

IBM Tivoli Composite Application Manager
Version 7.1.0.2

*Agent for WebSphere Applications Data
Collector Installation and Configuraton
Guide for IBM i*



IBM Tivoli Composite Application Manager
Version 7.1.0.2

*Agent for WebSphere Applications Data
Collector Installation and Configuraton
Guide for IBM i*



Note

Before using this information and the product it supports, read the information in “Notices” on page 87.

2010

This 2010 edition applies to ITCAM for Application Diagnostics 7.1.0.2 and all subsequent releases and modifications until otherwise indicated in new editions.

© **Copyright IBM Corporation 2006, 2010.**

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

Contents

Figures	v
--------------------------	----------

Tables	vii
-------------------------	------------

About this publication ix

Intended audience	ix
Publications	ix
ITCAM for Application Diagnostics library	ix
Related publications	x
Accessing terminology online	xi
Accessing publications online	xi
Ordering publications	xi
Accessibility	xii
Tivoli technical training	xii
Support information	xii
Conventions used in this publication	xiii
Typeface conventions	xiii
Variables for directories	xiii

Part 1. Introduction to ITCAM Agent for WebSphere Applications Data Collector on IBM i 1

Chapter 1. IBM Tivoli Composite Application Manager Agent for WebSphere Applications IBM i Data Collector 3

Overview of the monitoring and diagnostic capabilities	3
IBM i Data Collector component.	5

Part 2. Installing and Configuring the Data Collector on IBM i 7

Chapter 2. Prerequisites and pre-installation tasks for the Data Collector on IBM i 9

System and software prerequisites	9
Required tasks before installation	9
Permissions.	9
Data Collector installation path	9
Removing -Xrun parameters	9
Increasing the heap size	10
Adjusting for ports being blocked by your firewall or being used by other applications	10
Installing and configuring the monitoring agent	10

Chapter 3. Silent Installation and configuration for the Data Collector on IBM i 13

Silent installation and initial configuration of the Data Collector	13
From the local IBM i computer	14
Using a remote attached Linux client	15
Using a remote attached Windows client	16
Installation of a new Data Collector maintenance level.	17
Silent configuration of the Data Collector	18
Preparing a response file	19
Additional configuration tasks	23
Setting up a secure connection to the Managing Server	23
IBM WebSphere Application Server 6.1: enabling MQ requests	23
IBM WebSphere Portal Server: increasing heap size and Transaction Service properties	24
Matching time zone settings with the Managing Server	24
Completing and verifying Data Collector configuration	24

Chapter 4. Uninstalling the Data Collector on IBM i 25

Part 3. Advanced configuration of the Data Collector 27

Chapter 5. Customization and advanced configuration for the Data Collector 29

Properties files for the Data Collector.	29
Tuning Data Collector performance and monitoring scope	30
Data Collector internal buffering and turbo mode settings.	31
Enabling instrumentation and monitoring of RMI/IIOP requests between application servers	32
Disabling various types of Byte Code Instrumentation for J2EE APIs	33
Controlling instrumentation of application classes for lock analysis, memory leak analysis, and method profiling and tracing	35
Defining custom requests.	42
Enabling Asynchronous Bean request monitoring	44
Customizing monitoring of custom MBeans	45
Modifying Performance Monitoring Infrastructure settings	47
Enabling Performance Monitoring Infrastructure settings for the Service Integration Bus	48
Enabling and disabling instrumentation of Web Services as new request types	49
Enabling and disabling memory monitoring	49
Changing Managing Server communication configuration	50

Steps to perform if the IP address of the application server host is to be changed	50
Steps to support changing application the type of virtual machine for the Java platform.	50
Moving the Data Collector to a different host computer	51
Configuring a Data Collector for multiple network interfaces	51
Customizing RMI garbage collection interval	52
Customizing CICS transaction correlation	52
Modifying the garbage collection log path	54
Overriding the Data Collector autoconfiguration	55
Properties for communication with a Deployment Manager	55

Part 4. Appendixes. 57

Appendix A. Setting up security 59

Node Authentication	59
Script to run if your SSL certificates have expired	59
Node Authentication on the Managing Server	59
Data Collector custom properties file changes	60
Keystore management and populating certificates	60
Secure Socket Layer communications	62
Password encryption and Kernel property file encryption.	63
Enabling Secure Socket Layer at the Data Collector level	64
Verifying secure communications	64
Privacy filtering	65
Enabling privacy filtering.	65
Script to run if your SSL certificates have expired.	66
Settings for the Data Collector if Java 2 security is enabled.	67

Appendix B. Starting and stopping the monitoring environment 69

Disabling and re-enabling a Data Collector	69
Restarting the application server	69
Restarting the application server in a non-Network Deployment environment	69
Restarting the application server in a Network Deployment environment.	70

Starting the application server	70
Starting the application server in a non-Network Deployment environment.	70
Starting the application server in a Network Deployment environment.	71
Stopping the application server.	71
Stopping the application server in a non-Network Deployment environment.	71
Stopping the application server in a Network Deployment environment.	72

Appendix C. Using regular expressions 73

Regular expression library	73
Frequently used regular expressions	73
Specifying exclusions with the bang (!) operator (Quality of Service listening policies only)	74

