

Tivoli IBM Tivoli Composite Application Manager for Application
Diagnostics
Version 7.1.0.1

*ITCAM Agent for J2EE: Configuring and
using TTAPI*

IBM

Tivoli IBM Tivoli Composite Application Manager for Application
Diagnostics
Version 7.1.0.1

*ITCAM Agent for J2EE: Configuring and
using TTAPI*

IBM

2010

This 2010 edition applies to ITCAM Agent for J2EE version 6.1.0.5 and all subsequent releases and modifications until otherwise indicated in new editions.

© **Copyright IBM Corporation 2009, 2010.**

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

Contents

Figures	v	Chapter 3. Integration	13
Chapter 1. Overview	1	Servers, components, and application workspaces	13
Concepts and components	1	Transactions workspace	16
Key components	1	Servlet and JSP Request Integration	17
Prerequisites	2	RMI and IIOP Request Integration.	18
Integrated Requests	2	Web Services request integration	19
Chapter 2. Installation.	5	MQI request integration	20
Enabling Integration	5	EJB integration	22
Disabling Integration	6	Message Driven Bean integration	23
Monitoring JDBC nested requests	6	Custom request integration	23
Enabling and disabling MQ tracking	7	JDBC nested request integration	24
Enabling and disabling JDBC tracking at MOD Level		JMS messaging topology integration	24
1	9	Logging and tracing	26
Enabling and disabling JMS tracking at MOD Level		Logging and tracing for the Data Collector	26
1	11	Logging and tracing for the TTAPI and TEMA	26
		Appendix. Accessibility	29

Figures

1. Choose Data Collector Agents window	5	12. Web Services topology view in Transactions workspace	20
2. Enter TTAS Information window	6	13. MQI topology view in Transactions workspace	21
3. Enabling and disabling MQ tracking in the Modify Configuration window	8	14. MQI topology view in Components workspace	22
4. Enable TTAPI for JDBC window	10	15. EJB topology view in Transactions workspace	23
5. Enable TTAPI for JMS window	12	16. JDBC nested request view in Transactions workspace	24
6. Servers workspace	14	17. JMS example topology: queue sender and queue receiver	25
7. Components workspace	15	18. JMS example topology: topic publisher and topic subscriber	25
8. Applications workspace	16	19. JMS example topology: message sender and message driven bean	26
9. Transactions workspace	17		
10. Servlet and JSP transactions	18		
11. RMI/IIOP topology view in Transactions workspace	19		

Chapter 1. Overview

IBM Tivoli Composite Application Manager for Transactions (ITCAM for Transactions) is an IBM Tivoli Monitoring-based product that provides a unified, end-to-end transaction tracking solution for the IT Operations segment.

ITCAM for Transactions tracks transactions within and among applications. The product determines the time spent by the transaction in each application and, where possible, the time spent communicating between applications. You can use the product to observe transactions across products, providing easier integration between different products through the Tivoli Enterprise Portal interface.

Through the Transaction Tracking Application Programming Interface (TTAPI), the ITCAM Data Collector (DC) for J2EE provides request and transaction data to ITCAM for Transactions and enables seamless integration between the ITCAM for Application Diagnostics and ITCAM for Transactions products.

Concepts and components

High-level outline of the components and concepts behind ITCAM for Transactions.

ITCAM for Transactions includes the following functions:

- Comprehensive domain tracking across IBM middleware and mainframe systems.
- Proactive monitoring of business transactions and Internet services.
- Real user Web and client application monitoring.
- An integrated solution built on the IBM Tivoli Monitoring platform.

Key components

Key components and concepts of ITCAM for Transactions.

Key components and concepts of ITCAM for Transactions include:

- *ITCAM for Transaction Tracking*: Consumes data from application server, MQ, and custom instrumentation to show topology and isolate problems.
- *Transaction Collector Tivoli Enterprise Monitoring Agent*: Receives raw event data, stores windows of raw data, and aggregates over time. The monitoring agent responds to calls from the Transaction Reporter for aggregate or instance data.
- *Transaction Reporter Tivoli Enterprise Monitoring Agent*: Connects the data gathered by the Transaction Collector and constructs topology using linking and stitching criteria.
- *Transaction Tracking Application Programming Interface (TTAPI)*: A lightweight, low-latency client API through which Data Collectors construct and send tracking events to the Collector Tivoli Enterprise Monitoring Agent.
- *Linking*: Connects events that occur within a tracking domain, for example ARM and MQ.
- *Stitching*: Rather than using tokens, dynamically connects events that occur between tracking domains, for example MQ to WebSphere Application Server, to enable comprehensive domain tracking across IBM middleware & mainframe systems

In ITCAM for Transactions, request data and transactions are reported through the following four workspaces:

- *Servers*: This workspace provides aggregated data for all transactions and requests that occur in a physical computer over a period of time called the Aggregation Period. Transactions and requests can come from application servers, Web servers, and WebSphere MQ products. The workspace is mainly used to isolate a performance problem to a physical computer.
- *Components*: This workspace provides aggregated data for all transactions and requests that occur in a major software component over a period of time. The main software components in a production system may include an application server, WebSphere MQ, Web server, and so on. Data reported by this workspace is mainly used to isolate a performance problem to a software component.
- *Applications*: This workspace provides aggregated data for all transactions and requests that occur in a major runtime instance within a period of time. For example, for the WebSphere Application Server component, the runtime instance is a WebSphere Application Server process; for WebSphere MQ, it is a queue manager process. The workspace is mainly used to isolate a performance problem to a runtime instance.
- *Transactions*: This workspace provides aggregated data for transaction instances. The aggregation is done over each distinctive Unit of Work (UOW) or request. For example, in a J2EE application server environment, a transaction or UOW is usually identified as a URI for a Servlet or JSP page. Data reported by this workspace can be used to isolate a performance problem to a specific transaction or request.

Within each workspace, the Transactions Topology view provides a visual representation of the connections between servers, components, applications, and transactions that have occurred within a set time frame.

Prerequisites

The following prerequisites must be in place to allow for this integration.

- ITCAM Agent for J2EE version 6.1.0.5/6.2.0.5 or later must be installed.
- ITCAM for J2EE Managing Server version 6.1.0.5 or later or ITCAM for Application Diagnostics Managing Server version 7.1.0.1 or later must be installed.
- ITCAM for Transactions version 7.1 Fix Pack 2 or later Transaction Tracking Collector and Transaction Reporter must be installed.
- Clock must be accurate and synchronized (using something like Network Time Protocol) across servers.
- For application server instances that GET and PUT messages through the MQI interface, MQ monitoring must be enabled in the ITCAM for Application Diagnostics User Interface (Visualization Engine) configuration for the Data Collectors.
- RMI-IIOP instrumentation must be enabled on both client and server application server instances if both sides have Data Collector configured and enabled.
- Web Service instrumentation must be enabled on both requester and provider application server instances if both sides have Data Collectors configured and enabled.

Integrated Requests

Requests supported by this integration.

Integration between ITCAM Agent for J2EE Data Collector and ITCAM for Transactions supports all composite requests that generate events to Global Publishing Server (GPS). It also supports top-level EJB and Custom Edge requests. In addition, it supports top-level Servlet and JSP requests to integrate with the ITCAM for Transactions Robotic Response Time agent (T6) and (WebSphere Application Server-supported) Web Servers with ARM-enabled plug-ins:

- Custom Request
- EJB (including Message Driven Bean)
- MQI transactions (but MQv7 JMS is not supported)
- RMI/IIOP
- Servlet/JSP
- Web Services

JDBC nested requests are also supported.

JMS links are displayed in the Topology View.

In the Topology View, an entity participating in transactions instrumented via both ITCAM for Application Diagnostics (Agent for J2EE) and ITCAM for Service Oriented Architecture will be displayed as a single node.

Chapter 2. Installation

Follow the standard ITCAM Agent for WebSphere Applications installation procedure to install the Data Collector.

Enabling Integration

How to enable ITCAM for Application Diagnostics and ITCAM for Transactions integration through TTAPI.

To enable Data Collector TTAPI integration, use the Data Collector configuration process.

Configure or reconfigure ITCAM Agent for J2EE Data Collector for the application server instance. In the **Choose Data Collector Agents** window, select **Data Collection for ITCAM for Transaction Tivoli Enterprise Portal interface**.

