in the **Diag** column indicates that the business transaction has been successfully enabled for monitoring:



Reference

Transaction Configuration Page, Monitoring Tab

This tab enables you to define a business transaction based on HP Diagnostics server requests.

To access	**1** Select **Admin** > **Transaction Management** > **Monitoring.** **2** (left pane) Click the **Transaction Configuration** button. **3** Navigate to the business transaction on which you want to enable transaction monitoring. **4** (right pane) Click the **Monitoring** tab.
Important information	➤ HP Diagnostics must be deployed to the BSM deployment environment. ➤ Business Transactions set up for monitoring in this way appear in the **Transactions - Business Transactions** view in HP Diagnostics. See the *HP Diagnostics User's Guide*.

Available Top-Level Server Requests Area

This area provides two ways to choose the server requests that comprise the business transaction: browse and select, or search.

User interface elements are described below:

UI Element	Description
Browse tab	Use this tab to browse to the top-level Server request on which the transaction is based. The available top-level server requests are provided by the HP Diagnostics Server. Select the desired server request, then click the right arrow to add it to the business transaction monitoring definition.
Search tab	You can use this tab to locate top-level Server requests. **Search Criteria.** Enter any of the following search criteria: ➤ **Probe.** The name of the probe. ➤ **Probe group.** The name of the probe group. ➤ **Server Request.** The name of the server request. ➤ **Type.** You can qualify the probe or server request by choose a type from this drop-down list. **Search Results.** After you click the Search button, the Server Request list is populated. Select the desired server request, then click the right arrow to add it to the business transaction monitoring definition.

Selected Top-Level Server Requests Area

This area builds the list of server requests that comprise the business transaction.

User interface elements are described below:

UI Element	Description
Server Request	The name of the server request.
Type	The type of the server request.
Probe	The name of the probe.
Probe Group	The name of the probe group.

17

Business Transaction Data Model

This chapter includes:

Concepts

➤ Business Transaction Data Model Overview on page 582

Tasks

➤ How to Add a Column of Data to the Transaction Data Model on page 584

Reference

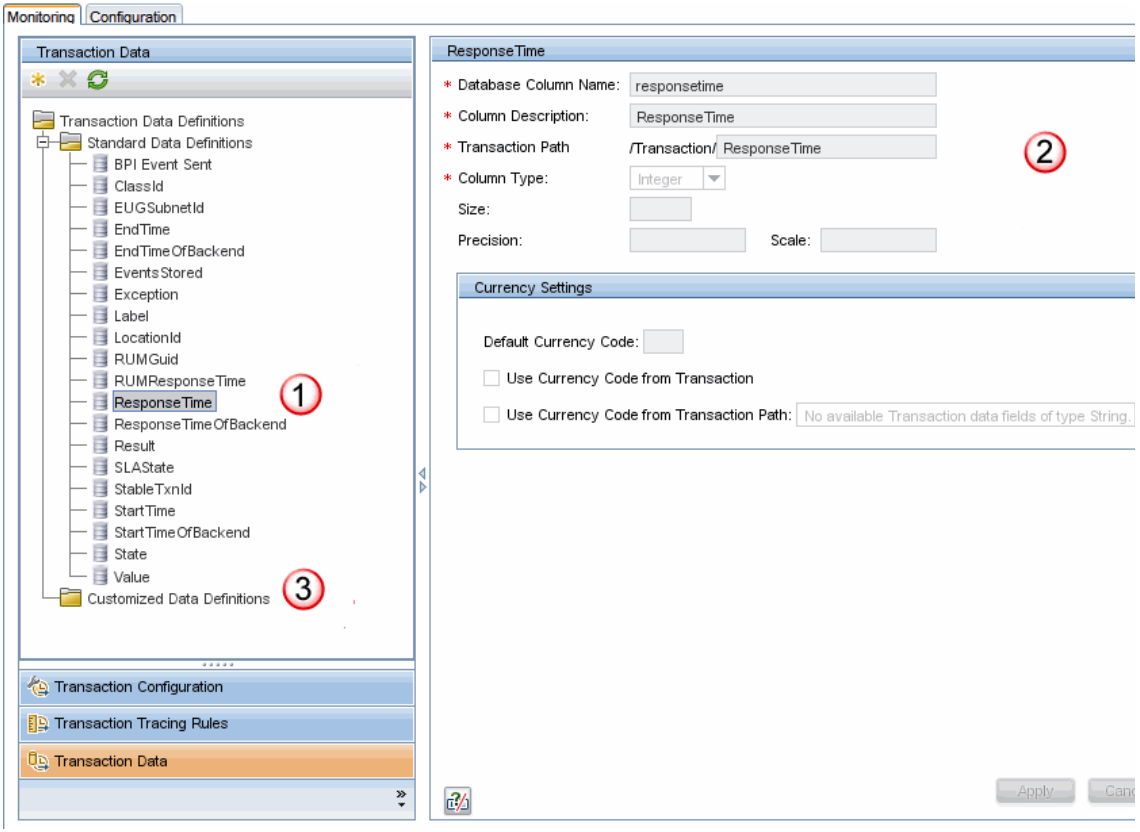
➤ Transaction Data Model User Interface on page 585

Concepts

Business Transaction Data Model Overview

Business Transactions CIs that are enabled for tracing have corresponding data structures that are used in the Analyzer cache and database schema to store the data.

You can view the columns of the data structure in the Transaction Management Monitoring tab:



The following table describes the locations in this page:

Callout	Description
1	The Standard Data Definition folder contains the predefined columns.
2	You can view the details of a selected column.
3	In some cases, you need to customize the structure. You can add new columns to the Customized Data Definitions folder.

When you define the database column you are defining the location where the information is actually stored. This allows for it to be used as an attribute in a transaction instance.

The XPath is the identifier used to reference this bit of data within TransactionVision.

Once the new attributes are added to the transaction data model, you can view them on the Transaction Tracking or Transaction Details reports.

Tasks

How to Add a Column of Data to the Transaction Data Model

This task describes how to add a custom attribute to a transaction. A new attribute is represented as a column in the Transaction Data model.

1 Determine the requirements of the new column. You will need to know how big it should be, its data type, and its currency type if applicable.

You can choose from these data types:

Currency
Date
Decimals
Double
Integer
String

2 Navigate to **Admin** > **Transaction Management** > **Monitoring** > **Transaction Data.**

3 (left pane) Click . The Create New Data Definition dialog box appears. For details about this interface, see "Create New Data Definition Dialog Box" on page 587.

4 Restart the Analyzer.

Reference

Transaction Data Model User Interface

This section describes:

➤ Transaction Data Page on page 585

➤ Create New Data Definition Dialog Box on page 587

Transaction Data Page

This page enables you to view the column details for the current business transaction data model.

To access	**1** Select **Admin** > **Transaction Management** > **Monitoring.** **2** (left pane) Click the **Transaction Data** button. **3** Navigate to the column for which you want to view or edit the definition.
Important information	➤ The columns appear in two categories: ➤ **Standard Data Definitions.** The predefined transaction data model. You cannot delete or modify these. ➤ **Customized Data Definitions.** The additions to the predefined transaction data model. You can add columns here. ➤ For more information about customizing the transaction data model, see the *HP TransactionVision Advanced Customization Guide* PDF.

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
Column Description	Description of the column.
Column Type	The data type of the column.
Currency Settings	For columns that represent currency, the location from which to take the currency code setting: **Default Currency Code**. **Use Currency Code from Transaction**. **Use Currency Code from Transaction Path**. See "Currency Columns" in the *HP TransactionVision Advanced Customization Guide* PDF.
Database Column Name	The column name.
Precision	For columns of numeric type, the precision.
Scale	For columns of numeric type, the scale.
Size	The size of the column.
Transaction Path	The XPath column path. This is how you reference the attributes.

Create New Data Definition Dialog Box

This page enables you to customize the business transaction data model by adding a new column.

To access	**1** Select **Admin** > **Transaction Management** > **Monitoring.** **2** (left pane) Click the **Transaction Data** button. **3** Click the **New** button.
Important information	➤ This dialog allows you to modify the model for the business transaction CI by adding a new column data. ➤ All transactions are affected by this model change. ➤ For more information about customizing the transaction data model, see the *HP TransactionVision Advanced Customization Guide* PDF.
Relevant tasks	"How to Add a Column of Data to the Transaction Data Model" on page 584

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
Column Description	These elements are the same that appear on the "Transaction Data Page" on page 585.
Column Type	
Currency Settings	
Database Column Name	
Precision	
Scale	
Size	
Transaction Path	

18

Custom Correlation

This chapter includes:

Concepts

- Custom Correlation Overview on page 590
- About Custom Event-to-Event Correlation on page 591
- About Custom Tracking ID Correlation on page 593
- About Modification Rules on page 594
- Managing Custom Correlation and Modification Rules on page 597

Tasks

- How to Create Custom Tracking ID Rules on page 599
- How to Replace a Standard Tracking ID With a Tracking ID Rule on page 601
- How to Clear a Standard Tracking ID Rule on page 603
- How to Create Event-to-Event Correlation Rules on page 604
- How to Create Modification Rules on page 607
- How to Test a Modification Rule's Regular Expression on page 612
- How to Use Event Modification With Custom Correlation - Work Flow on page 613

Reference

- Custom Correlation User Interface on page 617
- Troubleshooting and Limitations on page 632

Concepts

Custom Correlation Overview

TransactionVision performs event correlation by default to correlate events into a business transaction, based on criteria such as message ID, correlation ID, put time and other fields in these events. However, there may be times when these criteria are not sufficient to perform event correlation. In these cases, you can create custom correlation and modification rules to expand the criteria such as include or exclude other data fields or modify the event criteria.

TransactionVision provides the Event Customization Rules pages to assist in managing the following correlation and event modifier rules (for descriptions of the user interface, see "Custom Correlation User Interface" on page 617):

- **Correlation Rules:** Allow you to create custom rules for event-to-event and tracking ID correlation.
  - **Event-to-event** relationships are created by the Analyzer between two specific events, such as a JMS send and a JMS receive, usually to indicate a directional correlation flow from one event to another. For more details, see "About Custom Event-to-Event Correlation" on page 591.
  - **Tracking IDs** are generated by an agent and are used to tie two or more events together. One example would be a Servlet calling into several EJBs. The agent would generate the same unique correlation ID for all these events, and the Analyzer would correlate those events into a single business transaction. For more details, see "About Custom Tracking ID Correlation" on page 593.
- **Modification Rules:** Allow you to modify incoming events, such as extracting a value from payload data to be inserted in the event. For more details, see "About Modification Rules" on page 594.

Use Cases for Custom Correlation

- A transaction is broken across two instances and needs to be correlated by some common ID.
- Some custom events were added and need to be correlated by a common ID into a transaction instance.
- A user ID is located in payload data. A rule needs to be created to pull out the user data and consider it as the correlation ID.

About Custom Event-to-Event Correlation

Default Event-to-Event Correlation

By default and based on certain criteria such as message ID, correlation ID, PUT time and other fields in these events, the following event-to-event correlation occurs:

- PUT/GET scenarios
  - WebSphere MQ MQPUT and MQGET events
  - JMS-JMS
  - JMSPubSub-JMSPubSub
  - IMSBridge-IMSBridge
  - MQ-JMS
  - MQ-IMSBridge
- Technology-specific criteria
  - CICS-CICS
  - DataPower-DataPower
  - DataPower-Servlet

Custom Event-to-Event Correlation Rules

In cases where these criteria are not sufficient they need to be expanded to include other data fields, such as those from the event's user data, or may need to be relaxed to exclude some of the standard fields, or may need to be modified in other ways.

Following are some scenarios where event-to-event custom correlation rules may be required (see "How to Create Event-to-Event Correlation Rules" on page 604):

- TransactionVision agents may not be installed on some systems, such as those belonging to external agents. Therefore, the messages going out to the un-monitored systems would need to be correlated with the replies coming back from these systems.
- Unique message IDs or correlation IDs are not used by the applications. In this scenario, custom fields from the message data may need to be used to correlate message PUTs and GETs.
- An application that replies to a message swaps the message ID and correlation ID fields and this application is not monitored by TransactionVision agents.

This correlation can be done by defining custom event correlation rules in the Event Customization Rules user interface (see "Event Customization Rules Page" on page 618). Alternatively, if complex logic is required to be implemented, a Java bean can be written. For details of a bean implementation, see "4.5 Performing Event Analysis" in the *HP TransactionVision Advanced Customization Guide* PDF.

Event-to-Event Correlation Workflow

1. Establish a relationship between two events so that they are correlated into the same transaction instance and can see a flow link between them in the topologies.
2. Locate events by IDs:
   a. Locate an event ID to specify as the ID for the source event (PUT).
   b. Locate an event ID to specify as the ID for the target event (GET).

   To locate an event you can run a query to retrieve a set of events.

3 Set the match conditions for each event.

4 Set the location of the correlator ID for each event.

5 For the relationship type, choose message or control flow and the to/from direction.

6 Assign the rule to an Analyzer.

About Custom Tracking ID Correlation

The TransactionVision Agent generates standard tracking IDs for events that are discovered along a call or execution path, for example a Servlet calling into an EJB which is then making a JMS call. If this standard tracking ID correlation is not sufficient then custom tracking IDs are needed.

Following are some scenarios where custom tracking ID rules may be required:

- TransactionVision agents may not be installed on some systems. To capture the full transaction flow, custom tracking IDs based on common data in the payload may be used to tie events across those gaps together. See "How to Create Custom Tracking ID Rules" on page 599.
- TransactionVision agents may not be able to generate a tracking ID because the execution flow is continued in another thread or some other limitation. See "How to Create Custom Tracking ID Rules" on page 599.
- Standard TransactionVision events need to be tied to user events or vice versa. See "How to Replace a Standard Tracking ID With a Tracking ID Rule" on page 601.
- The default tracking IDs correlation is undesirable and needs to be broken up. See "How to Clear a Standard Tracking ID Rule" on page 603.

Correlation by Tracking ID Work Flow

1. Correlate two events based on an ID they have in common and some other conditions. If two events have the same ID, they are correlated to the same transaction.
2. Locate an event that has an instance of the ID for you to specify as the ID.

   To locate an event you can run a query to retrieve a set of events.
3. Assign the rule to an Analyzer.

About Modification Rules

The modification rules allow you to modify incoming events, such as extracting from payload data a value to be inserted in the event. The modification rule can govern what is extracted, how, and where the result is placed in the event being processed. For each event, the modification rule is applied to see if it matches the event. If so, a new element is added to the event based on the value specified in the rule.

The modification rules support both regular expressions and XPath expressions to denote the data to pull out of an event. By changing the rules, you can change what is processed in the incoming events, and what is extracted.

➤ For instructions about creating modification rules, see "How to Create Modification Rules" on page 607.

➤ For modification rule examples, see "Modification Rule Examples" on page 595.

Modification Rule Examples

Transactionvision provides two example modification rules on the "Event Customization Rules Page" on page 618. These rules provide samples of what an event modifier rule looks like and some of its capabilities.

The two example modification rules are:

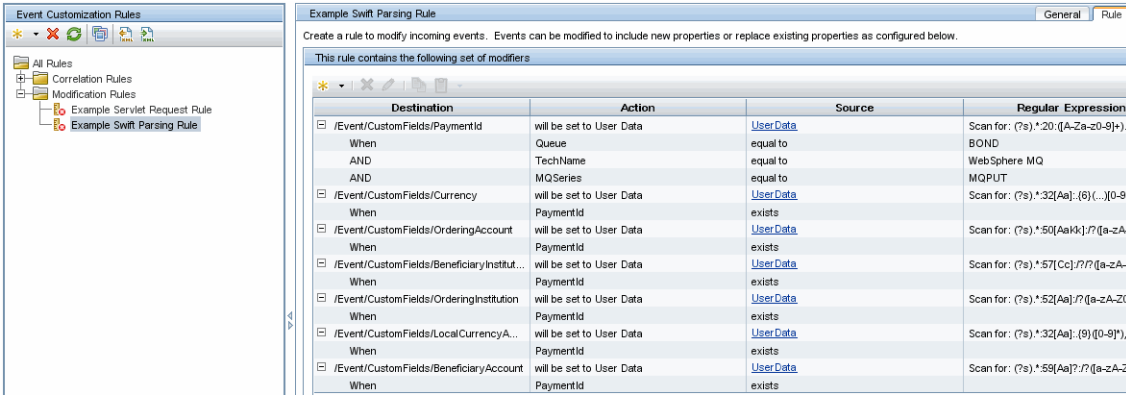
- "Example Swift Parsing Rule" on page 595
- "Example ServletRequestRule" on page 596

Example Swift Parsing Rule

This rule includes example rules for handling and extracting SWIFT payment message data from the payload of a WMQ event. These rules detect the presence of SWIFT format data by looking for a **:20:** field. In addition to indicating the data is a SWIFT message, this field provides a unique payment ID. The rules for populating the other fields are evaluated only if a payment ID field is found.

Note: It is important to consider whether to use extra conditions as shown in this example, where a regular expression test is done even though it is known that the expression is relevant. This is because excessive use of regular expression extracting could impact Analyzer performance.

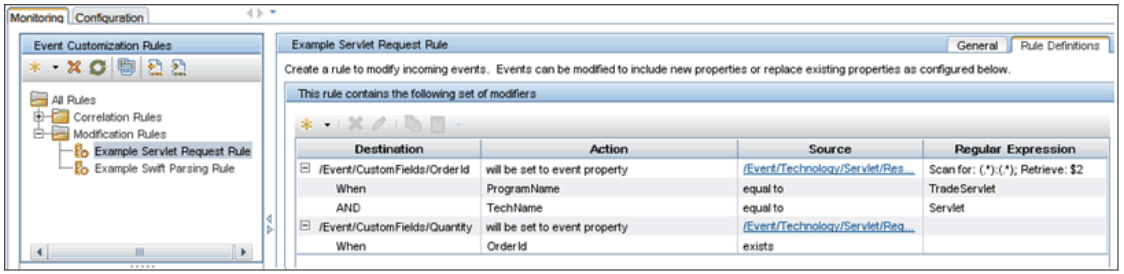
Following is an example of the Example Swift Parsing Rule:



Example ServletRequestRule

This rule is a simple example that matches and extracts an **OrderId** from a servlet response header. If this **OrderId** field exists, it then extracts a **Quantity** field.

Following is an example of the Example ServletRequestRule:



Managing Custom Correlation and Modification Rules

You can manage the correlation and modification rules on the "Event Customization Rules Page" on page 618 as follows:

- "Group/Ungroup Correlation and Modification Rules" on page 597
- "Copy Correlation and Modification Rules" on page 597
- "Enable Correlation and Modification Rules" on page 598
- "Import/Export Correlation and Modification Rules" on page 598

Group/Ungroup Correlation and Modification Rules

You can group rules in the left pane so that you can see at a glance:

- Each Analyzer and the Correlation and Modification rules assigned to each Analyzer.
- All unassigned Correlation and Modification rules.

When you ungroup the rules, they go back to the original order.

To group or ungroup the rules, click the **Group Rules Assigned Analyzer** toggle icon.

Copy Correlation and Modification Rules

Instead of creating a new rule and all its settings, you can copy a rule in the left pane and then edit it to create a new rule.

To copy a rule, right-click on the rule and select **Copy**. Rename the rule to a unique name on the General tab in the right pane, then edit and enable it. For instructions about creating rules, see:

- "How to Create Custom Tracking ID Rules" on page 599
- "How to Create Event-to-Event Correlation Rules" on page 604
- "How to Create Modification Rules" on page 607.

Enable Correlation and Modification Rules

Correlation and modification rules are disabled by default — they are preceded by red icons. A rule is enabled when it is assigned to one or more Analyzers. Enabling a correlation or modification rule results in the corresponding rule becoming active on that Analyzer and the icon turning green.

For instructions on enabling rules, see "How to Assign a Rule to the Appropriate Analyzers" on page 531.

Import/Export Correlation and Modification Rules

You can import and export correlation and modification rules from/to an xml file to create a backup of all the rules and configurations.

- When you export rules, the xml file includes the event criteria and Analyzer assignments. All rules currently on the Event Customization Rules page are exported.

  To export correlation and modification rules, from the **Admin** > **Transaction Management** > **Monitoring** tab, select **Event Customization Rules** (bottom-left pane), click the **Export All Event Customization Rule Definitions as XML to a file** button, and browse to the location where you want to place the **EventCustomizationRules.xml** file (you can change the default xml name).

- When you import exported rules, all event criteria and Analyzer assignments remain.

  To import transaction tracing rules, from the **Admin** > **Transaction Management** > **Monitoring** tab, select **Event Customization Rules** (bottom-left pane), click **Import Event Customization Rule Definition XML from a file** button, and select the existing **xml** rule file.

Tasks

How to Create Custom Tracking ID Rules

This task describes how to create custom tracking ID rules.

This task includes the following steps:

- "Create a correlation rule" on page 599
- "Enter the location of the identifier to use as a correlator" on page 599
- "Create a match condition or conditions for the tracking ID rule" on page 600
- "Assign the rule to the appropriate Analyzer or Analyzers" on page 600
- "Results" on page 601

1 Create a correlation rule

a Select **Admin** > **Transaction Management** > **Monitoring.**

b Click the **Event Customization Rules** button in the bottom-left pane.

c Click the New icon and select **New Correlation Rule**.

d Enter the name of the rule definition in the right pane and click **OK**.

2 Enter the location of the identifier to use as a correlator

In the **Rule Definitions** page in the right pane, if you know the XPath location of the ID to use as a correlator, enter it in the **Id to correlate by** text box.

If you do not know the correlation ID XPath, do the following:

a Click the **Edit event property used as id to correlate by** icon to open the **Edit XPath Value** dialog.

b Search or scroll for the ID, select it and click **Apply**. Or, you can search for the transactions that have the event data you want to use to search for the ID as described in the next step, Step c.

c If the ID is not available from the fixed list of properties, open the top **Filter Properties by Selected Events** collapsible area and click in the window to load the transaction events. You can also do one of the following:

- Select the **Load transaction events** icon. Loads all events based on the selected transactions.
- Select the **Load query events** icon. Loads all events based on the selected query defined on the Query Criteria tab of the Query page, see "Query Page, Query Criteria Tab" on page 266.

Select the appropriate criteria for the type of transaction or transactions with events you want to use to find the XPath ID.

For user interface details, see "Edit XPath Value Dialog" on page 626.

3 Create a match condition or conditions for the tracking ID rule

It is recommended that you create at least one match condition so that the tracking ID is found and used for specific events. Otherwise, the tracking ID will be found for any event encountered. For instructions, see "How to Create a Match Condition" on page 525.

4 Assign the rule to the appropriate Analyzer or Analyzers

After creating the correlation rule, it is disabled by default shown by the red icon next to the rule name. An enabled rule has a green icon.

Enable this rule by assigning it to an available Analyzer or Analyzers. You do this the same way you assign a Transaction Tracing rule to Analyzers as described in "How to Assign a Rule to the Appropriate Analyzers" on page 531.

5 Results

If you added match conditions, the new tracking ID is only used for incoming events when those conditions are met. This tracking ID is used in addition to any other custom or standard tracking IDs that meet the listed match conditions.

If you did not add match conditions, then the new tracking ID is used for all incoming events.

How to Replace a Standard Tracking ID With a Tracking ID Rule

This task describes how to replace a standard tracking ID with a tracking ID.

This task includes the following steps:

- "Create a correlation rule" on page 601
- "Replace the standard tracking ID with a custom tracking ID rule" on page 602
- "Create a match condition or conditions for the tracking ID rule" on page 602
- "Assign the rule to the appropriate Analyzer or Analyzers" on page 602
- "Results" on page 603

1 Create a correlation rule

a Select **Admin** > **Transaction Management** > **Monitoring.**

b Click the **Event Customization Rules** button in the bottom-left pane.

c Click the New icon and select **New Correlation Rule**.

d Enter the name of the rule definition in the right pane and click **OK**.

2 Replace the standard tracking ID with a custom tracking ID rule

a In the **Rule Definitions** page, click the Advanced Options icon to open the Advanced Options dialog opens.

b Select **Replace default Tracking Id** and click **OK**.

c Add a custom tracking ID in one of the following ways:

- If you know the correlation ID XPath name, enter it in the **Id to correlate by** text box.
- If you do not know the correlation ID XPath name, click the **Edit event property used as id to correlate by** icon to open the Edit XPath Value dialog. Follow the instructions in step 2 of "How to Create Custom Tracking ID Rules" on page 599.

d Click **Apply** and the custom tracking ID replaces the standard tracking ID.

3 Create a match condition or conditions for the tracking ID rule

It is recommended that you create at least one match condition so that the tracking ID is found and used for specific events. Otherwise, the tracking ID will be found in any event encountered. For instructions, see "How to Create a Match Condition" on page 525.

4 Assign the rule to the appropriate Analyzer or Analyzers

After creating the correlation rule, it is disabled by default shown by the red icon next to the rule name. An enabled rule has a green icon.

Enable this rule by assigning it to an available Analyzer or Analyzers. You do this the same way you assign a Transaction Tracing rule to Analyzers as described in "How to Assign a Rule to the Appropriate Analyzers" on page 531.

5 Results

If you added match conditions, the new tracking ID replaces all tracking IDs for incoming events that match any listed match conditions.

If you did not add match conditions, then the new tracking ID replaces the tracking IDs for all incoming events.

How to Clear a Standard Tracking ID Rule

This task describes how to clear a standard tracking ID rule.

This task includes the following steps:

- "Create a correlation rule" on page 603
- "Clear the standard tracking ID" on page 603
- "Create a match condition or conditions for the tracking ID rule" on page 604
- "Assign the rule to the appropriate Analyzer or Analyzers" on page 604
- "Results" on page 604

1 Create a correlation rule

a Select **Admin** > **Transaction Management** > **Monitoring.**

b Click the **Event Customization Rules** button in the bottom-left pane.

c Click the New icon and select **New Correlation Rule**.

d Enter the name of the rule definition in the right pane and click **OK**.

2 Clear the standard tracking ID

a In the **Rule Definitions** page, click the Advanced Options icon to open the Advanced Options dialog opens.

b Select **Clear default Tracking Id** and click **OK**.

c Click **Apply** and the custom tracking ID replaces the standard tracking ID.

3 Create a match condition or conditions for the tracking ID rule

It is recommended that you create a match condition to clear the tracking ID to specify for which events you want the tracking ID to be cleared. Otherwise, the tracking ID is cleared for all events. For match condition instructions, see "How to Create a Match Condition" on page 525.

4 Assign the rule to the appropriate Analyzer or Analyzers

After creating the correlation rule, it is disabled by default shown by the red icon next to the rule name. An enabled rule has a green icon.

Enable this rule by assigning it to an available Analyzer or Analyzers. You do this the same way you assign a Transaction Tracing rule to Analyzers as described in "How to Assign a Rule to the Appropriate Analyzers" on page 531.

5 Results

With the tracking ID cleared, this event will no longer be correlated to other events matching that tracking ID.

How to Create Event-to-Event Correlation Rules

This task describes how to create event-to-event correlation rules.

This task includes the following steps:

- "Create a correlation rule" on page 605
- "Enter the correlator ID for the first event" on page 605
- "Create a match condition or conditions for the first event" on page 605
- "Set the correlation flow between the events" on page 605
- "Locate the ID to correlate by for the second event" on page 606
- "Create match conditions for the second event" on page 606
- "Assign the rule to the appropriate Analyzer or Analyzers" on page 607
- "Results" on page 607

1 Create a correlation rule

a Select **Admin** > **Transaction Management** > **Monitoring.**

b Click the **Event Customization Rules** button in the bottom-left pane.

c Click the New icon and select **New Correlation Rule**.

d Enter the name of the rule definition in the right pane and click **OK**.

2 Enter the correlator ID for the first event

Add the correlator ID XPath name in one of the following ways:

- If you know the event ID XPath name, enter it in the **Id to correlate by** text box.
- If you do not know the event ID name, click the **Edit event property used as id to correlate by** icon to open the Edit XPath Value dialog. Follow the instructions in step 2 of "How to Create Custom Tracking ID Rules" on page 599.

3 Create a match condition or conditions for the first event

It is recommended that you create at least one match condition for the first event so that the correlator ID is found and used just for the first event with this match condition. Otherwise, the correlator ID will be found in any event encountered. For instructions, see "How to Create a Match Condition" on page 525.

4 Set the correlation flow between the events

a Select **Directional Correlation: Create a relationship between two specific events**.

b Specify the **Flow Direction** to determine which direction this relationship represents.

c Specify the flow **Type: Message Flow** or **Control Flow**. The flow types analyze and correlate the event in the same way. The main difference is how the Transaction Detail report shows the result of the relationship between events.

- **Message Flow.** When selected, indicates a direct message flow between the two events. That means the two events are associated with the same message data. For example, a MQPUT and MQGET call dealing with the same message produces a message path relation. For an example of viewing a message flow in the component topology, see "WebSphere MQ and JMS Events" on page 763.
- **Control Flow**. When selected, indicates a control flow between the two events. For example, a servlet calling some other web service produces a transaction path relation. For an example of viewing a control flow in the component topology, see "Servlet and EJB Events" on page 765.

d If you want the event flow to appear in the topology reports, check the **Show this event flow in the Topology reports** check box.

5 Locate the ID to correlate by for the second event

Follow the instructions in step 2.

6 Create match conditions for the second event

Create a match condition or match conditions for the second event. For instructions, see "How to Create a Match Condition" on page 525.

7 Assign the rule to the appropriate Analyzer or Analyzers

After creating the correlation rule, it is disabled by default shown by the red icon next to the rule name. An enabled rule has a green icon.

Enable this rule by assigning it to an available Analyzer or Analyzers. You do this the same way you assign a Transaction Tracing rule to Analyzers as described in "How to Assign a Rule to the Appropriate Analyzers" on page 531.

8 Results

The event is correlated into the same transaction. If the **Show this event flow in the Topology reports** check box is selected, the events will display in the topologies as linked nodes.

How to Create Modification Rules

This task describes how to create event modification rules.

This task includes the following steps:

- "Create a modification rule" on page 608
- "Create a new modifier and choose a destination property" on page 608
- "Select the Action to be taken by the rule" on page 608
- "Add the Source" on page 609
- "Add the XPath Regular Expression to match against the event - optional" on page 609
- "Test the Regular Expression - optional" on page 611
- "Create a match condition or conditions" on page 611
- "Assign the rule to the appropriate Analyzer or Analyzers" on page 611
- "Results" on page 611

1 Create a modification rule

a Select **Admin** > **Transaction Management** > **Monitoring**.

b Click the **Event Customization Rules** button in the bottom-left pane.

c Click the New icon and select **New Event Modification Rule**.

d Enter the name of the rule definition in the right pane and click **OK**.

2 Create a new modifier and choose a destination property

a Click the New icon and select **New Modifier** to open the **New Modifier** dialog.

b Add an event property to be the **Destination Property** in one of the following ways:

- Enter the modifier destination property XPath name in the **Destination Property** text box.
- Search or scroll for the destination property in the standard Available Properties tree, select the property and click **OK**.
- If your event data contains custom fields, open the top **Filter Properties by Selected Events** collapsible area and click in the window to load the transaction events or select the **Load transaction events** icon.

  Select the appropriate criteria for the type of transaction or transactions with events you want to use to find the destination XPath ID. You do this in the same way you search for the XPath for a match condition. For instructions, see "How to Create a Match Condition" on page 525.

3 Select the Action to be taken by the rule

Click in the **Action** field and select where the data is extracted from:

- **will be set to User Data**. When selected, a **User Data** link appears in the **Source** field to add the Source property, described in the next step.

- **will be set to constant.** When selected, you enter a fixed value in the **Source** field.
- **will be set to event property.** When selected, the **Edit Modifier Source** dialog opens to add the Source property, described in the next step. For user interface details, see "Edit Modifier Source Dialog" on page 631.

4 Add the Source

The source also specifies where the data is being extracted from. Do one of the following depending on the action you selected:

- If you selected **will be set by User Data**, a **User Data** link appears in the **Source** field.Clicking the **User Data** link opens the **Edit Modifier Source** dialog to add the regular expression, described in the next step.
- If you selected **will be set to event property**, the **Edit Modifier Source** dialog opens to add the Source Property. Add the Source Property in the same way you added the Destination Property in step 2 above.
- If you selected **will be set to constant**, sets the source data to be a fixed value.

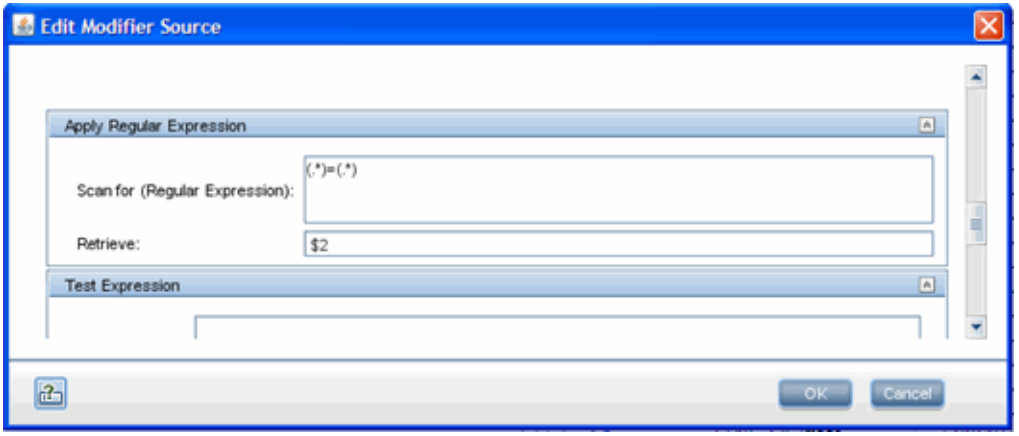
5 Add the XPath Regular Expression to match against the event - optional

If you just want the value of the source field, you do not need a regular expression. But, if you want to extract some part of your source field, then you need to specify the regular expression to pull the data out.

You can add regular expressions for the **will be set by User Data** and **will be set to event property** actions.

To add a regular expression:

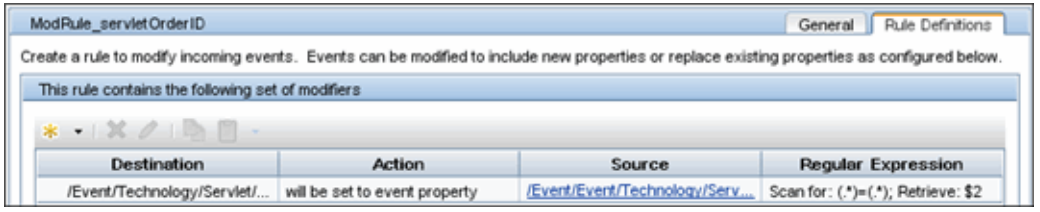
a Click the appropriate source link, or click in the Regular Expression field to open the **Edit Modifier Source** dialog as shown in the example below. For user interface details, see "Edit Modifier Source Dialog" on page 631.



b In the **Apply Regular Expression** area, add the regular expression in the **Regular Expression** text box.

In the **Retrieve** field, enter the value to extract from the matched regular expression.

c Click **OK** and the new modifier is created and the appropriate fields are populated as shown in the following example:



For examples of regular expressions, see the *HP TransactionVision Advanced Customization Guide* PDF.

6 Test the Regular Expression - optional

You can test the results of your regular expression to make sure it will work as expected. For instructions, see "How to Test a Modification Rule's Regular Expression" on page 612.

7 Create a match condition or conditions

It is recommended that you create a match condition or match conditions to limit the rules to be only for events that meet the match criteria. Otherwise, the modification rule is triggered for every event. For instructions, see "How to Create a Match Condition" on page 525.

8 Assign the rule to the appropriate Analyzer or Analyzers

After creating the correlation rule, it is disabled by default shown by the red icon next to the rule name. An enabled rule has a green icon.

Enable this rule by assigning it to an available Analyzer or Analyzers. You do this the same way you assign a Transaction Tracing rule to Analyzers as described in "How to Assign a Rule to the Appropriate Analyzers" on page 531.

9 Results

Data is saved into the specified destination according to what you configured to be extracted.

How to Test a Modification Rule's Regular Expression

This task describes how to test the regular expression associated with user data or event property actions for a modification rule. The test shows the behavior of the regular expression when run against the data and with the supplied regular expression rule. If the result is not as desired, you need to correct your supplied regular expression.

1 Create a modification rule.

For instructions, see "How to Create Modification Rules" on page 607.

The **Action** must be set to either **will be set to User Data** or **will be set to event property**.

2 Test the regular expression.

a Click the regular expression link to open the **Edit Modifier Source** dialog box.

b Open the **Test Expression** collapsible area.

c If your action is for user data, enter the user data in the User Data text box or open the **Select event to populate User Data** collapsible area to find an event associated with a transaction where you can see the user data.

d If your action is for an event property, enter the data you want to test your expression on in the **Property Value** text box.

e Click the **Test** button to test the results of your regular expression to make sure it worked as expected.

How to Use Event Modification With Custom Correlation - Work Flow

This work flow provides an example of using an event modification rule in conjunction with a custom correlation tracking ID rule.

In this example, you have an event that has within its user payload a lot of data. Part of this data is an ID that is unique and is required for some custom correlation.

This task includes the following steps:

- "Create a modification rule" on page 613
- "Create a new modifier and select a destination property" on page 613
- "For the Action, select the will be set to event property option" on page 614
- "Add a regular expression" on page 614
- "Test the regular expression" on page 614
- "Create a match condition or conditions" on page 616
- "Create a custom tracking ID correlation rule" on page 616
- "Results" on page 616

1 Create a modification rule

For instructions, see "How to Create Modification Rules" on page 607.

For example: **ModRule_servletOrderID**

2 Create a new modifier and select a destination property

This is the location to which the data will be extracted.

For example: **/Event/Technology/Servlet/OrderId**

3 For the Action, select the will be set to event property option

This indicates the source of data will be looking at the payload data of your chosen event.

This action opens the **Edit Modifier Source** dialog box so that you can select the Source Property to appear in the Source field.