Appendix D. Support information 75

Searching knowledge bases	75
Finding Release Notes.	75
Searching the information center on your local system or network	76
Searching the Internet	76
Obtaining fixes	76
Receiving Weekly support updates	77
Contacting IBM Software Support	77
Determining the business impact of your problem	78
Describing your problem and gathering background information	78
Submitting your problem to IBM Software Support	78
Tivoli Support Technical Exchange.	79

Appendix E. Accessibility. 81

Index 83

Trademarks. 85

Notices 87

Figures

1. Agent interaction with IBM Tivoli Monitoring 4
2. Agent interaction with ITCAM for Application
Diagnostics Managing Server 5

Tables

1. Removing service programs	25	9. Default Performance Monitoring Infrastructure instrumentation settings	48
2. Adding lines to toolkit_custom.properties	33	10. Procedures to customize instrumentation of the Performance Monitoring Infrastructure	48
3. Modifying lines in toolkit_custom.properties	34	11. JVM options for garbage collection logging	54
4. Byte Code Instrumentation configuration files	35	12. Location of the CYND4051I message	64
5. Parameters for the memory leak diagnosis configuration file.	39	13. Classification of the data processed on the CommandAgent channel	65
6. Parameters for the Level 3 method entry and exit analysis configuration file	41	14. Criteria for assessing the business impact of your problem	78
7. Parameters for the custom requests configuration file.	43		
8. Parameters for the JMX MBean configuration file	45		

About this publication

This publication provides information about installing, customizing, starting, and maintaining IBM® Tivoli® Composite Application Manager Agent for WebSphere® Applications Data Collector on IBM i.

Intended audience

This publication is for administrators or advanced users wanting to install or modify the configuration of ITCAM Agent for WebSphere Applications. The publication assumes that readers are familiar with maintaining operating systems, administering Web servers, maintaining databases, and general information technology (IT) procedures. Specifically, readers of this publication must have some knowledge of the following topics:

- Operating systems on which you intend to install product components
- Web servers, such as IBM HTTP Server and Apache HTTP Server
- Web application servers, such as IBM WebSphere
- Internet protocols such as HTTP, HTTPS, TCP/IP, Secure Sockets Layer (SSL), and Transport Layer Security (TLS)
- Digital certificates for secure communication

Publications

This section lists publications in the product library and related documents. It also describes how to access Tivoli publications online and how to order Tivoli publications.

ITCAM for Application Diagnostics library

The following publications are included in the ITCAM for Application Diagnostics library, available at http://publib.boulder.ibm.com/infocenter/tivihelp/v24r1/topic/com.ibm.itcamfad.doc_7101/ic-homepage.html:

- *IBM Tivoli Composite Application Manager for Application Diagnostics: Prerequisites*
Provides the hardware and software requirements for installing ITCAM for Application Diagnostics components.
- *IBM Tivoli Composite Application Manager for Application Diagnostics: User's Guide*
Provides the user overview, user scenarios, and Helps for every ITCAM for Application Diagnostics component.
- *IBM Tivoli Composite Application Manager for Application Diagnostics: Planning an Installation*
Provides the user with a first reference point for a new ITCAM for Application Diagnostics installation or upgrade.
- ITCAM Agent for WebSphere Applications Installation and Configuration Guides:
 - *IBM Tivoli Composite Application Manager: Agent for WebSphere Applications Installation and Configuration Guide*
 - *IBM Tivoli Composite Application Manager: Agent for WebSphere Applications Installation and Configuration Guide for z/OS*

- *IBM Tivoli Composite Application Manager: Agent for WebSphere Applications Data Collector Installation and Configuration Guide for IBM i*

Provide installation instructions for setting up and configuring ITCAM Agent for WebSphere Applications on distributed, z/OS®, and IBM i systems.

- ITCAM Agent for J2EE Applications Installation and Configuration Guides:

- *IBM Tivoli Composite Application Manager: Agent for J2EE Data Collector Installation and Configuration Guide*
- *IBM Tivoli Composite Application Manager: Agent for J2EE Monitoring Agent Installation and Configuration Guide*

Provide installation instructions for setting up and configuring ITCAM Agent for J2EE.

- *IBM Tivoli Composite Application Manager: Agent for HTTP Servers Installation and Configuration Guide*

Provides installation instructions for setting up and configuring ITCAM Agent for HTTP Servers.

- *IBM Tivoli Composite Application Manager for Application Diagnostics Managing Server Installation Guide*

Provides installation instructions for setting up and configuring ITCAM for Application Diagnostics Managing Server.

- *IBM Tivoli Composite Application Manager for Application Diagnostics: Troubleshooting Guide*

Provides instructions on problem determination and troubleshooting for ITCAM for Application Diagnostics.

- *IBM Tivoli Composite Application Manager for Application Diagnostics: Messaging Guide*

Provides information about system messages received when installing and using ITCAM for Application Diagnostics.