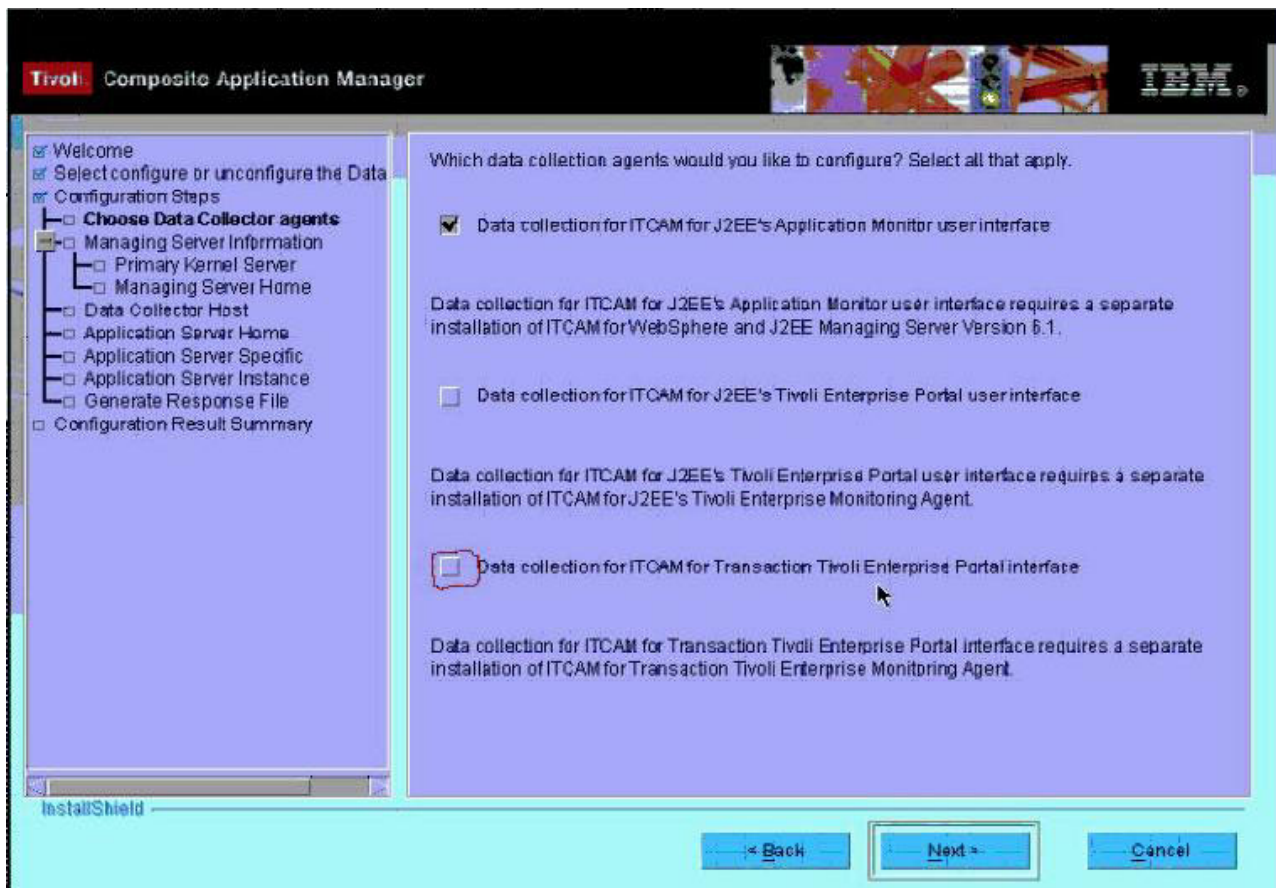


Figure 1. Choose Data Collector Agents window

Then in the **Enter TTAS Information** window, enter the host name and the port number of the ITCAM for Transaction Monitoring Agent.

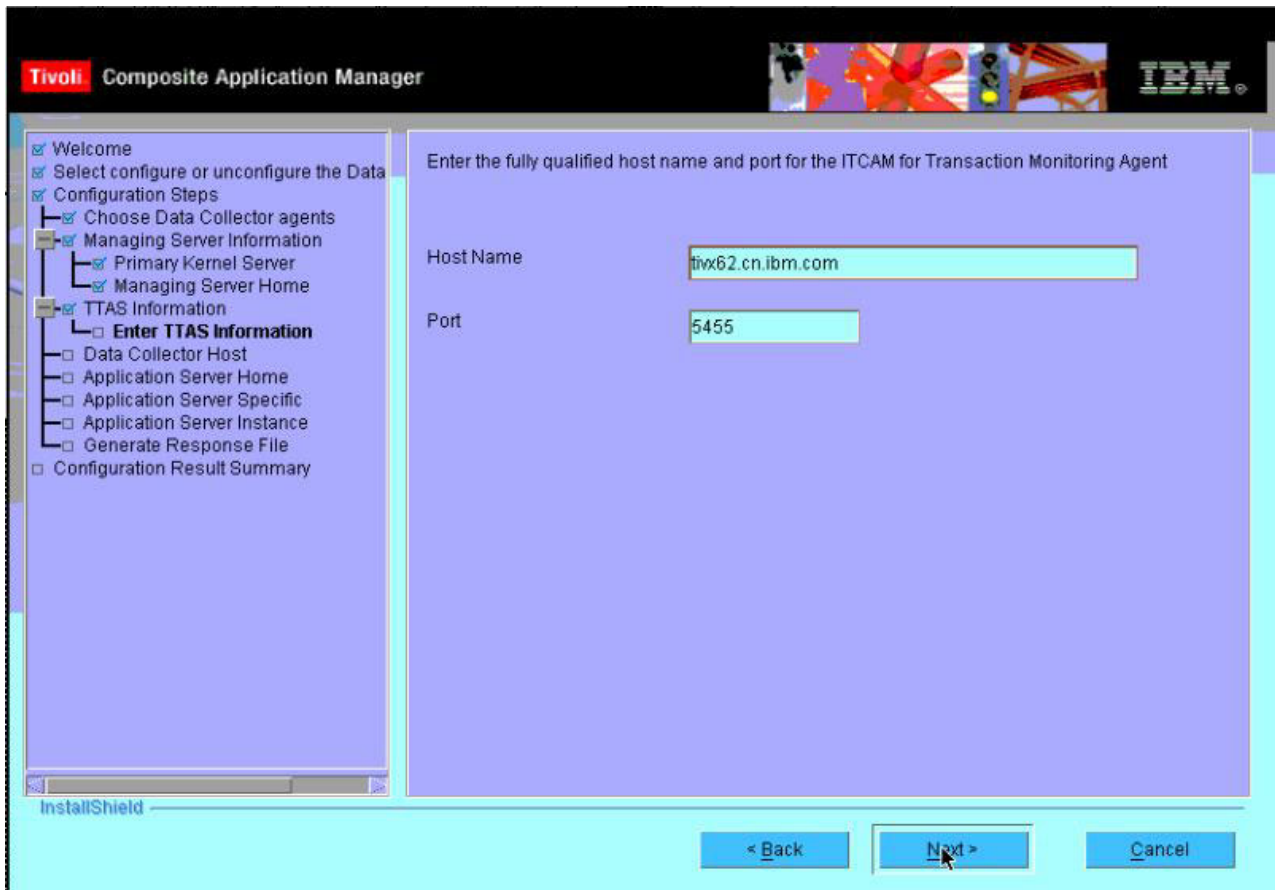


Figure 2. Enter TTAS Information window

Disabling Integration

Disabling TTAPI integration.

To disable Data Collector and TTAPI integration, set the following property in the `DCHOME/runtime/server/custom/toolkit_custom.properties` file:

```
com.ibm.tivoli.itcam.dc.ttapi.enable=false
```

To disable integration of the Data Collector with ITCAM for Transactions Web Response Time (T5) agent, set the following property in the `DCHOME/runtime/server/custom/toolkit_custom.properties` file:

```
com.ibm.tivoli.itcam.dc.ttapi.wrm.servlet.enabled=false
```

After making these changes, restart the application server instance.

Monitoring JDBC nested requests

How to control monitoring JDBC nested requests for TTAPI.

JDBC nested request monitoring is enabled by default when the Data Collector monitoring level, set in the Managing Server Visualization Engine, is L2 or L3. If the Data Collector monitoring level is L1, you can enable the JDBC nested request feature in the ITCAM web console by following these steps:

1. Choose **Administration -> Server Management -> Data Collector Configuration** and select **Enable TTAPI for JDBC**.
2. In the **TTAPI JDBC DISABLED DATA COLLECTORS** panel, select the Data Collectors that you want to enable for JDBC nested requests.
3. Click **Apply**.