For example: **/Event/Event/Technology/Servlet/Request/Parameters/Parameter[@name='account']/@value**

4 Add a regular expression

a Click the source property link in the Source field to open the **Edit Modifier Source** dialog.

This dialog allows you to edit the extraction rule for pulling out the user data. To help in developing the regular expression to extract your data you can find the exact event you want to work on and run a test against it.

b Open the panel that lets you browse for transactions and/or events, and use it to find the event that you want to work on.

On selecting the event, its user payload is shown in the testing window.

c By examining the payload, develop a regular expression to extract the data representing the correlation token.

For example: **(.*)=(.*)**

In this example, you have a servlet parameter in the form of **orderId=1234ABC**.

You write a regular expression of **(.*)=(.*)** and extract **$2** to get **1234ABCD**. You can then test it as described in the next step to be sure the result is **1234ABCD**.

5 Test the regular expression

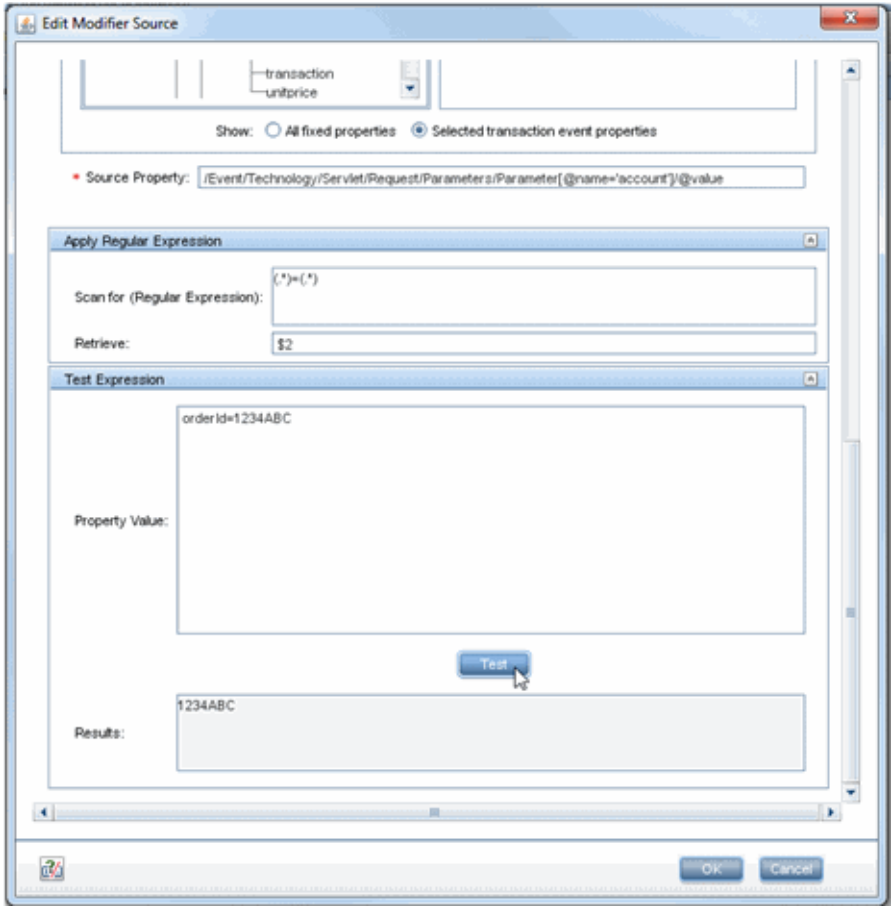
a Click the regular expression link to open the **Edit Modifier Source** dialog box.

b Open the **Test Expression** collapsible area.

c Enter the data you want to test your expression on in the **Property Value** text box.

d Click the **Test** button to test the results of your regular expression to make sure it worked as expected.

The following graphic shows the **Edit Modifier Source** dialog box with the tested regular expression and defined fields in the steps above.



6 Create a match condition or conditions

It is recommended that you create at least one match condition. For instructions, see "How to Create a Match Condition" on page 525.

For example: If **/Event/StdHeader/ProgramName** is **equal** to **TradeServlet**.

7 Create a custom tracking ID correlation rule

a In the Event Customization Rules left pane, click the New icon, select **New Correlation Rule**, enter the name of the rule definition in the right pane and click **OK**.

b In the **id to correlate by** field in the **Rule Definitions** pane, enter the destination you chose in step 2.

For example: **/Event/Technology/Servlet/OrderId**

c It is recommended that you create at least one match condition.

For example: If **/Event/StdHeader/ProgramName** is **equal to TradeServlet**.

d Enable this rule by assigning it to an available Analyzer or Analyzers. You do this the same way you assign a Transaction Tracing rule to Analyzers as described in "How to Assign a Rule to the Appropriate Analyzers" on page 531.

8 Results

You now have a tracking ID based correlation rule that correlates together all events that contain the same ID as was extracted from your payload from the modification rule you created.

Reference

Custom Correlation User Interface

This section describes:

- Event Customization Rules Page on page 618
- Event Customization Rules, General Tab on page 620
- Event Customization Rules, Rule Definitions Tab on page 621
- Correlation Rule, Rule Definitions Tab on page 622
- Edit XPath Value Dialog on page 626
- Modification Rule, Rule Definitions Tab on page 629

Event Customization Rules Page

The Event Customization Rules page enables you to define and assign custom behavior or event correlation.

To access	**1** Select **Admin** > **Transaction Management**. **2** Click the **Monitoring** tab in the left pane. **3** Click the **Event Customization Rules** button in the bottom-left pane.
Important information	The Event Customization Rules left pane includes folders for two types of rules: ➤ Correlation Rules. Lists custom event-to-event and tracking ID correlation rules. By default, the following correlation rules are provided: ➤ .NETMSMQRule ➤ .NETRemotingRule ➤ .NETRule ➤ .NETWCFRule ➤ Modification Rules. Lists rules that modify incoming events, such as extracting a value from payload data to be inserted in the event.
Relevant tasks	"How to Create Modification Rules" on page 607 "How to Use Event Modification With Custom Correlation - Work Flow" on page 613 "How to Copy a Rule to Create a New One" on page 532 "How to Group/Ungroup Rules" on page 532
See also	"Custom Correlation Overview" on page 590 "About Custom Event-to-Event Correlation" on page 591 "About Custom Tracking ID Correlation" on page 593 "About Modification Rules" on page 594 "Managing Custom Correlation and Modification Rules" on page 597

User interface elements are described below:

UI Element	Description
	New menu with the following commands: ➤ **New Correlation Rule**. For instructions, see "How to Create Custom Tracking ID Rules" on page 599 or "How to Create Event-to-Event Correlation Rules" on page 604. ➤ **New Event Modification Rule**. For instructions, see "How to Create Modification Rules" on page 607.
	Delete the selected item. Deletes the selected unassigned rule. Disabled by default until an assigned rule is selected.
	Refresh. Reloads all the rules.
	Group Rules Assigned Analyzer. Group/ungroup toggle to group/ungroup rules that are assigned to the same Analyzer or Analyzers.
	Export all Event Customization Rule Definitions as XML to a file. For details, see "Import/Export Correlation and Modification Rules" on page 598.
	Import Event Customization Rule Definition XML from a file. For details, see "Import/Export Correlation and Modification Rules" on page 598.

This page includes the following tabs:

➤ "Event Customization Rules, General Tab" on page 620

➤ "Event Customization Rules, Rule Definitions Tab" on page 621

➤ "Modification Rule, Rule Definitions Tab" on page 629

Event Customization Rules, General Tab

This tab enables you to assign a rule to an Analyzer.

To access	**1** Select **Admin** > **Transaction Management**. **2** Click the **Monitoring** tab in the left pane. **3** Click the **Event Customization Rules** button in the bottom-left pane. **4** Navigate to the Correlation or Modification rule for which you want to view or set properties. **5** Click the **General** tab in the right pane (selected by default).
Relevant tasks	➤ "How to Create Modification Rules" on page 607 ➤ "How to Assign a Rule to the Appropriate Analyzers" on page 531
See also	"Custom Correlation Overview" on page 590

User interface elements are described below:

UI Element	Description
	Move a selected Available Analyzer to the Assigned Analyzer column.
	Move a selected Assigned Analyzer back to the Available Analyzer column.
	Move all Available Analyzers to the Assigned Analyzer column.
	Move all Assigned Analyzers back to the Available Analyzer column.
Available Analyzers	The list of Analyzers in the deployment environment for which you have permission to manage. To add Analyzers, see "How to Add an Analyzer to the Deployment Environment" on page 106.

UI Element	Description
Assigned Analyzers	The Analyzers to which the rule is assigned. This Analyzer applies the rule continuously to the events it has collected to produce business transactions of the type.
Rule Definition Name	The name of the selected rule.

Event Customization Rules, Rule Definitions Tab

This page enables you to view or set custom correlation rules based on a correlation ID.

To access	**1** Select **Admin** > **Transaction Management**. **2** Click the **Monitoring** tab in the left pane. **3** Click the **Event Customization Rules** button in the bottom-left pane. **4** Navigate to the rule for which you want to view or set properties. **5** Click the **Rule Definition** tab in the right pane.
Important information	The Rule Definitions page change depending on which rule is selected: Correlation or Modification. You use this page to configure one of these types of rules.

Relevant tasks	"How to Create Custom Tracking ID Rules" on page 599 "How to Create Event-to-Event Correlation Rules" on page 604 "How to Create Modification Rules" on page 607 "How to Use Event Modification With Custom Correlation - Work Flow" on page 613 "How to Create a Match Condition" on page 525
See also	"About Custom Event-to-Event Correlation" on page 591 "About Custom Tracking ID Correlation" on page 593 "About Modification Rules" on page 594 "Managing Custom Correlation and Modification Rules" on page 597

For descriptions of the Rule Definitions tab, for each rule, see:

- "Correlation Rule, Rule Definitions Tab" on page 622
- "Modification Rule, Rule Definitions Tab" on page 629

Correlation Rule, Rule Definitions Tab

The Correlation rule, Rule Definitions tab allows you to create custom event-to-event or custom tracking ID correlation rules. For descriptions of each type of rule, see "About Custom Event-to-Event Correlation" on page 591 and "About Custom Tracking ID Correlation" on page 593.

User interface elements are described below:

UI Element	Description
Event 1	
Id to correlation by	Enter the property or XPath ID of the first event property to correlate by.

UI Element	Description
	Edit event property used as id to correlate by. If you entered the property ID and need the XPath ID, click this button next to the **Id to correlate by** text box to open the "Edit XPath Value Dialog" on page 626.
	Advanced Options. Opens the Advanced Options dialog box. For user interface details, see "Advanced Options Dialog Box" on page 625.
	New Match Condition. Opens the Create Match Condition dialog box. For user interface details, see "Create Match Condition Dialog" on page 564. For instructions on how to use this dialog box, see "How to Create a Match Condition" on page 525.
	Delete. Deletes the selected match condition.
	Edit selected item. Opens the Edit Match Condition dialog box, which is similar to the "Create Match Condition Dialog" on page 564.
	Copy match condition(s). Copies the selected match condition or conditions.
	Paste match condition(s). Pastes the copied match condition or conditions.
Property	Match condition property name. Displays when a match condition has been created. Right-click on the property name and select **Show complete condition path** to see the full XPath name.
Operator	Match condition operator. Displays when a match condition has been created. Click on the operator to open the list of operators and select a different one.
Directional Correlation: Create a relationship between two specific events.	Unselected by default. ➤ Select to create an event-to-event correlation rule. When selected, the options are enabled underneath. See "How to Create Event-to-Event Correlation Rules" on page 604. ➤ Unselect to create a tracking ID correlation rule. See "How to Create Custom Tracking ID Rules" on page 599.

UI Element	Description
Flow Direction: Event1 <to/from> Event 2	Disabled by default. Becomes enabled when the **Directional Correlation** check box is selected to create an event-to-event correlation rule. The direction of the arrow in the message flow indicates whether a message flow starts or ends from the first event. Arrows pointing right indicate a message flow start. Arrows pointing left indicate a message flow end.
Type	Disabled by default. Becomes enabled when the **Directional Correlation** check box is selected to create an event-to-event correlation rule. Choose one of the following flow types: ➤ **Message Flow**. Indicates a direct message flow between the two events. This means the two events are associated with the same message data. For example, a MQPUT and MQGET call dealing with the same message bears a message path relation. ➤ **Control Flow**. Indicates a control flow between the two events. The second event corresponds to the destination event in the control flow. For example, a servlet calling some other web service.
Show this event flow in the Topology reports	Disabled by default. Becomes enabled when the **Directional Correlation** check box is selected to create an event-to-event correlation rule. When selected, the two events are displayed in all the Topology reports as linked nodes.
Event 2	
<Toolbar elements>	Same toolbar elements as the ones in Event 1 except the **Advanced Options** button and all options underneath the match condition table.
<Table elements>	Same table columns as in Event 1 when a match condition is created.

Advanced Options Dialog Box

The Advanced Options dialog box allows you to specify how the standard tracking ID is to be used.

User interface elements are described below:

UI Element	Description
Add to existing Id(s) <default>	Selected by default, allows you to add other tracking IDs to the event specified in the correlation tracking ID rule.
Replace default Tracking Id	Replaces the standard tracking ID with a custom tracking ID you specify. For instructions, see "How to Replace a Standard Tracking ID With a Tracking ID Rule" on page 601.
Clear default Tracking Id	Clears the default standard tracking ID so that the event is no longer correlated to other events matching that tracking ID. For instructions, see "How to Clear a Standard Tracking ID Rule" on page 603.
Disable local transaction correlation	Disables all local transaction analysis. Allows you to disable the default behavior of how the Analyzer correlates events from the TransactionVision agents.

Edit XPath Value Dialog

This dialog enables you to select a tracking ID for a correlation rule. When you edit the tracking ID, this same dialog displays but is called the Edit XPath Value dialog.

To access	For a rule associated with a Business Transaction CI: **1** Select **Admin** > **Transaction Management** > **Administration** > **Monitoring.** **2** Click the **Event Customization Rules** button in the bottom-left pane. **3** Navigate to a particular Correlation rule. **4** Click the Edit icon in the Rules Definition pane.

User interface elements are described below:

UI Element	Description
Filter Properties by Selected Events Collapsible area in which you can select an existing transaction's events as candidates for property names and values in the Names panel.	
	Load events. Do one of the following: ➤ **Load transaction events**. Opens the **Select Transaction Events** dialog in which you specify criteria for retrieving a set of events. For descriptions of the fields, see "Select Transaction Events Dialog" on page 566. ➤ **Load query events**. Retrieves a set of events depending on the query. The dropdown lists all queries defined in the system.

UI Element	Description
	Show transaction events properties and values. When this button is selected, the **Name** and **Values** panes do not show names and values for a single selected transaction event. Instead, the panes display the union of names and values for all events in the transaction. For example, if you have a transaction with one MQSeries and one CICS event, selecting a single event (such as MQSeries) only displays MQSeries under the **Technology** entry in the **Names** pane. And fields such as **StdHeader ExitTime** list the time for this specific event under **Values**. But if you select **Show transaction events properties and values**, the **Technology** entry lists both **MQSeries** and **CICS** and the **ExitTime** displays the time values for both events.
Available Properties	
Names	Shows all available XPath properties. If you selected a transaction and/or event, all XPaths that occur within that event transaction display. For information on populating this tree with a selected transaction's event properties, see "Select Transaction Events Dialog" on page 628.
Values	The value of the selected event property. Only displays when the **Selected transaction event properties** option is enabled and selected.
All fixed properties	Displays all default XPath properties that can occur in Transaction Vision events.

UI Element	Description
Correlate By Id Property	The XPath name of the selected property to correlate the event by. When you click **OK**, this XPath appears in the **Id to correlation by** field in the Rules Definitions tab.
Selected transaction event properties	Displays the XPaths relevant to the selected transaction event as well as the standard fixed XPaths. Selecting this option also shows the value of each custom XPath so that you can choose the value directly. This option is disabled by default. It is enabled when a transaction is selected.

Select Transaction Events Dialog

For user interface details, see "Select Transaction Events Dialog" on page 566.

Modification Rule, Rule Definitions Tab

The Modification rule Rule Definitions tab allows you to create a rule to modify incoming events and configure the properties of the events.

User interface elements are described below:

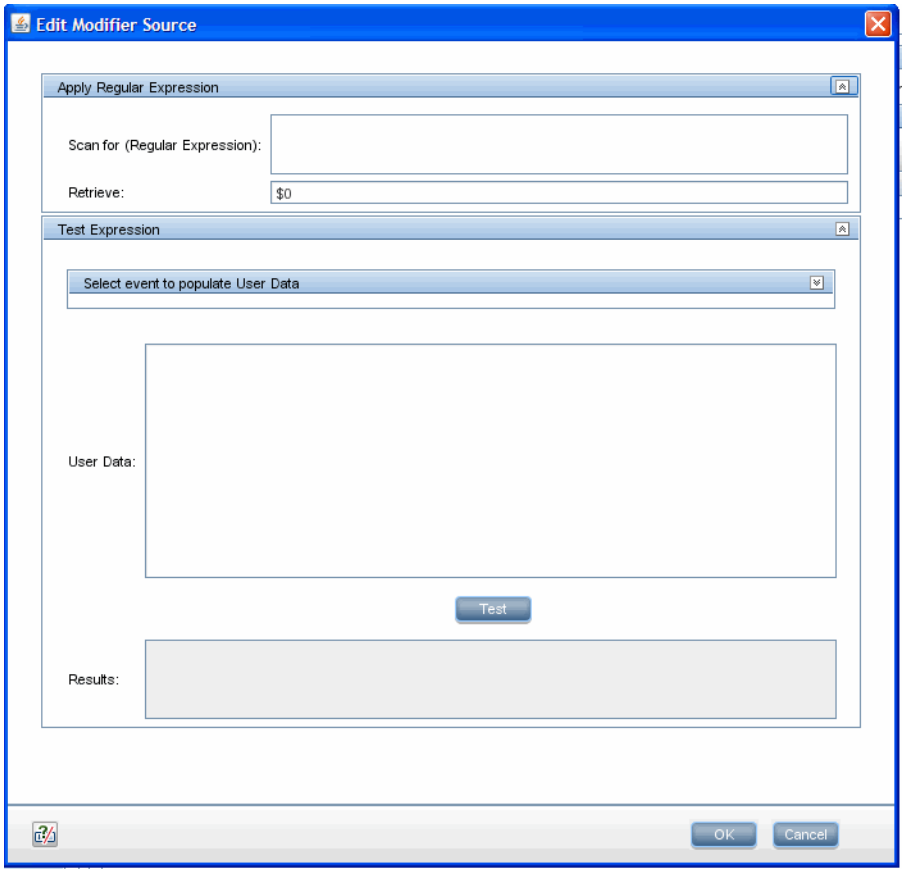
UI Element	Description
	New menu with the following commands: ➤ **New Modifier.** Opens the **New Modifier** dialog box, in which you select a source destination property from the standard events or from events associated with a selected transaction. You use this dialog in a similar way you use the "Create Match Condition Dialog" on page 564. The selected property appears in the **Destination** column. ➤ **New Match Condition.** Opens the **Create Match Condition** dialog box. For a description of the user interface, see "Create Match Condition Dialog" on page 564. For instructions on how to use this dialog box, see "How to Create a Match Condition" on page 525.
	Delete. Deletes the selected match condition.
	Edit selected item. Opens the **Edit Match Condition** dialog box, which is similar to the "Create Match Condition Dialog" on page 564.
	Copy match condition(s). Copies the selected match condition or conditions.
	Paste match condition(s). Pastes the copied match condition or conditions.
Destination	Modifier destination property XPath ID.

UI Element	Description
Action	Specifies where the data is extracted from. Select one of the following actions: ➤ **will set to User Data**. When selected, a **UserData** link appears in the Source column. ➤ **will be set to constant**. Default action. Set to a fixed value. ➤ **will be set to event property**. When selected, opens the **Edit Modifier Source** dialog box to change the modifier in the Destination column. For user interface details, see "Edit Modifier Source Dialog" on page 631.
Source	Specifies where the data is being extracted from: ➤ **User Data**. The **UserData** link displays in the Source field if the action is **will set to User Data**. Click the User Data link to open the **Edit Modifier Source** dialog box, to define a regular expression to apply to the event's User Data. ➤ <Constant value>. Enter the value if the action is **will be set to constant**. ➤ <Event property>. If the action is **will be set to event property**, the **Edit Modifier Source** dialog box opens to specify the Source Property, which displays in the Source field.
Regular Expression	(Optional) Regular expression to pull the data out of your source field. You can add regular expressions for the **will be set by User Data** and **will be set to event property** actions. Double-click on the action to open the **Edit Modifier Source** dialog box. Enter the regular expression information and click **OK** to display it in the Regular Expression field. You can test this regular expression by clicking the regular expression link. For instructions see "How to Test a Modification Rule's Regular Expression" on page 612.

Edit Modifier Source Dialog

The Edit Modifier Source dialog allows you to add a regular expression, which extracts some part of the source field.

Following is an example of the Edit Modifier Source dialog:



User interface elements are described below:

UI Element	Description
Filter Properties by Selected Events	
<elements>	Same as in "Edit XPath Value Dialog" on page 626.

UI Element	Description
Available Properties	
<elements>	Same as in "Edit XPath Value Dialog" on page 626.
Source Property	XPath name of the source ID to specify from where the data is being extracted.
Apply Regular Expression	
Regular Expression	Regular expression to pull the data out of your source field.
Retrieve	Value to extract from the matched regular expression.
Test Expression	
Select event to populate User Data	
<toolbar elements>	Displays if **will be set by User Data** is the action. Collapsible area to find an event associated with a transaction where you can see the user data. When you select the user data, it appears in the User Data field.
	Test. Click to test the regular expression. The results appear in the Results window.
User Data	User data on which you want to test the regular expression.
Property Value	Data on which you want to test the regular expression.
Results	Result of the tested regular expression.

Troubleshooting and Limitations

Use the **CorrelationUtil** utility to troubleshoot and identify the cause of correlation issues. For syntax and how to use this utility, see "CorrelationUtil" on page 373.

Part III

Reports and Topologies

19

Introducing Transaction Management Reports and Topologies

This chapter includes:

Concepts

- ➤ Reports and Topologies Overview on page 636
- ➤ How the Reports and Topologies Are Related on page 637
- ➤ Drilldown Workflows on page 637
- ➤ Key Information Reported For a Transaction on page 641
- ➤ Reports and Topologies Workflows on page 645

Concepts

Reports and Topologies Overview

The Transaction Management reports and topologies present the information that has been collected from the TransactionVision agents and processed by the TransactionVision Analyzers.

Some reports and topologies can also include HP Diagnostics or Real User Monitor (RUM) information related to the business transaction.

Topologies are graphical flow maps of a transaction that show the transaction execution from frontend real user experience through the back-end server level components (such as application servers, message middleware servers, database servers, mainframe transaction and batch jobs).

Transaction Management provides the following reports and topologies, grouped by tabbed categories in the Transaction Management application:

Category Tab	Reports/Topologies
Transaction Summary	➤ "Transaction Summary Report" on page 652 ➤ "Web Session Report" on page 664
Transaction Analysis	➤ "Transaction Over Time Report" on page 666 ➤ "Transaction Tracking Report" on page 675 ➤ "Transaction Detail Report" on page 686 ➤ "Event Analysis Report" on page 691
Transaction Topology	➤ "Aggregated Topology Page" on page 740 ➤ "Component Topology" on page 755
Transaction Infrastructure	➤ "Application Server Statistics Report" on page 708 ➤ "WebSphere MQ and JMS Statistics Report" on page 712

How the Reports and Topologies Are Related

The Transaction Management reports are designed to first present the highest level view of transaction information and offer drilldowns to more detailed reports for further analysis. For drilldown details, see the following section, "Drilldown Workflows" on page 637.

Drilldown Workflows

This section shows the drilldown workflows within Transaction Management and between Transaction Management and other specific applications in the following topics:

- "Drilling Down To Transaction Management Reports and Topologies" on page 637
- "Drilling Down To/From Specific Applications" on page 638

Drilling Down To Transaction Management Reports and Topologies

The following table provides the drilldown workflows from high-level Transaction Management reports and topologies to more detailed reports and topologies.

From: Transaction Management Report/ Topology	To Transaction Management Report/Topology:
Transaction Summary	➤ Aggregated Topology ➤ Transaction Over Time ➤ Transaction Tracking
Transaction Tracking	➤ Transaction Detail ➤ Instance Topology
Transaction Scatter Graph (part of the Transaction Tracking report)	➤ Transaction Detail ➤ Instance Topology ➤ Component Topology

From: Transaction Management Report/ Topology	To Transaction Management Report/Topology:
Transaction Detail (drilldown from the Transaction Tracking report)	➤ Event Analysis ➤ Aggregated Topology ➤ Component Topology
Transaction Over Time	➤ Aggregated Topology ➤ Transaction Tracking
Event Analysis	➤ Event Details ➤ Component Topology ➤ Transaction Detail (only when events are viewed as Local Transaction or Business Transaction)
Aggregated Topology	➤ Transaction Tracking

Drilling Down To/From Specific Applications

The following tables provide the drilldown workflows from Transaction Management reports and topologies to specific applications, and to Transaction Management reports and topologies from specific applications.

Drilldowns from Transaction Management:

From: Transaction Management Report/ Topology	To Application:
Transaction Tracking report	End User Management's RUM End User Group Summary report
Transaction Detail (part of the Transaction Tracking report)	Diagnostics Server Requests view

From: Transaction Management Report/ Topology	To Application:
Event Analysis report	➤ Diagnostics Hosts view, which provides system metrics like CPU utilization and memory utilization that can help you determine if the problem is a system issue. ➤ Diagnostics Server Requests view, which can help you study the performance of these server requests over time and look at detailed instance trees (call profiles, methods, exceptions and faults) associated with the server requests, to identify where the issue lies. From the Server Request data you can navigate to the corresponding probe entity and look at the application server metrics (for example, JMX metrics) to see if the issue lies with the application server.
Transaction Summary report	➤ End User Management's RUM Transaction Summary report ➤ Diagnostics Business Transactions Path view
Aggregated Topology	Diagnostics

Drilldowns to Transaction Management:

From Application:	To Transaction Management Report/Topology:
BPM Performance Matrix End User Management's RUM Session Analyzer End User Management's RUM Session Detail	Transaction Tracking report

Additional Information Provided by Drill-Down to HP Diagnostics

- You can drill down from the Transaction Summary report to the Diagnostics Business Transactions Path view.
- You can drill down from the Transaction Detail (part of the Transaction Tracking report) to the Diagnostics Server Requests view.
- You can drill down from the Event Analysis report to Diagnostics to get the following types of information:
  - The drill down to a Diagnostics Hosts view provides system metrics like CPU utilization and memory utilization that can help you determine if the problem is a system issue.
  - The drill down to a Diagnostics Server Requests view can help you study the performance of these server requests over time and look at detailed instance trees (call profiles, methods, exceptions and faults) associated with the server requests, to identify where the issue lies.

    From the Server Request data you can navigate to the corresponding probe entity and look at the application server metrics (for example, JMX metrics) to see if the issue lies with the application server.

Additional Information Provided by Drill-Down to EUM

- You can drill down from the Transaction Summary report to the EUM Transaction Summary report.
- You can drill down from the Transaction Tracking report to the EUM End User Summary report.

Key Information Reported For a Transaction

As TransactionVision collects and analyzes transactions, it gathers statistics. related to transactions, events, and infrastructure elements. These statistics as well as other data appears on the various reports and topologies.

The frequently-appearing information related to transactions is as follows:

- "Transaction Name" on page 641
- "Transaction Count (or Volume)" on page 641
- "Response Time" on page 642
- "Transaction States" on page 643
- "Threshold" on page 644
- "Value" on page 644

Transaction Name

The transaction name is the business transaction name that is specified when the business transaction is defined. See "How to Create a Business Transaction CI" on page 525.

Transaction Count (or Volume)

The transaction count is the number of completed instances of a particular business transaction in the specified time frame.

The transaction count is also referred to as transaction volume in some reports.

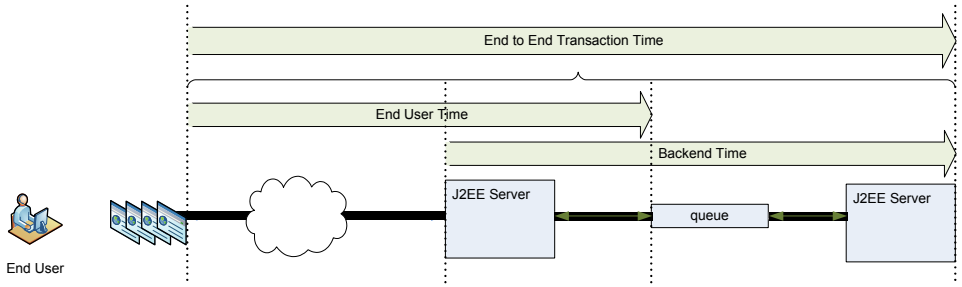
You can see this information on the Transaction Summary, Web Session, or Transaction Over Time report.

Response Time

TransactionVision provides the transaction response time of a transaction instance. Only completed transaction instances can provide a transaction response time.

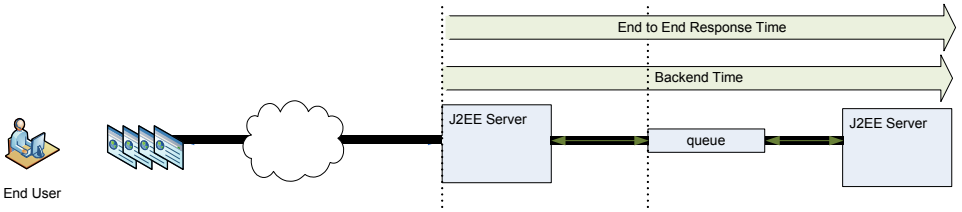
To calculate the response time, TransactionVision subtracts the start time of the first event in the transaction from the last event in the transaction (after correlation has occurred). The response time is reported in units of milliseconds or seconds; the column heading in the report indicates the unit.

If an application is monitored by RUM and TransactionVision, the end user response time is available. The end user response time is combined with the back end response time (reported by TransactionVision and HP Diagnostics, if present) to produce the total end-to-end response time:



There can be overlap between end user response time and back end response time because all products (RUM, TransactionVision, HP Diagnostics) measure the time it takes the J2EE Server to complete the request. The overlap also includes the time it takes to reach the J2EE Server from the end user;, backend time additionally includes the time spent processing any parts of the transaction that occur asynchronously to the request.

If RUM is not monitoring the application then the transaction end-to-end response time does not include the end user time:



Note: One important way the response time is used is to flag whether a transaction has exceeded its response time threshold This causes the transaction to be flagged as late and be seen as such in reports and System Health. For information about specifying target response times, see "Transaction Configuration Page, Tracing Tab" on page 550.

You can see the response times on the Transaction Summary, Transaction Over Time, Transaction Tracking, and Event Analysis reports. The end-to-end transaction time is sometimes referred to as the transaction response time.

Transaction States

Each transaction instance has the following state information:

- **Completion State**. Indicates whether the transaction completed. The completion is determined by classification rules. See "Transaction Tracing Rules Page" on page 560.
- **Result State**. For completed transactions, indicates whether the transaction met the criteria for a successful completion. See the Transaction Result success and failure rules on "Transaction Tracing Rules Page" on page 560.
- **SLA State**. For completed transactions, indicates whether the transaction met the Threshold criteria. For information about the Threshold, see "Transaction Configuration Page, Tracing Tab" on page 550.

Transaction states appear on the Transaction Tracking report.

Threshold

The number of milliseconds representing the target end-to-end response time of the transaction. The Threshold is specified when the business transaction CI is configured. See "Transaction Configuration Page, Tracing Tab" on page 550.

You can see the Threshold on the Transaction Detail area of the Transaction Tracing report.

Value

The monetary value of the transaction. The Value is specified as part of a data rule when the business transaction CI is configured. See "Transaction Tracing Rules Page" on page 560.

The Value is affected by the currency code. Currency code is a 3-letter code as defined by the IS0 4217 standard. Some common codes are:

USD
EUR
ILS
HKD
INR

The currency code is specified as a property of the transaction when the business transaction CI is configured. See "Transaction Tracing Rules Page" on page 560.

You can see the Value on the Web Session report, Transaction Over Time report, and Transaction Tracking report.

Reports and Topologies Workflows

There are many workflows supported by the Transaction Management reports and topologies. Some of the key workflows are as follows:

Show me the status of my transaction

Go to the Transaction Tracking Report, and use the Transaction Filter to choose the Business Transaction by its name. Run the report and view the Completion state column to see whether the transaction is completed.

Drill to the Transaction Detail for more information.

Show me all transaction instances with specific criteria

Go to the Transaction Tracking report, access the Transaction Filter Metrics tab to specify the Business Transaction attributes you want to base the search on. For example, those that have violated their threshold or those that are still in-process.

Show me the high and low runs of my transaction

Go to the Transaction Tracking Report, and use the Transaction Filter to choose the Business Transaction by its name. Run the report and view the scatter graph.

Hover over the highest point in the graph and view the response time metrics. Repeat for the lowest point.

Show me which application infrastructure elements my transaction is flowing across

Run the Aggregated Topology report.

How does the performance of one transaction instance compare to the average performance?

Go to the Transaction Tracking Report, and use the Transaction Filter to choose the Business Transaction by its name. Run the report and drill to the Instance topology.

Enable the labels for each performance metric you want to see. Instance and aggregate performance metrics are shown.

What is the performance of the different application infrastructure elements for transaction instance X?

Go to the Transaction Tracking Report, and use the Transaction Filter to choose the Business Transaction by its name. Run the report and drill to the Component topology.

These reports display information for servers, message queue latency, infrastructure performance, SQL statement statistics and WebSphere MQ and JMS components statistics. See "Component Topology" on page 755 for details.

20

Transaction Management Reports

This chapter includes:

Concepts

➤ Reports Overview on page 648

Tasks

➤ How to View Metrics Aggregated by Transaction Type on page 649
➤ How to View Metrics Aggregated by Web Session on page 649
➤ How to View Metrics for a Particular Transaction Type on page 649
➤ How to View Metrics at the Event Level on page 650
➤ How to View Metrics Aggregated by Application Server on page 650

Reference

➤ Reports User Interface on page 651

Troubleshooting and Limitations on page 715

Concepts

Reports Overview

Each Transaction Management report is designed for different levels of transaction or event analysis:

If you want to:	Access this report:
Perform aggregated transaction level analysis; aggregated by transaction CI type.	Transaction Summary
Perform aggregated transaction level analysis; aggregated by web session.	Web Session
View graphs of aggregated transaction level information; aggregated by transaction CI type.	Transaction Over Time
Perform individual transaction level analysis.	Transaction Tracking
Perform event-level analysis.	Event Analysis
Perform aggregated event level analysis; aggregated by application server components.	Application Server Statistics
Perform Queue and Queue Manager level analysis; aggregated by WebSphere MQ and JMS infrastructure activity.	WebSphere MQ and JMS Statistics

Drilling Down To/From Transaction Management Reports and Topologies

You can drill down from Transaction Management reports and topologies to related reports and topologies for further analysis. You can also drill down to specific applications, such as Diagnostics and End User Management. And, from these applications you can drill down to specific Transaction Management reports and topologies.

For tables that show the drilldown details, see "Drilldown Workflows" on page 637.

Tasks

How to View Metrics Aggregated by Transaction Type

For All Transactions

To view Business Transaction metrics aggregated by each Business Transaction CI type, access the Transaction Summary report.

For information about this report, see "Transaction Summary Report" on page 652.

For a Specific Transaction Type

To view time slices of Business Transaction metrics aggregated by a single Business Transaction CI type, access the Transaction Over Time report.

For information about this report, see "Transaction Over Time Report" on page 666.

How to View Metrics Aggregated by Web Session

To view Business Transaction metrics aggregated by Web Session, access the Web Session report.

For information about this report, see "Web Session Report" on page 664.

How to View Metrics for a Particular Transaction Type

To view Business Transaction metrics for a particular instance of a Business Transaction type, access the Transaction Tracking report.

For information about this report, see "Transaction Tracking Report" on page 675.

How to View Metrics at the Event Level

To view event data that has been collected, access the Event Analysis report.

For information about this report, see "Event Analysis Report" on page 691.

How to View Metrics Aggregated by Application Server

To view business transaction metrics aggregated by application server, access the Application Server Statistics report.

For information about this report, see "Application Server Statistics Report" on page 708.