Related publications

The following documentation also provides useful information:

- IBM Tivoli Documentation Central:

Information about IBM Tivoli Documentation is provided on the following Web site:

<http://www.ibm.com/tivoli/documentation.html>

- IBM WebSphere Application Server:

Information about IBM WebSphere Application Server is provided on the following Web site:

<http://www.ibm.com/software/webservers/appserv/was/library>

- IBM DB2®:

Information about IBM DB2 is provided on the following Web site:

<http://www.ibm.com/software/data/sw-library/>

- IBM Tivoli Enterprise Console®:

Information about IBM Tivoli Enterprise Console is provided on the following Web site:

<http://submit.boulder.ibm.com/tividd/td/EnterpriseConsole3.9.html>

- IBM Tivoli Data Warehouse:

Information about IBM Tivoli Data Warehouse is provided on the following Web site:

<http://publib.boulder.ibm.com/infocenter/tivihelp/v3r1/index.jsp?toc=/com.ibm.tivoli.tdwi.doc/toc.xml>

- IBM Tivoli Change and Configuration Management Database:
Information about IBM Tivoli Change and Configuration Management Database is provided on the following Web site:
http://publib.boulder.ibm.com/infocenter/tivihelp/v10r1/index.jsp?toc=/com.ibm.ccldb.doc/ccldb_ic.xml
- IBM Support Assistant:
Information about IBM Support Assistant is provided on the following Web site:
<http://www.ibm.com/software/support/isa/index.html?rccs=rtlrrr>

Accessing terminology online

The *Tivoli Software Glossary* includes definitions for many of the technical terms related to Tivoli software. The *Tivoli Software Glossary* is available at the following Tivoli software library Web site:

<http://publib.boulder.ibm.com/tividd/glossary/tivoliglossarymst.htm>

The IBM Terminology Web site consolidates the terminology from IBM product libraries in one convenient location. You can access the Terminology Web site at the following Web address:

<http://www.ibm.com/software/globalization/terminology>

Accessing publications online

The documentation CD contains the publications that are in the product library. The format of the publications is PDF, HTML, or both.

IBM posts publications for this and all other Tivoli products, as they become available and whenever they are updated, to the Tivoli software information center Web site. Access the Tivoli documentation center at the following Web address:

<https://www.ibm.com/developerworks/wikis/display/tivolidoccentral/Home>

Access the Tivoli Information Center for ITCAM for Application Diagnostics at the following Web address:

http://publib.boulder.ibm.com/infocenter/tivihelp/v24r1/topic/com.ibm.itcamfad.doc_7101/ic-homepage.html

Note: If you print PDF documents on other than letter-sized paper, set the option in the **File > Print** window that enables Adobe Reader to print letter-sized pages on your local paper.

The IBM Software Support Web site provides the latest information about known product limitations and workarounds in the form of technotes for your product. You can view this information at the following Web site:

<http://www.ibm.com/software/support>

Ordering publications

You can order many Tivoli publications online at the following Web site:

<http://www.elink.ibm.link.ibm.com/public/applications/publications/cgibin/pbi.cgi>

You can also order by telephone by calling one of these numbers:

- In the United States: 800-879-2755
- In Canada: 800-426-4968

In other countries, contact your software account representative to order Tivoli publications. To locate the telephone number of your local representative, perform the following steps:

1. Go to the following Web site:
<http://www.elink.ibm.link.ibm.com/public/applications/publications/cgibin/pbi.cgi>
2. Select your country from the list and click **Go**.
3. Click **About this site** in the main panel to see an information page that includes the telephone number of your local representative.

Accessibility

Accessibility features help users with a physical disability, such as restricted mobility or limited vision, to use software products successfully. With this product, you can use assistive technologies to hear and navigate the interface. You can also use the keyboard instead of the mouse to operate all features of the graphical user interface.

For additional information, see Appendix E, “Accessibility,” on page 81.

Tivoli technical training

For Tivoli technical training information, refer to the following IBM Tivoli Education Web site:

<http://www.ibm.com/software/tivoli/education/>

Support information

If you have a problem with your IBM software, you want to resolve it quickly. IBM provides the following ways for you to obtain the support you need:

Online

Go to the IBM Software Support at the following Web site:

<http://www.ibm.com/software/support/>

Follow the instructions.

IBM Support Assistant

The IBM Support Assistant (ISA) is a free local software serviceability workbench that helps you resolve questions and problems with IBM software products. The ISA provides quick access to support-related information and serviceability tools for problem determination. To install the ISA software, see the instructions for installing ISA in the Data Collector installation guide.

Troubleshooting Guide

For more information about resolving problems, see the corresponding part in *IBM Tivoli Composite Application Manager for Application Diagnostics: Troubleshooting Guide*.

Conventions used in this publication

This publication uses several conventions for special terms and actions, operating system-dependent commands and paths, and margin graphics.

Typeface conventions

This publication uses the following typeface conventions:

Bold

- Lowercase commands and mixed case commands that are otherwise difficult to distinguish from surrounding text
- Interface controls (check boxes, push buttons, radio buttons, spin buttons, fields, folders, icons, list boxes, items inside list boxes, multicolumn lists, containers, menu choices, menu names, tabs, property sheets), labels (such as **Tip:**, and **Operating system considerations:**)
- Keywords and parameters in text

Italic

- Citations (examples: titles of publications, diskettes, and CDs)
- Words defined in text (example: a nonswitched line is called a *point-to-point line*)
- Emphasis of words and letters (words as words example: "Use the word *that* to introduce a restrictive clause."; letters as letters example: "The LUN address must start with the letter *L*.")
- New terms in text (except in a definition list): a *view* is a frame in a workspace that contains data.
- Variables and values you must provide: ... where *myname* represents....