If exceptions (failed requests) occur within a reporting period, they are reported via TTAPI, and the status of the transaction is set to Fail. The user is able to inspect individual exceptions. To limit the amount of JDBC exceptions displayed for a top-level transaction, set the following property in the `DCHOME/runtime/server/custom/toolkit_custom.properties` file:

```
com.ibm.tivoli.itcam.dc.ttapi.maxExceptions=number
```

By default, this amount is limited to 10.

Enabling and disabling MQ tracking

You can use TTAPI to track MQ transactions (both MQI and JMS). To do this, you must enable MQ tracking using the Visualization Engine; it is disabled by default.

About this task

To track MQ requests on application server instances, make sure MQ tracking is enabled in the configuration applied to Data Collectors monitoring each of the instances. For each of the required Data Collector configurations, perform the following procedure.

Tip: For more information on configuring Data Collectors in the Visualization Engine, refer to *ITCAM for Application Diagnostics User Guide*.

Procedure

1. Log on to the Visualization Engine as a user with administrator permissions.
2. From the top navigation, click **Administration > Server Management > Data Collector Configuration**. The Configured Data Collector Overview page opens.
3. Click **Configuration Library** on the left navigation pane. The Data Collector Configuration List page opens.
4. Click the Modify icon next to the configuration you want to modify. The Modify page opens.

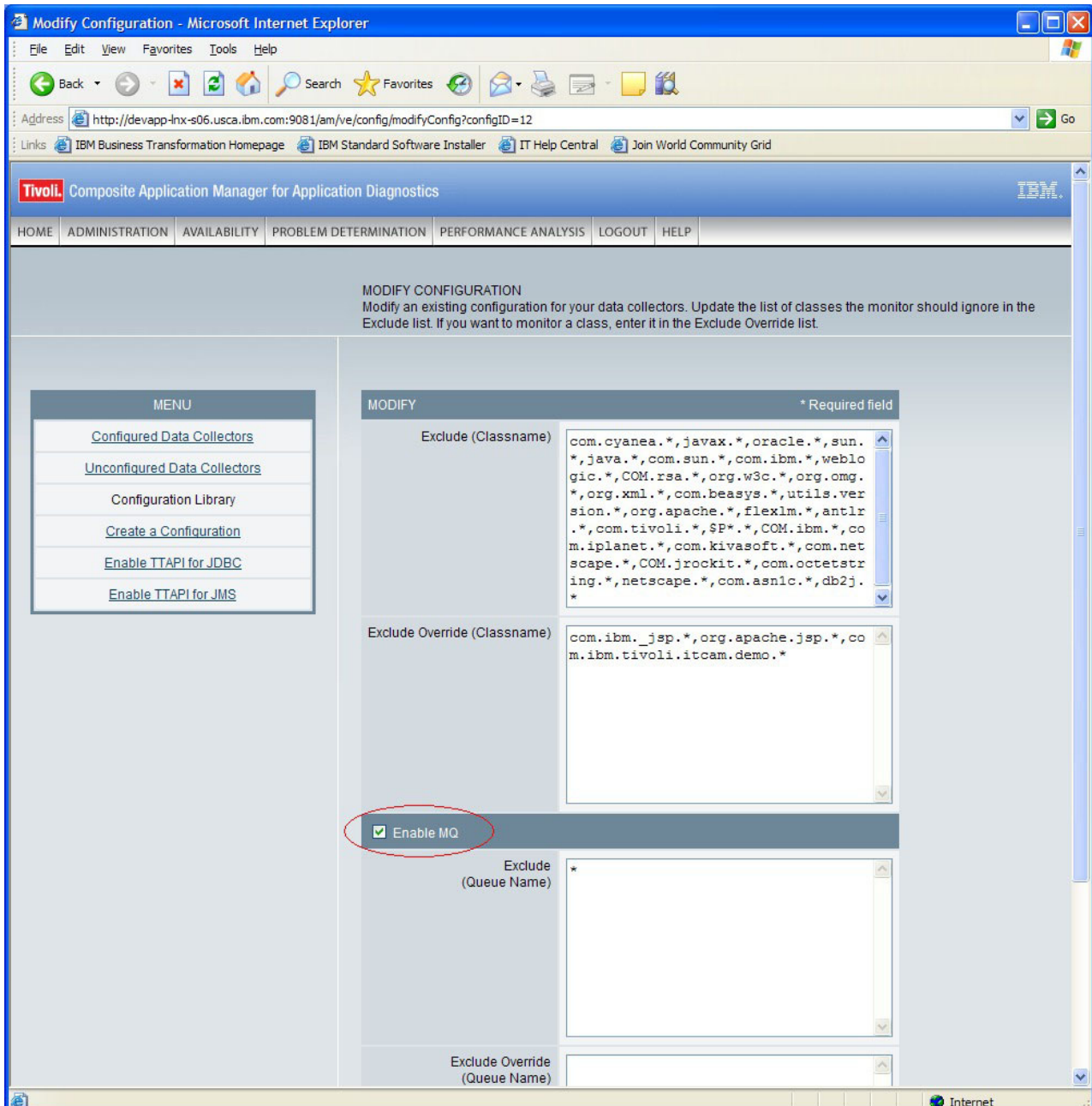


Figure 3. Enabling and disabling MQ tracking in the Modify Configuration window

5. You can perform any of the following changes:
 - To enable MQ transaction tracking for all queues, select the **Enable MQ** box and clear the **Exclude** window.
 - To disable MQ tracking for specific queues, enter the queue names in the **Exclude** window. You can use the * wildcard in the queue name; use the comma (,) to separate the queue names.
 - To enable MQ tracking for queues that would be disabled in the **Exclude** window, enter the queue names in the **Exclude Override** window.
 - To disable MQ transaction tracking for all queues, clear the **Enable MQ** box.
6. Click **Save** to save your modifications to the configuration. The Configured Data Collector Configuration List displays with the updated information.

Enabling and disabling JDBC tracking at MOD Level 1

You can use TTAPI to track JDBC transactions. However, by default, a Data Collector will not monitor JDBC transactions when it is set to MOD Level 1 by the Managing Server. If you need JDBC tracking on MOD L1, you must enable it.

About this task

Use the Visualization Engine to enable JDBC tracking at MOD L1.

Tip: For more information on configuring Data Collectors in the Visualization Engine, refer to *ITCAM for Application Diagnostics User Guide*.

Procedure

1. Log on to the Visualization Engine as a user with administrator permissions.
2. From the top navigation, click **Administration > Server Management > Data Collector Configuration**. The Configured Data Collector Overview page opens.
3. Click **Enable TTAPI for JDBC** on the left navigation pane. The Enable TTAPI for JDBC page opens.