Reference

Reports User Interface

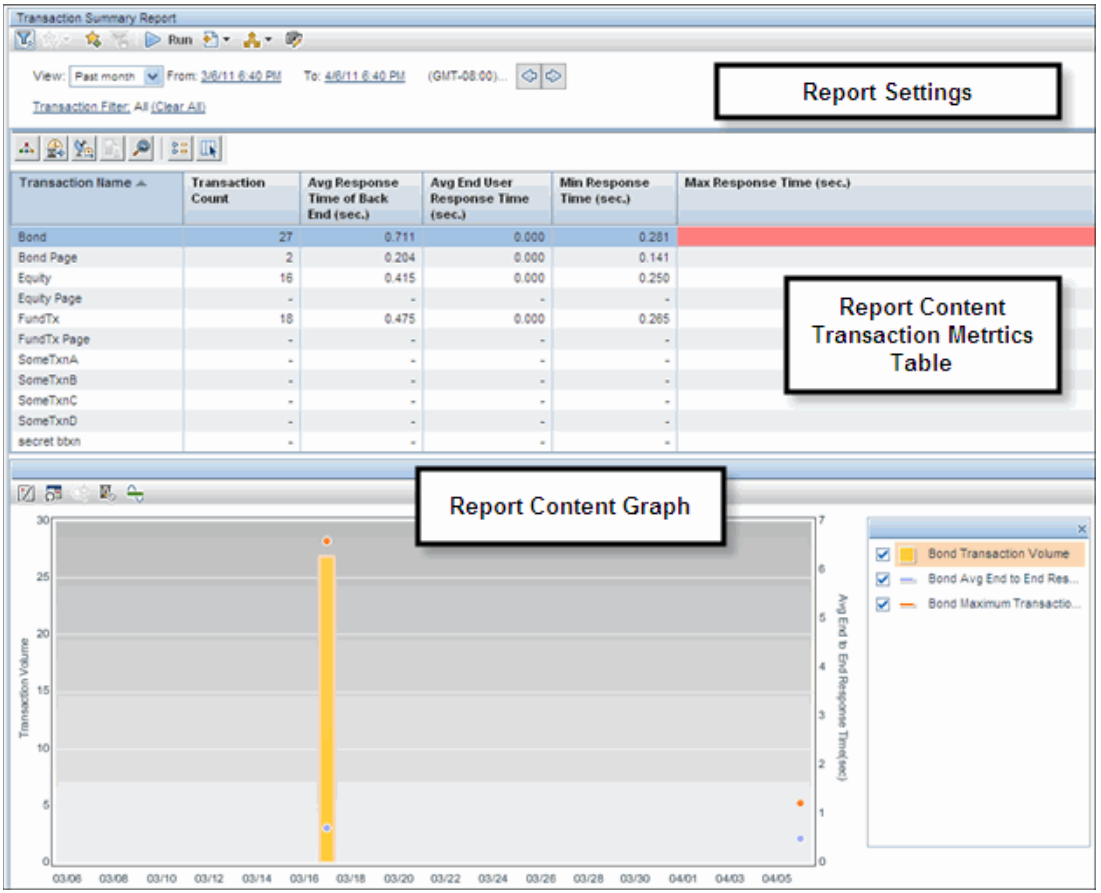
This section describes:

- Transaction Summary Report on page 652
- Transaction Filter Dialog Box on page 657
- Web Session Report on page 664
- Transaction Over Time Report on page 666
- Transaction Tracking Report on page 675
- Transaction Detail Report on page 686
- Event Analysis Report on page 691
- Application Server Statistics Report on page 708
- WebSphere MQ and JMS Statistics Report on page 712

Transaction Summary Report

This report displays volume and response time metrics for transactions and a graph of these metrics for a selected transaction. Metrics are aggregated by Business Transaction CI type.

The following is an example of the Transaction Summary report.



To access	Select **Applications** > **Transaction Management** > **Transaction Summary** > **Transaction Summary.** This report is also accessed from RUM reports.
Important information	➤ Transactions that have completed in the specified **View** time range are displayed in alphabetical order by business transaction name. You can change the sort order and the column on which the transactions are sorted by clicking the column heading in the Transaction Metrics table. ➤ You can limit which transactions appear by using the Transaction Filter. ➤ This report is based on combined data from all products monitoring or tracing the transaction: TransactionVision, HP Diagnostics, or Real User Monitor.
Relevant tasks	"How to View Metrics Aggregated by Transaction Type" on page 649
See also	"Key Information Reported For a Transaction" on page 641

Report Settings

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
<Common report settings>	For details, see "Common Report Elements" in *Reports*.
View	The reporting time period. Choose an option from the drop-down list. For some time periods, also choose: **From**. The date and/or time for the beginning of the reporting time period; click to access a calendar from which to select the date/time. **To**. The date and/or time for the ending of the reporting time period.

UI Element	Description
Transaction Filter	Launches the Transaction Filter dialog box, where you can select the business transactions to include in the report. See "Transaction Filter Dialog Box" on page 657. **Default:** All business transactions in the specified View range are selected.
(Clear All)	Clears any active filters. All business transactions in the reporting time period are selected.

Report Content: Transaction Metrics Table

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
	Drill to the Aggregated Topology for the selected business transaction.
	Drill to the Transaction Over Time report for the selected business transaction.
	Drill to the Transaction Tracking report for the selected business transaction. Only enabled if the transaction is being traced by TransactionVision.
	Drill to the RUM Transaction Summary report. Only enabled if the transaction is being monitored by RUM.
	Drill to the HP Diagnostics Business Transaction Path view. Only enabled if the transaction is being monitored by HP Diagnostics.
	Reset column widths.
	Select columns to show in the report. For available columns, see the section below, "Report Settings: Transaction Summary Columns Available for Display" on page 655.

Report Settings: Transaction Summary Columns Available for Display

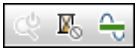
The available columns are described below:

UI Element	Description
Transaction Name	Name of the transaction. Select a transaction name to determine what is shown in the graphs below.
Transaction Count	Sum of transaction instances that have completed in the selected time frame.
Avg End to End Response Time (sec.)	Average end-to-end response time of transaction instances that have completed in the selected time frame. This values is a sum of the backend and end user response times. See "Key Information Reported For a Transaction" on page 641. The cell is colored red if the threshold is exceeded, otherwise, the default color scheme is used.
Avg Response Time of Back End (sec.)	Average response time of the transaction over monitored application components, such as application servers, message middleware servers, database servers, mainframe transactions and batch jobs.
Avg End User Response Time (sec.)	Average response time of end user as measured by RUM. If RUM is not monitoring the application, this value is 0 or blank.
Min Response Time (sec.)	Minimum end-to-end transaction response time of transaction instances that have completed in the selected time frame.
Max Response Time (sec.)	Maximum end-to-end transaction response time of transaction instances that have completed in the selected time frame. The cell is colored red if the threshold is exceeded, otherwise, the default color scheme is used.

UI Element	Description
Total Value	Represents the summed monetary value of the classified transaction class in whatever currency is specified in the Transaction class configuration. You can configure the currency in the Transaction Configuration Tracing tab described in "Transaction Configuration Page, Tracing Tab" on page 550.
% Exceptions	Percentage of transactions over the specified period that resulted in an exception condition.
% Late	Percentage of transactions over the specified time that were late according to SLA requirements.
% Failures	Percentage of transactions in the specified time period that failed.

Report Content: Volume and Response Time Graph

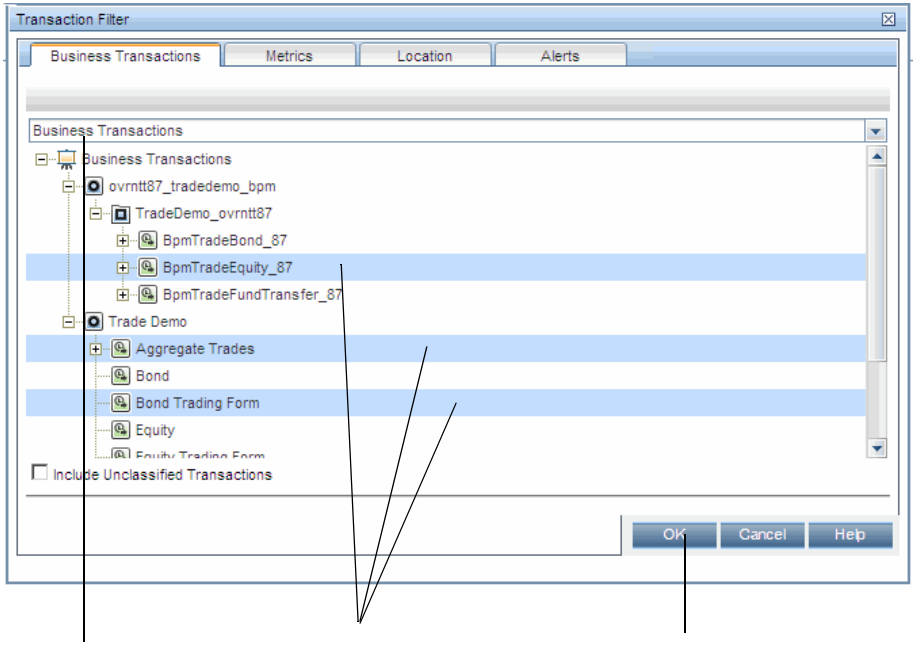
The bar graph indicates the transaction volume and the line graph indicates the average end-to-end response time for the business transaction selected in the Transactions Metrics table.

Note: The last three icons on the toolbar of the Report Content Graph are not supported for this report: 

Transaction Filter Dialog Box

This dialog box enables you to find and select a business transaction on which a a report or topology is based. The Transaction Filter dialog includes different tabs depending on where it was launched from. All tabs are described here.

The following is an example of the Transaction Filter dialog box and the three basic steps in using it.



1. Choose the CI Type of Business Transaction.

2. Control-click to select one or more business transactions, transaction flows, or applications.

3. Click OK.

To access	The **Transaction Filter** link on any of the following: ➤ Transaction Summary report ➤ Transaction Over Time report ➤ Transaction Tracking report ➤ Aggregated Topology
Important information	➤ The Metrics, Locations, and Alerts tabs are included on the Transaction Filter dialog only when it is launched from the Transaction Tracking report. ➤ Filtering business transactions is optional. If no filter is specified then all business transactions are used to generate the report. ➤ The Alerts tab allows you to filter transactions by alerts described in "Alerts Tab" on page 662.
See also	"Business Transactions in Applications or Transaction Flows" on page 508. The "Security" chapter in the *HP TransactionVision Deployment Guide* PDF.

This dialog can contain the following tabs depending on in which report this dialog is launched:

➤ "Browse Views Tab" on page 659

➤ "Search CIs Tab" on page 660

➤ "Metrics Tab" on page 661

➤ "Location Tab" on page 662

➤ "Alerts Tab" on page 662

Browse Views Tab

User interface elements are described below:

UI Element	Description
<business transaction tree>	Displays the configured applications, groups, transaction flows and transactions for which you have permission to view. Select one or more of these items that you want to include in the filter. **Note**: When you select an item in the hierarchical tree, its children are also selected.
Include Unclassified Transactions	Check to include unclassified instances of transactions. Unclassified transactions are transaction instances that TransactionVision has identified but not mapped to a business transaction CI. This option appears only when the filtering for the Transaction Tracking reports.
Exclude BPM-only Transactions	BPM generated transactions do not provide information that is relevant to the Transaction Management reports or topologies so by default these types of transactions are excluded. If you want to include them, clear the check box.

Search CIs Tab

User interface elements are described below:

UI Element	Description
Search CIs tab	Click the Search CIs tab to search for CI names that contain a specific string, or for CIs of a specific CI type. Search results are displayed in the lower part of the search tab. Right-click a CI in the search results table to open a Shortcut menu and click: **Select in tree**. Displays the hierarchical tree with the CI's check box selected. **Note**: The search is made on the hierarchical tree of relevant CIs displayed in the Browse Views tab. The search is not case sensitive. When searching for a CI name, you can use the asterisk (*) wildcard in your string to match one or more words of text.
	Click to start the search.
Type	Choose a CI type of **business_transaction**. Only business transactions are eligible for the reports and topologies. Click the Search CIs tab to search for CI names that contain a specific string, if desired. Search results are displayed in the lower part of the search tab. Right-click a CI in the search results table to open a Shortcut menu and click **Select in tree** to display the hierarchical tree with the CI's check box selected. **Note**: ➤ The search is not case sensitive. ➤ When searching for a CI name, you can use the asterisk (*) wildcard in your string to match one or more words of text.

UI Element	Description
Exclude BPM-only Transactions	Check to exclude BPM generated transactions. BPM generated transactions do not provide information that is included on the Transaction Management reports or topologies so typically you would not include these types of transactions.
Name	The search results. Choose the items you want to include.

Metrics Tab

The Metrics tab is included on the Transaction Filter dialog only when it is launched from the Transaction Tracking report.

User interface elements are described below:

UI Element	Description
Completion State	Check one or more of the following options to filter the report or topology by the Completion State of a transaction: ➤ **Unknown** ➤ **Processing** ➤ **Completed**
Result State:	Check one or more of the following options to filter the report or topology by the Result State of a transaction: ➤ **Unknown** ➤ **Success** ➤ **Failed**
SLA State:	Check one or more of the following options to filter the report or topology by the SLA State of a transaction: ➤ **None** ➤ **Violated** ➤ **Aged Out**

Location Tab

The Location tab is included on the Transaction Filter dialog only when it is launched from the Transaction Tracking report.

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
<location tree>	Displays locations of business transactions that are both both enabled for TransactionVision tracing data and EUM monitoring. Select the locations for transactions you want to include. The TransactionVision deployment environment must be set up for RUM data. See "How to Verify or Modify Communication Between BSM and TransactionVision" on page 364.

Alerts Tab

The Alerts tab is included on the Transaction Filter dialog only when it is launched from the Transaction Tracking report.

User interface elements are described below:

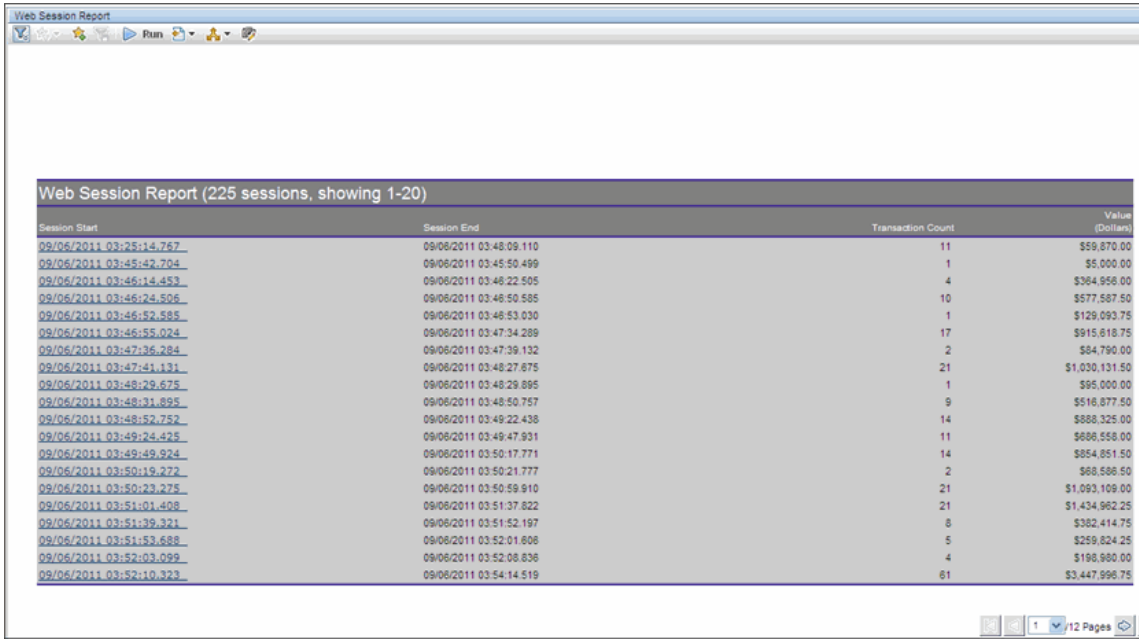
UI Element	Description
Show Only Transactions With Alerts	Click to limit the report to display only business transactions with alerts. When selected, the alert options below are enabled.
All Alerts	Displays all business transactions for which any alert is triggered.
Transaction Incomplete Alert	Displays all business transactions for which the **Incomplete Transaction Violation Alert** is triggered. For details, see "Tracing Tab, Alert Rules Tab" on page 557.

UI Element	Description
Transaction Threshold Alert	Displays all business transactions for which the **Response Threshold Violation Alert** is triggered. For details, see "Tracing Tab, Alert Rules Tab" on page 557.
<custom alert>	If one or more custom alerts were created, each one appears as an option in this alerts list. For details on creating custom alerts, see "How to Create Custom Alerts" on page 536. Displays all business transactions for which a custom alert is triggered.

Web Session Report

The Web Session report displays transaction volume and value for each completed Web session.

The following is an example of the Web Session report.



To access	Select **Applications** > **Transaction Management** > **Transaction Summary** > **Web Session.**
Important information	➤ Only agents that monitor servlets can collect Web session data. To generate this report the Analyzer must be configured to extract and store Session ID information. See "How to Enable Session Tracking" on page 336. ➤ You can drill down to the Transaction Tracking report. ➤ This report is based on data from TransactionVision only.
See also	"How to Enable Session Tracking" on page 336

Report Settings

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
<Common report elements>	For details, see "Common Report Elements" in *Reports*.
View	The reporting time period. Choose an option from the drop-down list. For some time periods, also choose: **From**. The date and/or time for the beginning of the reporting time period; click to access a calendar from which to select the date/time. **To**. The date and/or time for the ending of the reporting time period.
Sort by Attribute	Choose the field on which you want to sort from the drop-down list.

Report Content

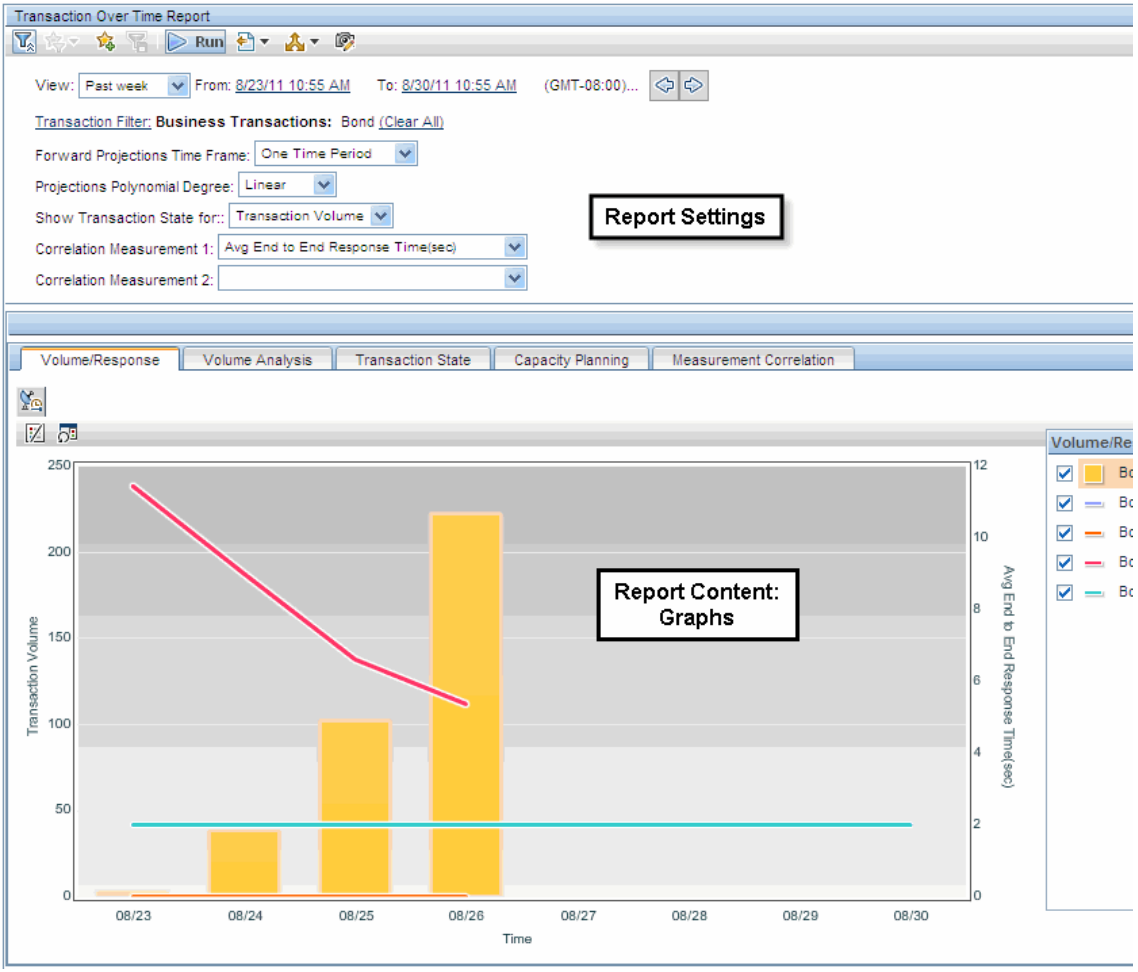
User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
Session Start	The start time for the first event within this session. Click on this link to display the Transaction Tracking report on the transactions within this session. See "Transaction Tracking Report" on page 675 for details.
Session End	The session end time.
Transaction Count	The total number of transactions within this session.
Value (Dollars)	The dollar value of all transactions within this session.

Transaction Over Time Report

Displays graphs of various combinations of the business transaction metrics, including forward projections of the metric values. Metrics are aggregated by transaction CI type.

The following is an example of the Transaction Over Time report.



To access	➤ Select **Applications** > **Transaction Management** > **Transaction Analysis** > **Transaction Over Time.** ➤ Drill-down from the Transaction Summary report.
Important information	➤ You can limit which transactions appear by using the Transaction Filter. ➤ This report is based on combined data from all products monitoring or tracing the transaction: TransactionVision, HP Diagnostics, or Real User Monitor. ➤ This report includes the following graphs: ➤ Volume/Response ➤ Volume Analysis ➤ Transaction State ➤ Capacity Planning ➤ Transaction Measurements Correlation ➤ You can drill down to the Transaction Tracking report from any of these graphs by right-clicking on a bar to view the timeframe and criteria of the corresponding transaction instance.
See also	"Key Information Reported For a Transaction" on page 641

Report Settings

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
<Common report settings>	For details, see "Common Report Elements" in *Reports*.
View	The reporting time period. Choose an option from the drop-down list. For some time periods, also choose: **From**. The date and/or time for the beginning of the reporting time period; click to access a calendar from which to select the date/time. **To**. The date and/or time for the ending of the reporting time period.
Transaction Filter	Launches the Transaction Filter dialog box, where you can select the business transactions to include in the report. See "Transaction Filter Dialog Box" on page 657. **Default:** The last business transactions that was specified is selected. If this is the initial viewing of the report, you are prompted to specify the business transaction.
(Clear All)	Clears any active filters. All business transactions in the reporting time period are selected. The reporting time period is determined by the View setting on this dialog.
Forward Projections Time Frame	The amount of time (relative to the selected time frame for the report) over which projections occur in the Capacity Planning graph. Options are: **One Time Period.** **Two Time Periods.** **Three Time Periods.** For example, if a user is viewing a one hour report and specifies Two Time Periods, then the graph spans 3 hours, 1 hour of data and 2 hours (1 hour times 2 time periods) of projections.

UI Element	Description
Projections Polynomial Degree	Alters the projection algorithm for the Capacity Planning graph. Changing the value modifies the way the projections are calculated and result in different projection values. Specify **Linear** to use the linear based algorithm; choose **Quadratic** to use the quadratic based algorithm.
Show Transaction State for	Options are: **Transaction Value.** The graphs display results based on the monetary value of each instance for the business transaction. **Transaction Volume.** The graphs display results based on the volume of instances for the business transaction.
Correlation Measurement 1 and Correlation Measurement 2	Choose one or two measurements to display in the Measurements Correlation graph: **Avg End User Response Time (sec.)** **Avg Response Time of Backed (sec.)** **End to End Response Time (sec)** **In Process Count** **In Process Total Value** **Max End User Response Time (sec)** **Max Response Time of Backend (sec)** **Maximum Transaction Response Time (sec)** **Min End User Response Time (sec)** **Min Response Time of Backend (sec)** **Minimum Transaction Response Time (sec)** **Total Value Exception** **Total Value Failed** **Total Value Late** **Transaction Volume** For example, you can compare **In Process Count** with **Maximum Transaction Response Time** to see if there is a relationship.

Report Content Common to All Tabs

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
	Drill to Transaction Tracking. Opens the Transaction Tracking report for the selected event.
	Show/Hide Legend. Displays or hides the graph's legend.
	Dock Legend. You can move the legend and then click **Dock Legend** to lock it in place until you move it again.

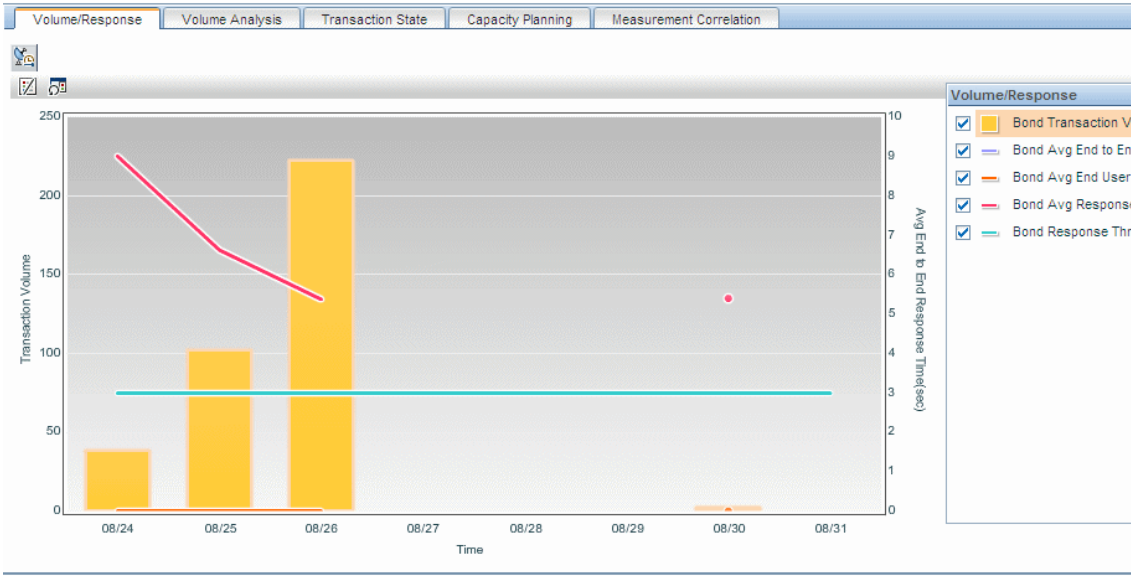
Report Content: Volume/Response Tab

The Volume/Response tab displays a graph of transaction volume and end-to-end response time for the selected transaction.

The horizontal line indicates the response time threshold of the transaction defined in **Admin** > **Monitoring tab** > **Transaction Configuration** > **Tracing tab** > **Tracing Properties** (see "Tracing Tab, Tracing Properties Tab" on page 551). You can quickly see (as in the graphic below) where the response times go over or under the response time threshold (the green horizontal line).

From a bar in this report you can right-click and drill down to the Transaction Tracking report to see more information about the transactions associated with that bar. For example, if you drill down from the Bond Failed Transactions bar only those failed transaction instances appear on the Transaction Tracking report.

The following is an example of the Volume/Response tab.



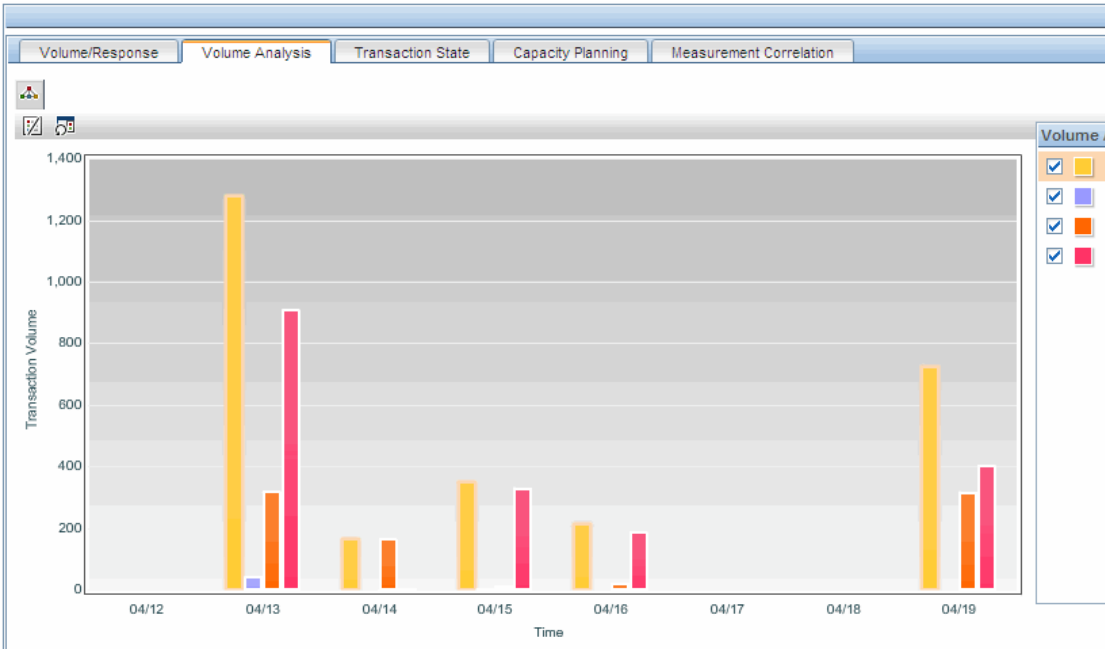
Report Content: Volume Analysis Tab

The Volume Analysis tab displays transaction volume and counts of successful, late, and failed transactions.

From a bar in this report you can drill down to the Transaction Tracking report to see more information about the transactions associated with that bar. For example, if you drill down from the Bond Failed Transactions bar only those failed transaction instances appear on the Transaction Tracking report.

To perform the drill down, right-click the bar and choose **Drill to Transaction Tracking Report**.

The following is an example of the Volume Analysis tab.



Report Content: Transaction State Tab

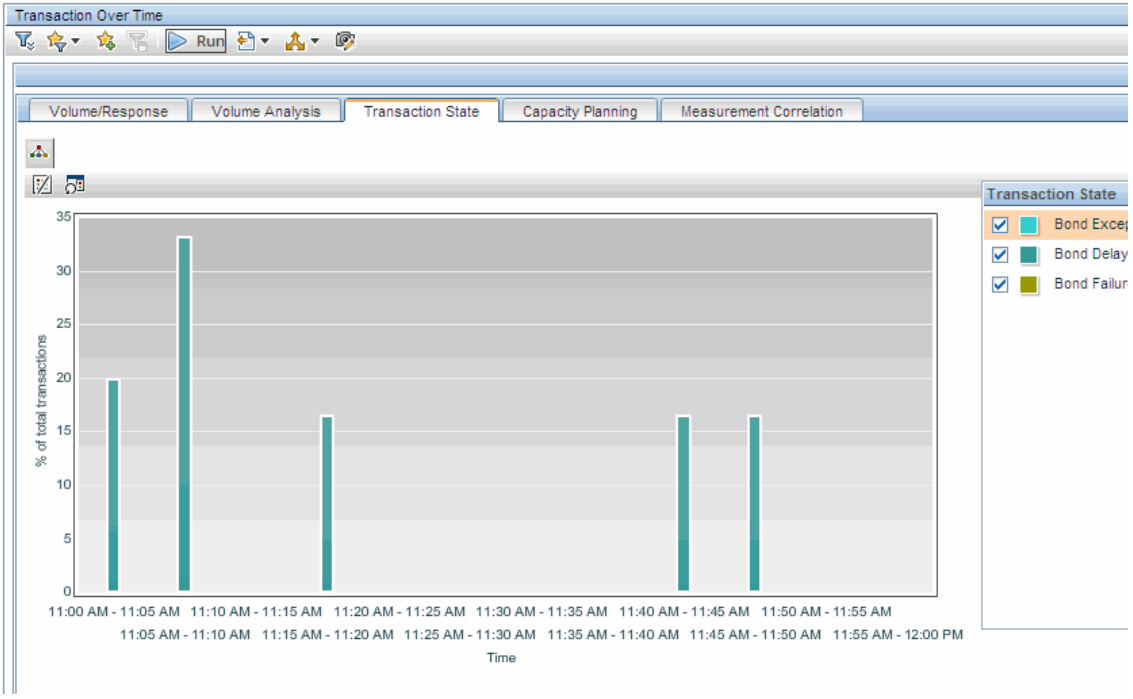
The Transaction State tab displays the time period on the x-axis, and percentage on the y-axis. A separate bar is shown for each transaction state during each time period included in the graph.

For information about the transaction states, see "The Transaction States Relevant to Service Health" on page 351.

From a bar in this report you can drill down to the Transaction Tracking report to see more information about the transactions associated with that bar. For example, if you drill down from the Bond Failed Transactions bar only those failed transaction instances appear on the Transaction Tracking report.

To perform the drill down, right-click the bar and choose **Drill to Transaction Tracking Report**.

The following is an example of the Transaction State tab.



Report Content: Capacity Planning Tab

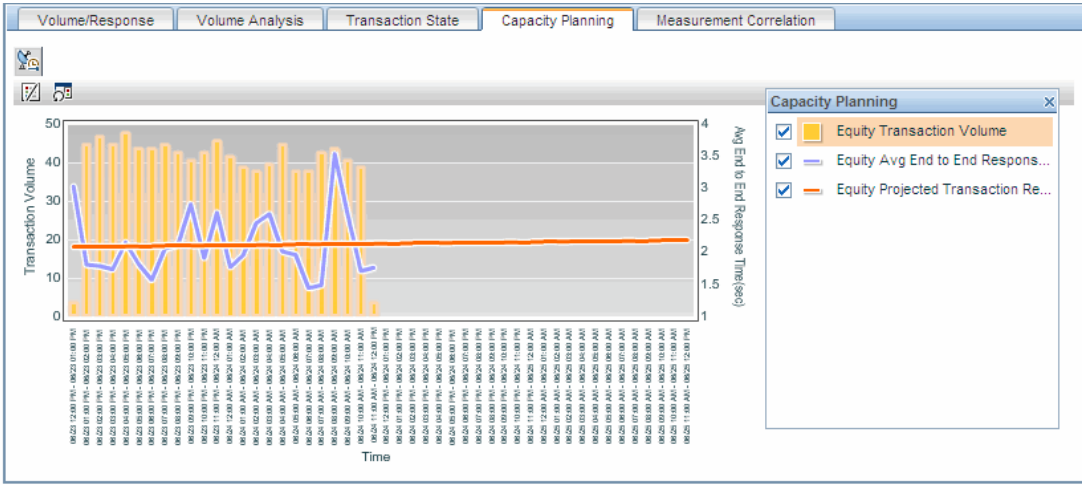
The Capacity Planning tab displays volume and response time metrics, and projects future response time levels based on the algorithm you specified in the report settings.

For example, the following graph shows a projection over two time periods. The flat line indicates that the response time is stable, or more importantly, not heading in the direction that the response time threshold will soon be exceeded. So at this time you do not need more capacity in your infrastructure to handle the current trends in performance.

From a bar in this report you can drill down to the Transaction Tracking report to see more information about the transactions associated with that bar. For example, if you drill down from the Bond Failed Transactions bar only those failed transaction instances appear on the Transaction Tracking report.

To perform the drill down, right-click the bar and choose **Drill to Transaction Tracking Report**.

The following is an example of the Capacity Planning tab.



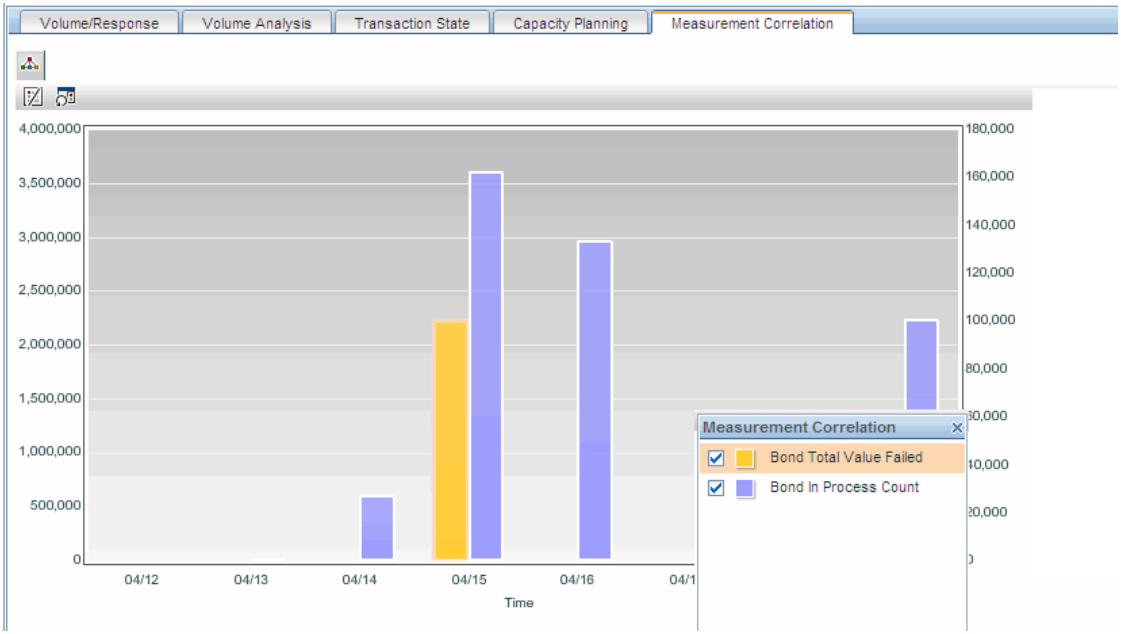
Report Content: Measurement Correlation Tab

The Measurement Correlation tab displays the time period on the x-axis, and the units for the selected measurements on the y-axes. Each point on the graph shows the total or average for the preceding time period.

From a bar in this report you can drill down to the Transaction Tracking report to see more information about the transactions associated with that bar. For example, if you drill down from the Bond Failed Transactions bar only those failed transaction instances appear on the Transaction Tracking report.

To perform the drill down, right-click the bar and choose **Drill to Transaction Tracking Report**.

The following is an example of the Measurement Correlation tab configured to show the Value of the Failed transactions and how many transactions are in-process.