Monospace

- Examples and code examples
- File names, programming keywords, and other elements that are difficult to distinguish from surrounding text
- Message text and prompts addressed to the user
- Text that the user must type
- Values for arguments or command options

Variables for directories

This guide refers to the following variables:

- *DC_home*: the directory where the Data Collector files are installed. The location is set at installation time.
- *AppServer_home*: the directory where the application server core product files are installed.

Example: /QIBM/UserData/WebSphere/AppServer/V61/ND/

Part 1. Introduction to ITCAM Agent for WebSphere Applications Data Collector on IBM i

Chapter 1. IBM Tivoli Composite Application Manager Agent for WebSphere Applications IBM i Data Collector

This chapter introduces the ITCAM Agent for WebSphere Applications IBM i Data Collector and explains how it can help you monitor, administer, and diagnose your IBM i systems that run IBM WebSphere Application Server.

Overview of the monitoring and diagnostic capabilities

IBM Tivoli Composite Application Manager Agent for WebSphere Applications is a component of ITCAM for Application Diagnostics, Version 7.1.0.2. It is also a component of ITCAM for Applications Version 6.2.3. If you are using ITCAM for Applications, the Managing Server (deep dive) functionality is not available; please ignore all references to this functionality in this document.

ITCAM Agent for WebSphere Applications can function within two different infrastructures: IBM Tivoli Monitoring and ITCAM for Application Diagnostics Managing Server.

The IBM Tivoli Monitoring environment places this agent into the context of the IBM Tivoli Monitoring family, a suite of products used to monitor a mixed-systems environment. With IBM Tivoli Monitoring, the user can:

- Monitor for alerts on the managed systems
- Trace the causes leading up to an alert
- Monitor processing time for various requests within WebSphere applications
- Establish your own performance thresholds
- Create custom situations, which are conditions that IBM Tivoli Monitoring automatically monitors
- Create and send commands to control system monitoring using the Take Action feature
- Create comprehensive reports about system conditions
- Define your own queries, using the attributes provided with ITCAM Agent for WebSphere Applications, to monitor conditions of particular interest to you

The Tivoli Enterprise Portal is the user interface for the IBM Tivoli Monitoring environment. It provides an overall view of the enterprise network; from this view, the user can "drill down" to examine components of the environment more closely. The Portal includes information from different agents that monitor various parts of the environment; ITCAM Agent for WebSphere Applications is one of them.

For details on capabilities of IBM Tivoli Monitoring, and information on deploying the IBM Tivoli Monitoring infrastructure, see *IBM Tivoli Monitoring: Installation and Setup Guide*.

Figure 1 on page 4 shows how ITCAM Agent for WebSphere Applications interacts with other IBM Tivoli Monitoring components.

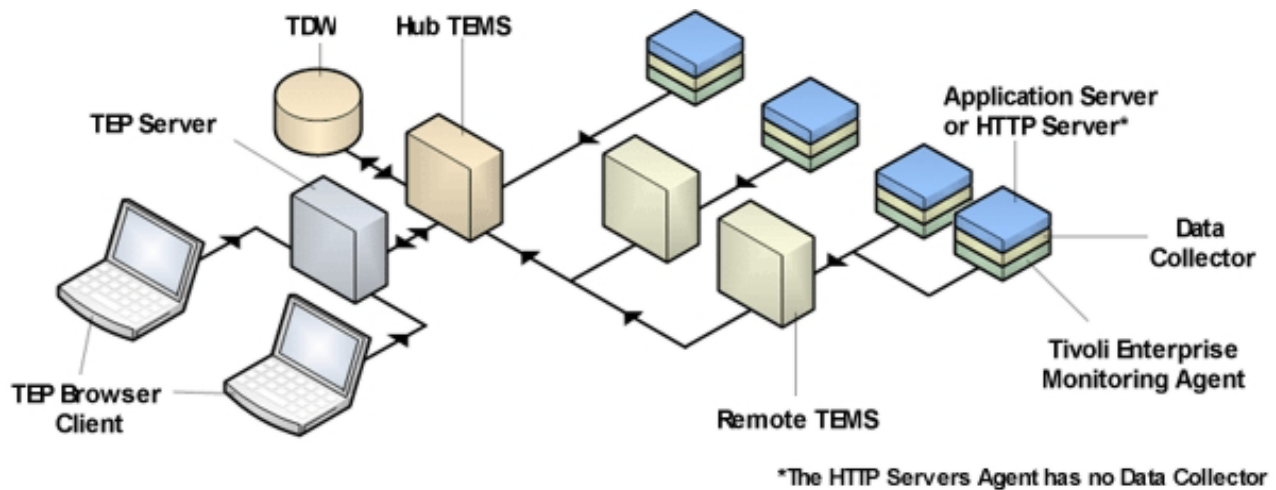


Figure 1. Agent interaction with IBM Tivoli Monitoring

The Managing Server is a component of ITCAM for Application Diagnostics. Its Visualization Engine provides a user interface for "deep dive" diagnostics information. The user can "click through" or "launch in context" to the Visualization Engine from the Tivoli Enterprise Portal when detailed information is required. The Visualization Engine can also be used as a stand alone user interface; this user interface is a good solution for software developers and performance analysts.