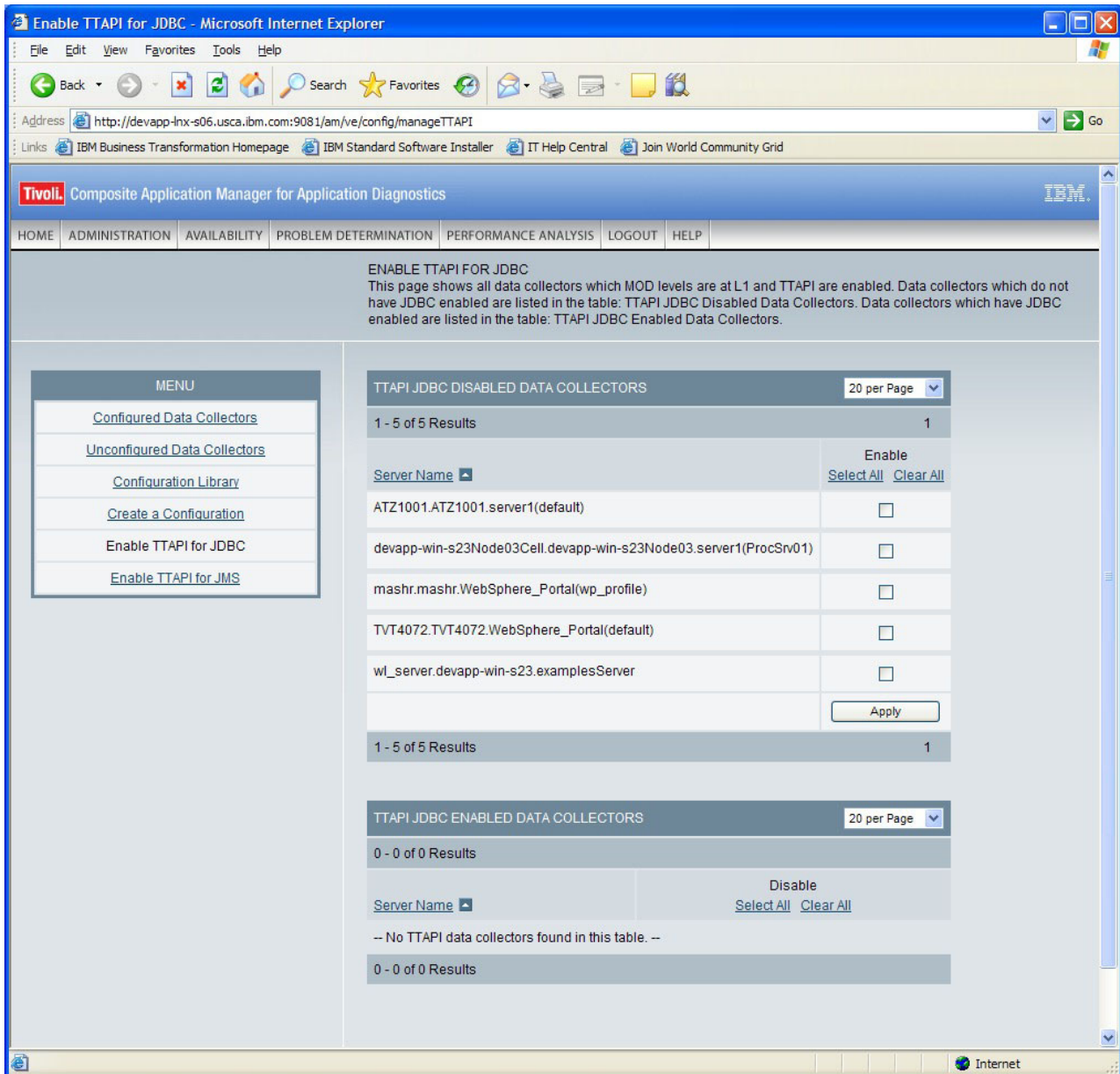


Figure 4. Enable TTAPI for JDBC window

4. You can perform any of the following changes:
 - To enable JDBC transaction tracking on MOD L1 for any monitored application servers, select the boxes next to the application server names in the **TTAPI JDBC DISABLED DATA COLLECTORS** table, and click the **Apply** button below this table.
 - To disable JDBC transaction tracking on MOD L1 for any monitored application servers, select the boxes next to the application server names in the **TTAPI JDBC ENABLED DATA COLLECTORS** table, and click the **Apply** button below this table.

Enabling and disabling JMS tracking at MOD Level 1

For the WebLogic application server, you can use TTAPI to display topology for JMS transactions. However, by default, a Data Collector will not track JMS topology when it is set to MOD Level 1 by the Managing Server. If you need to view JMS topology on MOD L1, you must enable it.

About this task

Use the Visualization Engine to enable JMS tracking at MOD L1. (JMS tracking is enabled by default when the Data Collector is at MOD L2 or MOD L3).

Tip: For more information on configuring Data Collectors in the Visualization Engine, refer to *ITCAM for Application Diagnostics User Guide*.

Procedure

1. Log on to the Visualization Engine as a user with administrator permissions.
2. From the top navigation, click **Administration > Server Management > Data Collector Configuration**. The Configured Data Collector Overview page opens.
3. Click **Enable TTAPI for JMS** on the left navigation pane. The Enable TTAPI for JMS page opens.

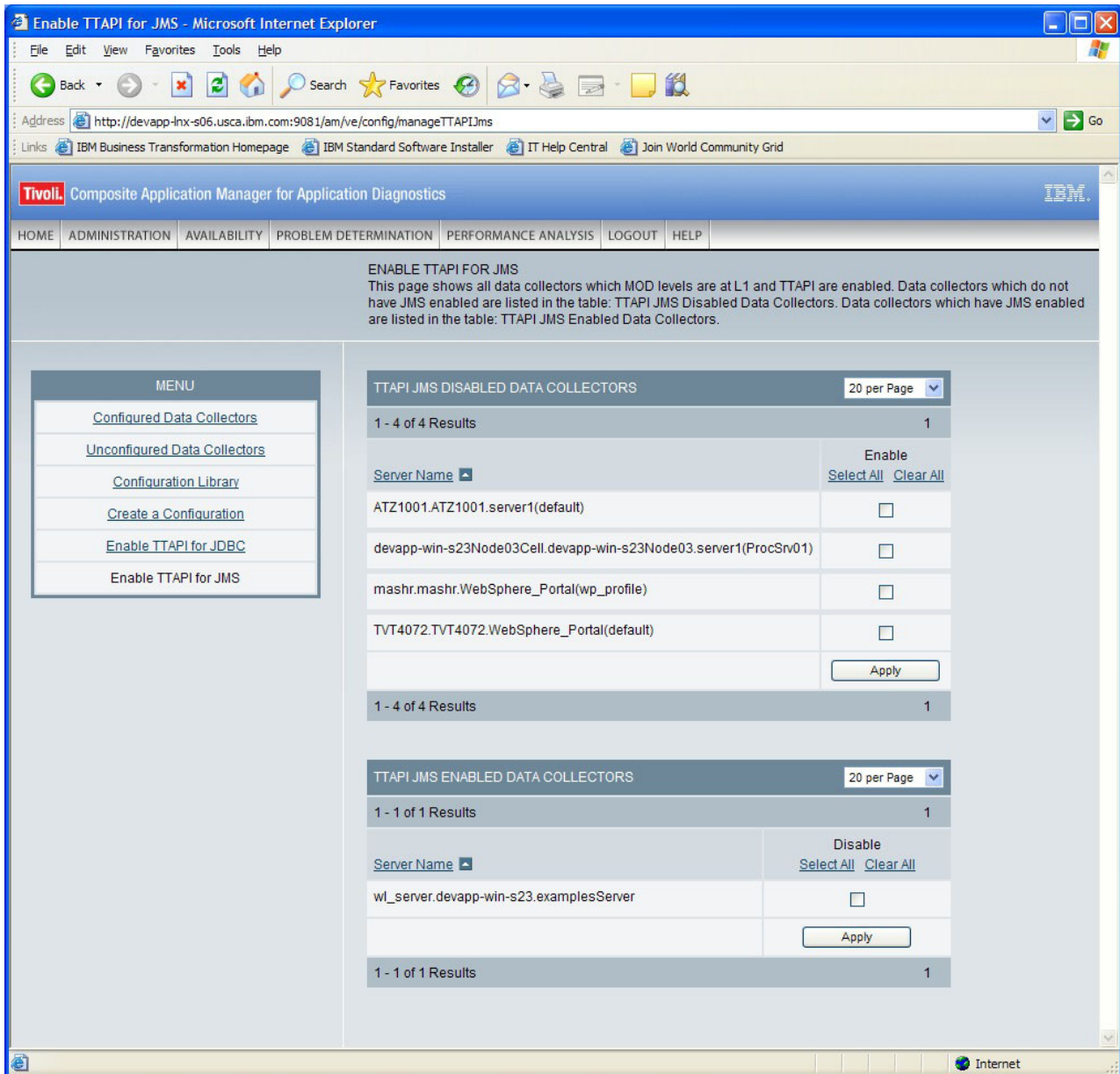


Figure 5. Enable TTAPI for JMS window

4. You can perform any of the following changes:
 - To enable JMS transaction tracking on MOD L1 for any monitored application servers, select the boxes next to the application server names in the **TTAPI JMS DISABLED DATA COLLECTORS** table, and click the **Apply** button below this table.
 - To disable JMS transaction tracking on MOD L1 for any monitored application servers, select the boxes next to the application server names in the **TTAPI JMS ENABLED DATA COLLECTORS** table, and click the **Apply** button below this table.

Chapter 3. Integration

This chapter covers the required servers, components, and application workspaces and their integration.

Servers, components, and application workspaces

Servers, Components, and Application workspaces display aggregated transactions and request data.

The following table shows the values set by the Data Collector for these workspaces to identify servers, components, and applications.

Table 1.

TTEMA TEP Workspace	Value of Name	Notes
Servers	Short DNS Name	
Components	The product name of the Application Server. For example, <code>weblogic:9, JBoss:4</code>	
Applications	The name of the Server instance. Examples: <ul style="list-style-type: none">• For WebLogic, <code>exampleServer</code>• For JBoss, <code>default</code> or <code>all</code>	The default value set by Data Collector

These values remain the same for all integrated requests. Figure 6 on page 14, Figure 7 on page 15, and Figure 8 on page 16 show the values displayed on the IBM Tivoli Monitoring Tivoli Enterprise Portal interface.