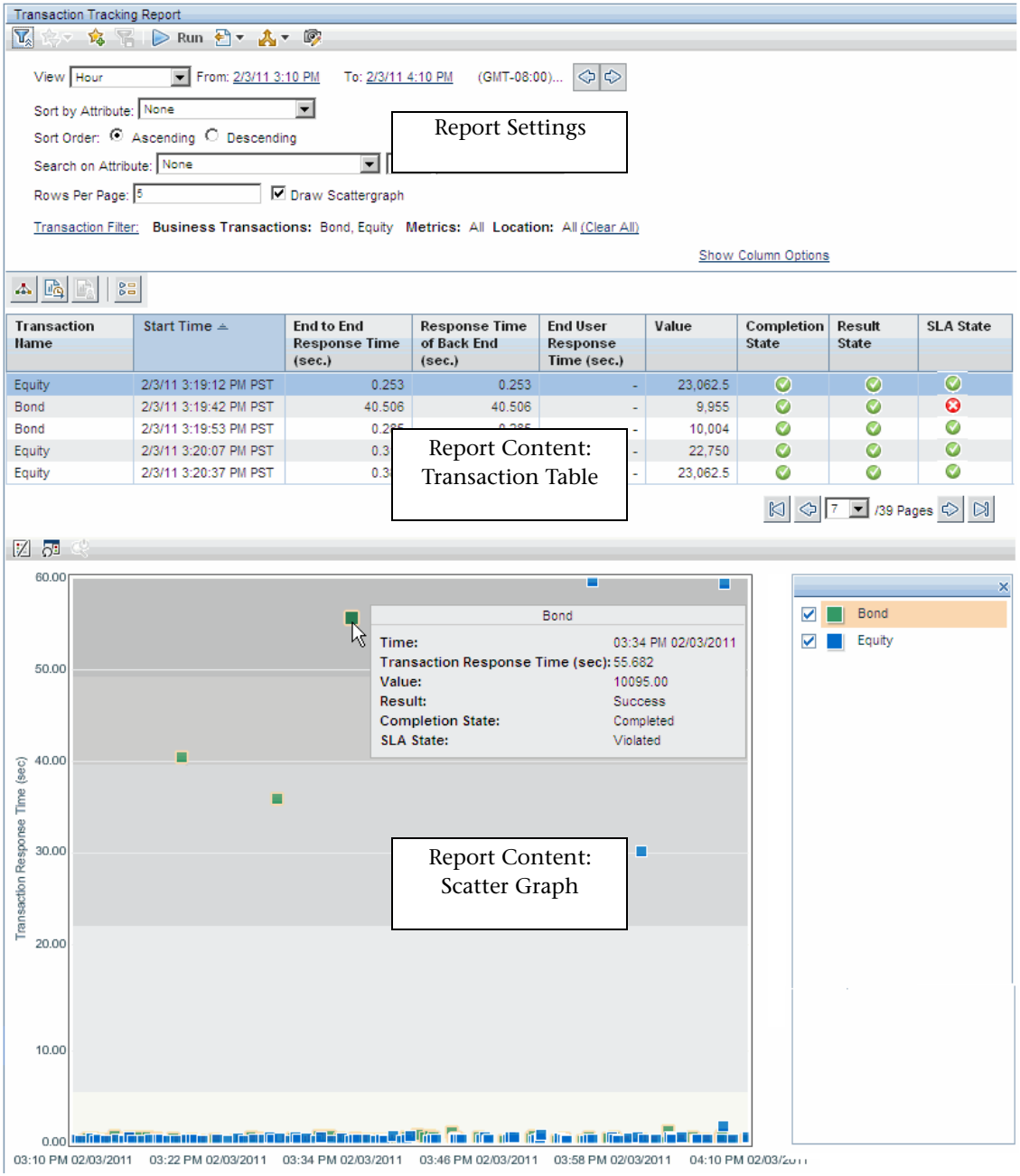


Transaction Tracking Report

The Transaction Tracking report displays transaction information for all transactions that were active in the specified time period. The report filtering criteria can limit the displayed results to only those transactions of a certain completion state, result state, SLA state, Alert state , or location (if the application is also monitored by RUM).

This report offers drill-down analysis for individual transactions from the Transaction Table or from the Scatter Graph (if enabled).

The following is an example of the Transaction Tracking report.



To access	➤ Select **Applications** > **Transaction Management** > **Transaction Analysis** > **Transaction Tracking Report**. ➤ Drill-down from the Transaction Summary report. ➤ Drill-down from the Event Details report. ➤ Drill-down from the Transaction Over Time report.
Important information	➤ This report can show unclassified transactions. These are transaction instances that TransactionVision tracks in the same way it tracks named business transactions but they have not been classified. You may want to view unclassified transactions to see what is being collected so that you can examine them and determine which filtering or classification rules are needed. ➤ Drill down from the Transaction Table to: ➤ The "Transaction Detail Report" on page 686 of a selected transaction, by clicking the **Drill to Transaction Detail** icon. ➤ The Instance Topology of a selected transaction by clicking on the **Drill to Instance Topology** icon. ➤ The RUM End User Summary report, if there is RUM data in a selected transaction, by clicking the **Drill to RUM End User Summary** report icon. ➤ Drill down from the Scatter Graph to: ➤ The Transaction Detail Report by right clicking on a point and selecting **Transaction Detail**. ➤ The Instance Topology by right-clicking on a point and selecting **Intance Topology**, see Chapter 21, "Transaction Topologies." ➤ The Component Topology by right-clicking on a point and selecting **Component Topology**, see Chapter 22, "Component Topology." ➤ This report is based on data from TransactionVision only.

Relevant Tasks	"How to View Metrics for a Particular Transaction Type" on page 649
See also	"Key Information Reported For a Transaction" on page 641

Report Settings

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
<Common report elements>	For details, see "Common Report Elements" in *Reports*.
Sort Order	Choose **Ascending** or **Descending**.
View	The reporting time period. Choose an option from the drop-down list. For some time periods, also choose: **From**. The date and/or time for the beginning of the reporting time period; click to access a calendar from which to select the date/time. **To**. The date and/or time for the ending of the reporting time period from a calendar control.
Sort by Attribute	Select an attribute to sort results by, if desired. Custom attribute columns are available if custom data definitions have been added. See "Business Transaction Data Model" on page 581. **Note**: If the report is set to sort on the **Class ID** attribute, results are sorted in the order in which the Transaction CI was defined.
Search for	Enter text for a specific search.
Rows per Page	The number of transactions that are displayed in the Transaction Table. **Default:** 25

UI Element	Description
Draw Scattergraph	Enables/disables the Scatter Graph portion of the report. Disabling the Scatter Graph allows the report to render more quickly. For more details, see "Report Content: Scatter Graph" on page 682. All transactions are represented in the Scatter Graph, regardless of how many transactions appear in the transaction table on the current page of the report. **Default:** On
Transaction Filter	Launches the Transaction Filter dialog box, where you can select the business transactions to include in the report. See "Transaction Filter Dialog Box" on page 657. **Default:** All business transactions in the specified View range are selected.
(Clear All)	Clears any active filters. All business transactions in the reporting time period are selected. The reporting time period is determined by the View setting on this dialog.
Show Column Options	Click to enable the column selection fields where you choose which columns appear on the report. For information about the columns, see "Report Content: Transaction Table" on page 680. Click **Hide Column Options** to hide the column selection fields.

The Transaction Tracking report consists of the following areas:

- "Report Content: Transaction Table" on page 680
- "Report Content: Scatter Graph" on page 682

Report Content: Transaction Table

You can choose which columns to display. The User interface elements are described below:

UI Element	Description
	Drill to Instance Topology.
	Drill to Transaction Detail.
	Drill to RUM End User Summary report. Enabled only when there is RUM data. **Default**: Disabled
	Reset column widths.
	Select columns to show in the report.
Transaction Name	Name of the transaction. Select a transaction name to determine what is shown in the table below.
Start Time	The start time of the transaction.
End to End Response Time (sec.)	End-to-end transaction response time of transaction instances that have completed in the selected time frame. This value is a sum of the backend and end user response times. See "Key Information Reported For a Transaction" on page 641.
Response Time of Back End (sec.)	Response time of the transaction over TransactionVision monitored application components, such as application servers, message middleware servers, database servers, mainframe transactions and batch jobs.
End User Response Time (sec.)	Response time of end user as measured by RUM. If RUM is not configured, this value is 0 or blank.
Value	The value of the transaction as determined by the Value property.

UI Element	Description
Completion State	The completion state of the transaction: Completed. Processing. Unknown.
Result State	Displays the result state of the transaction: Success. Unknown. Only transactions where the result state matches that selected in the report settings display in the report.
SLA State	The SLA state of the transaction: None. Violated.
End Time	End time of the transaction.
These columns can optionally be added to the report:	
End Time of Backend	End time of the backend portion of the transaction.
EUG Subnet	For transactions monitored be RUM, the end-user subnet ID.
Exception	Any exception that occurred for the transaction. This value appears only if the Exception rule is defined for the transaction. For information about setting the Exception rule, see "Tracing Tab, Tracing Properties Tab" on page 551.
Label	The label that was specified for the transaction, if any. The label is defined as a Data rule for the transaction. For information about setting the Label data rule, see "Tracing Tab, Data Rules Tab" on page 554.
Location	For transactions monitored be RUM, the location being monitored as reported by RUM.
Start Time of Backend	The start time of the backend portion of the transaction.

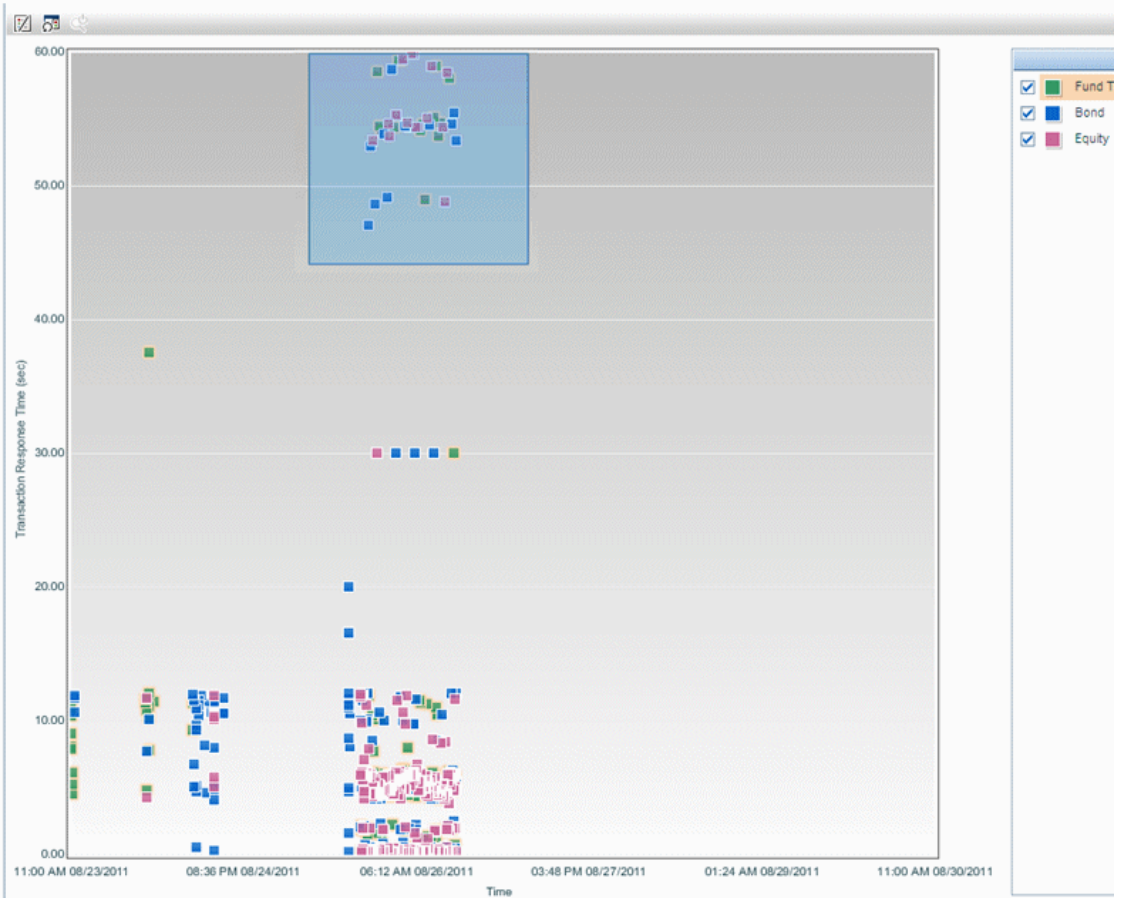
Report Content: Scatter Graph

All transactions in the Transaction Table are represented in the Scatter Graph, regardless of how many transactions appear on the current page of the Transaction Table.

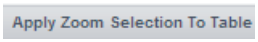
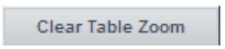
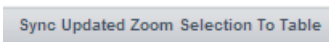
The Scatter Graph allows you to see at a glance which transactions (points on the graph) took longer to complete than other transactions. You can select a group of points to zoom in on the transactions you want to see, or right-click on a point to drill down to the Transaction Detail report, Instance toplogy or Component Topology.

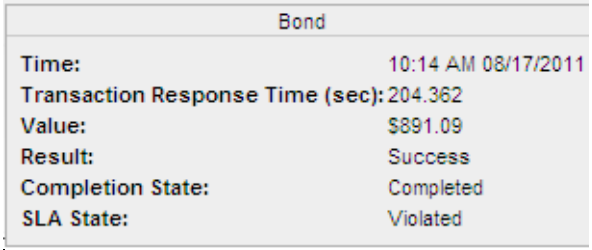
After using the zoom feature in the Scatter Graph, you can narrow the selection of transactions in the Transaction Table above by clicking the **Apply Zoom selections to Table** button. If additional zoom selections are made the table may be updated again using the **Sync Updated Zoom Selection To Table** button. These additional filter criteria may be cleared at any time by clicking the **Clear Table Zoom** button.

The following is an example of the Scatter Graph with a selected group of points.



User interface elements are described below (unlabeled elements are shown in angle brackets):

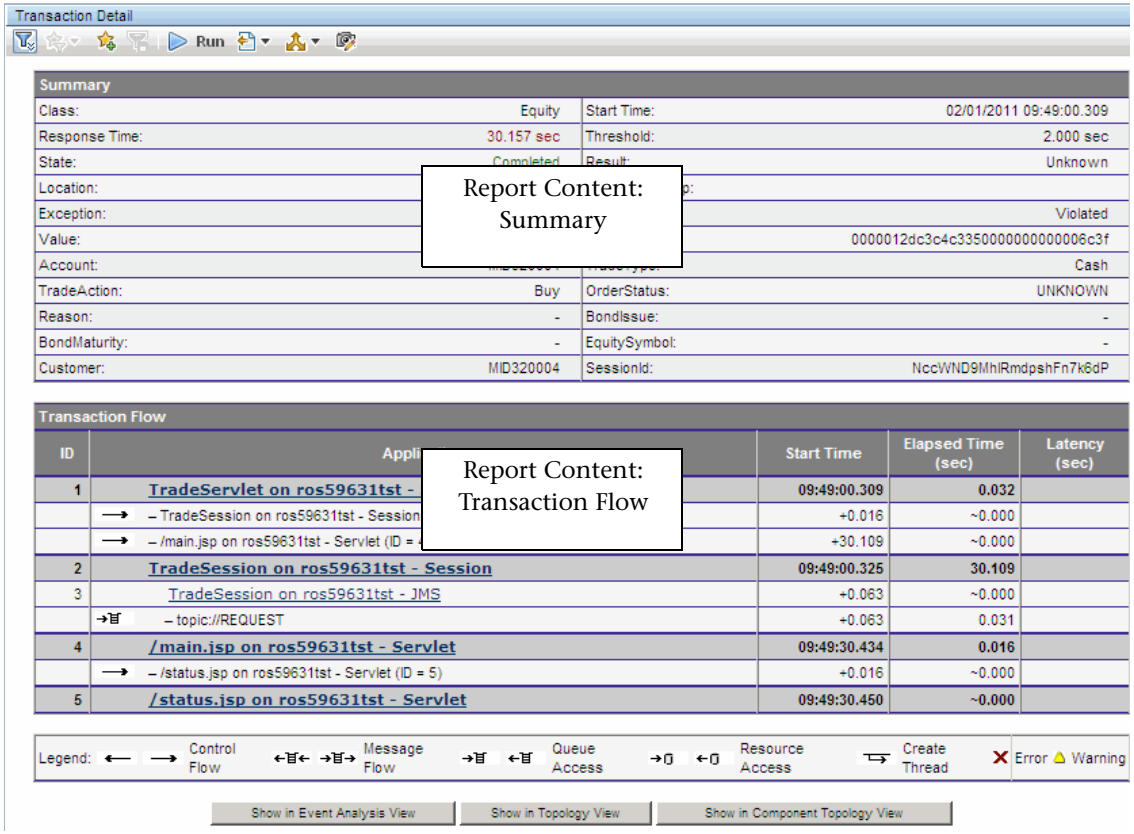
UI Element	Description
	Show/Hide Legend. Displays or hides the graph's legend.
	Dock Legend. You can move the legend and then click **Dock Legend** to lock it in place until you move it again.
	Reset Zoom in. Resets the zoom level of the graph to its original state, showing all transactions in the filtered period.
	**Apply Zoom Selection To Table**. Click after zooming in a group of transaction points to sync the Transaction Table with the zoomed in Scatter Graph.
	**Clear Table Zoom**. Click to return the zoomed in Transaction Table and Scatter Graph to show all transactions.
	**Sync Updated Zoom Selection To Table**. Replaces the **Apply Zoom Selection To Table** button after it is selected. Appears after applying the first zoom selection of points to the table, which syncs the table with the graph. so that the table reflects the same transaction instances. , and then zooming in fewer points. Click to sync the newly zoomed points with the Transaction Table.

UI Element	Description
<tooltip>	Hover over a point to see the metrics of the transaction instance (also reflected in the Transaction Table for that instance). For example: 
View menu	Right-click on a point in the Scatter Graph for the following drill down options: **Instance Topology**. See Chapter 21, "Transaction Topologies." **Component Topology**. See Chapter 22, "Component Topology." **Transaction Detail**. See "Transaction Detail Report" on page 686.

Transaction Detail Report

The Transaction Detail report displays summary and flow information for a selected business transaction from the Transaction Tracking report.

The following is an example of the Transaction Detail report.



To access	From the Transaction Tracking report, click the **Drill to Transaction Detail** button for a selected business transaction.
Important Information	From the Transaction Details report you can drill down to the following reports: ➤ "Event Analysis Report" on page 691 by clicking the **Show an Event Analyzsis View** button to see all local transactions with the business transaction. ➤ "Aggregated Topology Page" on page 740 by clicking the **Show in Topology View** button. ➤ "Component Topology" on page 755 by clicking the **Show in Component Topology View** button. In the Event Details, for a message flow, separate panes show a side-by-side comparison of the departure and arrival user data buffer. For a control flow, only the event details for the event that corresponds to the destination event in the control flow are shown. For user interface details of the Transaction Flow table, see "Report Content: Transaction Details Transaction Flow Table" on page 688.
See also	➤ "Key Information Reported For a Transaction" on page 641 ➤ "Transaction Tracking Report" on page 675 ➤ "Event Analysis Report" on page 691

Report Settings

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
<Common report elements>	For details, see “Common Report Elements” in *Reports*.
Show in Event Analysis View button	Display the current business transaction in the Event Analysis report. See "Event Analysis Report" on page 691.

UI Element	Description
Show in Topology View button	Display the current business transaction in the Aggregated Topology. See "Aggregated Topology Page" on page 740.
Show in Component Topology View button	Display the current business transaction in the Component Topology. See "Component Topology" on page 755.

Report Content: Transaction Details Summary Table

The Summary table displays information about the transaction: the name and several metrics including response time, status, start time, result, and the value for any custom data definitions for the transaction.

If there is diagnostics data for the specific transaction, the **Drill to Diagnostics** icon becomes available, which opens the Server Requests Snapshots Analysis view.

Report Content: Transaction Details Transaction Flow Table

The Transaction Flow table displays the local transaction ID, application name, start time, and processing time for each local transaction within the selected business transaction.

Note: Processing time for each program does not include the CICS TASK_START and TASK_END events, since these events do not contain a program name. Therefore, the processing time in this report differs slightly from the execution times in the Transaction Analysis view.

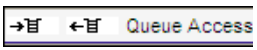
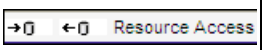
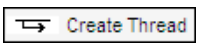
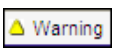
Following is an example of the Transaction Flow table:

Transaction Flow

ID	Application	Start Time	Elapsed Time (sec)	Latency (sec)
1	**TradeServlet on bsavm2 - Servlet**	11:07:35.795	0.188	
	→ – TradeSession on bsavm2 - Session (ID = 2)	+0.180	~0.000	
	→ – /main.jsp on bsavm2 - Servlet (ID = 19)	+11.281	~0.000	
2	**TradeSession on bsavm2 - Session**	11:07:35.975	11.274	
3	TradeSession on bsavm2 - JMS	+0.213	~0.000	
	→▮→ – topic://REQUEST Listener on bsavm2 - JMS (ID = 4)	+0.213	0.123	0.152
18	TradeSession on bsavm2 - JMS	+0.228	~0.000	
	←▮← – MDBReplyBroker on bsavm2 - JMS (ID = 17)	+0.228	10.775	0.067
4	**topic://REQUEST Listener on bsavm2 - JMS**	11:07:36.340	~0.000	
	←▮← – TradeSession on bsavm2 - JMS (ID = 3)	+0.000	~0.000	0.152
	→ – MDBRequestBroker on bsavm2 - Message Driven (ID = 5)	+4.790	~0.000	
5	**MDBRequestBroker on bsavm2 - Message Driven**	11:07:41.130	1.067	
6	MDBRequestBroker on bsavm2 - JMS	+1.003	~0.000	
	→▮→ – queue:///FX Listener on bsavm2 - JMS (ID = 7)	+1.003	0.031	0.102
7	**queue:///FX Listener on bsavm2 - JMS**	11:07:42.235	0.117	
	←▮← – MDBRequestBroker on bsavm2 - JMS (ID = 6)	+0.000	~0.000	0.102
	→ – FundTransferProcessor on bsavm2 - Message Driven (ID = 8)	+0.116	~0.000	
8	**FundTransferProcessor on bsavm2 - Message Driven**	11:07:42.351	0.952	
14	FundTransferProcessor on bsavm2 - JMS	+0.901	~0.000	
	→▮→ – topic://REPLY Listener on bsavm2 - JMS (ID = 15)	+0.901	0.050	0.050
9	**QualifyTrade on bsavm2 - Session**	11:07:42.463	0.007	
10	**Order on bsavm2 - Entity**	11:07:42.567	~0.000	
11	**Order on bsavm2 - Entity**	11:07:42.732	0.025	
12	**Order on bsavm2 - Entity**	11:07:43.061	0.010	
13	**Order on bsavm2 - Entity**	11:07:43.074	0.001	
15	**topic://REPLY Listener on bsavm2 - JMS**	11:07:43.302	~0.000	
	←▮← – FundTransferProcessor on bsavm2 - JMS (ID = 14)	+0.000	~0.000	0.050
	→ – MDBReplyBroker on bsavm2 - Message Driven (ID = 16)	+3.821	~0.000	
16	**MDBReplyBroker on bsavm2 - Message Driven**	11:07:47.123	0.025	
17	MDBReplyBroker on bsavm2 - JMS	+0.001	~0.000	
	→▮→ – TradeSession on bsavm2 - JMS (ID = 18)	+0.001	0.019	0.067
19	**/main.jsp on bsavm2 - Servlet**	11:07:47.256	0.003	
	→ – /status.jsp on bsavm2 - Servlet (ID = 20)	+0.001	~0.000	
20	**/status.jsp on bsavm2 - Servlet**	11:07:47.257	0.005	

Legend: ← → Control Flow ←▮← →▮→ Message Flow →▮ ←▮ Queue Access →0 ←0 Resource Access Create Thread ✕ Error ⚠ Warning

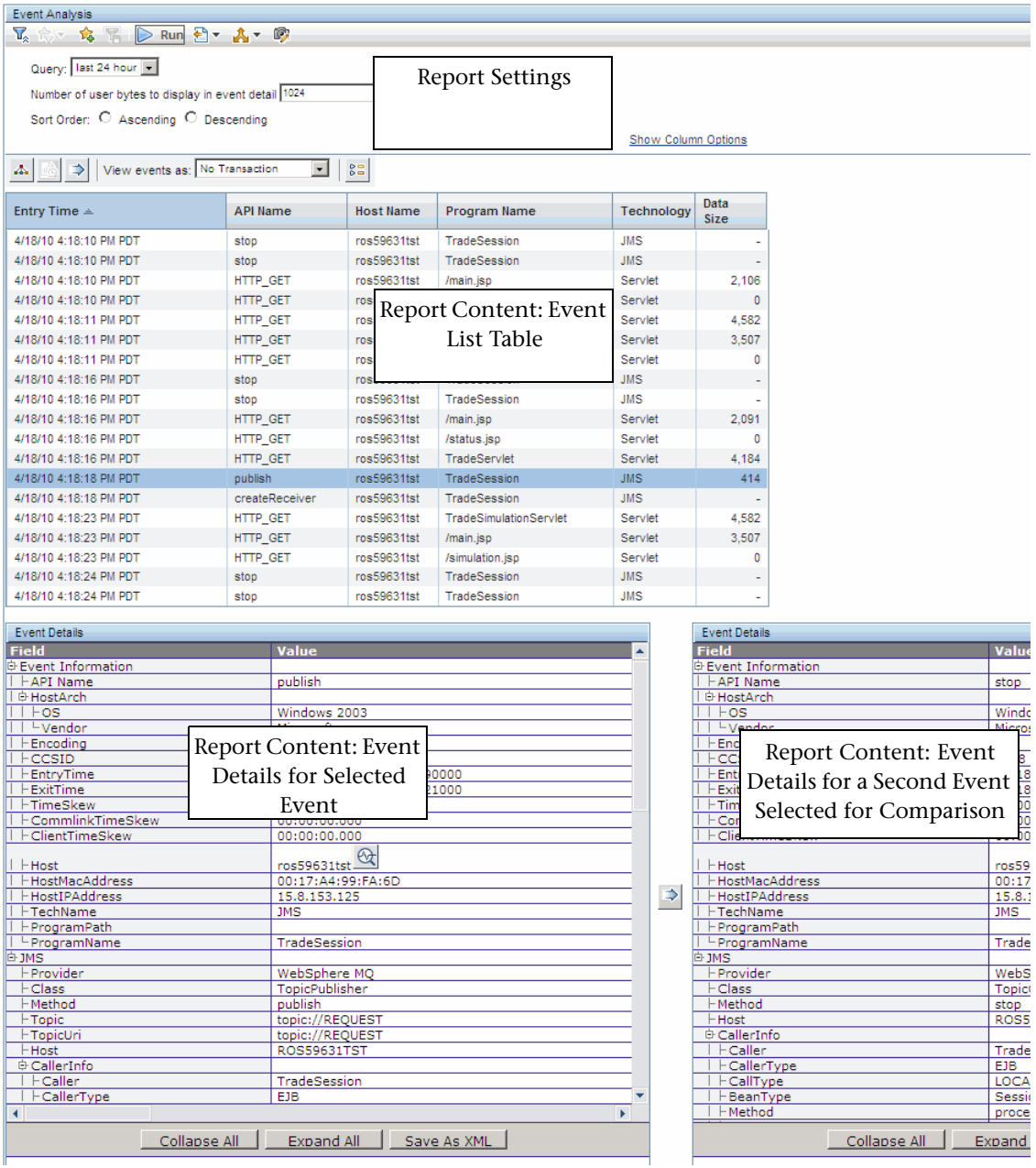
User interface elements are described below:

UI Element (A-Z)	Description
	The control flow icon is followed by the name of the application where the control flow starts (right arrow) or ends (left arrow). If a control flow starts from the application in a local transaction, a sub-row appears. The direction of the arrows in the sub-row indicates whether a control flow starts or ends from the application. Click the application name in the sub-row to navigate to the local transaction where the control flow ends.
	The message flow icon is followed by the name of the application where the message flow starts or ends. If a message flow starts from the application in a local transaction, a sub-row appears. The direction of the arrows in the message flow icon in the sub-row indicates whether a message flow starts or ends from the application.
	Both the Queue Access icon and the application name link to the local transaction where the message flow ends. Click this link to navigate to the local transaction where the message flow ends.
	Accesses a resource.
	Creates a thread for the local transaction.
	Indicates an error in the processing time.
	

Event Analysis Report

Displays all of the events in the database that match the query you specify. The query reduces the number of events displayed by restricting the hosts, queue managers, programs, times, or APIs associated with displayed events. You can also specify which data of the events display. For details on specifying query criteria for the events displayed on the Event Analysis report, see Chapter 6, "Queries."

The following is an example of the Event Analysis report.

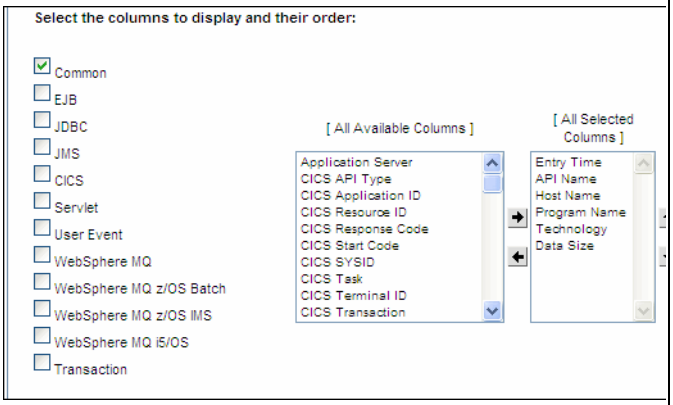


To access	Access in one of the following ways: ➤ Select **Applications** > **Transaction Management** > **Transaction Analysis** > **Event Analysis Report**. ➤ Select **Applications** > **Transaction Management** > **Transaction Analysis** > **Transaction Tracking Report** > click the **Drill to Transaction Detail** button > click **Show in Event Analysis View** in the Transaction Flow pane.
Important information	➤ This report includes data from TransactionVision only. ➤ If the events are sorted, an arrow in the column heading indicates the column and direction on which sorted occurs. Click another column heading to change the sort column. Sorting is for the current page only. ➤ The Event Detail area of this report requires that the Event XML document is stored in the database. See the **Store event document** option on "Miscellaneous Information Page" on page 199. ➤ If the associated data collection filter assigned is configured to collect the API Names Only data range, the Event Detail area does not display details for events recorded with that filter. For more information, see "Data Collection Filters" on page 165. ➤ To view event details, you must have permission to view User Data. For more information, see "Security" in the *HP TransactionVision Deployment Guide* PDF.
Relevant tasks	"How to View Metrics at the Event Level" on page 650
See also	➤ "Key Information Reported For a Transaction" on page 641. ➤ This report can be customized to show custom fields that have been collected and stored in the Event XML. See "Extending the User Interface" in the *HP TransactionVision Advanced Customization Guide* PDF.

Report Settings: General

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
<Common report elements>	For details, see "Common Report Elements" in *Reports*.
	Report filter. Move the cursor over the filter icon or click the collapsible arrows to view the filter options described in this table.
Query	Select one of the following: ➤ All (a built-in query) ➤ Last 24 Hours (a built-in query) ➤ A custom query. For information about custom queries, see "Queries" on page 259.
Number of User Bytes to Display in Event Data	The number of bytes of user buffer data to display in the Event Details report. **Default value**: 1024
Rows Per Page	Number of rows displayed on the page. **Default:** 20
Sort Order	Check to sort events chronologically when the number of events exceeds the threshold.

UI Element	Description
Show Column Options	Check to enable the column selection fields. Check the categories of columns first, then choose from those which ones to move to the Selected Columns list:  By default, the Common options are included on the report. See the table below for information about the columns. This report can also display custom columns. See "Extending the User Interface" in the *HP TransactionVision Advanced Customization Guide* PDF
Ascending	Click to display the events in ascending order by date and time.
Descending	Click to display the events in descending order by date and time.

Report Settings: Event Analysis Columns Available for Display

The categories and columns are described below.

Category	Available Columns
Common	**API Name.** The WebSphere MQ or JMS API name, the HTTP request, or EJB methods name. **Data Size.** For WebSphere MQ events, The size in bytes of the user data buffer for MQGET, MQPUT, and MQPUT1 calls. For JMS events, the size of JMS user data in send and receive calls. For Servlet events, the size of the returning HTML page. **Entry Time.** The time stamp of when the event initiated. **Event Time.** The time used to order the event chronologically. It may be either the value of the event entry time or the event exit time, depending on the API. For some event types such as those that receive messages, the important time is not when the event was first called, but when it returned (when the message was received). For those events, the event time is the event exit time; for all other events, the event time is the event entry time. **Exit Time.** The time stamp of when the event finished. **Host IP.** The host IP address of the machine the events occurred on. **Host Name.** The host running the application that called the WebSphere MQ or JMS API or the host the WebSphere application server is running on. **Program Name.** The name of the program making the event. **Technology.** The technology used by the event. Currently, WebSphere MQ, JMS, Servlet, EJB, JDBC, CICS, and User Events technologies are supported.
EJB	**EJB Name.** For EJB events, the EJB name associated with the event. If the Program Name column is displayed, the EJB name also appears as the program name.

Category	Available Columns
JDBC	**Database.** For JDBC events, the database name that these events are connected to. **JDBC Class.** For JDBC events, the JDBC class associated with the event. **SQL Objects.** For JDBC events, tables, stored procedures, views, this event accesses. **SQLState.** For JDBC events, SQL State values used by JDBC. The values are based on the underlying database vendor. **SQL Type.** For JDBC events, type of SQL statement — batched SQL statements such as DELETE, INSERT, MERGE, SELECT, Stored Procedure, UPDATE. **SQLCode.** For JDBC events, SQL code used by JDBC. The code is based on the underlying database vendor.
JMS	**JMS Class.** For JMS events, the JMS class associated with the event. **JMS Exception Code.** For JMS events, the exception code associated with the event. **JMS Queue.** For JMS events, the name of the JMS queue associated with the event. **JMS Topic.** For JMS events, the name of the topic associated with the event.

Category	Available Columns
CICS	**CICS API Type.** For CICS events, the API type (File Control, Interval Control, Program Control, Program Start, Start Attach, Task End, Task Start, Temporary Storage, and Transient Data). **CICS Application ID.** The name by which CICS is known to the network. **CICS Resource ID.** For CICS events, the resources used by CICS transactions. Currently, these resources are file name, transient data (TD) queue name, and temporary storage (TS) queue name. **CICS Response Code.** For CICS events, the CICS response code associated with the event. **CICS Start Code.** A two character code indicating how the CICS transaction associated with the event was started. Refer to the CICS Application Programming Reference (ASSIGN command) for the range of Startcode values and meanings. **CICS SYSID.** The CICS SYSID associated with the event. **CICS Task.** The CICS task associated with the event. **CICS Terminal ID.** The CICS Terminal ID associated with the event. **CICS Transaction.** The CICS transaction ID associated with the event.

Category	Available Columns
Servlet	**Application Server.** For servlet events, the name of the WebSphere MQ application server associated with the event. For WebSphere Application Server 5.0 Express Edition, the application server name default is server1; other names are an alias for this server name. **Servlet.** For servlet events, the name of the servlet associated with the event. **Status Code.** For servlet events, the status code associated with the event. Note: The TransactionVision servlet agent is unable to get the response status code and any headers in the response object for servlet events generated from an HTTP_HEAD request. Therefore, the status code value for HTTP_HEAD events is always N/A. **Web Application.** For servlet events, the name of the web application associated with the event.

Category	Available Columns
User Event	**DataPower Backend URL**. For DataPower events, the URL called by the DataPower service on the backend. **DataPower Frontend URL**. For DataPower events, the URL that represents the frontend of the DataPower service. **Generic Event Completion Code.** For generic events, the completion code associated with the event. For information about implementing generic events in your application, see the *HP TransactionVision Advanced Customization Guide* PDF. **Generic Event Action.** For generic events, the action associated with the event. **Generic Event Custom Object.** For generic events, the custom system model object representing the event. **Generic Event Status.** For generic events, the status associated with the event. **Generic Event Technology.** For generic events, the technology associated with the event. **User Event Class.** For user events, the class associated with the event. For information about implementing user events in your application, see the *HP TransactionVision Advanced Customization Guide* PDF. **User Event Completion Code.** For user events, the completion code associated with the event. **User Event Object.** For user events, the object associated with the event. **User Event Queue**. For user events, custom User Event Agents can use this field for any purpose. For Tuxedo user events, this is the Tuxedo queue. **User Event Queue Space**. For user events, custom User Event Agents can use this field for any purpose. For Tuxedo user events, this is the Tuxedo queue space. **User Event Service.** For user events, custom User Event Agents can use this field for any purpose. For DataPower events, this is the DataPower service. **User Event Status.** For user events, the status associated with the event. **User Event Technology.** For user events, the specific user event technology such as Tuxedo, .NET, DataPower.

Category	Available Columns
WebSphere MQ	**Completion Code.** The completion code in the WebSphere MQ API call return from the WebSphere MQ library. **Connection Name.** For all technologies, the connection name represents the connection the event is using. For WebSphere MQ, the connection name and queue manager are typically the same, except in the case where you open a queue on one queue manager using a connection (specified with the hConn parameter) to a different queue manager over a transmission queue. For example, suppose your program opens queue QM1.Q using a connection to queue manager QM1, then later opens QM1.Q using a connection to queue manager QM2. This event shows QM1.Q as the object name, QM1 as the queue manager, and QM2 as the connection name. **Object Name.** The WebSphere MQ object (queue, distribution list, namelist, and so on) associated with the event. **Queue Manager.** The WebSphere MQ queue manager associated with the WebSphere MQ event. The queue may not be always be the same as the connection name. For example, it is different when you are putting to a queue on a different queue manager from your connection over a transmission queue. **Reason Code.** The reason code in the WebSphere MQ API call return from the WebSphere MQ library. **Reply To Queue.** The WebSphere MQ or JMS queue in the Reply To field. **Reply To Queue Manager.** The WebSphere MQ or JMS queue manager in the Reply To field.
WebSphere MQ z/OS Batch	**z/OS Job Name.** The z/OS Batch job name associated with the event. **z/OS Job Step.** The z/OS Batch step name associated with the event.