Most information provided by ITCAM Agent for WebSphere Applications and available through the Tivoli Enterprise Portal can also be viewed through the Visualization Engine. The Visualization Engine also provides additional diagnostic information, including:

- Method entry/exit and stack tracing,
- Lock analysis,
- Heap object analysis for memory leak diagnosis,
- Thread information,
- "In-flight" request analysis to detect malfunctioning applications.

For details on the capabilities of ITCAM for Application Diagnostics Managing Server, and information on deploying it, see *IBM Tivoli Composite Application Manager for Application Diagnostics Managing Server Installation Guide*.

The diagram on Figure 2 on page 5 shows how ITCAM Agent for WebSphere Applications interacts with the components of the Managing Server. (The Data Collector is a component of the Agent).

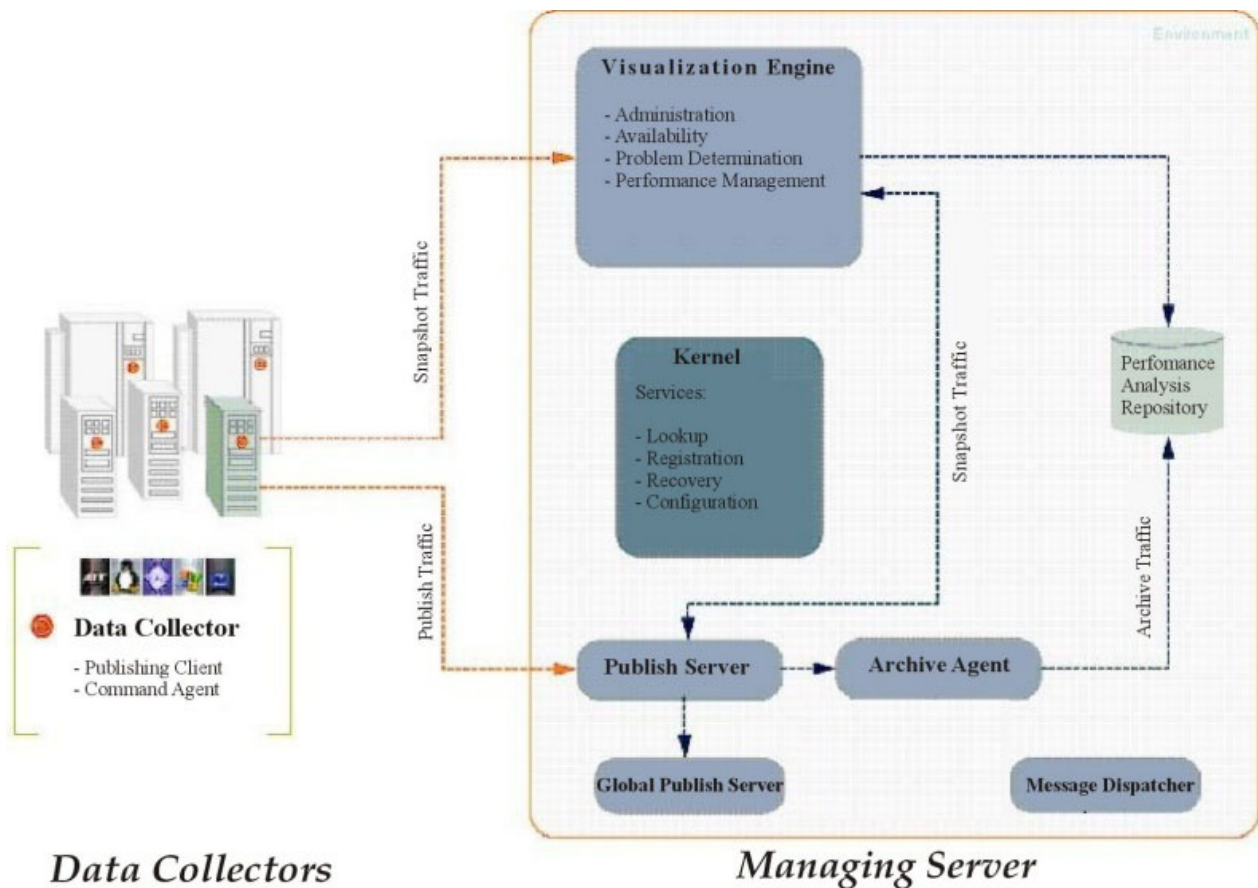


Figure 2. Agent interaction with ITCAM for Application Diagnostics Managing Server

IBM i Data Collector component

ITCAM Agent for WebSphere Applications on IBM i consists of two components: the *Data Collector* and the *Managing Server*. On an IBM i host, only the Data Collector is deployed. This Data Collector can interact with ITCAM for Application Diagnostics Managing Server directly. To work with IBM Tivoli Monitoring, the Data Collector uses a Monitoring Agent installed on a Windows, Linux, or UNIX host.