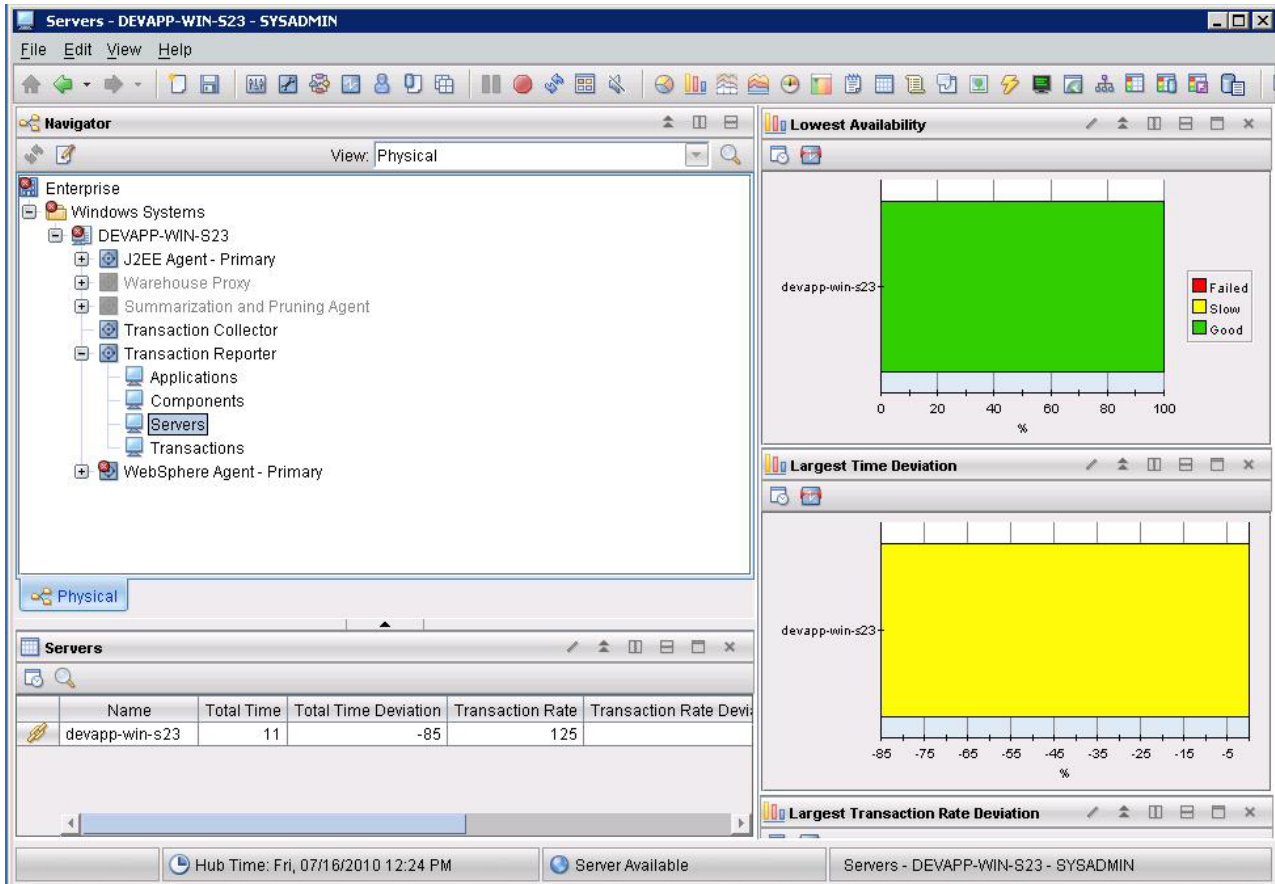


Figure 6. Servers workspace

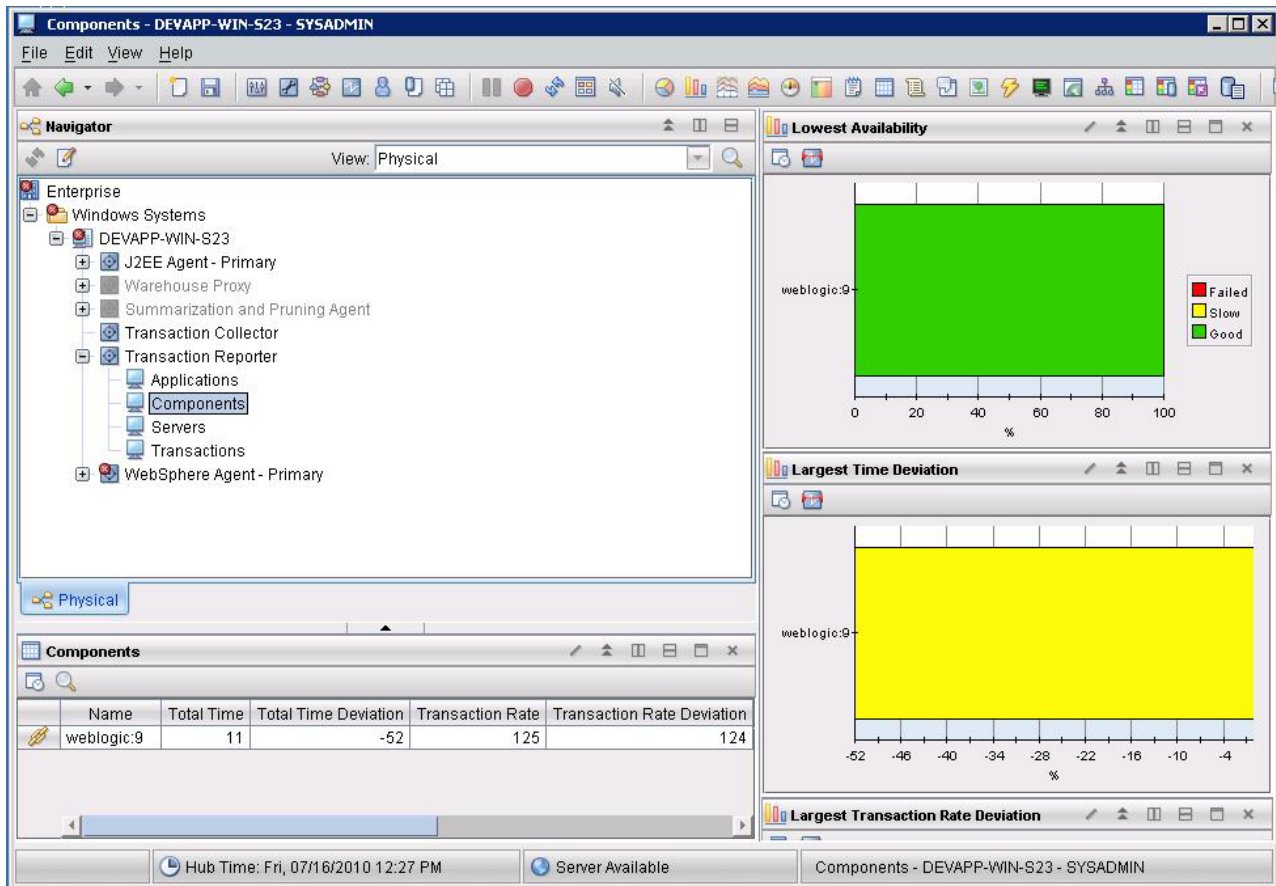


Figure 7. Components workspace

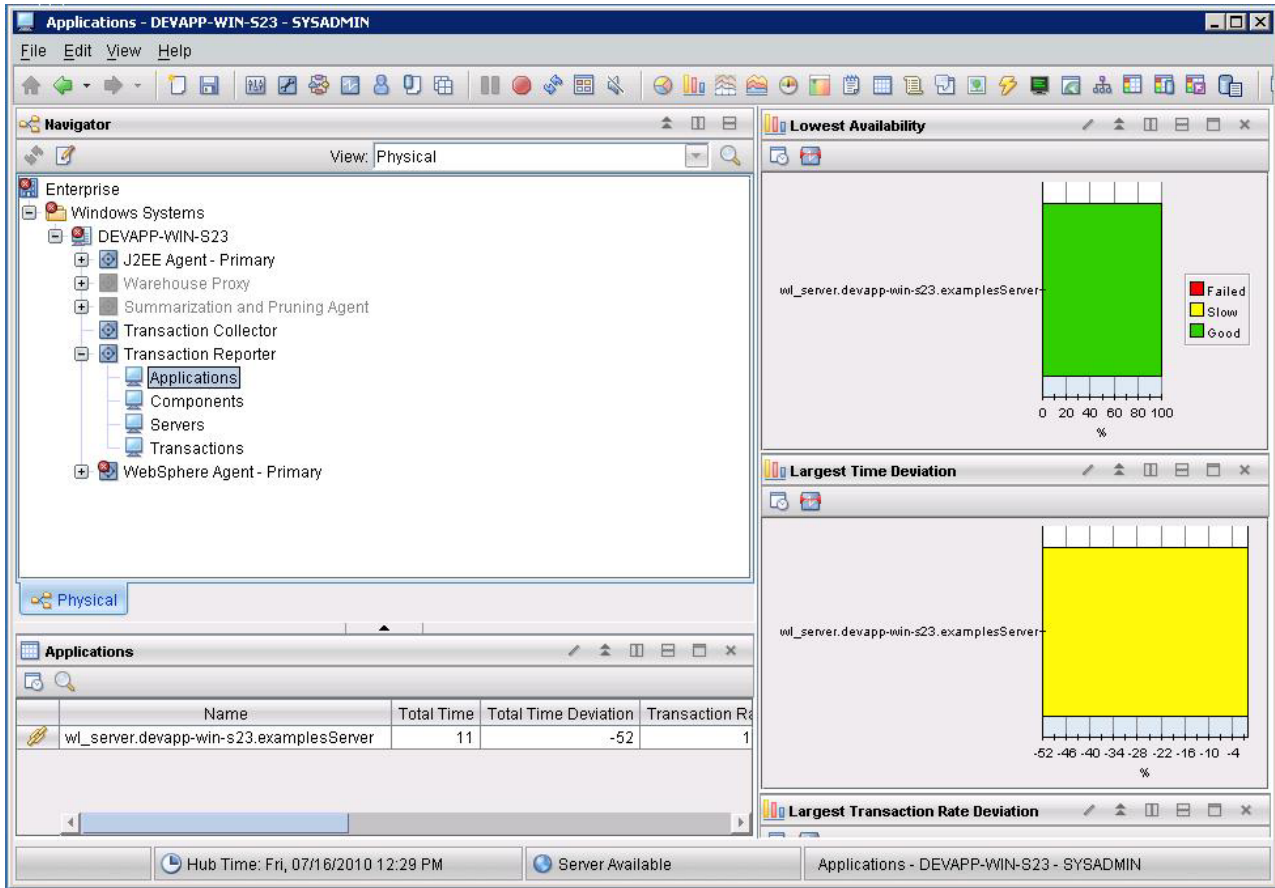


Figure 8. Applications workspace

Transactions workspace

Use the Transactions workspace to view the aggregated transaction information, and to access additional information in ITCAM for Application Diagnostics.

The Transactions workspace provides a list of transactions fitting certain criteria. Aggregated information is displayed for every transaction name.

To access the link menu for a transaction, right-click the chain icon at the left of the line in the transaction table. Use the link menu to view additional transaction information, including topology and response time statistics.

You can also use the link menu to access detailed information in ITCAM for Application Diagnostics:

- **Request Analysis** displays the Request Analysis workspace.