Category	Available Columns
WebSphere MQ z/OS IMS	**IMS Id.** The z/OS IMS identifier associated with the event. **IMS PSB.** The z/OS IMS PSB name associated with the event. **IMS Region ID.** The z/OS IMS region identifier associated with the event. **IMS Region Type.** The z/OS IMS region type associated with the event. **IMS Transaction.** The z/OS IMS transaction name associated with the event.
WebSphere MQ i5/OS	**i5/OS Job Name.** The i5/OS job name associated with the event.
Transactions	**Txn Class.** The business transaction CI name. **Txn End Time.** The transaction end time. **Txn ID.** The transaction identifier. **Txn Label.** The events from transactions based on matching the transaction label. **Txn Response Time.** The end-to-end transaction response time, in milliseconds. **Txn Result.** The events from transactions that have a Failed/Success/Unknown result. **Txn SLA State.** The SLA state of Aged out/None/Violated. **Txn Start Time.** The transaction start time. **Txn State.** The events from transactions that are in a Completed/Processing/Unknown state. **Txn Value.** The data **value** attribute of a transaction that was set from executed data rules. For details on using data rules, see "Data Rules" on page 517.

Report Content: Events List Table

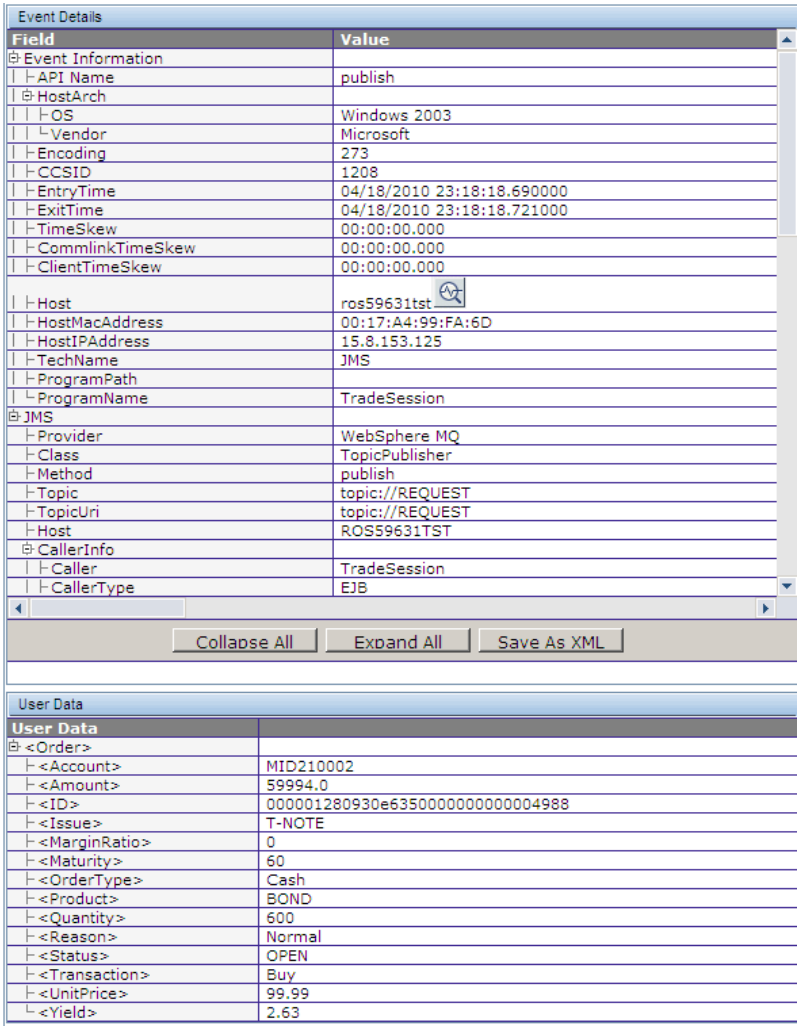
User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
	Drill to the Component Topology for the events in the selected business transaction.
	Drill to the Transaction Detail report for the selected event. Applies only when the events are viewed as Local Transaction or Business Transaction.
	Send the selected event to the right-hand pane so that it is available for a side-by-side comparison. Select a different event in the event list view to populate the left-hand pane.
	Reset column widths.
<navigation links>	If the Event Analysis report contains more than one page of events, use the navigation links to navigate between pages.
View events as:	**No transaction**. Show all events. **Local transaction**. Filter the events to show only those in the same local transaction. **Business transaction**. Filter the events to show only those in the same classified business transaction.

Report Content: Event Details

The Event Details displays details of a selected event in the Events List Table.

The event details are displayed in a hierarchy, enabling you to view different levels of detail.



Event Details

Field	Value
Event Information	
API Name	publish
HostArch	
OS	Windows 2003
Vendor	Microsoft
Encoding	273
CCSID	1208
EntryTime	04/18/2010 23:18:18.690000
ExitTime	04/18/2010 23:18:18.721000
TimeSkew	00:00:00.000
CommlinkTimeSkew	00:00:00.000
ClientTimeSkew	00:00:00.000
Host	ros59631tst
HostMacAddress	00:17:A4:99:FA:6D
HostIPAddress	15.8.153.125
TechName	JMS
ProgramPath	
ProgramName	TradeSession
JMS	
Provider	WebSphere MQ
Class	TopicPublisher
Method	publish
Topic	topic://REQUEST
TopicUri	topic://REQUEST
Host	ROS59631TST
CallerInfo	
Caller	TradeSession
CallerType	EJB

Collapse All | Expand All | Save As XML

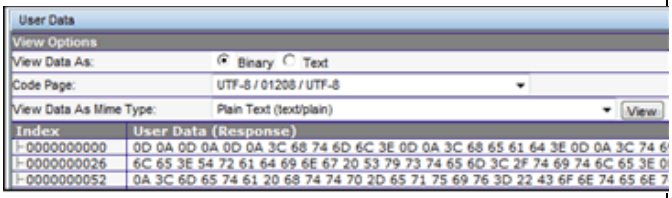
User Data

User Data	
<Order>	
<Account>	MID210002
<Amount>	59994.0
<ID>	000001280930e6350000000000004988
<Issue>	T-NOTE
<MarginRatio>	0
<Maturity>	60
<OrderType>	Cash
<Product>	BOND
<Quantity>	600
<Reason>	Normal
<Status>	OPEN
<Transaction>	Buy
<UnitPrice>	99.99
<Yield>	2.63

To access	**1** Select **Applications** > **Transaction Management** > **Transaction Analysis** > **Event Analysis.** **2** Select an event in the Event table of the EVent Analysis report. and scroll down to reach the Event Detail tables.

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
(from Host)	**Drill to HP Diagnostics**. The Hosts view opens with the TransactionVision host selected, data for the host is shown in the graph, and metrics for the host are displayed in the details pane. This is useful for troubleshooting as you see system metrics and can identify if the problem is a system issue. Some of the metrics available in the Event Details pane for the host system include: Average CPU Utilization, Average Memory Utilization, Average Disk IO in Bytes/Second, Average Network IO in Bytes/Second. From this Hosts view in Diagnostics you can see the performance of the host system charted over various time periods. Also you can check to see if the system has recently experienced a performance degradation or if the system has consistently under performed.
(from URI)	**Drill to HP Diagnostics**. The Analyze Snapshot view opens.
	Send the selected event to the right-hand pane so that it is available for a side-by-side comparison. Select a different event in the event list view to populate the left-hand pane.
Event Details	The field names that comprise the event and values are displayed. Click - or + to collapse or expand the field.
Collapse All	Click to collapse all nodes.
Expand All	Click to expand all nodes.

UI Element	Description
Save as XML	Click to save a copy of the current event as an XML document.
User Data	For an event that includes user data, the user data content is also displayed. Data is displayed in XML format or plain text.  XML data below the default size of 50000 bytes is displayed in tree form. XML data above that size is displayed as plain text to improve rendering speed in the Web browser. **View Data As**. ➤ **Binary**. View the data as hexadecimal. ➤ **Text**. View the data as plain text. **Default**: Binary

UI Element	Description
User Data (continued)	**Code Page**. View event payload as if it had been encoded using a different encoding method. This setting is for event payloads that were created and stored in a non-standard encoding. Select the encoding method from the drop-down list. This setting is typically used for non-UTF8 environments. **Default**: UTF-8/01208/UTF-8 **View Data as MIME Type**. View event payload using the appropriate format for the event. For example, for an event that contains a JPEG-formatted image as its payload, select a MIME type that can read an image, such as JPEG, BMP, or GIF. Otherwise, another type of format cannot display it properly. Select the MIME type from the drop-down list and click **View**. **Default**: Plain Text (text/plain) **Payload may also be viewed via**. View a large payload by clicking the **Download** button to select in which file format you want to view the data. This setting displays when a large payload (over 10,000, for example) is selected in the Event Details table.

Application Server Statistics Report

Displays statistics on the activities of EJB and Servlet components including average method execution time and call counts.

The following is an example of the Application Server Statistics report.

EJB Statistics

EJB	EJB Method	Success Call count	Error Call count	Average Method time (sec)
BondTradeProcessor				
	onMessage	4	0	1.01
EquityTradeProcessor				
	onMessage	4	0	0.57
FundTransferProcessor				
	onMessage	2	0	7.96
MDBReplyBroker				
	ejbCreate	1	0	0
	ejbRemove	1	0	0
	onMessage	6	0	0.15

Servlet Statistics

Servlet	Success Call count	Error Call count	Average Method time (sec)
/bond.jsp	2	0	0
/fx.jsp	4	0	0
/main.jsp	16	0	0.17
/status.jsp	6	0	0

To access	Select **Applications** > **Transaction Management** > **Transaction Infrastructure** > **Application Server Statistics**.
Important information	➤ You can filter the EJBs or Servlets for which you want to view statistics. ➤ This report includes data from TransactionVision only.

Report Settings

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
<**Common report elements**>	For details, see "Common Report Elements" in *Reports*.
View	The reporting time period. Choose an option from the drop-down list. For some time periods, also choose: **From**. The date and/or time for the beginning of the reporting time period; click to access a calendar from which to select the date/time. **To**. The date and/or time for the ending of the reporting time period from a calendar control.
Servlet Filter	Restrict the report by setting the filter. Specify a ';' delimited list of servlets to include or include wildcards such as '*servlet' or 'servlet*'. By default all servlets from this project are selected.
EJB Filter	Restrict the report by setting the filter. Specify a ';' delimited list of EJBs to include or include wildcards such as '*EJB' or 'EJB*'. By default all EJBs from this project are selected.

Report Content: Statistics Tables

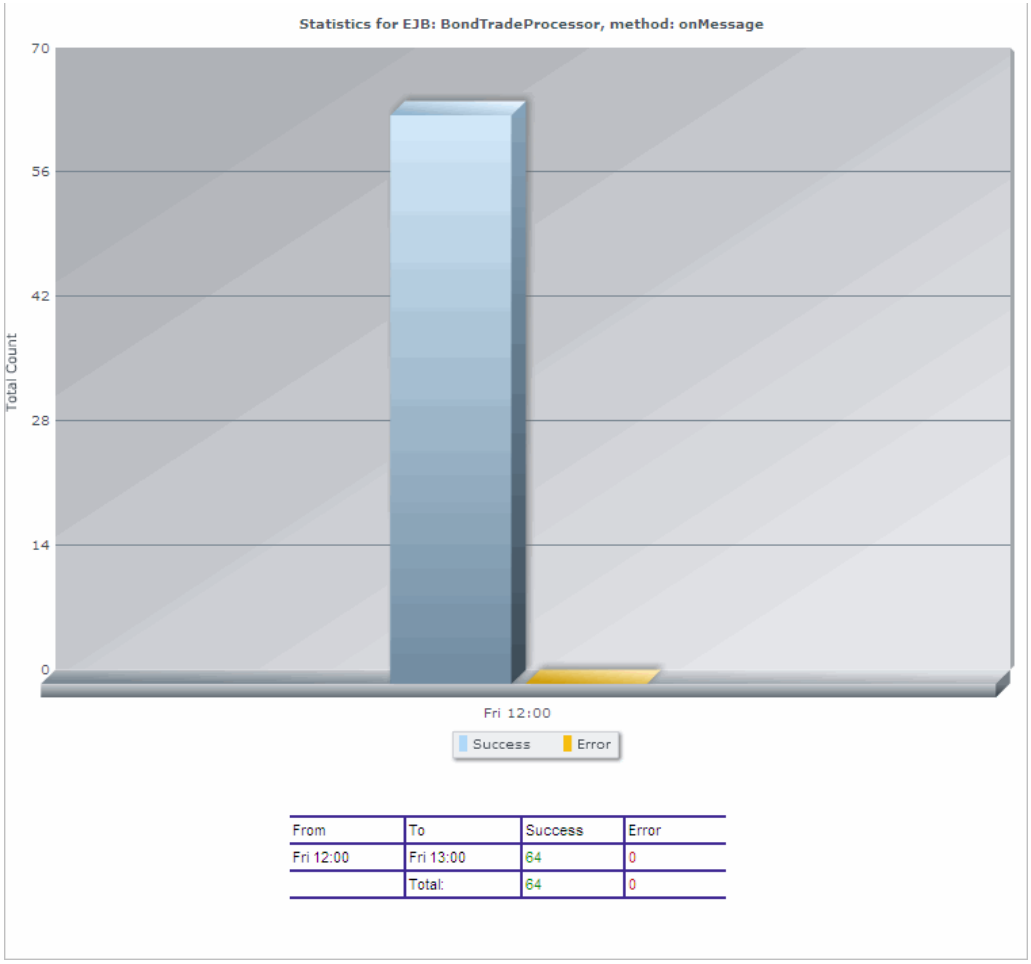
User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
EJB Statistics	**EJB**. EJB name. **EJB method**. The method in the EJB that was called. **Success Call Count**. The number of calls to the method that did not have an error. Click the call count link to view the total call count bar graph. **Error Call Count**. The number of calls to the method that did have an error. Click the call count link to view the total count bar graph. **Average Method Time (sec)**. Average method execution time, in seconds, for the method.
Servlet Statistics	**Servlet**. Servlet name. **Success Call Count**. The number of calls to the servlet request call that did not have an error. Click the call count link to view the total call count bar graph. **Error Call Count**. The number of servlet request calls that did have an error. Click the call count link to view the total count bar graph. **Average Method Time (sec)**. Average method execution time, in seconds, for the servlet.

Report Content: Total Call Count Graph

The Total Call Count graph displays the number of successful and error calls.

The following is an example of the Total Call Count graph for an EJB.



From	To	Success	Error
Fri 12:00	Fri 13:00	64	0
	Total:	64	0

WebSphere MQ and JMS Statistics Report

Displays statistics on WebSphere MQ and JMS infrastructure activity, assisting successful queue management.

The following is an example of the WebSphere MQ and JMS Statistics report.

WebSphere MQ Statistics

Queue Manager	Queue	Applications Interacting	Average Latency (sec)	Puts [success/warning/error]
bsavm2.tv1.manager		5	2.4	489 [489/0/0]
	SYSTEM.JMS.MODEL.QUEUE	1	-	1 [1/0/0]
	TV.REMOTE.QUEUE	1	-	1 [1/0/0]
	MULTIINCLUDE.CASE2.Q	1	-	4 [4/0/0]
	TV.LOCAL.Q1	3	9.47	13 [13/0/0]
	AMQ.4E52C88C095D0020	1	2.41	2 [2/0/0]
	SYSTEM.BROKER.CONTROL.QUEUE	1	-	107 [107/0/0]
	TRANSPATH.Q1	2	3.61	5 [5/0/0]
	MULTIINCLUDE.CASE5.Q1	1	-	3 [3/0/0]
	TRANSPATH.Q2	2	2.1	2 [2/0/0]
	SYSTEM.CHANNEL.SYNCQ	1	0	1 [1/0/0]
	MULTIINCLUDE.CASE4.Q	1	2.48	3 [3/0/0]
	ONMESSAGE.Q3	1	-	20 [20/0/0]
	SYSTEM.JMS.REPORT.QUEUE	1	0	176 [176/0/0]
	SYSTEM.DEFAULT.LOCAL.QUEUE	1	2.43	85 [85/0/0]
	TV.LOCAL.Q2	1	4.17	5 [5/0/0]
	TV.LOCAL.Q3	1	4.17	5 [5/0/0]
	bsavm2.tv2.manager	1	-	0 [0/0/0]
	TV.ALIAS.QUEUE	1	1.62	3 [3/0/0]
	TV.LOCAL.Q4	1	4.17	5 [5/0/0]
	MULTIINCLUDE.CASE3.Q	1	-	3 [3/0/0]
	ONMESSAGE.Q2	1	2.61	20 [20/0/0]
	SYSTEM.BROKER.DEFAULT.STREAM	1	-	2 [2/0/0]
	ONMESSAGE.Q1	1	8.66	4 [4/0/0]
	TV.JMS.REPLY.QUEUE	1	3.92	19 [19/0/0]

JMS Statistics

Queue Manager	Queue	Applications Interacting	Average Latency (sec)	Puts [success/warning/error]
TRADING		13	0.04	1,502 [1,502/0]
	REPLY_FOR_SESSION	2	0.16	374 [374/0]
	BOND	2	0	115 [115/0]
	EQUITY	2	0	102 [102/0]
	MQ/MQJMS/TT/414D512054524144494E4720202020209FB3304E20016502/1	1	-	1 [1/0]
	SYSTEM.DEFAULT.MODEL.QUEUE	2	-	2 [2/0]
	MQ/MQJMS/TT/414D512054524144494E4720202020209FB3304E2000FE02/1	1	-	1 [1/0]
	REQUEST	2	0	375 [375/0]

To access	Select **Applications** > **Transaction Management** > **Transaction Infrastructure** > **WebSphere MQ and JMS Statistics Report**.
Important information	Click an item in the **Total** link in the table or the **Total Count** bar in the chart, to view transaction details for all the transactions for a given day, in a new window.

Report Settings

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
<Common report elements>	For details, see "Common Report Elements" in *Reports*.
View	The reporting time period. Choose an option from the drop-down list. For some time periods, also choose: **From**. The date and/or time for the beginning of the reporting time period; click to access a calendar from which to select the date/time. **To**. The date and/or time for the ending of the reporting time period from a calendar control.
Queue Name Filter	By default all queues in the project are included in the report. To restrict the report, check the queue filter and enter the desired queues (or the * wildcard character), separating queues with commas.
Queue Manager Filter	By default all queue managers in the project are included in the report. To restrict the report, check the queue manager filter and enter the desired queue managers (or the * wildcard character), separating them with commas.

Report Content

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element A-Z	Description
Applications Interacting	The number of applications interacting within this queue or queue manager.
Average Latency (sec)	The average latency for this queue or queue manager.
Gets (success/ warning/error)	Click the event count link for a queue to view a chart and table showing additional information about the **Get** events for that queue. The results display in a new window.
Puts (success/ warning/error)	Click the event count link for a queue to view a chart and table showing additional information about the **Put** events for that queue. The results display in a new window.
Queue	The queue for which the data is displayed.
Queue Manager	The queue manager for which the data is displayed.

Troubleshooting and Limitations

This section describes troubleshooting and limitations for the Transaction Management reports.

Errors Retrieving Content for Transaction Detail Reports

In some cases the Transaction Detail report does not render when drilled to from the Transaction Tracking report. It appears to hang, and after a few minutes (possibly as much as 5) returns an error message **Error retrieving content for this report...**.

This is typically caused by inadequate database sizing or tuning. For example, the built-in database provided with TransactionVision does not perform well with large amounts of data.

For reasonable performance the number of events to be stored should not exceed more than 100,000 - 200,000 events when using the built-in database. For more information about database sizing and tuning, see the *HP TransactionVision Planning Guide* on the HP Software Product Manuals Web site at:

http://h2030.www2.hp.com/selfsolve/manuals

You must have an HP Passport account. Choose **TransactionVision**, the desired product version and operating system, and click **Search**.

JMS Icons are Missing from Event Analysis

In some cases TIBCO JMS events do not show JMS queue/topic name in the Event Analysis view and there are no JMS icons in the Component Topology view. The Sensor error log contains the following entry:

```
com.tibco.tibjms.admin.TibjmsAdminException: Unable to connect to server.
Root cause:
javax.jms.JMSSecurityException: invalid name or password, or not authorized to
connect as administrator
```

To enter the appropriate EMS Server credentials, modify the following file in the Java Agent installation directory:

<java_agent_install_dir>/TransactionVisionAgent/config/sensor/SensorConfiguration.xml

Enter the lines in bold below:

...
<!-- for object resolution, we need admin right to access servers
the incoming url could be tcp://heineken:7222 or
tcp://heineken.bristol.com:7222 or using ssl -->
<TibcoEMSServers>
<TibcoEMSServer name="ems_server_name" loginName="XXX"
password="YYY">
<ServerUrl>tcp://ems_server_host_name:7222</ServerUrl>
</TibcoEMSServer>
</TibcoEMSServers>
...

21

Transaction Topologies

This chapter includes:

Concepts

- Transaction Topology Overview on page 718
- A Closer Look at the Flow Map on page 720

Tasks

- How to View the Aggregated Topology of a Business Transaction on page 738
- How to View the Instance Topology of a Business Transaction on page 739

Reference

- Transaction Topologies User Interface on page 740

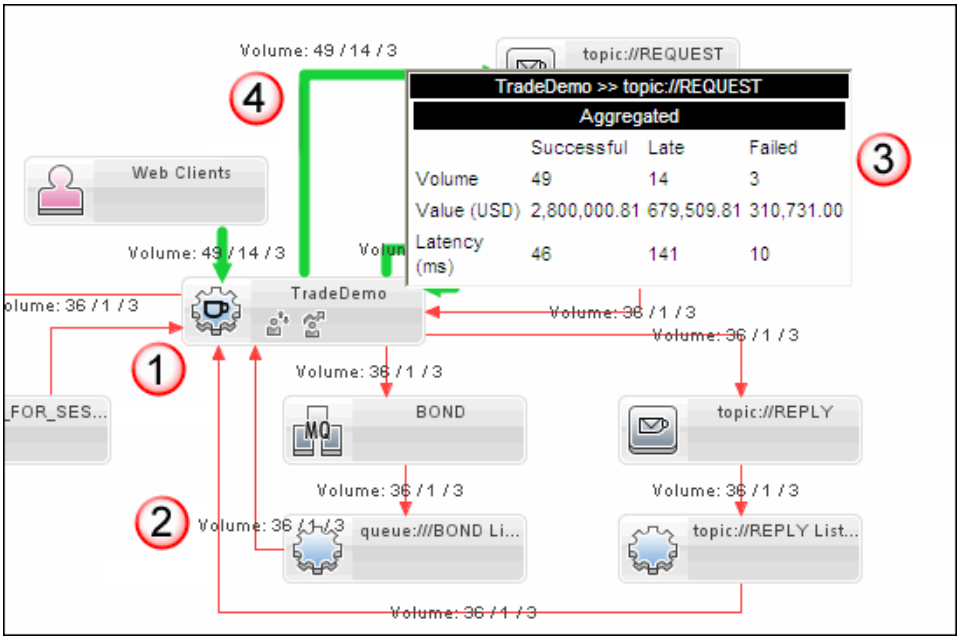
Troubleshooting and Limitations on page 753

Concepts

Transaction Topology Overview

A transaction topology is a flow map for a business transaction CI. That is, it shows the components through which the business transaction executes. The flow map includes metrics about the data flow at specific points and properties of the components through which the data flows.

The metrics display in arrow labels and properties display in tooltips:



The following table describes the diagram callouts:

Callout	Description
1	Each component appears as a node with arrows indicating message flows between components.
2	Labels can be enabled that indicate details of the data flow between components.
3	Tooltips provide more details for an arrow or a component.
4	Links can be customized to display varying link widths and colors to quickly see the metrics and state of each link.

You specify the Business Transaction CI type for which you want to view a transaction topology as well as the time frame for which you want to view data.

Along with the flow map, the Transaction Topology includes detail information about selected components, including KPI information where applicable. Components that do not have KPIs or are not RTSM components do not present this information.

Aggregated Versus Instance Transaction Topologies

There are two versions of the transaction topology:

- The Aggregated Topology displays information about all transaction instances that exist in the time frame. The metrics that display are aggregated. For example, the data flowing from the TradeDemo to the Bond component shown above had two instances, one completed successfully and one completed late.
- The Instance Topology displays information about a single transaction instance along with the aggregated information described above. The instance is specified by drilling down from the Transaction Tracking report.

HP Diagnostics Data and the Transaction Topology

Transactions Monitored by Diagnostics provide basic topology information. Where the same transaction is monitored by multiple products the topology is merged together. Link highlighting by volume, value, and latency is not available on transactions that are only monitored by Diagnostics.

Real User Monitor and the Transaction Topology

Transaction Monitored by RUM provides basic topology information.Where the same transaction is monitored by multiple products the topology is merged together. Link highlighting by volume, value, and latency is not available on transactions that are only monitored by RUM. Some RUM transactions provide additional statistical information in the Statistics tab below the topology for certain links in the graph when they are clicked.

A Closer Look at the Flow Map

The following sections describe the key areas of the flow map:

- "Completed Versus In-Process Data" on page 721
- "Components" on page 721
- "Connecting Arrows" on page 722
- "Link Labels" on page 723
- "Link Highlighting" on page 726
- "Tootips" on page 729
- "Groups/Plain Modes" on page 732
- "Check Marks for Instance Topologies" on page 737
- "Start and End Nodes for the Instance Topologies" on page 737

Completed Versus In-Process Data

The Aggregated Topology divides all the transaction instances into these categories, each represented by a tab:

- **Completed**: Metrics and flow map are only for transaction instances that have completed in the given time frame. A transaction is completed when the Transaction State property is **completed.**
- **Currently In-Process**: Metrics and flow map are only for transactions that have not completed in the given time frame. You can view this tab to locate any instances that appear to be bottlenecked and on which component that occurs. You can also compare the metrics of a given instance with the average reported on the Completed tab.

The Instance Topology does not have multiple transaction instances, so the tabs do not appear.

Components

Each component that appears in the flow map is one through which the selected business transaction passed data. You can select a component in the flow map to display additional information about that component in the tables below the flow map. These details include:

- RTSM properties display if the selected object is known to the RTSM.
- KPIs display if there are any being monitored on the selected object.
- Transaction level health indicators (HIs) and key performance indicators (KPIs) information display in the Properties area of the Aggregated Topology (see "Aggregated Topology Page" on page 740). The HIs appear on the **Transaction HIs** tab and the KPIs appear on the **Transaction KPIs** tab. These tabs are equivalent to similarly named tabs in Service Health when a transaction is selected in that component using the Model Explorer.
- Additional custom details per application display.

Components in the flow map are multi-tiered. That is, application servers are shown inside of hosts, queues are shown inside of queue managers, and so forth.

Components on the map can be re-arranged, and the map itself can be panned and zoomed. The toolbar contains buttons to change the labeling of components, as well as pan, center, zoom, and select components in the map.

The component icons each represent a specific type of software or hardware element that the transaction instances have passed through. The flow map always shows the back-end server-level components such as application servers, message middleware servers, database servers, mainframe transaction and batch jobs.

Other icons are used for elements that originated from an instance of the transaction, such as a Web Client. If RUM is also monitoring the transaction, the flow map displays an End User Group component icon if the transaction originates from an end user group.

Not all component types have a specific icon that appears in the flow map.

The icon for each component type is determined by the RTSM class type for that component. For a list of the component icons used in the Transaction Topology flow map, see "Aggregated Topology Page" on page 740.

Connecting Arrows

Communication between components in the flow map is represented by links (directed arrows). The following link patterns between the component icons can be seen in the flow map:

- A directional link between two components

  A directional arrow shows direction of message flow or a call from one component to another.

- Two links between two components, each link going in the opposite direction

  Many times this shows an exchange of request and response data flow.

- A component with a self-directed link

  The transaction spent multiple processing steps at the same component. Such a link is common when viewing nodes that may contain lower level components (for example one Web Application can contain several Servlets and EJBs which call each other).

- A flow with several end points

  Many times this can represent an asynchronous transaction and some components either did not send a response or they sent a response to a component that is not monitored by TransactionVision.

- A circular flow

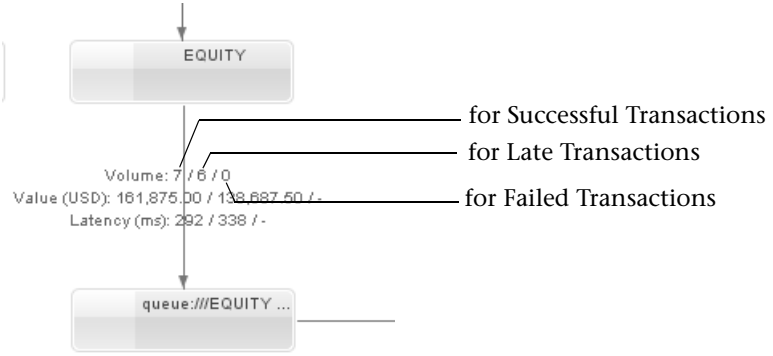
  May indicate that the request was answered by something other than the component that took the request.

- Unconnected flow paths

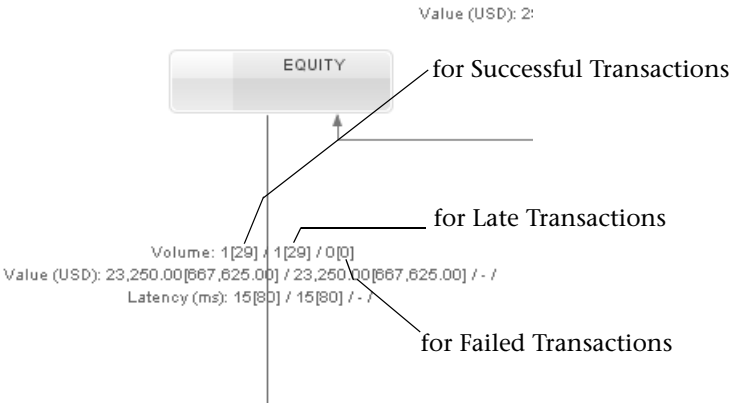
  This can occur when there are elements that cannot be interpreted by TransactionVision, for example, if the technology is not supported.

Link Labels

Labels can be enabled on each link using the **Link Labels** menu on the flow map toolbar. You can display metrics related to the flow of transaction instances (either completed or currently in-process, as relevant) between the two components connected by the arrow. The metrics are totaled for each transaction state:



For the Instance Topology, the instance-level metrics appear alongside the aggregate-level metrics in brackets:



Data You Can Display as Labels

By default, no labels display. You can choose to display any or all of the following labels:

Label/Metric	Description
Volume	Displays the count of the transaction instances that either passed between the two components or are currently in-process of being passed. The count is displayed in each of the three transaction state categories: Successful, Late, and Failed.
Value	Displays the total monetary value associated with the transaction instances that either passed between the two components or are currently in-process of being passed. The count is displayed in each of the three transaction state categories: Successful, Late, and Failed.
Latency (average)	Displays the average latency (response time) for the transaction instances that either passed between the two components or are currently in-process of being passed (in milliseconds). The count is displayed in each of the three transaction state categories: Successful, Late, and Failed.

The Business Transaction States

The Business Transaction state categories are:

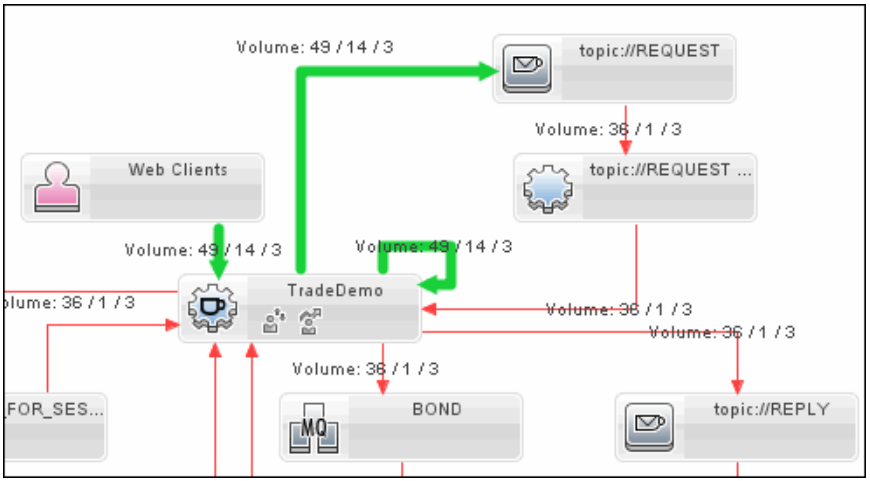
Transaction State Category	Description
Successful	The total count of successful transaction instances.
Failed	The total count of failed transaction instances, that is, those that met the transaction's Failure rule. For information about specifying the failure data rule, see "Tracing Tab, Data Rules Tab" on page 554.
Late (Also referred to as delayed.)	The total count of transaction instances with a response time that exceeded the threshold defined for the business transaction. For information about setting the Threshold, see "Transaction Configuration Page, Tracing Tab" on page 550.

Note: Transactions can be classified in more than one state category, for example, as both late and as successful.

If a metric is not applicable for a link, or if there is no criteria set in TransactionVision for late status or failed status, then the relevant metric shows a dash.

Link Highlighting

You can use the **Link Rendering** menu on the flow map toolbar to change the width and/or color of the links to quickly see flow problems. The following example shows which connections have a larger volume of transaction that use them (wide green links) and which connections have failed transactions (red links).



Notes:

- If there is no TransactionVision transaction data, then the **Link Rendering** menu does not appear.
- Link highlighting does not display when the groups are collapsed in Groups Mode (see "Groups Mode: Show/Hide Tiers" on page 732).

Link Widths to Display Metrics

By default, the links are all the same width in black. You can choose to show link width for any of the following metrics available on the flow map:

Label/Metric	Description
Volume	Displays wider links for connections between components that have a larger volume of transactions that use them. Narrower links are displayed for connections that are used less often.
Value	Displays link widths based on the monetary value of transactions across a particular link. Links that have little or no monetary activity are shown as narrow links, whereas links that have substantial monetary activity are wider.
Latency (average)	Displays links wider or thinner based on their absolute latency for the transaction instances that either passed between the two components or are currently in-process of being passed (in milliseconds).
Latency (relative)	Displays wide links that are relative to the median, rather than by absolute value for the transaction instances that either passed between the two components or are currently in-process of being passed (in milliseconds).

Link Colors to Display States

You can use the **Link Rendering** button on the flow map toolbar to color code the following link states. The color of the highlighting reflects the state of the transaction:

Transaction State Category	Description
Dominant Transaction State	Links are displayed in the color corresponding to the most common transaction state across that link. ➤ If more transactions are successful across a link than were failed or late, the link is green. ➤ If more transactions are late, the link is yellow. ➤ If more transactions failed than succeeded or are late, then the link is red. Note that if 51% of transactions are successful and 49% failed then the link is green, which shows the dominant state.
Failed	Displays red links for any of the following selected options: ➤ **Show Failed Transactions Above Configured Threshold** ➤ **Any Failed Transactions** ➤ <percent> **or more Failed Transactions** You can set the <percent> value for the **Above Configured Threshold** option in the Topology Properties dialog by selecting **Link Rendering** > **Threshold Properties**.
Late (Also referred to as delayed.)	Displays yellow links for any of the following selected options: ➤ **Show Late Transactions Above Configured Threshold** ➤ **Any Late Transactions** ➤ <percent> **or more Late Transaction**s You can set the <percent> value for the **Above Configured Threshold** option in the Topology Properties dialog by selecting **Link Rendering** > **Threshold Properties**.

Tootips

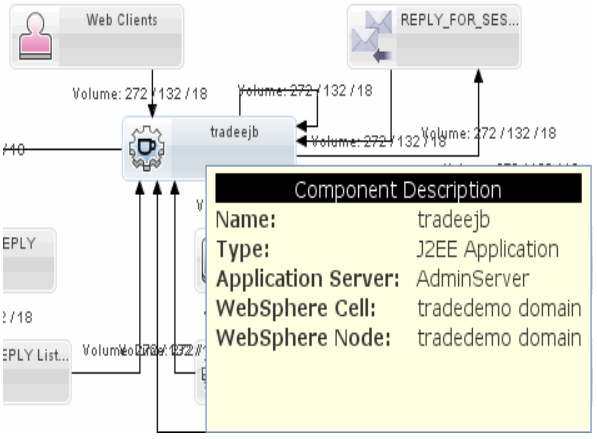
You can hover over icons or arrows in the flow map to display tooltips, which provide additional details for an arrow or a component. The tooltip provides node or group type (RTSM type) at the top of the tooltip and details such as:

- Metrics showing volume, value, latency (average), and latency (relative).
- State of the transaction across that link in the flow map, indicating the transaction is successful, late, or failed.
- Information relating to end user transactions that are mapped to the Business Transaction for the Web Clients and End User Group components.
- Status indicators on relevant CIs. KPI status displays on nodes on which the KPIs have been configured. Nodes that are displayed as groups since they have nested elements do not display indicators. The KPIs are color-coded in the same way they are in Service Health (green for OK, red for CRITICAL, and other colors consistent with Service Health color coding).
- Additional information if the transaction is also monitored by HP Diagnostics or RUM.

Example of Tooltip Metrics for a Component

The component tooltip contains more information about how the components is relevant to the transaction.

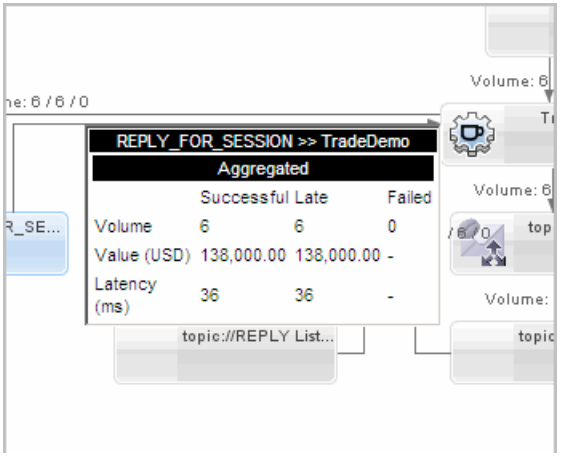
For example:



The component tooltip may include additional data from the software or hardware upper layers. For example, if a transaction instance passed through a J2EE application, the map shows a flow component for the J2EE application, and may include additional data for the application server and the host in the tooltip.