The Data Collector collects monitoring and diagnostics information from the application server using the following methods:

- In *Byte Code Instrumentation (BCI)*, the Data Collector injects monitoring calls into the Java code that processes application requests. Data is collected on request processing time and on different types of J2EE API calls within each request. These requests are sometimes referred to as *nested requests*. BCI monitoring requirements can be different; while on a production system request level monitoring might be sufficient, on a test or development system, or when a problem is being investigated, it can be used to instrument application method entry and exit, synchronized methods, and object allocation. BCI uses a certain amount of system resources, depending on the amount of injected calls. The level of detail, and thus the use of resources, is determined by the *monitoring level*, which can be set for every monitored application server. With IBM Tivoli

Monitoring, levels L1 and L2 are supported; with ITCAM for Application Diagnostics Managing Server, the additional level L3 is available. The monitoring level can be set independently for IBM Tivoli Monitoring and the Managing Server, and for every application server instance.

- *Performance Monitoring Interface (PMI)* is an API provided by IBM WebSphere Application Server, supplying a number of performance metrics.
- *Garbage Collection logs* are written by IBM WebSphere Application Server, and contain detailed information about the garbage collection process. Such information can be useful for application monitoring and enhancement.

You must install the Data Collector on the monitored IBM i host. If the Managing Server infrastructure is used, set up the Data Collector to communicate to the Managing Server.

To monitor WebSphere on the IBM i host using IBM Tivoli Monitoring, you need a Monitoring Agent installed on a Windows, Linux, or UNIX host. Configure the Data Collector to communicate to this Monitoring Agent. One Monitoring Agent can support Data Collectors on multiple IBM i hosts.

Attention: Because the monitoring agent is not available on IBM i, application server log information is not reported.

Part 2. Installing and Configuring the Data Collector on IBM i

Chapter 2. Prerequisites and pre-installation tasks for the Data Collector on IBM i

This chapter includes tasks that need to be performed before installing a Data Collector on IBM i.

Begin this chapter by verifying that your computer meets the system and software prerequisites. See “System and software prerequisites.”

System and software prerequisites

The software and hardware requirements before installing ITCAM for Application Diagnostics are listed at the following Web site:

<https://www.ibm.com/developerworks/wikis/display/tivolimonitoring/Prerequisites+for+ITCAM+for+Application+Diagnostics+7.1>

Required tasks before installation

Perform the tasks in each of the following sections before you attempt to install the Data Collector on IBM i.

Permissions

The user who installs the Data Collector must have at least *SYSADMIN privileges.

In addition, add by command line the following privileges for the application server user:

```
GRTOBJAUT OBJ(QSYS/STRDBG) OBJTYPE(*CMD) USER(QEJBSVR) AUT(*USE)
GRTOBJAUT OBJ(QSYS/ENDSRVJOB) OBJTYPE(*CMD) USER(QEJBSVR) AUT(*USE)
GRTOBJAUT OBJ(QSYS/STRSRVJOB) OBJTYPE(*CMD) USER(QEJBSVR) AUT(*USE)
GRTOBJAUT OBJ(QSYS/DMPJVM) OBJTYPE(*CMD) USER(QEJBSVR) AUT(*USE)
GRTOBJAUT OBJ(QSYS/ANZJVM) OBJTYPE(*CMD) USER(QEJBSVR) AUT(*USE)
```

In IBM i version 6.1, also add the following privileges for all users:

```
GRTOBJAUT OBJ(QSYS/QPLPMGT) OBJTYPE(*SRVPGM) USER(*public) AUT(*use)
```

Remote installation

The IBM i user running a remote installation must have the *ALLOBJ authority.

Data Collector installation path

Make sure the planned Data Collector installation path (for example, /QIBM/ProdData/itcamfwas/server1was60) does not exist in the Integrated File System (IFS) before the installation.

Removing -Xrun parameters

Perform the following procedure to remove -Xrun parameters from Generic JVM arguments:

1. Log into the IBM WebSphere Application Server administrative console where the target server instance resides.

2. Click **Server > Application Servers** and select the *server_name*.
3. In the **Configuration** tab, navigate to **Server Infrastructure > Java and Process Management > Process Definition > Additional Properties: Java Virtual Machine**.
4. If there are any **-Xrun** parameters in the **Generic JVM arguments** field, delete them.

Increasing the heap size

For each of the application servers to be monitored by the Data Collector, increase the heap size by performing the following procedure:

1. Log into the IBM WebSphere Application Server administrative console.
2. Click **Server > Application Servers** and select the *server_name*.
3. In the **Configuration** tab, navigate to **Server Infrastructure > Java and Process Management > Process Definition > Additional Properties: Java Virtual Machine**.
4. Edit the field **Maximum Heap Size**. For best Data Collector performance, set it to ***NOMAX**. If the value is not specified, ***NOMAX** is the default.

Adjusting for ports being blocked by your firewall or being used by other applications

At various times during the installation you will need to specify or accept the defaults for port numbers used by the Data Collector.

By default, the Data Collector will communicate in the following ways:

- If the IBM Tivoli Monitoring infrastructure is used, the Data Collector will make outbound connections to the Monitoring Agent host. The default port for these connection is 63335.
- If the ITCAM for Application Diagnostics Managing Server is used, and the Data Collector is configured for one or more application server instances, it will need to open ports in the 8200 to 8399 range for inbound communication.
- With WebSphere Network Deployment or Extended Deployment, the Data Collector will make outbound connections to the Deployment Manager host. The port number is available in the Deployment Manager administrative console.