- **Diagnostics Recent Completed Requests** displays the recent request detail in the Managing Server Visualization Engine. This information is available only if the Deep Dive diagnostics infrastructure (Managing Server) is installed.

Important: If the Managing Server used for monitoring an application server has been changed, the link to the Visualization Engine may not work. To enable the link again, perform the Forget Topology Take Action in the Transactions workspace. Then ensure that data is still sent from the Data Collector, and wait

for four aggregation periods (by default, for 20 minutes). The aggregation period is set in the Translation Reporter configuration.

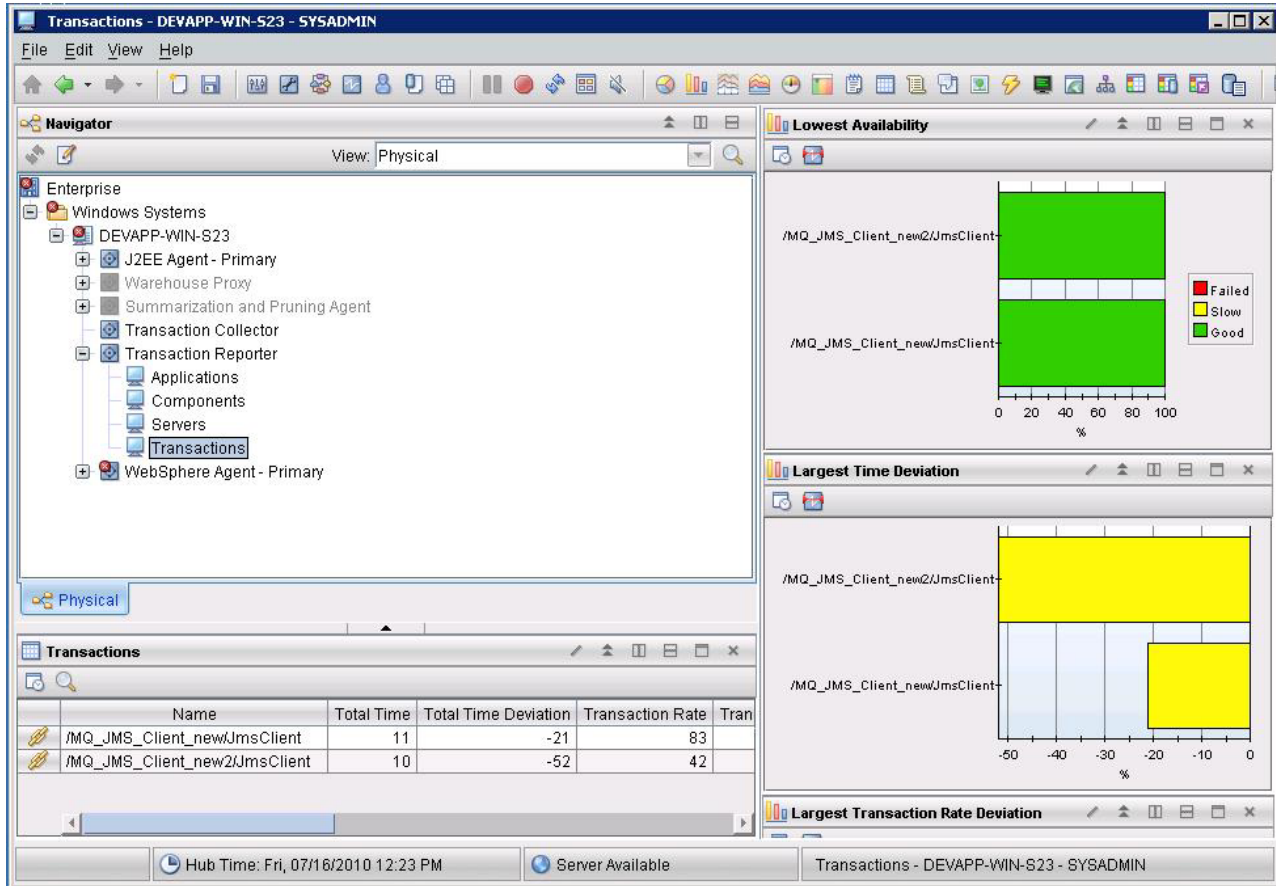


Figure 9. Transactions workspace

Servlet and JSP Request Integration

Details on integrating the monitoring of Servlet and JSP requests.

Servlet and JSP requests are integrated for JBoss, NetWeaver, and WebLogic application servers.

Table 2.