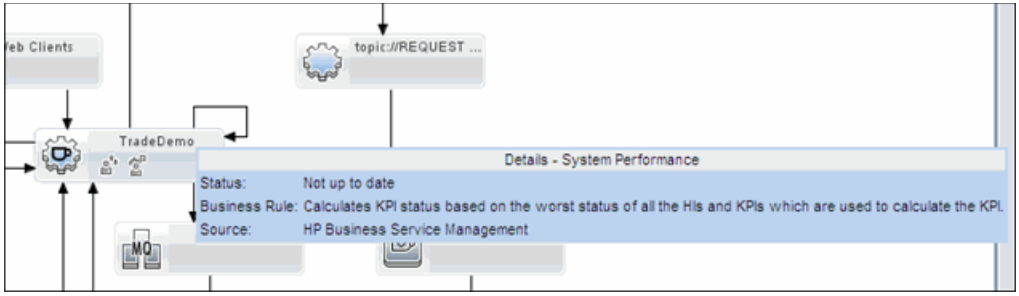
Example of Tooltip Metrics for an Aggregated Topology Arrow

This example shows a tooltip for an Aggregated Topology link showing the Late and Failed metrics with no criteria set.



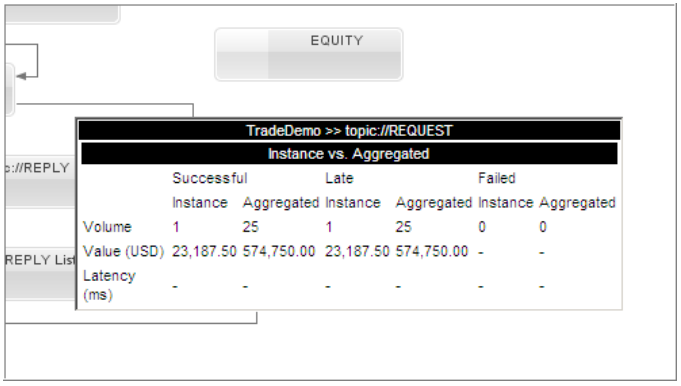
Example of Tooltip Status Indicators for an Aggregated Topology CI

This example shows a tooltip for an Aggregated Topology link showing the KPI is not up to date. User Configured KPIs are displayed on nodes in the topology. These can include status and performance KPIs and are color coded based on their current state according to the same rules that apply in the Service Health topology. The title shows which KPI is on display. **Status** indicates the current status for the indicated KPI. **Business Rule** is the business logic used to calculate the status. **Source** indicates which product provided the data which is used to calculate the KPI status.



Example of Tooltip Metrics for an Instance Topology Arrow

This example shows a tooltip for an Instance Topology link that compares the metrics of the path taken by a Business Transaction instance with the path taken by other instances of that Business Transaction. The dotted line represents the Aggregated Topology of the Business Transaction with respect to the Instance Topology, which is a solid line.



Groups/Plain Modes

You can enable or disable displaying the components in their associated tiers by toggling between the **Groups Mode** and **Plain Mode** buttons on the "Aggregated Topology Page" on page 740.

Groups Mode: Show/Hide Tiers

Users can view connections between components at different tiers of infrastructure by expanding or collapsing groups that represent that tier of infrastructure component. A group representing a host, if collapsed, shows all connections into or out of that host. But if the group is expanded, those connections are shown as connected to the server tier components, which are shown as connected to the application tier.

The following table shows the tiers that appear when the tiers are enabled.

Topology Tier	Content in the tier
Host Tier	**host**. Generic host. **nt**. Windows host. **unix**. UNIX host. **mainframe**. Mainframe host.

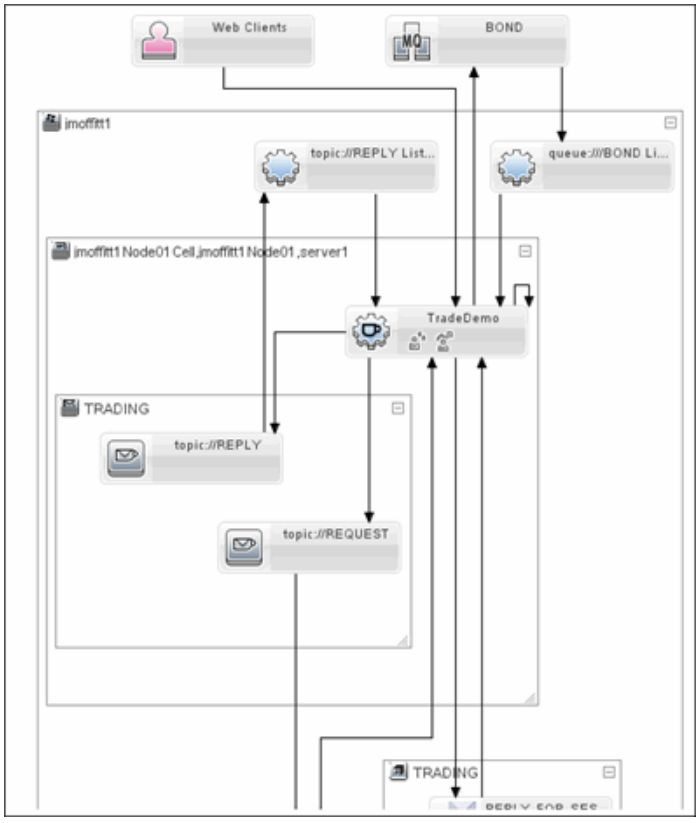
Topology Tier	Content in the tier
Server Tier	Expanded from mainframe: **ims_program**. IMS Program. **batch_job**. Batch job. **cics_region**. CICS Region.
	Expanded from host: **websphereas.** Websphere Application Server **weblogicas**. WebLogic Application Server **sqlserver**. SQL server database instance **iis**. Windows internet information services instance **j2eeserver**. Generic java enterprise server **db2**. DB2 database instance **oracle**. Oracle Database instance **jmsserver**. Generic JMS messaging server **running_software**. Software application **customer**. User **rum**. End user group (a collection of users specified by rules in RUM) **mqqueuemanager**. WebSphere MQ Queue Manager cics_region.

Topology Tier	Content in the tier
Application Tier	Expanded from jmsserver: **jms_queue**. JMS message queue **jms_topic**. JMS message topic **jmsdestination**. JMS destination
	Expanded from j2eeserver: **j2eeapplication**. Software application running on a J2EE server
	Expanded from iis: **webservice**. Software application running on an IIS server
	Expanded from mqqueuemanager: **mqqueue.** Generic WebSphere MQ Queue **mqqueuelocal.** WebSphere MQ Local Queue **mqqueueremote.** WebSphere MQ Remote Queue **mqqueuealias.** WebSphere MQ Alias Queue
	Expanded from mainframe **cics_transaction**. CICS transaction.
Other Types	**mqcluster**. WebSphere MQ Cluster Queue This node has no parent and is at the Host level.

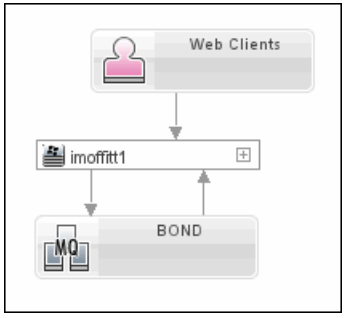
Example Groups Mode

In Groups Mode an application is surrounded by a group that represents the Application Server where that application is running, and that Application Server group is surrounded by a node representing the host/server where it is running.

Following is an example of an expanded Aggregated Topology in Groups Mode:



Following is an example of a collapsed Aggregated Topology in Groups Mode:

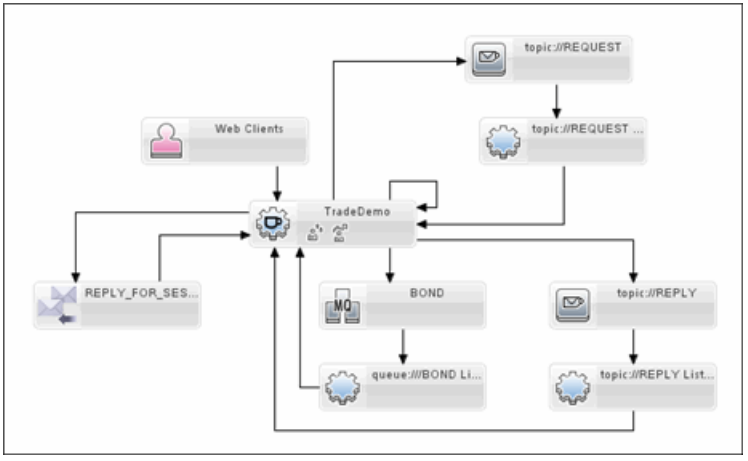


Plain Mode

Plain Mode only displays components that are at the lowest known level of infrastructure without their parent components.

For example, in Plain Mode an application is shown as a standalone node with links entering and exiting.

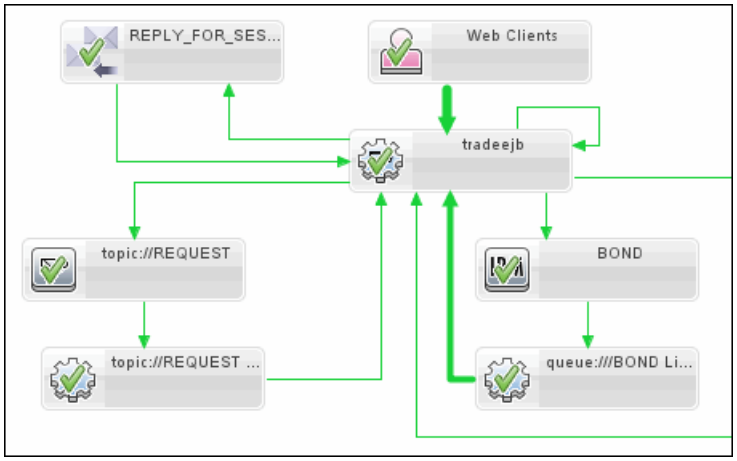
Following is an example of an expanded Aggregated Topology in Plain Mode:



Check Marks for Instance Topologies

In the instance topologies only, check marks appear in the component icons to show which nodes/groups a transaction went through.

Following is an example of an instance topology:



Start and End Nodes for the Instance Topologies

When viewing the Instance Topology from a business transaction drill down (from the "Transaction Tracking Report" on page 675 or the "Transaction Detail Report" on page 686), start and end nodes may be displayed. This helps you quickly see where the start and end points are located, especially for very complex topologies.

An icon with a **Play** symbol represents the starting point of the instance topology and is linked to the actual starting node. Similarly, the end point is represented by an icon with a **Stop** symbol and is linked to the ending node. For an example of the Instance Topology with the start and end nodes, see "Instance Topology Page" on page 750.

Note: The start and end points are determined by event timestamps, and may or may not be accurate. This is especially true if the graph has more than one start and/or end point.

Tasks

How to View the Aggregated Topology of a Business Transaction

This task includes the following tasks:

- "Determine the Transaction Category You Need" on page 738
- "View the topology for Completed Business Transactions" on page 738
- "View the Topology for In-process Business Transactions" on page 739

Determine the Transaction Category You Need

The Aggregated Topology can display the topology for completed or in process business transactions. Review the information "Completed Versus In-Process Data" on page 721 to determine which category you need.

View the topology for Completed Business Transactions

1 Select **Applications** > **Transaction Management** > **Transaction Topology** > **Aggregated Topology.**

By default the **Completed** tab should be displayed, if not click it.

The topology displays based on the last business transaction that was specified. If this is the initial viewing of the report, use the Transaction Filter link to specify the business transaction.

2 To specify a different business transaction, access the Transaction Filter dialog box. See "Transaction Filter Dialog Box" on page 657 for details.

View the Topology for In-process Business Transactions

1 Select **Applications** > **Transaction Management** > **Transaction Topology** > **Aggregate Topology.**

2 Click the **In Process** tab.

The topology displays based on the last business transaction that was specified. If this is the initial viewing of the report, use the Transaction Filter link to specify the business transaction.

3 To specify a different business transaction, access the Transaction Filter dialog box. See "Transaction Filter Dialog Box" on page 657 for details.

How to View the Instance Topology of a Business Transaction

1 Run the Transaction Tracking report so that the transaction instance for which you want to view the topology is on the report.

For details about this interface, see "Transaction Tracking Report" on page 675.

2 Select the transaction instance and click the **Drill to Instance Topology** button.

Reference

Transaction Topologies User Interface

This section includes (in alphabetical order):

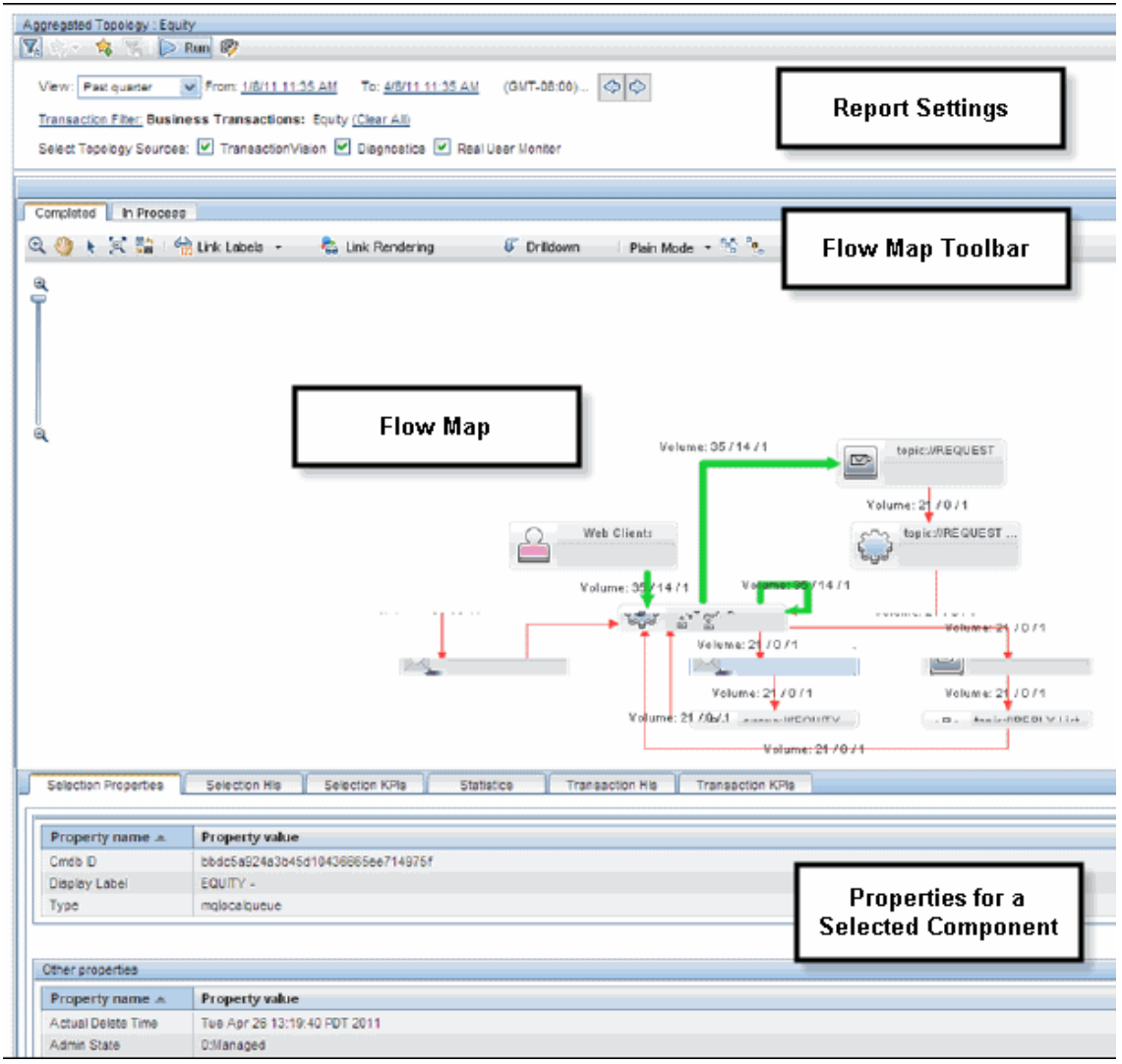
➤ Aggregated Topology Page on page 740

➤ Instance Topology Page on page 750

Aggregated Topology Page

The Aggregated Topology page displays a flow map and response time and volume metrics for a business transaction.

The following is an example of the Aggregated Topology.



To access	Aggregated Topology: ➤ Select **Applications** > **Transaction Management** > **Transaction Topology** > **Aggregated Topology.** ➤ From the Transaction Over Time or the Transaction Summary report, click the **Drill to Aggregated Topology** button.
Important information	➤ The Aggregated Topology can be based on combined data from all products monitoring or tracing the transaction: TransactionVision, HP Diagnostics, or Real User Monitor. Choose the source before running the report. ➤ The Instance Topology is similar to the Aggregated Topology, except it shows metrics for a single Business Transaction instance. For details, see "Instance Topology Page" on page 750.
Relevant Tasks	➤ "How to View the Aggregated Topology of a Business Transaction" on page 738 ➤ "How to View the Instance Topology of a Business Transaction" on page 739
See also	➤ "Transaction Topology Overview" on page 718 ➤ "A Closer Look at the Flow Map" on page 720

Aggregated Topology Settings

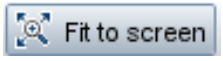
User interface elements are described below (unlabeled elements are shown in angle brackets):

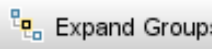
UI Element	Description
<Common report settings>	For details, see "Common Report Elements" in *Reports*.
View	The reporting time period. Choose an option from the drop-down list. For some time periods, also choose: **From**. The date and/or time for the beginning of the reporting time period; click to access a calendar from which to select the date/time. **To**. The date and/or time for the ending of the reporting time period.
Transaction Filter	Launches the Transaction Filter dialog box, where you select the business transaction on which to base the transaction topology. See "Transaction Filter Dialog Box" on page 657.
(Clear All)	Clears any active filters.
Select Topology Sources	The transaction on which the topology is based can be monitored by any or all of these products. You can choose which product's data the topology is based on.

Flow Map Toolbar

The following toolbar options are included:

UI Element	Description
	Interactive Zoom. Click the button and then drag to the area of the flow map you want to zoom in to or out from.
	Pan. Click the button and then click and drag the flow map to view different areas of the flow map.

UI Element	Description
	Select. ➤ click an entity in the map to select it. ➤ click and drag an entity in the map to reposition it within the flow map. For example, you can click on the elbow of a link arrow and drag it to a different location. ➤ click and drag over an area to select the entities within that area. Click again outside the selected area to deselect.
	**Fit to screen**. Click to resize the flow map to fit the available viewing space.
	Toggle presentation type. Click to change the component rectangles to icons, which can sometimes make the topology easier to view.
	Link Labels. Choose the data you want to display on the link arrows: ➤ **Value.** Displays value metrics for each link, alongside each link arrow. ➤ **Volume.** Displays volume metrics for each link, alongside each link arrow. ➤ **Latency.** Displays latency metrics for each link, alongside each link arrow. For more information on these options, see "Link Labels" on page 723. **Default value**: no data displays on the link arrows. The data is available on the tooltip however.

UI Element	Description
	Link Rendering. Choose the option or options for displaying varying link widths and colors to quickly see the metrics and state of each link: ➤ **No Link Highlighting.** Displays links as thin, black lines. ➤ **Width Rendering Options.** Displays the links in varying widths depending on the metric chosen. ➤ **Color Rendering Options.** Displays the links in the colors associated with the state on each link. ➤ **Threshold Properties.** Opens the Topology Properties dialog to set the percentage for highlighting error and late thresholds. For more information on these options, see "Link Highlighting" on page 726. **Default value**: links are not highlighted. The data is available on the tooltip however.
	Drilldown. Drill to the Transaction Tracking report. If HP Diagnostics is monitoring the transaction, you can also drill to the Business Transaction Paths view.
	Plain/Groups Mode. Click to enable or disable showing the components in their associated tiers. For details, see "Groups/Plain Modes" on page 732. ➤ **Plain Mode** shows the lowest tier components for each connection. ➤ **Groups Mode** encapsulates components in groups representing higher level entities such as servers and hosts.
	Collapse Group. Collapses all groups so only the highest level tiers are showing. For more details, see "Groups/ Plain Modes" on page 732.
	**Expand Group.** Expands the groups so that all tiers are showing. For more details, see "Groups/Plain Modes" on page 732.

Aggregated Topology Content: Flow map

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
<Component icon>	Each flow component (for example, Application, Servlet, MQ Queue, and so forth) is represented by an icon: **Tooltip:** Hold the pointer over a component icon to display a tooltip containing information for that component. For the **Web Clients** and **End User Group** components, the tooltip displays information relating to end user transactions that are mapped to the Business Transaction. For more information on the data in the tooltips, see "Tootips" on page 729.
	Program component. Generic events that are used in transactions are displayed as program components. For information about implementing generic events, see the *HP TransactionVision Advanced Customization Guide* PDF.
	Application. A generic application that has not been identified as being of a more specific application type.
	Batch Job component
	CICS Region.
	CICS Transaction component
	DB2 database
	End User Group component

UI Element	Description
	IMS Program component
	J2EE application component
	J2EE application server
	JMS Destination. The endpoint of a JMS message.
	JMS Queue component
	JMS Server
	JMS Topic component
	Mainframe. A generic mainframe component.
	MySQL database
	Other database types
	Oracle database
	SQL Server database
	Sybase database

UI Element	Description
	Web Clients component
	Weblogic Application Server
	WebSphere Application Server
	WebSphere MQ Queue component
	WebSphere MQ Cluster Queue component
<Status indicator>	Configured Health Indicators and KPIs are displayed in the lower right quadrant of the node.
<Link arrow>	The link arrows, each connecting two component icons, indicate the direction of the transaction flow. If the Transaction Topology flow map is set to show data (according to the selections made in the **Link Labels** toolbar menu), then the Transaction Topology flow map displays these metrics, broken down by three transaction state categories, along the edge of each link arrow. The format **<successful>/<late>/<failed>** is used to display the metrics across each transaction state category, for example: Volume: 100/2/1. For more information see "Link Labels" on page 723. **Tooltip:** Hold the pointer over a link arrow to display a tooltip that shows the metrics broken down by transaction state. For more information, see "Tootips" on page 729.

Properties For a Selected Component

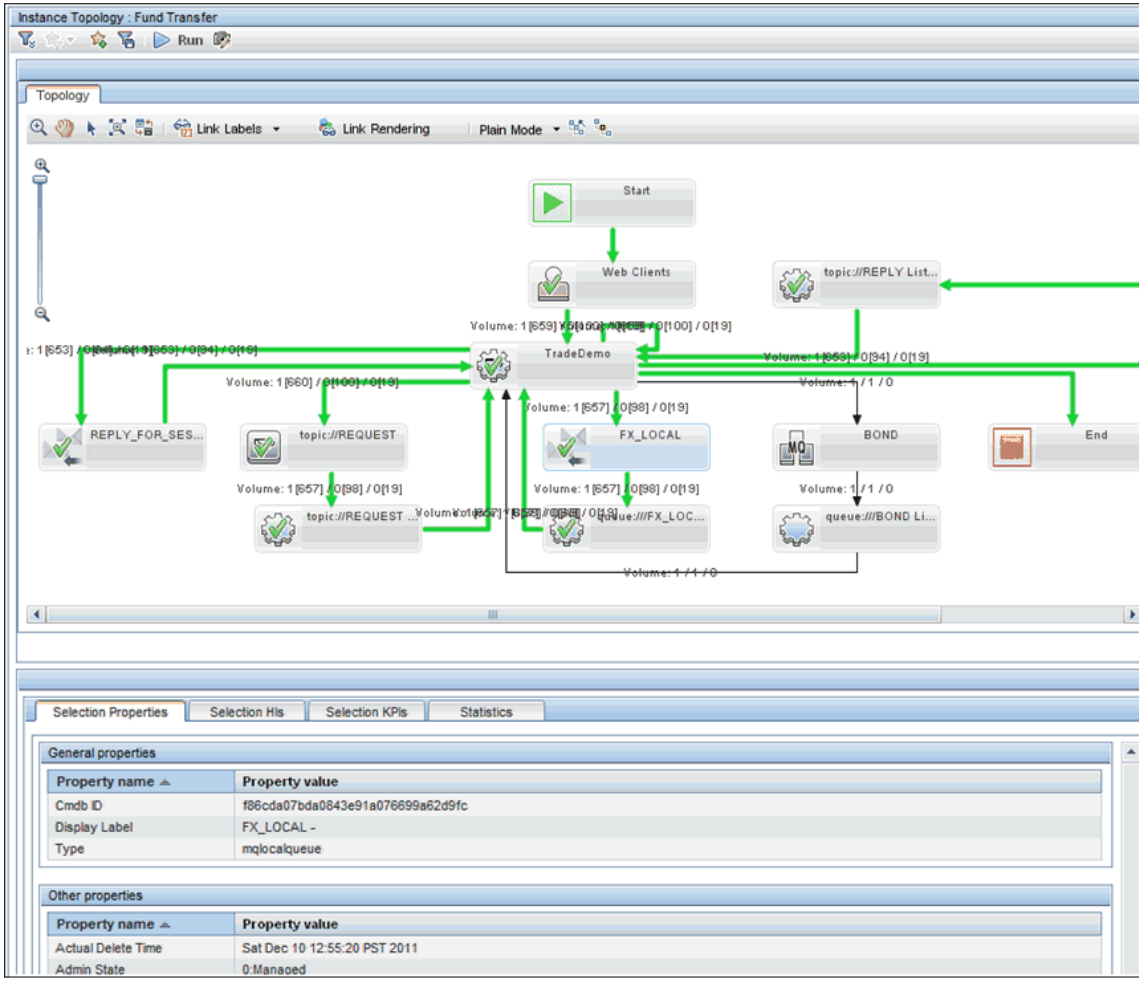
User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
Selection Properties tab	CMDB properties of the selected element.
Selection HIs tab	Health indicators of the selected element.
Selection KPIs tab	Key Performance Indicators of the selected element.
Statistics tab	Applies only to certain connections (links) of RUM transactions.
Transaction HIs tab	Health indicators for the transaction class as a whole.
Transaction KPIs	Key Performance Indicators for the transaction class as a whole.

Instance Topology Page

The Instance Topology page displays a flow map and response time and volume metrics for a business transaction.

The following is an example of the Instance Topology.



To access	➤ From the Transaction Tracking or the Transaction Detail report, click the **Drill to Instance Topology** button.
Important information	The Instance Topology is similar to the Aggregated Topology. It shows metrics for a single Business Transaction instance along with aggregated metrics for the associated business transaction.
Relevant Tasks	➤ "How to View the Instance Topology of a Business Transaction" on page 739
See also	➤ "Transaction Topology Overview" on page 718 ➤ "A Closer Look at the Flow Map" on page 720

Instance Topology Settings

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
<Common report settings>	For details, see "Common Report Elements" in *Reports*.

Instance Topology Content: Flow map

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
<Component icon>	Each flow component (for example, Application, Servlet, MQ Queue, and so on) is represented by an icon. For icon descriptions, see "Aggregated Topology Content: Flow map" on page 746. Check marks appear in the component icons to show which nodes/groups a transaction went through. For details, see "Check Marks for Instance Topologies" on page 737. **Start point**. Shows where the start point is located. This is linked to the starting node with a dotted line. Determined by event timestamps, which may or may not be accurate. **End point**. Shows where the end point is located. This is linked to the ending node with a dotted line. Determined by event timestamps, which may or may not be accurate.
<Link arrow>	The link arrows indicate the direction of the transaction flow between connecting component icons. For details about the link arrows, see "Connecting Arrows" on page 722.

Properties For a Selected Component

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
Selection Properties tab	CMDB properties of the selected element.
Selection HIs tab	Health indicators of the selected element.
Selection KPIs tab	Key Performance Indicators of the selected element.
Statistics tab	Applies only to certain connections (links) of RUM transactions.

Troubleshooting and Limitations

There is a limit on how many instances can be retrieved from the database to present the transaction topology. For TransactionVision, the number of transaction instances retrieved is limited to 100,000.

22

Component Topology

This chapter includes:

Concepts

- Component Topology Overview on page 756
- Static Mode Versus Dynamic Mode on page 758
- A Closer Look at the Graph on page 759

Tasks

- How to View the Component Topology on page 770
- How to Change the Layout of the Graph on page 772
- How to Modify Component Groupings on page 776
- How to Configure Arrow and Node Labels on page 781
- How to Print the Component Topology Graphs on page 787

Reference

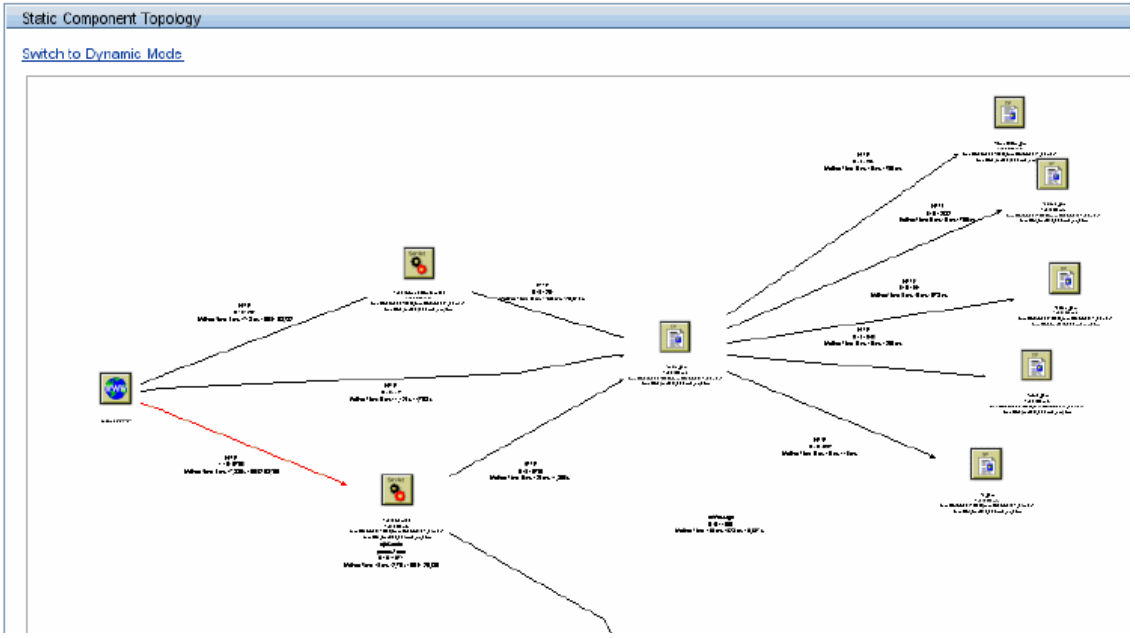
- Component Topology Page on page 792

Troubleshooting and Limitations on page 806

Concepts

Component Topology Overview

The Component Topology provides a graphical representation of all system components from which agents have collected events. These events are correlated and used to draw a visual map of the flow of information throughout the system:

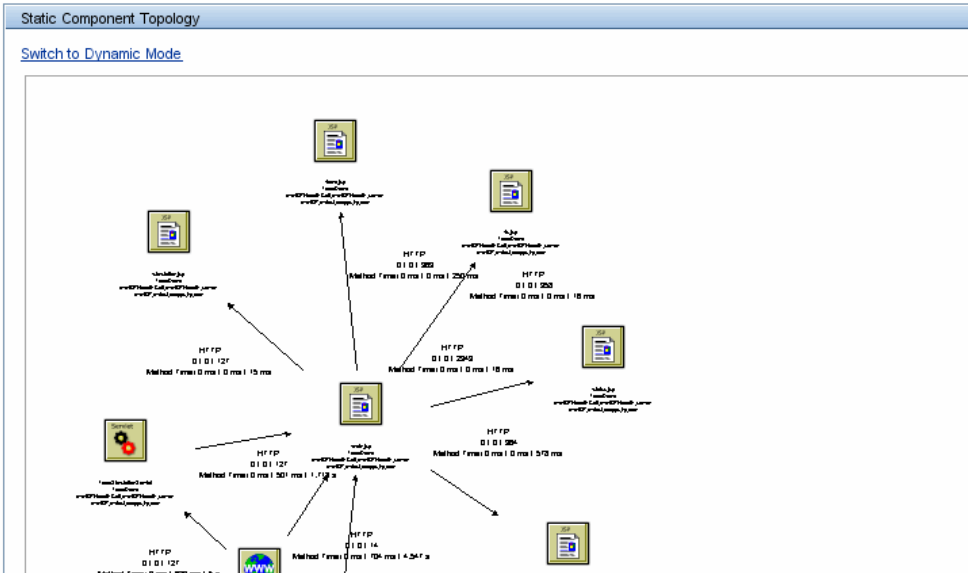


The Component Topology is helpful in determining which components have a bottleneck or on which component the flow has stopped.

Note: Unlike other metrics and data presented in the Transaction Management reports and topologies, these statistics are not aggregated by the transaction. However you can drill in-context to the Component Topology from the Event Analysis report which allows you to see data only for those events related to a business transaction or local transaction instance.

By default, the flow is presented in a hierarchical layout as shown above so that you can easily see the overall flow from component to component.

You can change the layout by symmetrical instead as in the following example.



This layout allows the information on the arrows to be easier to read in some cases.

Static Mode Versus Dynamic Mode

You choose one of two modes for the Component Topology: Static mode or Dynamic mode.

Static mode

In Static mode, the Component Topology graph is created from statistics gathered automatically during event collection. You specify the time range for which you want to view statistics.

Static mode includes data about the relationships between the components. This data is aggregated by component to component relationship. For example, if program A puts 100 messages on JMS queue B, then 100 events of that put operation are collected by the corresponding TransactionVision agent. The Analyzer creates a statistic metric representing the Program A-to-JMS Queue B operation with its count, in this case, 100.

In Static mode, the granularity of data is at the program level or higher, whereas the granularity of data in Dynamic mode is at the program instance level and higher. For more information about granularity of data, see "How to Modify Component Groupings" on page 776.

By default, the Component Topology displays in Static mode. The graph is rendered much faster in Static mode than in Dynamic mode.

The statistics used to support Static mode are stored in a specific cache. Two configuration options are provided for this cache

- See the **Flush Interval** and options in "Configuration Tab, Cache Tab" on page 127.
- See **topologyStateTimeslice** in "Configuring the Caches" on page 87.

Dynamic mode

In Dynamic mode, the Component Topology graph is created based on events returned by a query. Only the correlation data related to those events is shown in the graph. You define the query to be used; the query can choose events of a certain technology, host, program, or other criteria.

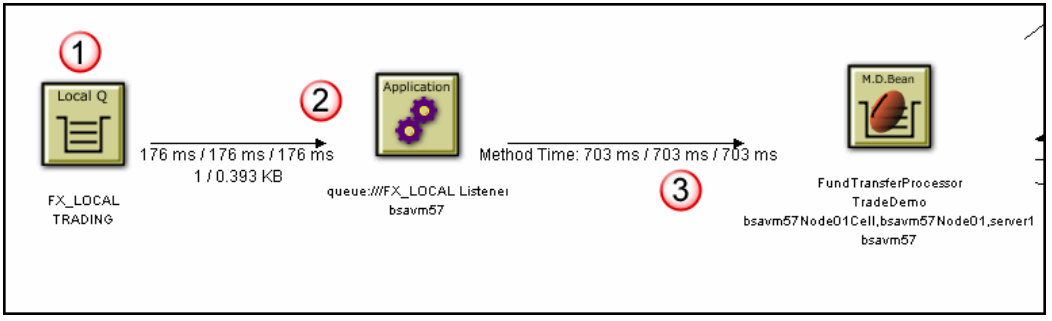
This mode enables you to view very specific information in the graph, but takes longer than Static mode to render.

For information about writing queries, see "Queries" on page 259.

A Closer Look at the Graph

By default, the Component Topology is arranged in a hierarchical layout according to the flow of messages through your system.

The flow starts with the left most component representing the first message in the requested time frame, and flows to other components. In the following example, the web page calls the **TradeServlet** and **TradeSimulationServlets**. These each in turn call the same Java server page.



The following table describes the callouts in this screenshot:

Callout	Description
1	A component on which the first event occurred.
2	Arrows show the relationship between components.
3	Labels on the arrow provide data about the flow such as latency or event delivery status.

This section includes the following topics:

- "Resource Icons" on page 760
- "Start and End Nodes" on page 760
- "Arrows" on page 761

Resource Icons

Resource icons represent system components and appear as nodes in the graph. For a list of the resource icons, see "Resource Icons" on page 802.

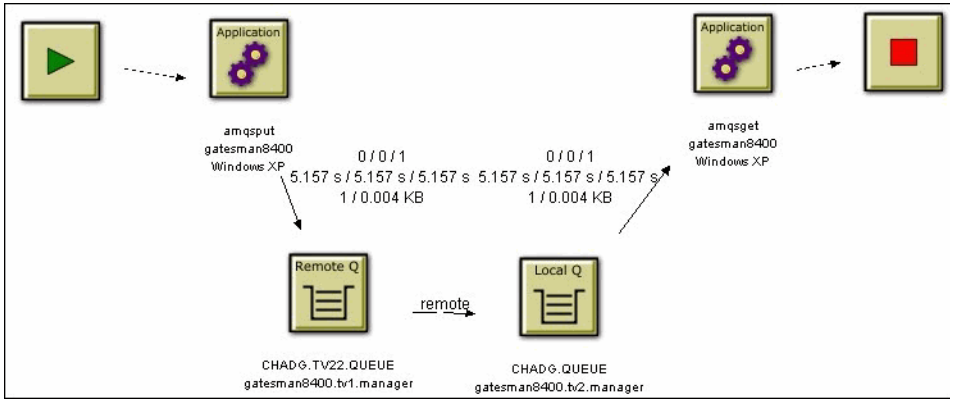
Start and End Nodes

When viewing the Dynamic Component Topology from a business transaction drill down (from the "Transaction Tracking Report" on page 675 or the "Transaction Detail Report" on page 686), you can select the **Edge & Node** > **Show Start/End Nodes** option to show special nodes that show the start and end points to a component topology. This helps you quickly see where the start and end points are located, especially for very complex topologies.