You need to ensure that these connections are not blocked by a firewall. If they are blocked, you must either modify the communication settings during installation and configuration of the Data Collector, or change the settings of the firewall. To determine the connections that your firewall may block, see the documentation supplied with the firewall.

If you are using ITCAM for Application Diagnostics Managing Server, you also need to make sure that ports used for inbound communication are not used by other applications. If they are used by other applications, you will need to change the ports for Data Collector inbound communication when configuring the Data Collector. To list the ports used by other applications, run the command **netstat *CNN** from the IBM i main menu.

Installing and configuring the monitoring agent

If you need to monitor WebSphere servers on IBM i using the IBM Tivoli Monitoring infrastructure, you need to install a Monitoring Agent on a Windows, Linux, or UNIX system. The Data Collector will communicate with this Monitoring Agent.

The Monitoring Agent is installed as a part of ITCAM Agent for WebSphere Applications. You may use one installation of the monitoring agent to support Data Collectors on several IBM i hosts. The same Monitoring Agent can also support the Data Collectors on its own Windows, Linux, or UNIX host.

Make sure that the port the monitoring agent uses to listen for Data Collector communication (by default, 63335) is not blocked by a firewall.

Also, make sure that the monitoring agent is configured to communicate with a Tivoli Enterprise Monitoring Server, and that Application Support files for ITCAM Agent for WebSphere Applications are installed on the hub Tivoli Enterprise Monitoring Server, Tivoli Enterprise Portal Server, and all Tivoli Enterprise Portal desktop clients.

For information on installing ITCAM Agent for WebSphere Applications, see *IBM Tivoli Composite Application Manager: Agent for WebSphere Applications Installation and Configuration Guide*.

Chapter 3. Silent Installation and configuration for the Data Collector on IBM i

You can use commands and responses file to install and configure the Data Collector on IBM i. Installation through the response file without user interaction is called *silent installation*. This is the only installation method available on IBM i. You must also perform additional tasks on every instance of the application server that you configure for monitoring by the Data Collector.

To install the Data Collector on an IBM i host where it is not already installed, or to upgrade from the Data Collector for ITCAM for WebSphere version 6.1 or ITCAM For WebSphere Application Server version 7.0 or 7.1 (shipped with WebSphere Application Server), use the silent installer utility, supplied in the full installation image. See “Silent installation and initial configuration of the Data Collector.”

To install a new maintenance level of the Data Collector on an IBM i host where the ITCAM for Agent for WebSphere Applications Data Collector version 7.1 is already installed, use the update installer, supplied in the fix pack image. See “Installation of a new Data Collector maintenance level” on page 17.

To perform other configuration tasks, use the silent configuration utility. See “Silent configuration of the Data Collector” on page 18.

Silent installation and initial configuration of the Data Collector

On IBM i, use the silent installer utility to install the Data Collector files. At the same time, the utility can also configure the Data Collector for application server instances. If the Data Collector from ITCAM for WebSphere version 6.1 or ITCAM For WebSphere Application Server version 7.0 or 7.1 (shipped with WebSphere Application Server) is installed, use the silent installer to upgrade to the new version of the Data Collector.

To use the silent installer utility, perform one of the following procedures:

- “From the local IBM i computer” on page 14
- “Using a remote attached Linux client” on page 15
- “Using a remote attached Windows client” on page 16

Within the procedures, you will need to prepare a response file by editing a copy of one of the samples provided in the installation package. This file determines the installation and configuration tasks that will be performed. In the response file, you must set the installation directory and the parameters for communication with the Managing Server and the Tivoli Enterprise Monitoring Agent, as well as the information on the application server instances to be monitored. See “Preparing a response file” on page 19.

Do not use the silent installer utility to perform any configuration tasks when the Data Collector is already installed. Instead, use the silent configuration utility. See “Silent configuration of the Data Collector” on page 18.

If the Data Collector for ITCAM Agent for WebSphere Applications version 7.1 is already installed, do not use the silent installer utility to install a new maintenance

level. Use the update installer provided in the fix pack package; see “Installation of a new Data Collector maintenance level” on page 17.

You can run the silent installer utility from a local IBM i computer, or using a Windows or Linux client.

From the local IBM i computer

Perform the following procedure if you are performing the silent installation on the local IBM i computer:

Procedure

1. Log on to the computer on which you want to install the Data Collector as a user with the proper permissions. (See “Permissions” on page 9.)
2. Copy the installation images to an IFS directory on the IBM i computer.
3. Use the Start TCP/IP (STRTCP) command to start TCP/IP.
4. Use the Start Host Servers (STRHOSTSVR) command to start Host Servers.
5. Enter the QSH environment by running the STRQSH command.
6. If you plan to configure or unconfigure the Data Collector for monitoring application server instances:
 - For a stand-alone deployment, verify that the instance of the application server that will be monitored by the Data Collector is running. For each instance, at a command prompt, enter the following command from the *AppServer_home/profiles/profile_name/bin* directory:

```
serverStatus server_name
```