TTEMA TEP Workspace	Value of Name	Notes
Transactions	URI	Default value set by Data Collector
	URI + QueryString	If <code>com.ibm.tivoli.itcam.dc.ttapi.servlet.include.querystring=true</code> is specified in the server specific <code>toolkit_custom.properties</code> file

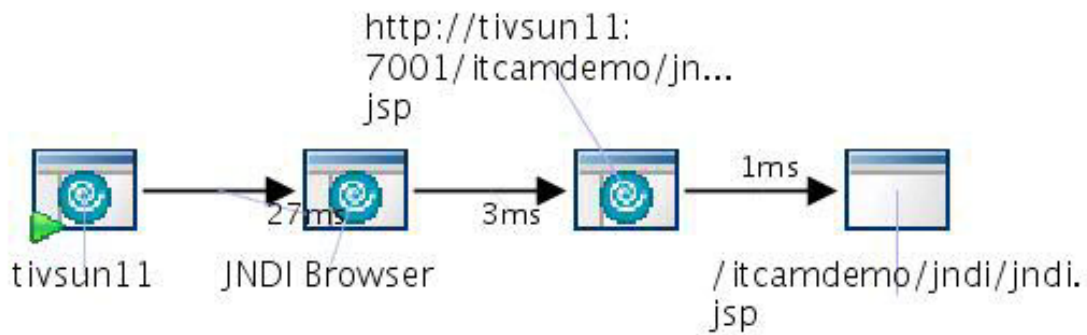


Figure 10. Servlet and JSP transactions

RMI and IIOP Request Integration

Details on integrating the monitoring of RMI and IIOP requests.

RMI and IIOP requests are integrated for JBoss and NetWeaver application servers.

Table 3.

TTEMA TEP Workspace	Value of Name	Notes
Transactions	RMI Client: Invoking calling Servlet/JSP URI or ClientRequestInterceptor:MethodName	
	RMI Server: ServerRequestInterceptor:MethodName	

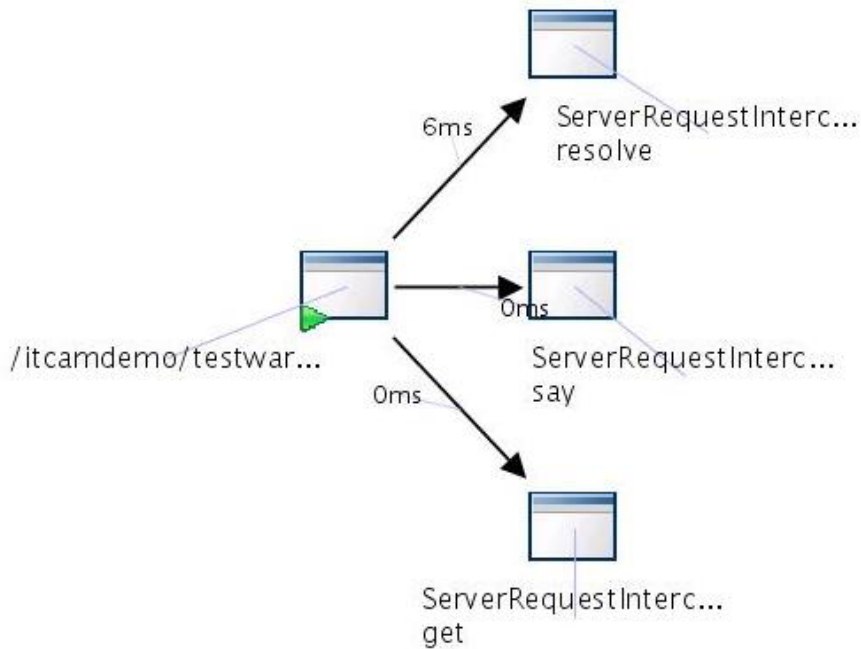


Figure 11. RMI/IIOP topology view in Transactions workspace

Web Services request integration

Details on integrating the monitoring of Web Services requests.

Web Services requests are integrated for JBoss and WebLogic application servers. For JBoss, only AXIS Web Services are supported.

Table 4.

TTEMA TEP Workspace	Value of Name Column	Note
Transactions	Client Side: Invoking Servlet/JPS or WS:WebServicePort:OperationName	
	Server Side: WS:WebServicePort:OperationName	



Figure 12. Web Services topology view in Transactions workspace

MQI request integration

Details on integrating the monitoring of MQI requests.

MQI requests are integrated for JBoss, NetWeaver, and WebLogic application servers. MQv7 JMS requests are not supported.

Table 5.

TTEMA TEP Workspace	Value of Name	Notes
Transactions	PUT/GET: Invoking Servlet/JPS or QueueManagerName:QueueName	

MQI request detail are available in the Managing Server Visualization Engine.

To track MQI transactions, you must enable MQ tracking using the Visualization Engine. See “Enabling and disabling MQ tracking” on page 7.

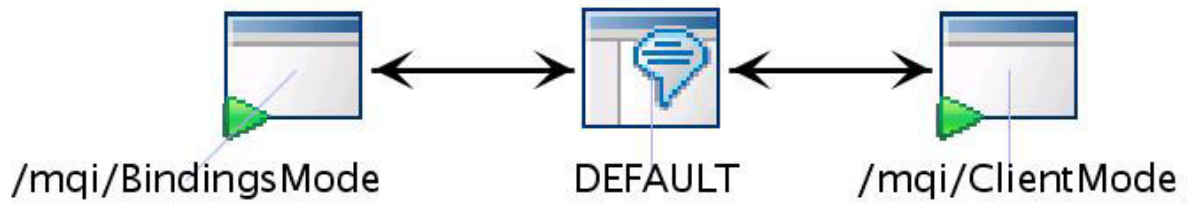


Figure 13. MQI topology view in Transactions workspace

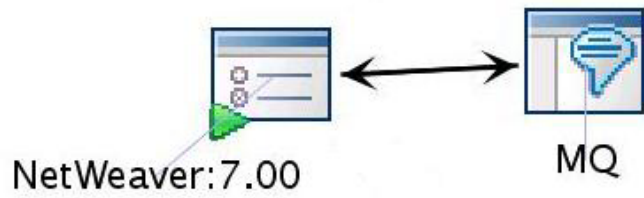


Figure 14. MQI topology view in Components workspace

EJB integration

Details on integrating the monitoring of EJB requests.

EJB requests are integrated for JBoss, NetWeaver, and WebLogic application servers.

Table 6.

TTEMA TEP Workspace	Value of Name	Notes
Transactions	EJB ClassName.methodName	Only the top-level EJB request is displayed.

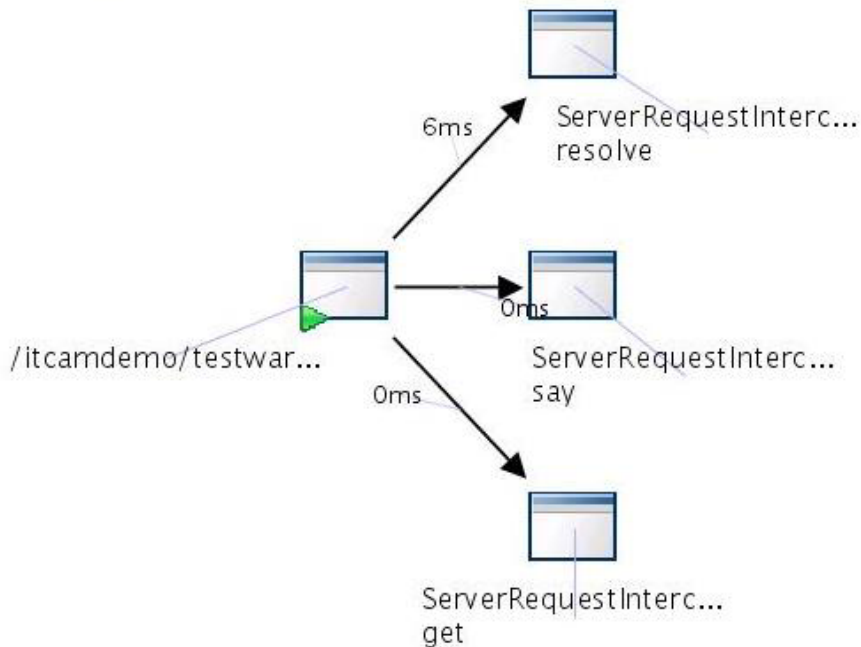


Figure 15. EJB topology view in Transactions workspace

Message Driven Bean integration

Details on integrating the monitoring of Message Driven Bean (MDB) requests.

MDB requests are integrated for JBoss, NetWeaver, and WebLogic application servers.

Table 7.

TTEMA TEP Workspace	Value of Name	Notes
Transactions	MDB className.onMessage	

Custom request integration

Details on integrating the monitoring of custom requests.

Custom requests are integrated for JBoss, NetWeaver, and WebLogic application servers.

Table 8.

TTEMA TEP Workspace	Value of Name	Notes
Transactions	RequestName defined in custom request configuration XML file.	Only the top-level Custom Request is displayed.

JDBC nested request integration

Details on integrating the monitoring of JDBC nested requests.

JDBC requests are integrated for JBoss, NetWeaver, and WebLogic application servers.

Table 9.