An icon with a **Play** symbol represents the starting point of the component topology and is linked to the actual starting node with a dotted line. Similarly, the end point is linked with a dotted line and is represented by an icon with a **Stop** symbol .

Note: The start and end points are determined by event timestamps, and may or may not be accurate. This is especially true if the graph has more than one start and/or end point.

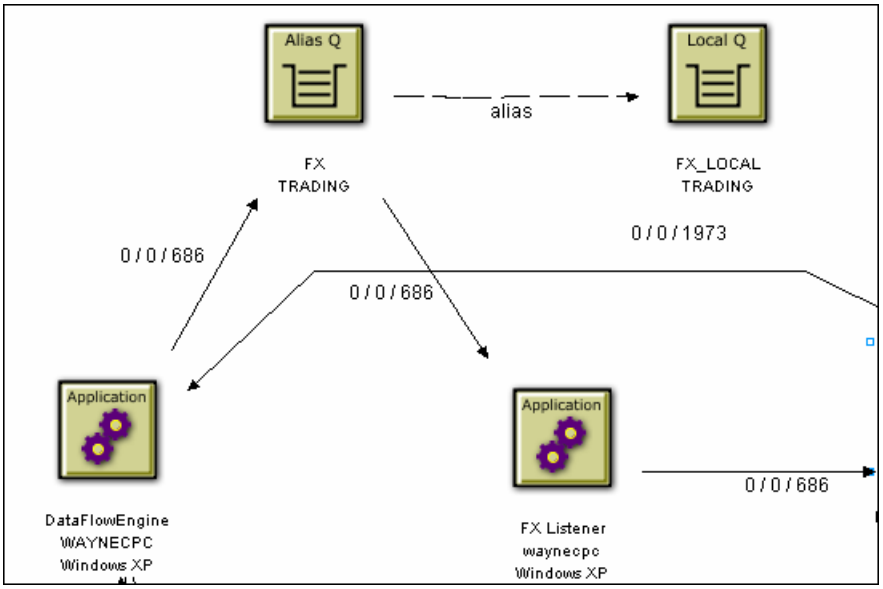
Following is an example of a component topology's start and end nodes:



Arrows

For WebSphere MQ, JMS, and CICS events, solid arrows represent message flows. For servlet, Java user events, and EJB events, arrows represent one component calling another component. For JDBC events, arrows represent access to database resources.

Dashed arrows represent a relationship between components, such as between a remote, alias, or model queue and the local queue it is associated with. In the following example, the DataFlowEngine and FX Listener programs write to and read from the FX queue, respectively. The FX queue is an alias queue for the local FX_LOCAL queue.

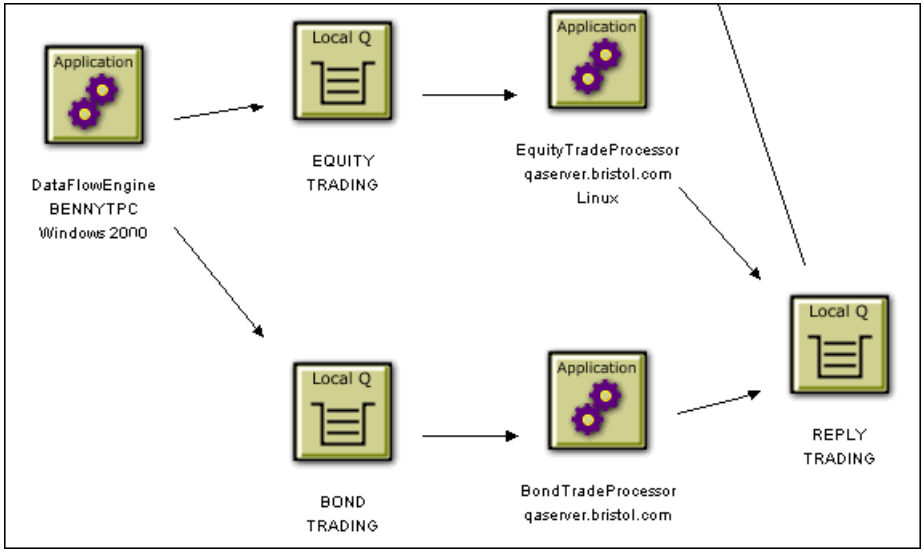


Other arrow labels are dependent on the technology of the following associated components:

- "WebSphere MQ and JMS Events" on page 763
- "CICS Events" on page 764
- "Servlet and EJB Events" on page 765
- "JDBC Events" on page 766
- "WebSphere MQ-IMS Bridge Events" on page 766
- "Java User Events" on page 767
- "IBM WebSphere DataPower Events" on page 768
- "Generic Events" on page 769

WebSphere MQ and JMS Events

For WebSphere MQ and JMS events, an arrow represents message flow. For example, the following diagram shows that the DataFlowEngine application puts messages on the EQUITY and BOND queues of the TRADING queue manager. The EquityTradeProcessor and BondTradeProcessor programs retrieve messages from the EQUITY and BOND queues, respectively, and put a message on REPLY.



When monitoring JMS events from stand-alone Java applications, the Component Layout view displays the top level class name as the program name. The top level class name is the outmost name in the stack trace. For example, the class name in the following stack trace is taken from the last line in the following example:

```
at java.lang.Thread.dumpStack(Thread.java:993)
at Test.<init>(Test.java:29)
at Test.main(Test.java:62)
```

However, this class does not necessarily contain the public static void main(String[] args). Instead, it might be a class derived from java.lang.Thread, which runs in other threads than the main thread.

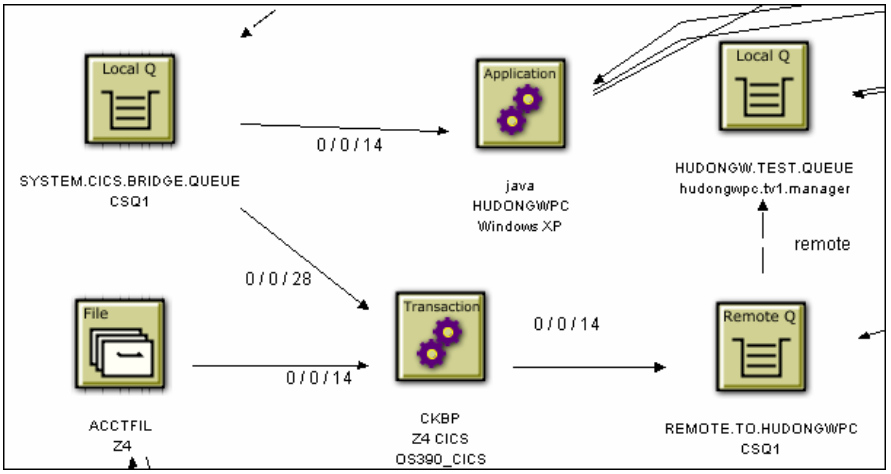
If the correct program name is not displayed, set the com.bristol.tvision.programname system property in the command line when running the stand-alone application. For example, to set the program name to MyProgram, start it as follows:

```
java -Dcom.bristol.tvision.programname=MyProgram ...
```

CICS Events

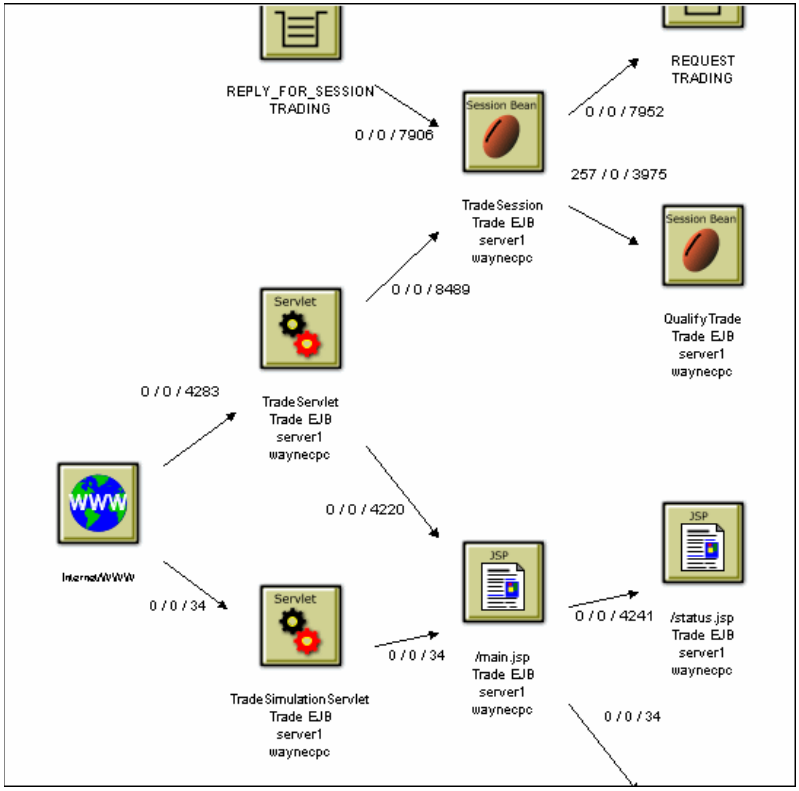
Like WebSphere MQ and JMS events, arrows represent message flow for CICS events. For example, in the following diagram, the CKPB CICS transaction reads from the SYSTEM.CICS.BRIDGE.QUEUE queue and the ACCTFIL file.

It then writes to the REMOTE.TO.HUDONGWPC queue, which is a remote queue for HUDONG.TEST.QUEUE.



Servlet and EJB Events

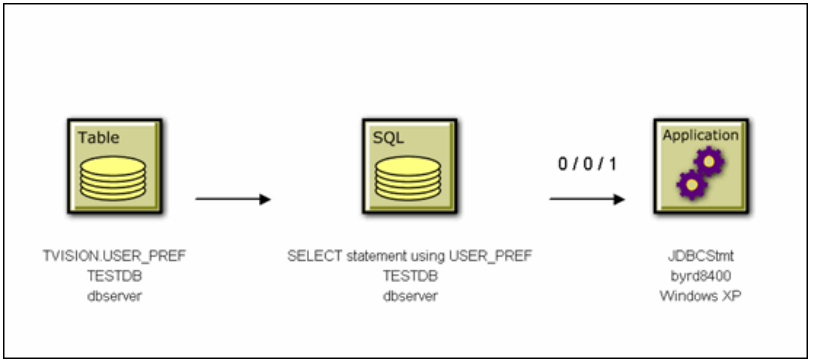
For servlet and EJB events, arrows represent control flow. In the following example, the web page calls the TradeServlet and TradeSimulationServlets. TradeServlet calls the TradeSession EJB, while TradeSimulationServlet calls a Java server page.



JDBC Events

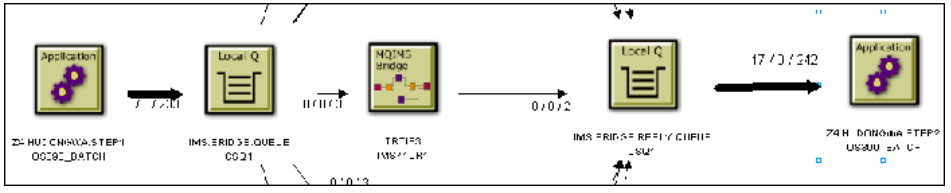
For JDBC events, arrows represent access to database resources. An application has a link to an SQL object, which represents the type of SQL statement and which tables it accesses. Various statistics on this operation are available on this arrow. Additionally, an SQL object has links to database table objects that were accessed by the SQL statement.

In the following example, you see the JDBCStmt application execute an SQL statement that does a SELECT on the table TVISION.USER_PREF.



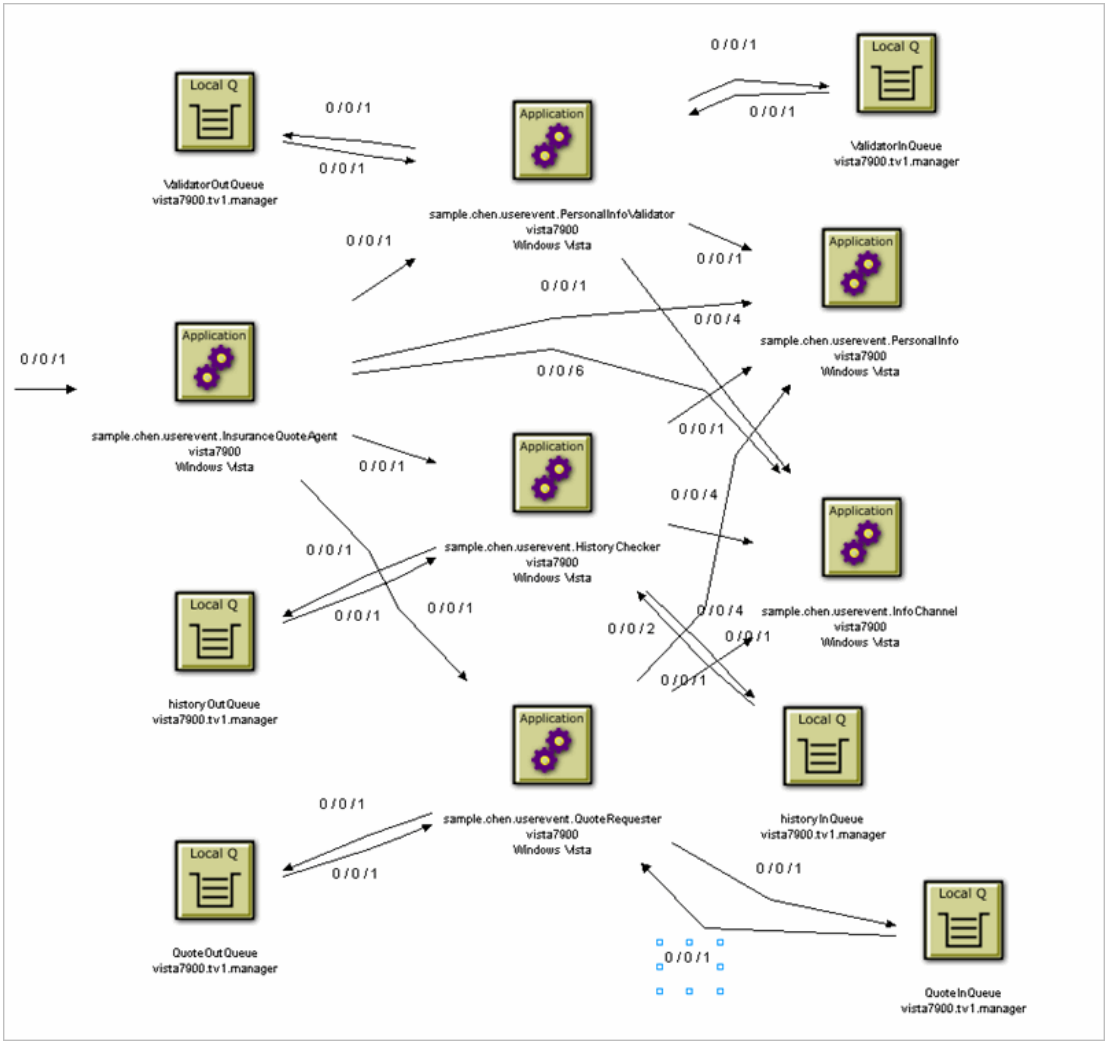
WebSphere MQ-IMS Bridge Events

The following example shows a batch job HUDONGWA with a job step, STEP 1, putting messages onto a MQ-IMS bridge input queue. On receiving these messages on the input queue, the WebSphere MQ-IMS bridge invokes an IMS transaction TRTIB3 in IMS job IMS71CR1. That transaction places reply messages on the IMS.BRIDGE.REPLY.QUEUE, which is read by the job step, STEP 2 of the job HUDONGWA.



Java User Events

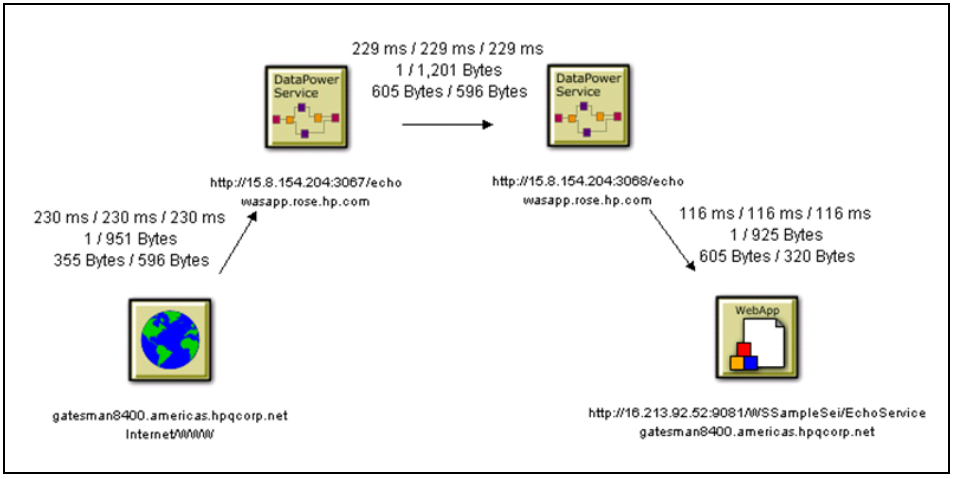
When monitoring Java User events, arrows show calling relationship. For example:



IBM WebSphere DataPower Events

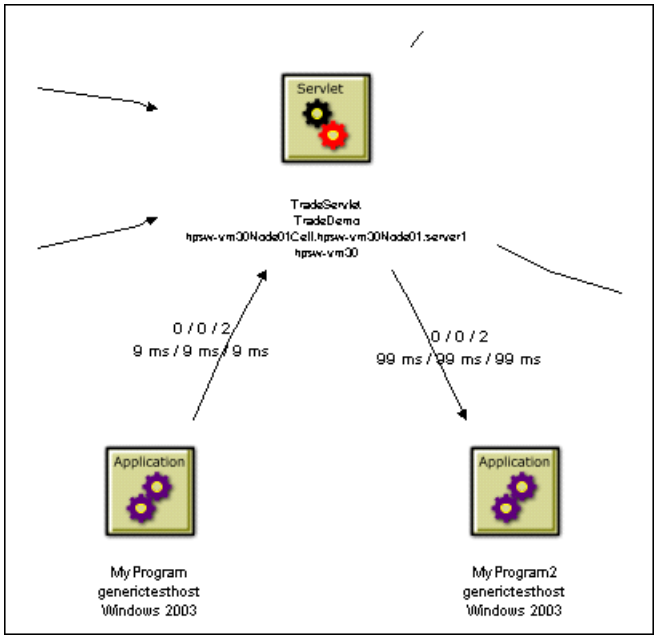
For IBM WebSphere DataPower events, edges represent control flow. In the following example, a web client calls a DataPower service which calls another DataPower service. The second DataPower service then calls a back-end web application.

The first row in the edge label represents Min/Max/Avg Latency. The second row shows the volume count and total bytes passed (including both request and response). The third row shows Request/Response byte counts separated.



Generic Events

For generic events, arrows represent calling relationships. In the following example, MyProgram calls TradeServlet and TradeServlet calls MyProgram2.



For information about implementing generic events in your application, see the *HP TransactionVision Advanced Customization Guide* PDF.

Tasks

How to View the Component Topology

This task includes the following steps:

- "Determine the View Mode You Need" on page 770
- "View the graph in Static Mode" on page 770
- "View the graph in Dynamic Mode" on page 771

Determine the View Mode You Need

By default, the Component Topology is created in static mode. Review the information "Static Mode Versus Dynamic Mode" on page 758 to determine which mode you need.

View the graph in Static Mode

1. Select **Applications** > **Transaction Management** > **Transaction Topology** > **Component Topology.**

   If the graph displays in Dynamic mode, click **Switch to Static Mode**.

   The Graph displays based on the last time period that was specified.

2. Change the time period as desired:

   - To display all events, set the **Show all statistics** check box and click **Run**.
   - To display events from a predefined time period, choose a time period from the **Set Time to** drop-down list and click **Run**.

➤ To display a custom time period, choose Custom from the **Set Time to** drop-down list, enter the desired start and end dates and times in the **Display Static Data from** and **to** fields and then click **Run**.

Note: For custom time periods, the time slice interval in effect for the Analyzer is the smallest increment of time period in which you can view your data in static mode. See "Configuring the Caches" on page 87.

View the graph in Dynamic Mode

1 Select **Applications** > **Transaction Management** > **Transaction Topology** > **Component Topology.**

If the graph displays in Static mode, click **Switch to Dynamic Mode**.

The Graph displays based on the last query that was specified or if this is the initial viewing, based on the All Events built-in query.

2 To specify a different query, choose the query name from the Query drop-down list and click **Run**.

The list includes all the existing queries for which you have permission to view.

For information about creating or modifying queries, see Chapter 6, "Queries."

How to Change the Layout of the Graph

The layout of the graph is highly customizable. The following sections describe the types of layout customizations you can perform.

This task includes the following optional steps:

- "Move Components" on page 772
- "Show/Hide the Status Bar" on page 772
- "Adjust the Zoom Level" on page 773
- "Set Background Color" on page 775
- "Set White Space Options" on page 775
- "Spacing Model" on page 775
- "JDBC" on page 779

Move Components

To move any component in the graph, select the component in the view area and drag it to the desired location. The arrows automatically adjusts to and from the component.

Show/Hide the Status Bar

Use the **Show/Hide** arrow buttons to show, hide, or resize the status message pane.

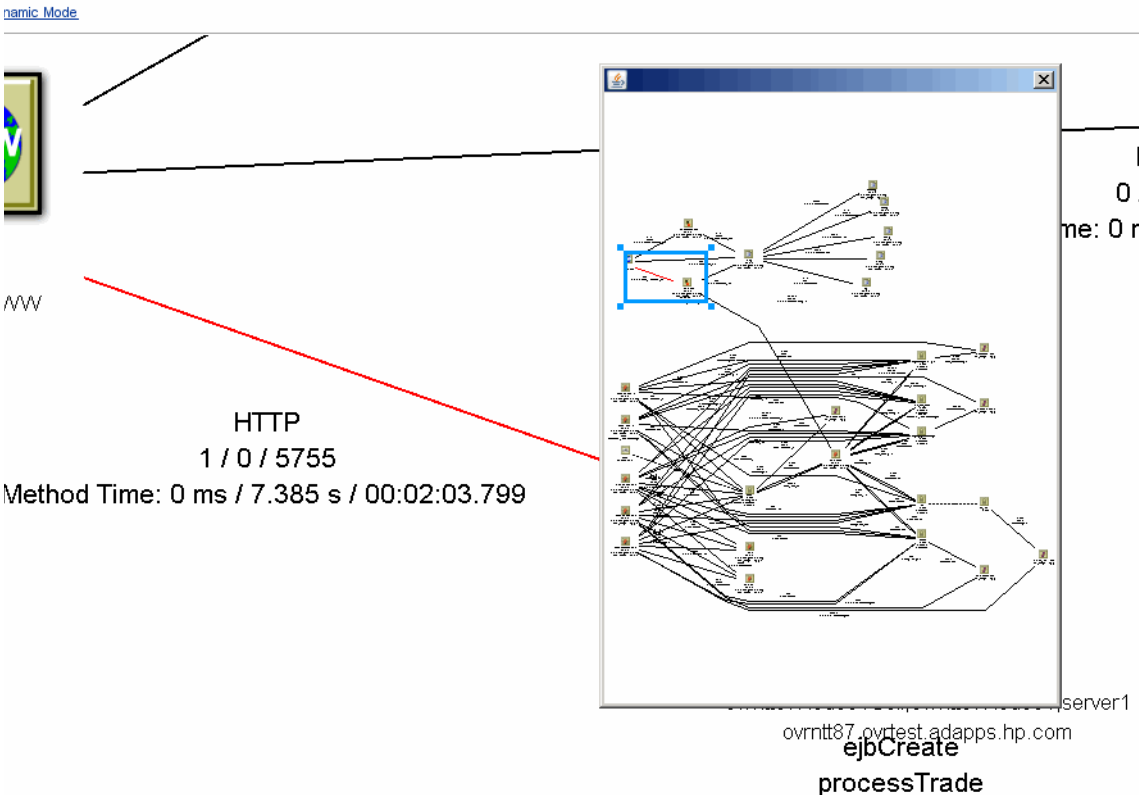
To hide the status bar, click the down arrow button. To show a hidden status bar, click the up arrow button. To resize the status bar, point the cursor to either arrow button so that the cursor changes to a double arrow and drag.

Adjust the Zoom Level

By default, the Component Topology is sized to fit in the view window. There are several ways to adjust the zoom level:

- Choose the **Zoom** > **Zoom** > **Zoom In** menu item to increase the zoom level.
- Choose the **Zoom** > **Zoom** > **Zoom Out** menu item to decrease the zoom level.
- Click the **Selective Zoom** icon or choose **Zoom** > **Selective Zoom** to change to selective zoom mode. The cursor changes to the selective zoom cursor. Click on the view area and then drag the cursor to select an area to zoom. TransactionVision adjusts the zoom level so that the selected area fills the view area. Click the **Selection** icon to return to selection mode.
- Click the **Interactive Zoom** icon or choose **Zoom** > **Interactive Zoom** to change to interactive zoom mode. The cursor changes to the interactive zoom cursor. Click in the view area and drag the cursor to the top or left to decrease the zoom level or drag to the bottom or right to increase the zoom level. Click the **Selection mode** icon to return to selection mode.
- Click the **Zoom Fit Window** icon or choose **Zoom** > **Zoom** > **Fit Window** to adjust the zoom level so that the entire graph appears in the view window.
- Enter a custom zoom percentage or choose a predefined zoom percentage from the zoom percentage list.
- Choose the **Zoom** > **Show Overview Window** menu item to open a small secondary window that contains a read-only view of the view window. Change the view in the view window by moving the blue selection box in the Overview window. This allows you to see in-context areas of the graph.

Following is an example of the Overview Window:



- To resize the selection box, click one of the corner handles and drag your cursor to the desired size. Note that the selection box maintains a constant aspect ratio to match the size of the active view window.
- To move the selection box, click inside the selection box and drag your cursor to the desired location.
- To create a new selection box, click outside of the current selection box and drag your cursor to the desired size.

Note: Any time the Component Topology is refreshed, the zoom level reverts to Fit Window.

Set Background Color

You can specify the background color of the Component Topology graph. See "Layout Menu" on page 797.

Set White Space Options

You can specify options relating to the spring model of the graph. See "Layout Properties Dialog, General Properties Tab" on page 799.

Spacing Model

You can specify options that determine the amount of white space around nodes in the graph. Two spacing models are available:

- **Proportional Spacing.** The amount of white space around the perimeter of an object is based on the area of the object. For example, if a component's width and height increase, the white space between the component and an adjacent component also increases. The margins of the graph can be controlled through the Graph Margin Spacing fields. You can set independent proportional spacings for the left, right, bottom, and top sides of the graph.
- **Constant Spacing.** A constant space is allocated between like objects in a graph. For example, in hierarchical layout a fixed distance is set between each pair of adjacent components within a level, regardless of the width and height of the nodes. The margins of the graph can be controlled through the Graph Margin Spacing fields below. You can set independent constant spacing values for the left, right, bottom, and top sides of the graph.

Note: Additional spacing controls are available on the Layout Properties dialog.

Label Positions

This option allows you to tailor the automatic positioning of arrow labels. See "Layout Properties Dialog, General Properties Tab" on page 799.

How to Modify Component Groupings

Changing component groupings changes the granularity of the information displayed in the view, depending on the level of information you need. For example, viewing each process thread as a separate component provides very fine granularity, but can make it difficult to spot an information flow problem between programs.

Note that proxy objects do not support grouping by thread, process, transaction name, transaction ID, TCB, and PCB. Selecting these grouping criteria have no effect on proxy objects. For more information about proxy objects and the proxy agent, see the *HP TransactionVision Deployment Guide* PDF.

Select the grouping options from the Component Topology **Grouping** menu to configure the display granularity.

This task describes how to modify grouping in the following components:

- "Queues and Queue Managers" on page 777
- "Distributed Platforms" on page 777
- "z/OS CICS" on page 778
- "z/OS Batch" on page 778
- "z/OS IMS" on page 778
- "z/OS IMS Bridge" on page 779
- "Servlet/JSP/EJB" on page 779
- "JDBC" on page 779
- "Web Client" on page 780
- "Show Publish-Subscribe Topic" on page 780
- "Group Method Calls On The Same EJB" on page 780

Queues and Queue Managers

Select one of the following **Queue Grouping Criteria** menu items:

- **WebSphere MQ Object.** Shows each WebSphere MQ object as a separate component. This is the default.
- **WebSphere MQ Resolved Object.** Shows only resolved WebSphere MQ objects. Alias and remote queues are combined into their resolved queue as one node. The resolved queue node label contains all alias and remote queue names (in parenthesis) that have been combined into the node. Note, remote queue names include both queue manager and queue names.
- **Queue Manager.** Shows all objects on a single queue manager as a single component.
- **Show Model Queue Resolution.** Turns model queue resolution on/off. A model queue is a template of a queue definition that is used when creating a dynamic queue. When you call MQOPEN on a model queue, the queue manager creates a temporary or permanent local dynamic queue using the attributes of the model queue. Select this menu item to show model queues in the Component Topology.

The arrows between a model queue node and its model queue definition, remote queue definition, alias queue definition, or cluster queue node representation are displayed as dashed arrows because they represent a relationship rather than a message flow.

Distributed Platforms

For UNIX and Windows NT hosts, select one of the following **Distributed Program Grouping Criteria** menu items:

- **Program Name.** Displays each program name on all hosts as a single component.
- **Host**. Displays all programs on a single host as a single component.
- **Program Name/Host Combination.** Shows the default component grouping.

z/OS CICS

For z/OS CICS hosts, select one of the following **z/OS CICS Program Grouping Criteria** menu items:

- **Transaction ID/Region/Host Combination.** Shows the default component grouping.
- **Transaction ID/Host Combination**. Shows the transaction and host combination for all regions as a single component.
- **Region/Host Combination.** Shows all transaction IDs for a single region on a host as a single component.
- **Host.** Shows all transaction IDs for all regions on a single host as a single component.

z/OS Batch

For z/OS Batch hosts, choose one of the following **z/OS Batch Program Grouping Criteria** menu items:

- **Individual Job Step.** Shows each job step as a separate component.
- **Individual Job.** Shows each job as a separate component.
- **Host.** Shows all TCBs on a single host as a single component.
- **Host/Job Name/Step Name Combination.** Shows the job name, job step, and host combination as a single component. This is the default.
- **Host/Job Name Combination.** Shows all jobs on a host as a single component.

z/OS IMS

For z/OS IMS hosts, select one of the following **z/OS IMS Program Grouping Criteria** menu items:

- **Individual PSB Name.** Shows each PSB as a separate component. This is the default.
- **IMS Transaction Name.** Shows each transaction as a separate component.
- **IMS Region Identifier.** Shows each region as a separate component.

- **IMS Identifier.** Shows each IMS identifier as a separate component.
- **Job Name.** Shows each job name as a separate component.
- **IMS Type.** Shows each IMS type as a separate component.

z/OS IMS Bridge

For z/OS IMS hosts, select one of the following **z/OS IMS Bridge Grouping Criteria** menu items:

- **Transaction Name.** Shows each transaction name as a separate component. This is the default.
- **Job Name.** Shows each job name as a separate component.

Servlet/JSP/EJB

For Servlet/JMS/EJB events, select one of the following **Servlet/JSP Grouping Criteria** menu items:

- **Servlet/JSP/EJB.** Shows servlet, JSP, and EJB icons. This is the default.
- **Web Application.** Shows web application icons.
- **Application Server.** Shows WebSphere Application Server icons.
- **Host.** Shows all programs on a single host as a single component.

JDBC

For JDBC events, select one of the following **Grouping** > **Database Grouping Criteria** menu items:

- **Database**. Groups JDBC components by which database they access. This is the default.
- **Database object.** Groups JDBC components by SQL statements, and shows the tables on which the SQL statements operate.
- **Sql Object.** Groups JDBC components by SQL statements.

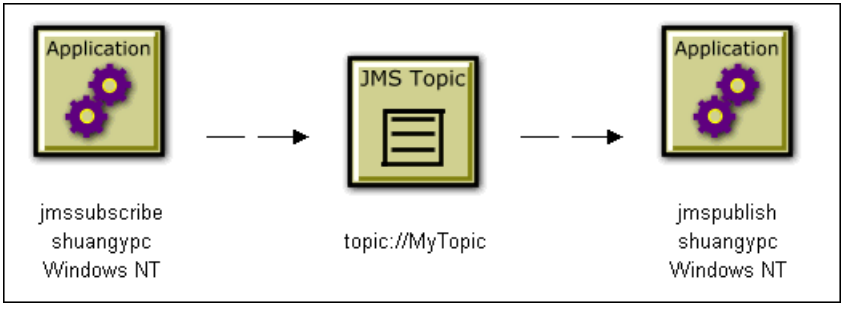
Web Client

For Servlet and DataPower events, select one of the following Web Client Grouping Criteria menu items:

- **Client Host**. Shows each individual client host as separate components. To display the statistics at this level, the Analyzer must be configured to collect Client Host System Model Objects. See "Configuration Tab, Events Tab" on page 130.
- **Internet/Intranet**. Shows all Web Clients as a single component.

Show Publish-Subscribe Topic

TransactionVision is able to correlate these publish-subscribe topics into a single transaction as follows:



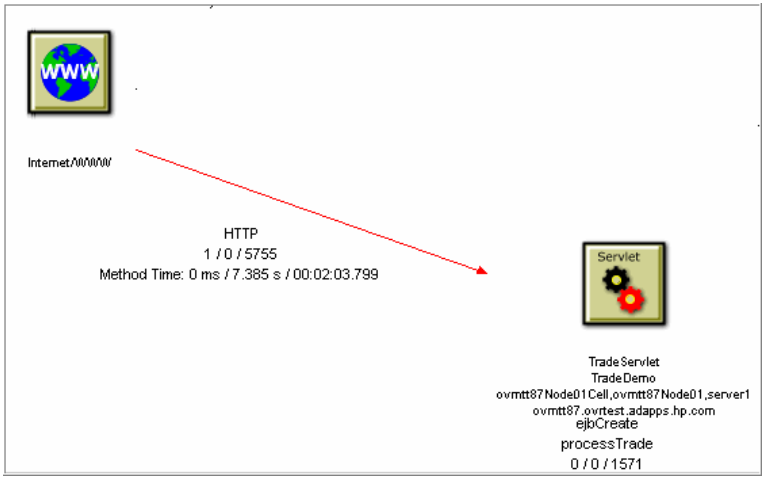
You can change the JMS icon from the underlying queue to topic. Choose **Grouping** > **Show Publish-Subscribe Topic**.

Group Method Calls On The Same EJB

To group similar method calls on the same EJB as a single arrow, choose **Grouping** > **Group Method Calls On the Same EJB**. Otherwise, an arrow is drawn on the graph to represent each method call between EJB beans.

How to Configure Arrow and Node Labels

Labels are the text that appear on an arrow or next to a node:



You can configure arrow and node labels to provide more information about the components and events they represent as follows:

- "Remove/Show Labels" on page 781
- "Show Start/End Nodes" on page 782
- "Message Count/Byte Count Data" on page 782
- "Error, Warning, and Success Count Data" on page 783
- "Latency Times" on page 784
- "Arrow (Edge) Connection Name" on page 785
- "EJB Methods" on page 785
- "JDBC Methods" on page 786

Remove/Show Labels

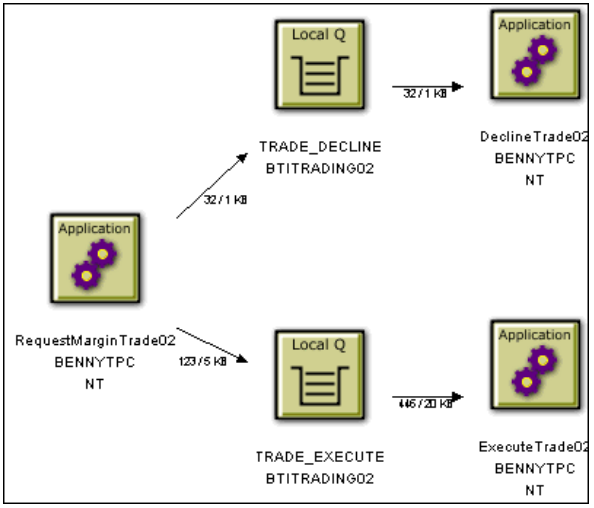
Use the **Edge & Node** menu to toggle the display of various types of labels that appear on the arrows and nodes. Each menu option shows or hides the label, and the sub-menu options specify the label details.

Show Start/End Nodes

When available, this option turns on and off display of the Start and End Nodes. For details, see "Start and End Nodes" on page 760.

Message Count/Byte Count Data

This label indicates the total number of successful message calls between the resources, along with the number of bytes transferred. Note that MQIMS_BRIDGE_ENTRY and MQIMS_BRIDGE_EXIT calls are not included in the message count/byte count.



To specify a unit of measure for the byte count, choose one of the following from the **Edge & Node** > **Show Byte Counts** menu:

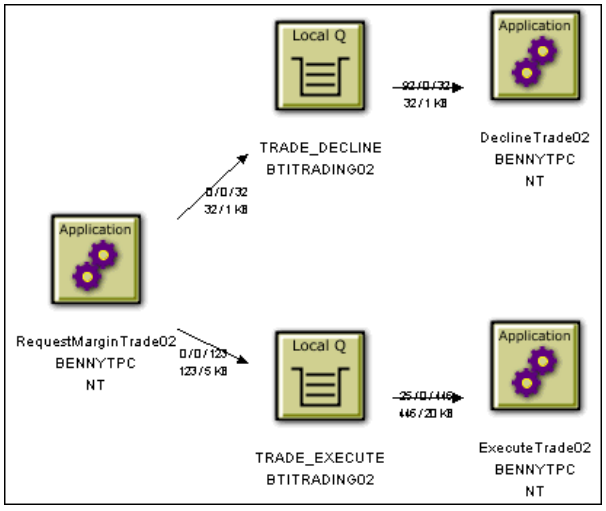
- In MegaBytes
- In KiloBytes
- In Bytes

This menu item is only enabled if the message count/byte count is displayed.