TTEMA TEP Workspace	Value of Name	Notes
Transactions	JDBC:dataSourceName JDBC:dataSourceName:hostName	

Attention: the host name is only reported if JDBC type 4 drivers are used. To track JDBC transactions when a Data Collector is at MOD Level 1, you must enable this tracking in the Visualization Engine; see “Enabling and disabling JDBC tracking at MOD Level 1” on page 9. If you do not enable it, JDBC transactions will be tracked for an application server instance only when it is monitored at MOD L2 or MOD L3. The monitoring level can be set in the Visualization Engine; see *ITCAM for Application Diagnostics User Guide*.



Figure 16. JDBC nested request view in Transactions workspace

Important: If a JDBC transaction reports any failures within a reporting period, the percentage of failed transactions will be displayed for the transaction. You can view individual successful and failed instances using the Transaction Instances view.

JMS messaging topology integration

Details on support for JMS messaging topology.

For the WebLogic application server, when JMS tracking is enabled (see “Enabling and disabling JMS tracking at MOD Level 1” on page 11), JMS links will be displayed between the following top-level request types:

- EJB, see “EJB integration” on page 22.
- Servlet

- Custom Request

The following JMS providers are supported:

- WebSphere MQ
- WebLogic JMS server

JMS topology for WebSphere MQ transactions is not displayed to avoid duplication. If the transaction target is set to MQ, enable MQ tracking to display its topology (see “Enabling and disabling MQ tracking” on page 7).

The following JMS messaging scenarios are supported for TTAPI:

- **queue Sender and queue Receiver.** A top-level request invokes the queueSender API to send a message to a queue. A top-level request invokes the queueReceiver API to receive a message from the queue. The application URLs displayed for the sender and receiver are the URLs of the top-level requests.

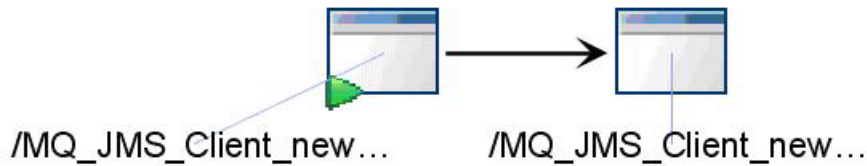


Figure 17. JMS example topology: queue sender and queue receiver

- **Topic Publisher and Topic Subscriber.** A top-level request invokes the TopicPublisher API to send a message to a Topic. One or several top-level requests may invoke the TopicSubscriber API to receive a message from the Topic. The application URLs displayed for the sender and receivers are the URLs of the top-level requests.

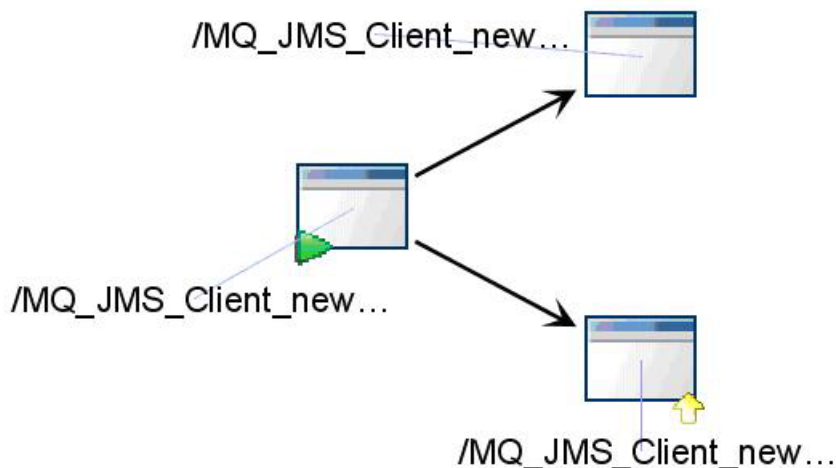


Figure 18. JMS example topology: topic publisher and topic subscriber

- **Message Sender and Message Driven Bean.** A top-level request sends a message to a queue or Topic. A Message Driven Bean that listens to the queue or Topic gets a callback on its onMessage method and receives the message. For the sender, the application URL is displayed; it is the URL of the top-level requests. For the receiver, the Message Driven Bean class name and method name are

displayed.

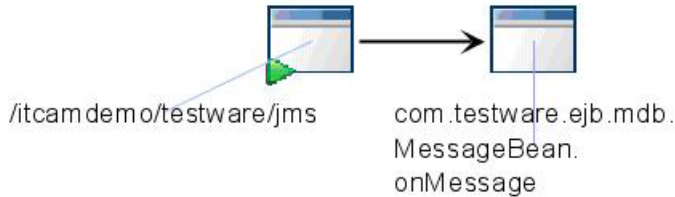


Figure 19. JMS example topology: message sender and message driven bean

Logging and tracing

Details on logging and tracing for Data Collector and Transaction Tracking Application Programming Interface (TTAPI) integration.

Logging and tracing for the Data Collector

How to enable logging and tracing for the Data Collector for the ITCAM for Application Diagnostics and TTAPI integration.

The following entries can be added to the `DCHOME/toolkit/etc/cynlogging.properties` file to trace Data Collector and Transaction Tracking Application Programming Interface integration:

```
# dc and ttapi integration tracing
CYN.trc.shared.datacollector.ttapi.TTAPIUtil.level=DEBUG_MAX
CYN.trc.shared.datacollector.ttapi.TTAPIUtil.logging=true
```

The following entry can be added to the `DCHOME/runtime/<platform.node.server>/custom/toolkit_custom.properties` file to record event information when the writing of the events fail:

```
com.ibm.tivoli.itcam.dc.ttapi.logExceptionEventRecs=true
```

The standard logging locations are:

For Windows® systems:

```
C:\Program Files\IBM\tivoli\common\CYN
```

For Linux® and Unix® systems:

```
/var/ibm/tivoli/common/CYN
```

Logging and tracing for the TTAPI and TEMA

TTAPI and Transaction Collector Tivoli Enterprise Monitoring Agent (TEMA) communication.

TTAPI can use the IBM Tivoli Monitoring standard RAS1 logging package to log error and debug messages at significant points in the process of initializing, shutting down, and sending events to a TTAS and various states in between. Logging can be controlled by the following environment variables:

Table 10.

Environment variable	Description
KBB_RAS1=ALL	Enable logging of all messages.

Table 10. (continued)

Environment variable	Description
KBB_RAS1=ERROR	Enable logging of error messages
KBB_RAS1=	Disable all message logging. This is the default value.
KBB_RAS1_LOG=	Log to standard output.
KBB_RAS1_LOG=. . .	Set the log file name and other parameters. See the information following this table.
KBB_VARPREFIX=%	Set the prefix for variables specified in KBB_RAS1_LOG

KBB_RAS1_LOG uses the following format:

```
KBB_RAS1_LOG=<filename> [INVENTORY=<inventory filename>] [COUNT=<count>]
[LIMIT=<limit>] [PRESERVE=<preserve>] [MAXFILES=<maxfiles>]
```

where:

<count> is the maximum number of log files to create in one invocation of the application.

<inventory> is a file in which to record the history of log files across invocations of the application.

<limit> is the maximum size per log file.

<maxfiles> is the maximum number of log files to create in any number of invocations of the application. This only takes effect when <inventory> is specified.

<preserve> is the number of log files to preserve when log files wrap over <count>.

For WebSphere Application Server, you can set the environment variables for RAS1 tracing through AdminConsole, as shown in the following example:

1. Navigate to **Server > Application Servers** and select the *ServerName*.
2. In the Configuration tab, navigate to **Server Infrastructure > Java and Process Management > Process Definition > Additional Properties: Custom Properties**.
3. Set the following environment variables:

```
KBB_RAS1=ALL
KBB_RAS1_LOG=c:\itcam71\tt71\ras1.1
```

If your server instance belongs to a Network Deployment cell, synchronize your change with NodeAgent.

Appendix. Accessibility

Accessibility features help a user who has a physical disability, such as restricted mobility or limited vision, to use software products successfully. These are the major accessibility features you can use with ITCAM for Application Diagnostics when accessing it via the IBM® *Personal Communications* terminal emulator:

- You can operate all features using the keyboard instead of the mouse.
- You can read text through interaction with assistive technology.
- You can use system settings for font, size, and color for all user interface controls.
- You can magnify what is displayed on your screen.

For more information on viewing PDFs from Adobe®, go to the following Web site:
<http://www.adobe.com/enterprise/accessibility/main.html>



Printed in USA