Error, Warning, and Success Count Data

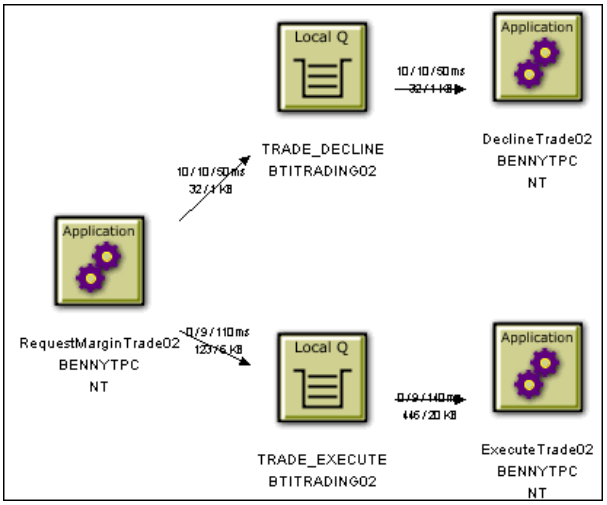
This label indicates the number of error events, warning events, and successful events between components. Events include the following:

- All WebSphere MQ MQGET, MQPUT, MQPUT1, MQINQ, MQSET, MQOPEN, MQIMS_BRIDGE_ENTRY, and MQIMS_BRIDGE_EXIT calls.
- All JMS receive, receiveNoWait, OnMessage, OnException, createReceiver, createSubscriber, createBrowser, publish, send, createSender, createPublisher, and nextElement calls.
- All servlet HTTP_PUT, HTTP_GET, and HTTP_POST requests.



Latency Times

This label indicates the minimum, average, and maximum transaction times for each connection. Use this information to identify potential performance bottlenecks. Transaction times are defined as the time from when a message is put onto the queue and when it was pulled off that queue. For transactions involving a proxy object, TransactionVision shows a latency of N/A.

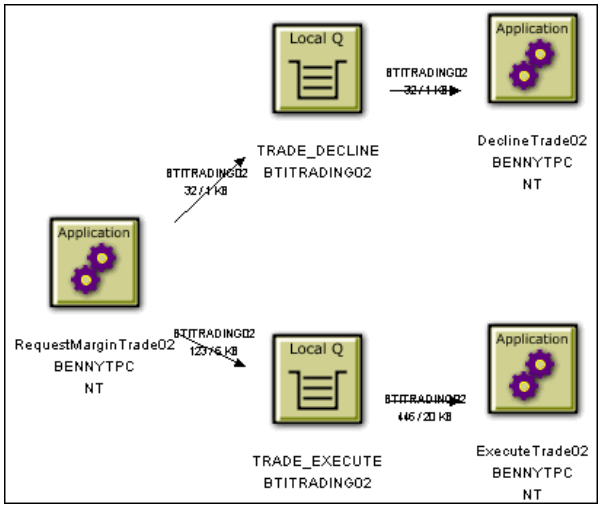


Note: For EJB and servlet events, the time on the arrow pointing to a bean represents the time spent in that bean's or servlet's methods; not a latency time.

By default, latency times are calculated using a resolution of 1/100 seconds. To modify the latency resolution, change the value of the Latency Resolution setting for the Analyzer. For more information, see "Configuration Tab, Events Tab" on page 130.

Arrow (Edge) Connection Name

Arrow labels indicate the connection associated with the arrow.



Click **Edge Label** to toggle the display of the connection name.

EJB Methods

This label provides details about the EJB Methods. Select the following menu items to display or hide the method information:

- **Show Standard EJB Methods.** Includes standard EJB methods in the arrow labels showing the error/warning/success count.
- **Show Access Methods.** Includes access methods in the arrow labels showing the error/warning/success count.
- **Show Bean Relationships.** Indicates EJB relationships.

JDBC Methods

This label provides details about the JDBC Methods. Select the following **JDBC Statistics** menu items to display or hide the method information:

- **Cursor Iteration/Get Count.** Indicates the number of result rows that were retrieved using this statement, and the number of calls made to the JDBC resultset to get results from the row.
- **Min/Avg/Max Execute.** Displays statistics showing timing of how long it took to execute this SQL statement.
- **Min/Avg/Max Cursor time.** Displays statistics showing the total time spent iterating through result rows and retrieving data from the JDBC resultset.
- **Avg Time (Cursor, Execute) per statement/Total Statement time.** Displays statistics showing the combined average time and total time spent on operations using this SQL statement.
- **Commit time (Database grouping only).** Displays statistics showing the commit time. Note that this aggregate data is not available on individual statements, so this statistic only displays when viewing your topology with the Database grouping level enabled.
- **Total time (Cursor, Execute, commit)(Database grouping only).** Displays statistics showing the total time spent in JDBC calls, including Cursor time, Execute time and commit time. Note that this aggregate data is not available on individual statements, so this statistic only displays when viewing your topology with the Database grouping level enabled.

How to Print the Component Topology Graphs

To print the Component Topology graph using the current print setup, click the Print icon and choose **Print Graph**. TransactionVision invokes the standard printing mechanism for your platform.

Note: To print in landscape mode, use the TransactionVision Page Setup dialog to set page orientation. The Windows Print dialog has no effect on orientation. For more information about this dialog, see "Page Setup" on page 790.

When you print a graph, a Java warning appears asking permission to print. This message is generated by the Java security feature to prevent unauthorized use of printers. The first time you print in a session, it appears multiple times; it appears once after the first time you print. To disable this message, add the following lines in the **java.policy** file in your home directory on the client computer:

```
grant {
   permission java.lang.RuntimePermission "queuePrintJob";
};
```

This task includes these optional steps:

- "Print Preview" on page 788
- "Print Setup" on page 789
- "Page Setup" on page 790
- "Caption Font" on page 790
- "Border Color" on page 791

Print Preview

To display a print preview, choose the > **Print Preview** menu item. The Print Preview window opens so you can see how changes in print setup options affect the graph.

Note: Do not change the printing orientation options on the **Print Setup** > **Page Setup** dialog through the Print Preview window. Instead, use the TransactionVision **Page Setup** dialog to change the orientation options. For more information about this dialog, see "Page Setup" on page 790.

In the Print Preview window, you can zoom in on a preview image by moving the mouse over it and clicking.

You can also use the following buttons to control the appearance of the print preview:

Button	Description
Print	Invokes the standard printing mechanism for your platform.
Print Setup	Opens the Print Setup dialog for setting printing options.
Zoom In	Increases the zoom level for the preview image.
Zoom Out	Decreases the zoom level for the previous image.
Zoom Percentage	Shows the current zoom percentage. You may select a pre-defined zoom percentage from the list or enter a custom zoom percentage.
Fit in Window	Scales the preview image to display the complete image in the preview window.
Close	Closes the Print Preview window.

Print Setup

To customize the printed graph, choose the > **Print Setup** menu item. The Print Setup dialog opens.

Set the following options, then click **OK** to apply your changes. Click **Cancel** to close the dialog without applying your changes, **Default** to revert to default values, or **Page Setup** to display the Page Setup dialog for selecting paper size, orientation, and margins.

Option	Description
Print	Select the part of the graph you want to print. You can print the entire graph, only the part of the graph currently visible in the view window, or only the selected graph component.
Scale By	Set options related to page size. Select Pages and specify the number of page rows and columns to scale the image to fit the specified page size. The image is not scaled if Actual Size is selected. Choose Zoom Level to scale the image to the zoom level specified in the view.
Print Caption	Select to print the page caption in the Caption field.
Caption	Enter text for the page caption. You must select Print Caption to enable caption printing. To select the font for the caption, click **Font** to display the Choose Font dialog.
Position	Select the position on the page for the caption to be printed.
Margins	Set the distances in inches between the edge of the paper and the edge of the graph. Set margins for the top, bottom, left, and right edges.
Print Page Numbers	Select to print page numbers on each page when printing multiple pages. Page numbers are printed just outside the crop marks.
Print Crop Marks	Select to print crop marks when printing multiple pages. Crop marks are the lines drawn on the margins of pages that connect to other pages.

Option	Description
Print Border	Select to print a border around the graph. Click **Color** to open the Choose Color dialog for setting the border color.
Print Background	Select to print the background color. To set the background color, choose the Layout > Background Color menu item on the Component Topology or Transaction Analysis view.
Print Grid	Select to print a background grid.

Page Setup

Click **Page Setup** on the Print Setup dialog to open the Page Setup dialog. Set the following options, then click **OK** to activate your changes or **Cancel** to close the dialog without applying the changes.

Option	Description
Paper Size	Select a page style from the list.
Paper Source	Select a source from the list.
Margins	Set the distance between the text and the edge of the printed page (in inches). Set margins for the top, bottom, left, and right edges.
Orientation	Select whether to use Portrait or Landscape for setting up the page. Portrait is default (the height of the page is greater than the width). For Landscape, the width of the page is greater than the height.
Sample	Displays a preview of the selected settings.

Caption Font

Click **Font** on the Print Setup dialog to open the Choose Font dialog.

Use this dialog to select the font to use for the captions when printing a Component Topology graph.

Select a font family, text attributes, and font size from the drop-down lists. The sample window shows how text on the graphic looks.

Click **OK** to make your changes active or **Cancel** to close the dialog without activating your changes.

Border Color

Click **Color** on the Print Setup dialog to open the Choose Color dialog. Use this dialog to set the border color for the printed Component Topology graph.

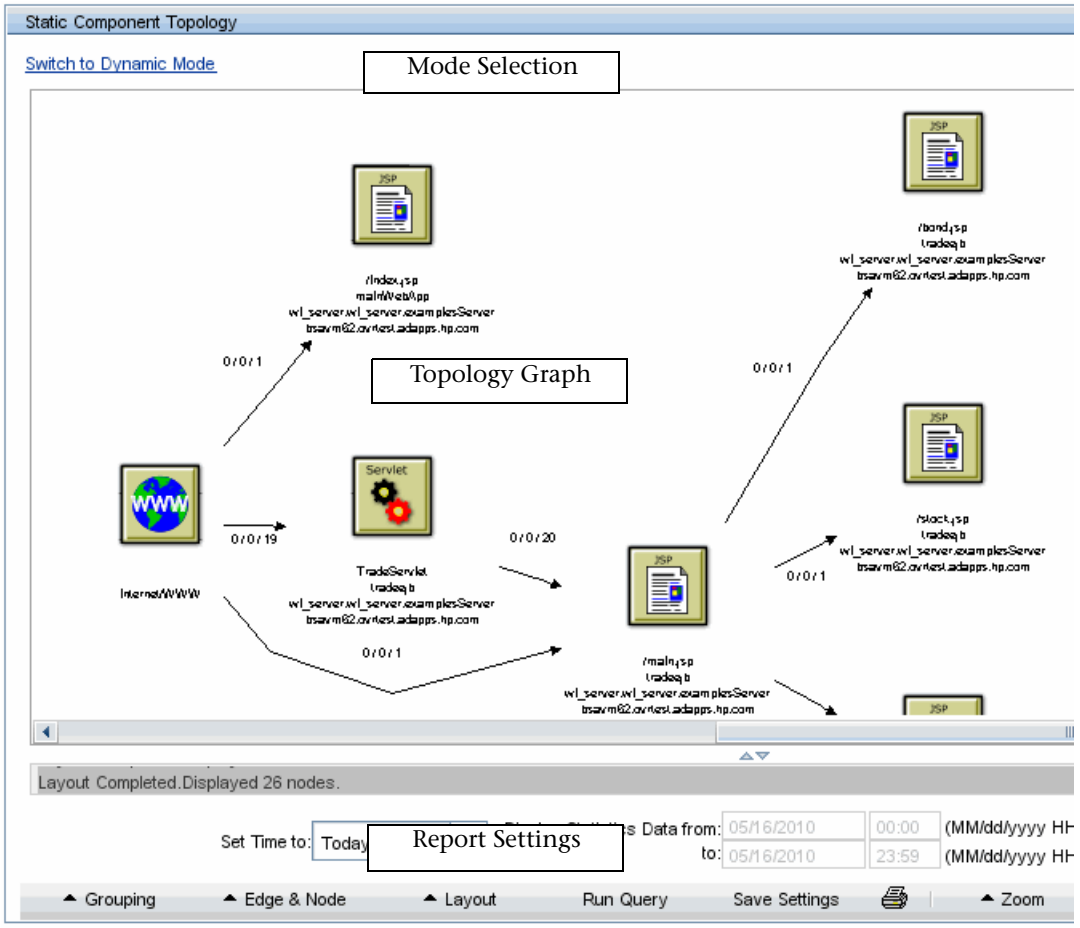
Set the border color in the same way you set the background color for the graph. For more information, see "A Closer Look at the Graph" on page 759.

Reference

Component Topology Page

The Component Topology page provides a graphical representation of all system components from which agents have collected events, and includes data about the relationships between the components.

The following is an example of the Component Topology in Static mode:



To access	Select **Applications** > **Transaction Management** > **Transaction Topology** > **Component Topology**
Important information	➤ This topology is based on data from TransactionVision only. ➤ There are two view modes: static and dynamic. See "Static Mode Versus Dynamic Mode" on page 758 ➤ The nodes in the graph are resource icons that represent system components. See "Resource Icons" on page 802. ➤ Edges refer to the arrows that show the interaction between two components.
Relevant tasks	➤ "How to View the Component Topology" on page 770 ➤ "How to Change the Layout of the Graph" on page 772 ➤ "How to Modify Component Groupings" on page 776 ➤ "How to Configure Arrow and Node Labels" on page 781 ➤ "How to Print the Component Topology Graphs" on page 787
See also	➤ "Component Topology Overview" on page 756 ➤ "Static Mode Versus Dynamic Mode" on page 758 ➤ "A Closer Look at the Graph" on page 759

User interface elements are described below (unlabeled elements are shown in angle brackets):

UI Element	Description
	Print the current Component Topology. See "How to Print the Component Topology Graphs" on page 787 for details.
	Enable selection mode. Allows you to drag a node.

UI Element	Description
	Enable interactive zoom mode. The cursor changes to the interactive zoom cursor. Click in the view area and drag the cursor to the top or left to decrease the zoom level or drag to the bottom or right to increase the zoom level. Click the selection mode button to return to selection mode.
	Enable selective zoom mode. The cursor changes to the selective zoom cursor. Click on the view area and then drag the cursor to select an area to zoom. The zoom level is adjusted o that the selected area fills the view area. Click the selection mode button to return to selection mode.
	Automatically adjust the zoom level so that the entire graph appears in the view window.
<zoom > drop-down	Select a zoom percentage or enter a custom zoom percentage to adjust the zoom level of the Component Topology graph.
Display Statistics Data from: <date and time> to: <date and time>	The time period on which the graph is based.
Edge & Node menu	See "Edge and Node Menu" on page 797.
Grouping menu	Menu options to define component groupings for the Component Topology graph. See "How to Modify Component Groupings" on page 776 for descriptions of each menu item.
Layout menu	See "Layout Menu" on page 797.
Run	Click to refresh a view with information about transactions that have completed since the graph was opened.
Save Settings	Click to save changes and make the current settings the default settings for your user ID.

UI Element	Description
Set Time to:	Select one of the following predefined time periods: ➤ Today ➤ Yesterday ➤ Past Week ➤ This Month ➤ Last 48 hours of data ➤ Custom. Choose a time period. The time period must be greater than or equal to the current statistics cache time slice setting. See "Configuring the Caches" on page 87. The date and time fields are automatically populated with the corresponding information.
Show all statistics	Check to use all events and click **Run** to view all statistics. Clear if you want to view a pre-defined or custom time period.
Zoom menu	Menu options to adjust the zoom level of the Component Analysis Topology graph. By default, the Component Topology graph is sized to fit in the view window. See "Tasks" on page 770 for descriptions of each menu item.

Edge and Node Menu

UI Element	Description
Edge Widths	See "How to Configure Arrow and Node Labels" on page 781.
Edge Color Coding	
Error/Warning/ Success Count	
Min/Avg/Max Latency Times	
Edge Label	
Show Byte Counts	
EJB Methods	
JDBC Statistics	
Node Label	
Show Start/End Nodes	See "Start and End Nodes" on page 760.

Layout Menu

UI Element	Description
Hierarchical Style	Components are arranged in a hierarchical organization based on the flow of messages through your system.
Symmetrical Style	Components are distributed evenly to minimize edge crossings.

UI Element	Description
Layout Properties	➤ Layout Properties appear on the following tabs: ➤ "Layout Properties Dialog, General Properties Tab" on page 799 ➤ "Layout Properties Dialog, Disconnected Tab" on page 799 ➤ "Layout Properties Dialog, Hierarchical Tab" on page 800 ➤ "Layout Properties Dialog, Symmetric Tab" on page 802
Set Background Color	There are three ways to select a background color: ➤ On the Swatches page, click the desired color. The preview area shows what your choice looks like. ➤ On the HSB page, click the desired color in the color area. Use the sliding bar to adjust the hue, saturation, and brightness levels individually. ➤ On the RGB page, enter specific values for red, green, and blue, or click the desired value on the scale. Click OK to make your changes active, Cancel to close the dialog without activating your changes, or Reset to reset the background color to the default.

Layout Properties Dialog, General Properties Tab

UI Element	Description
Spacing	Adjust the spacing between objects of a graph. **Constant Spacing**. Specify the minimum amount of white space between adjacent rows and columns. **Proportional Spacing**. Specify a factional value. The margin around disconnected objects is calculated as the product of this fractional value and half the height of the node. **Graph Margin Spacing.** This group of fields affects the margins for graphs. Separate frame values are stored for the Constant Spacing and Proportional Spacing models, but only one spacing model is in effect at any given time.
Labeling	Customize the automatic positioning of arrow labels. **Incremental Layout**. Maintains the relative positioning of labels from a graph's previous layout whenever the graph is modified. Incremental Layout results in graphs that maintain the same overall form when changes are made.

Layout Properties Dialog, Disconnected Tab

UI Element	Description
Components	Modify disconnected component properties. These options only have effect if more than one disconnected object exists within the graph.
Disconnected Nodes	Modify disconnected node properties. These options only have effect if more than one disconnected object exists within the graph.

Layout Properties Dialog, Hierarchical Tab

UI Element	Description
Orientation	**Left To Right**. Positions components so that the roots of the graph are near the left of the viewing area, and the leaves of the graph are near the right. The levels are vertical. **Top To Bottom**. Positions components so that the roots of the graph are near the top of the viewing area, and the leaves of the graph are near the bottom. The levels are horizontal. **Right To Left**. Positions components so that the roots of the graph are near the right of the viewing area, and the leaves of the graph are near the left. The levels are vertical. **Bottom To Top**. Positions components so that the roots of the graph are near the bottom of the viewing area, and the leaves of the graph are near the top. The levels are horizontal. **Default value**: Left to Right.
Level Alignment	Changes the alignment of components on the same level. It is similar to text alignment within a paragraph. If Orientation is horizontal (Top To Bottom or Bottom To Top), you can set the Level Alignment as either Top, Center or Bottom. If Orientation is vertical (Left To Right or Right To Left), you can set the Level Alignment as either Left, Center or Right.

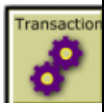
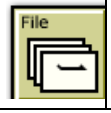
UI Element	Description
Spacing	Adjusts spacing between levels of components and between component within the same level in a graph by adjusting the frames that surround them. **Variable Level Spacing**. When enabled, variable level spacing considers the density of arrows between adjacent levels when adjusting the level spacing. For example, if too many arrows cross at the same point between levels, then the level spacing is automatically increased. Consequently, not all levels would have the same spacing. Variable level spacing is usually used in combination with orthogonal arrow routing. **Proportional Spacing.** Sets the spacing around each node to a fractional value (between 0 and 1) based on the node size. **Constant Spacing**. Allows the spacing around each node to be a constant value, independent of the size of the node. **Between Levels**. Allows you to set the spacing between each level in the graph. In left-to-right and right-to-left views, levels run vertically. In top-to-bottom and bottom-to-top views, levels run horizontally. **Between Nodes**. Allows you to set the spacing between adjacent components within a level.
Edge Routing	The following options allow you to tailor arrow routing: **Orthogonal Routing**. Each arrow is drawn as a sequence of line segments that run parallel to the x- and y-axes. arrows have bend points of 90 degrees only. The orthogonal routing style works best in combination with variable level spacing and ports. **Straight Edges**. Merges the incoming arrows of a node into a single fork-like structure. Clearer drawings result provided the graph is not too dense. This feature is only available when orthogonal routing is enabled.

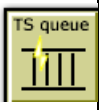
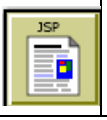
Layout Properties Dialog, Symmetric Tab

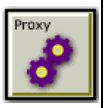
UI Element	Description
Spring Options	These controls allows you to set the options relating to the spring model of the graph. **Node Spacing**. This value is used in determining the white space between components. It specifies the length of each arrow as a function of the width and height of the components to which the arrow connects. A node spacing of zero specifies that arrow lengths be just long enough to keep components from touching — there is no white space between components. The larger the node spacing value, the greater the amount of white space between components. **Random Seed**. The initial layout of the graph is keyed to a random number sequence. The value set in this field is used to generate the random number sequence to set initial coordinates for each components. Each different non-negative value can produce a different layout. This field is active when Incremental Layout is not checked.

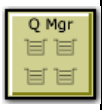
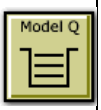
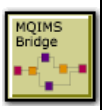
Resource Icons

Resource icons represent system components. Depending on your system, the following resource icons can appear in the Component Topology:

Icon	Resource Description
	CICS transaction. Note that TransactionVision automatically filters out all CICS transactions that begin with “C” because they are internal CICS transactions. One exception is CKBP, which denotes a WebSphere MQ-CICS bridge transaction. All CICS DPL programs invoked through MQ-CICS bridge have this transaction name.
	CICS file

Icon	Resource Description
	CICS Transient Data (TD) queue
	CICS indirect TD queue. Similar to a TD queue, the indirect TD queue is an alias to other TD queues.
	CICS temporary storage (TS) queue. Used for temporarily passing data, users can create and destroy TS queues on the fly.
	EJB session bean
	EJB entity bean
	EJB message-driven bean
	Internet
	JMS queue
	JMS topic
	Java server page

Icon	Resource Description
	Java servlet
	Process object
	Program. For generic events, this represents the event itself or a user defined system resource object. For information about implementing generic events, see the *HP TransactionVision Advanced Customization Guide* PDF.
	Proxy object. Proxy objects involve applications that are not monitored by a TransactionVision agent. For information about the proxy agent and proxy objects, see the *HP TransactionVision Deployment Guide* PDF.
	SQL queue. Represents access and execution of an SQL statement.
	Web application
	WebSphere web application server
	WebSphere MQ cluster node relating multiple queue instances in a clustering environment
	WebSphere MQ distribution list
	WebSphere MQ message queue

Icon	Resource Description
	WebSphere MQ remote queue
	WebSphere MQ alias queue
	WebSphere MQ dead letter queue
	WebSphere MQ namelist
	WebSphere MQ queue manager
	WebSphere MQ local queue
	WebSphere MQ model queue
	WebSphere MQ IMS Bridge node. If you do not have a custom bean to add queue and queue manager information for IMS Bridge events, an "unknown queue" node is shown for each IMS Bridge node associated with the MQIMS_BRIDGE_ENTRY event. For more information about the TransactionVision IMS Bridge Agent, see the *HP TransactionVision Deployment Guide* PDF.

Icon	Resource Description
	Start point. Only displayed when you select the **Edge & Node > Show Start/End Nodes** option, which is only available when you drill down to the component topology from the "Transaction Tracking Report" on page 675 or the "Transaction Detail Report" on page 686. For more details, see "Start and End Nodes" on page 760.
	End point. Only displayed when you select the **Edge & Node > Show Start/End Nodes** option, which is only available when you drill down to the component topology from the "Transaction Tracking Report" on page 675 or the "Transaction Detail Report" on page 686. For more details, see "Start and End Nodes" on page 760.

Troubleshooting and Limitations

Topology is Empty

If an empty graph appears when you view the Component Topology in dynamic mode, it is likely due to the associated query returning an extremely large number of events. The status window may contain a message with more information.

Modify the query or time period so that fewer events are selected and re-run the Component Topology.

JMS Icons are Missing

In some cases TIBCO JMS events do not show JMS queue/topic name in the Event Analysis view and there are no JMS icons in the Component Topology view. The Sensor error log contains the following entry:

```
com.tibco.tibjms.admin.TibjmsAdminException: Unable to connect to server.
Root cause:
javax.jms.JMSSecurityException: invalid name or password, or not authorized to
connect as administrator
```

To enter the appropriate EMS Server credentials, modify the following file in the Java Agent installation directory:

<java_agent_install_dir>/TransactionVisionAgent/config/sensor/SensorConfiguration.xml

Enter the lines in bold below:

...
<!-- for object resolution, we need admin right to access servers
the incoming url could be tcp://heineken:7222 or
tcp://heineken.bristol.com:7222 or using ssl -->
<TibcoEMSServers>
<TibcoEMSServer name="ems_server_name" loginName="XXX" password="YYY">
<ServerUrl>tcp://ems_server_host_name:7222</ServerUrl>
</TibcoEMSServer>
</TibcoEMSServers>
...

Index

Symbols

.Net Agent
  monitored APIs 476

A

Add job dialog 325
agent
  error messages 423
  in architecture 27
  monitored APIs 467
  stopping event collection 371
aggregated topology
  viewing 738
alert rules
  UI tab 557
alerts
  creating 536
Analyzer
  adding 106
  assigning communication links 107
  common log entries 142
  configuration concepts 85
  configuration settings 124
  enabling failure mode 112
  enabling for BPM 96
  enabling for RUM 94
  in architecture 26
  log files 141
  overview 40, 84
  recovering 111
  recovery mode 143
  removing 110
  starting and stopping 109
  transaction tracing rules assigned 562, 621
  viewing status 108, 120
Analyzer wizard 117
AnalyzerManager utility 368
API name
  in data collection filters 228
Application CI 508
Application CI Creation Dialog Box 546
Application Management View 509
application server name
  in data collection filters 240
Application Server Statistics report 708

B

BPI
  data sent from TransactionVision 358
  defining events to send 571
  drilldowns 358
  enabling communication with TransactionVision 365
  sending events to 534
  Transaction Management interaction 358
BPI rules
  UI tab 557
BPM
  Analyzer configuration 96
  drilldowns to Transaction Management 357
BSM
  exporting TransactionVision data to 363
BSM Gateway Server 343
BSM Profile database 361
BSM settings 343
Business Process Insight, see BPI.
Business Service Management, see BSM.

business transaction
see also transaction
creating a CI 525
creating custom alerts 536
enabling for monitoring 574
enabling for tracing 521
overview 507
sending events to BPI 534
stop tracing 533
Business Transaction CI Creation Dialog Box 545
Business Transaction Flow CI 508
Business Transaction Flow Creation Dialog Box 547

C

Caches
concept 87
CICS agent
monitored APIs 468
CICS events in component topology 764
CICS SYSID
in data collection filters 251
CICS Task
in data collection filters 252
CICS transaction
in data collection filters 252
classification rules
UI tab 553
client applications 483
client host/IP
in data collection filters 240
Communication Link wizard 182
communication links
assigning to an Analyzer 175
configuration queues 157, 158
create new 174
data collection filter assignment 177
default 171
event queues 159
exception message queues 159
in architecture 27
modifying 177
overview 156
removing 178, 180
SonicMQ configurations 161
testing 175
TIBCO EMS configurations 161
troubleshooting 205
viewing status 178
WebLogic JMS configurations 162
WebSphere MQ configurations 160
completion code
in data collection filters 243
component topology
configure edge labels 781
modify component groupings 776
modifying layouts and properties 759
overview 756
printing graphs 787
resource icons 802
viewing 770
configuration message expiry, for Analyzers 46
configuration queues, communication links 158
ConfigurationTool utility 371
correlation
customizing 373
CorrelationUtil utility 373
custom correlation
about event-to-event correlation 591
about tracking ID correlation 593
overview 590
troubleshooting 632

D

Data Collection Filter Summary page 219
data collection filters
assigning to a communication link 177, 215
creating 213
Edit page 257
modifying 215
overview 29, 208
removing 216
data purge operation
starting and stopping 123
viewing status 120

data rules
creating 568
UI tab 554
database
in architecture 27
database connection settings 45
DataPower Agent
communication link 172
monitored APIs 477
DataUtil 388
DeleteEvents utility 391

E

Edit Data Collection Filter page 257
Edit Job dialog 326
EJB
classes instrumented 473
events in component topology 765
methods
data collection filter 233
in data collection filters 233
methods instrumented 473
name
in data collection filters 225, 232
technology in the Java Agent
monitored APIs 473
troubleshooting 485
enable session tracking 336
enable_comm_between_BPI-TV 365
End User Management, see EUM.
EUM 355
Event Analysis report 691
event collection
emergency stop 371
starting and stopping 108
threads 200
event correlation 373
Event Customization Rules
General tab 620
Event Customization Rules page 618
event persistence 200
event queue manager requirement
MQ 164
event queue message persistency requirement
TIBCO EMS 169
WebSphere MQ 165
event queue storage requirement
SonicMQ 167
TIBCO EMS 169
event queue storage requirements
WebSphere MQ 165
event queues
communication links 159
maximum message length 197
requirements
SonicMQ 167
TIBCO EMS 169
event transport provider 184
event XML
overview 85
storing 201
events
overview 28, 29
packaging 224
purging 391
storing and using raw 146
troubleshooting missing events 484
viewing metrics 650
event-to-event correlation
about 591
exception message queue 143, 159, 200
exception setting
in data collection filters 232, 238, 240

F

failed state of a transaction 725, 728
Failure mode 89
firewall configuration 77
flow map 720
FlushStatusUtil 144, 399

G

generic events 769
Group CI Creation Dialog Box 548

H

host name
in data collection filters 225
HP Diagnostics 357
HP Software Support Web site 20
HP Software Web site 21

I

IBM AIX platform troubleshooting 486
IBM WebSphere DataPower Events 768, 769
IBM WebSphere MQ
communication link configuration 185
icons
component topology 802
IMS Identifier
in data collection filters 253
IMS PSB
in data collection filters 253
IMS region identifier
in data collection filters 254
IMS region type
in data collection filters 255
IMS transaction
in data collection filters 256
instance topology
viewing 739

J

Java programs
monitoring 481
Java user events in component topology 767
JDBC Agent
methods tracked 474
JDBC class name
in data collection filters 234
JDBC database name
in data collection filters 234
JDBC database object
in data collection filters 234, 235
JDBC events in component topology 766
JDBC method
in component topology 786
in data collection filters 235
JDBC result code
in data collection filters 235
JDBC Technology in the Java Agent
monitored APIs 474
JMS
methods tracked 471
troubleshooting 486
JMS class
in data collection filters 238
JMS connection name
in data collection filters 238
JMS events in component topology 763
JMS method
in data collection filters 238
JMS queue
in data collection filters 239
JMS technology in the Java agent
monitored APIs 471
JMS topic
in data collection filters 239
Job Manager
overview 40
viewing status 321
jobs
Add Job dialog 325
adding custom 318
built-in overview 316
custom 317
Edit Job dialog 326
Job Summary page 321
modifying built-in 318
viewing status 323

K

Knowledge Base 20

L

late state of a transaction 725, 728
LD_LIBRARY_PATH environment variable 206, 483
LIBPATH environment variable 206, 483
link rendering
for Aggregated Topology states 728

log files
Analyzer
common entries 142
logging
concept 88

M

ManageQueue 143
match condition 520
match conditions
creating 564
message length requirement
WebSphere MQ 163
message queue permissions
SonicMQ 168
TIBCO EMS 170
WebSphere MQ 166
message queue requirements 162
SonicMQ 167
TIBCO EMS 168
WebLogic JMS 170
WebSphere MQ 163
method
in data collection filters 244
metrics aggregated by application server 650
metrics at the event level 650
MigrateDButility 408
modification rules
Rule Definitions tab 629
MyBSM 353

N

nanny utility 408
Nanny Manager 408
NonStop TMF Agent
monitored APIs 476

O

online resources 20

P

PassGen utility 411, 413
PATH environment variable 206, 483
Perl programs
monitoring 481
Processing Server
create 48
firewall configuration 77
in architecture 26
key configuration settings 43
modify 49
overview 40
port conflicts 75
recover 50
remove 49
synchronize 50
processing server
troubleshooting 75
Processing Server Wizard 51
program name
in data collection filters 225
Progress SonicMQ communication link
configuration 187

Q

queries
adding 262
modifying 264
overview 260
testing 264
Queries Summary page 265
Query Engine
configure 53
overview 40
queue
in data collection filters 244
Queue depth requirement
WebSphere MQ 164
queue manager
troubleshooting channel limitations
485
queue manager channel limitations 485
queue object
in data collection filters 248
queue parameter requirements
SonicMQ 168
TIBCO EMS 169
WebSphere MQ 165

queue requirements
event queue manager 164
queue depth 164
queue space
in data collection filters 245
queue user permissions
WebSphere MQ 165

R

raw events
storing 201
rebind_agent utility 414
rebind_sensor 486
rebind_tux_sensor utility 415
reports
overview 636, 648
WebSphere MQ and JMS Statistics 712
Response 228
response code
in data collection filters 228
response time 642
RTSM
TransactionVision data 363
rules
importing/exporting 519
RUM
Analyzer configuration 94
data sent to TransactionVision 356
publish port 345
publish protocol 345
reports 356
runmqtrm 482
runSupportSnapshot utility 416

S

schemas
See database schema 27
SensorSetup 485
Service Health 348
Service Level Management 353
service strings
in data collection filters 245
servlet
instrumented classes 469
methods tracked 470
troubleshooting 485
servlet events in component topology 765
servlet method
in data collection filters 241
servlet name
in data collection filters 225
servlet technology in the Java Agent
monitored APIs 469
session tracking
enabling 336
SHLIB_PATH environment variable 206, 481, 483
SonicMQ
broker
settings 334
communication link 171
communication link configurations 161
DataPower communication link 172
Domain Manager service 408
HTTP communication link 173
message queue requirements 167
RUM communication link 171
SonicMQ Broker
service 408
SQL code
in data collection filters 236
SQL state
in data collection filters 236
SQL statement
in data collection filters 237
Standard mode 89
status code
in data collection filters 241
status strings
in data collection filters 246
successful state of a transaction 725
Synchronize Configuration command 50
SYSID
in data collection filters 229, 273

T

TD queue name
  in data collection filters 229
technology
  in data collection filters 246
terminal ID
  in data collection filters 229
TIBCO EMS
  communication link configurations 161, 186
  message queue requirements 168
time range
  in data collection filters 226
time server
  configuration 199
time skews
  concept 87
TimeServer utility 420
tooltips 729
top-level server request 579
topologies
  overview 636
tracing
  creating a business transaction CI 525
  setting up for a business transaction 521
  stopping 533
Tracing Properties tab 551
tracking ID correlation
  about 593
Trade Demo sample events 34
TradeTransaction.xdm 34
transaction
  count 641
  data model
    overview 582
  ID
    in data collection filters 230
  latency 724, 727
  name 641
  response time 642
  state 351, 725
  states 643
  threshold 644
  value 644, 724, 727
  volume 641, 724, 727
Transaction Configuration page
  General tab 549
  Tracing tab 550
  Tracing tab, Alert Rules 557
  Tracing tab, BPI Rules 557
  Tracing tab, Classification Rules 553
  Tracing tab, Data Rules 554
  Tracing tab, Tracing Properties 551
Transaction Data page 585
  Create New Data Definition dialog 587
transaction management
  overview 506
transaction monitoring
  overview 574
  setting up 576
Transaction Over Time report 666
Transaction Summary report 652
transaction topology
  overview 718
transaction tracing
  rules 563, 572
Transaction Tracking report 675
TransactionDefinition.xml 34
TransactionVision
  administration
    workflow overview 32
  architecture 26
  database
    updating manually 384
  demo
    overview 31
    setup 34
  integrations troubleshooting 365
  overview 26
triggered programs, monitoring 482
troubleshooting 479
  custom correlation 632
  displaying Perl and Java program names 481
  enabling SHLIB_PATH on HP-UX 481
  JMS Sensor 486
  missing events 484
  monitoring client applications 483
  monitoring Java programs 481
  monitoring Perl programs 481

monitoring triggered programs 482
monitoring WebSphere MQ triggered programs 482
optimizing performance 484
processing server 75
Processing Server port conflicts 75
queue manager channel limitations 485
TransactionVision integrations 365
triggered programs 482
Troubleshooting and Knowledge Base 20
TS queue ID
in data collection filters 230
Tuxedo Agent
monitored APIs 475
TVISION_BANNER environment variable 205
TVISION_REPORT_ARGS environment variable 481
TVISION_SYSLOG environment variable 206

U

Unicode
troubleshooting 494
URI
in data collection filters 241
user data
storing 201
user data matching
in data collection filters 227
user data range
in data colletion filters 223
user event class
in data collection filters 243
user ID
in data collection filters 231
user name
in data collection filters 226
User permissions 30
user-defined events processing 200

V

ValidateXml utility 421

W

Web application name
in data collection filters 233, 237, 239, 242
Web Session report 664
WebLogic communication links
configuration 188
WebLogic JMS
communication link configurations 162
WebLogic JMS message queue requirements 170
WebSphere API Name
in data collection filters 247
WebSphere completion code
in data collection filters 247
WebSphere MQ
communication link configurations 160
message queue requirements 163
rebinding sensor after support pac installation 486
WebSphere MQ agent
monitored APIs 468
WebSphere MQ and JMS Statistics report 712
WebSphere MQ API reason code 249
WebSphere MQ events in component topology 763
WebSphere MQ MQIMS Bridge API
in data collection filters 251
WebSphere MQ ReplyTo object
in data collection filters 250
WebSphere MQ-IMS bridge agent
monitored API 469
WebSphere MQ-IMS bridge events in component topology 766

X

XDM files
editing 342
modifying 334
overview 332

z

z/OS job steps
in data collection filters 256