If the instance of the application server is not started, perform the instructions in “Starting the application server” on page 70.
 - For a Network Deployment or Extended Deployment, verify that Deployment Manager and Node Agents for the instances of the application server that will be monitored by the Data Collector are running.
7. Create a temporary directory as the installation working directory. For example,

```
mkdir /tmp/itcam_i_dc_install
```
8. Copy the DC installation package (iDC_itcam71.tar) to the installation working directory.
9. Extract this file in the installation working directory. For example,

```
cd /tmp/itcam_i_dc_install  
/qopensys/usr/bin/tar xvf iDC_itcam71_full.tar
```
10. Prepare the response file for the installation and configuration tasks that you need to perform. Copy the *dcInputs_Sample_os400.txt* sample response file. See “Preparing a response file” on page 19. Save this file in the installation working directory as *dcInputs.txt*.
11. Set the JAVA_HOME environment variable to point to the classic JDK, making sure the java program is accessible at \$JAVA_HOME/bin/, for example:

```
export JAVA_HOME=/QIBM/ProdData/Java400/jdk14
```
12. Change the permissions for the installer script to make it executable. The script is located in the *Installer-DC/bin* subdirectory under the installation working directory. For example,

```
chmod +x /tmp/itcam_i_dc_install/Installer-DC/bin/install-DC-os400.sh
```
13. Run the installer script from the *Installer-DC/bin* subdirectory under the installation working directory. For example,

```
cd /tmp/itcam_i_dc_install/Installer-DC/bin
./install-DC-os400.sh hostname user password
```

The *hostname* specifies the IBM i host name.

The *user* specifies the IBM i user.

The *password* specifies the password for the IBM i user.

Note: The installation writes log files into a subdirectory named *installdate_time.txt* in the installation working directory. The log files from the latest installation are also copied into the installation directory itself; their names are *install-dc-msg.log* and *install-dc-trc.log*.

14. If you have configured the Data Collector to monitor any application server instances, perform the additional post-configuration tasks for these instances, See “Additional configuration tasks” on page 23.

Using a remote attached Linux client

Perform the following procedure if you are performing the silent installation using an attached remote Linux client:

Procedure

1. Copy the installation images to a hard disk on the Linux client.
2. Log on to the IBM i computer on which you want to install the Data Collector as a user with the proper permissions. (See “Permissions” on page 9.) Perform the following operations on this computer:
 - a. Use the Start TCP/IP (STRTCP) command to start TCP/IP.
 - b. Use the Start Host Servers (STRHOSTSVR) command to start Host Servers.
 - c. If you plan to configure or unconfigure the Data Collector for monitoring application server instances:
 - For a stand-alone deployment, verify that the instance of the application server that will be monitored by the Data Collector is running. For each instance, at a command prompt, enter the following command from the *AppServer_home/profiles/profile_name/bin* directory:

```
serverStatus server_name
```

If the instance of the application server is not started, perform the instructions in “Starting the application server” on page 70.
 - For a Network Deployment or Extended Deployment, verify that Deployment Manager and Node Agents for the instances of the application server that will be monitored by the Data Collector are running.
3. On the Linux client, create a temporary directory as the installation working directory. For example,

```
mkdir /tmp/itcam_i_dc_install
```
4. Copy the DC installation package (*iDC_itcam71.tar*) to the installation working directory
5. Extract this file in the installation working directory. For example,

```
cd /tmp/itcam_i_dc_install
tar xvf iDC_itcam71_full.tar
```
6. Prepare the response file for the installation and configuration tasks that you need to perform. Copy the *dcInputs_Sample_os400.txt* sample response file. See “Preparing a response file” on page 19. Save this file in the installation working directory as *dcInputs.txt*.

7. Set the JAVA_HOME environment variable, making sure the java program is accessible at \$JAVA_HOME/bin/.
8. Change the permissions for the installer script to make it executable. The script is located in the Installer-DC/bin subdirectory under the installation working directory. For example,

```
chmod +x /tmp/itcam_i_dc_install/Installer-DC/bin/install-DC-os400.sh
```

9. Run the installer script from the Installer-DC/bin subdirectory under the installation working directory. For example,

```
cd /tmp/itcam_i_dc_install/Installer-DC/bin
./install-DC-os400.sh hostname user password
```

The *hostname* specifies the IBM i host name.

The *user* specifies the IBM i user.

The *password* specifies the password for the IBM i user.

Note: The installation writes log files into a subdirectory named *installdate_time.txt* in the installation working directory. The log files from the latest installation are also copied into the installation directory itself; their names are *install-dc-msg.log* and *install-dc-trc.log*.

10. If you have configured the Data Collector to monitor any application server instances, perform the additional post-configuration tasks for these instances, See “Additional configuration tasks” on page 23.

Using a remote attached Windows client

Perform the following procedure if you are performing the silent installation using an attached remote Windows client:

Procedure

1. Copy the installation images to a hard disk on the Windows client.
2. Log on to the IBM i computer on which you want to install the Data Collector as a user with the proper permissions. (See “Permissions” on page 9.) Perform the following operations on this computer:
 - a. Use the Start TCP/IP (STRTCP) command to start TCP/IP.
 - b. Use the Start Host Servers (STRHOSTSVR) command to start Host Servers.
 - c. If you plan to configure or unconfigure the Data Collector for monitoring application server instances:
 - For a stand-alone deployment, verify that the instance of the application server that will be monitored by the Data Collector is running. For each instance, at a command prompt, enter the following command from the *AppServer_home/profiles/profile_name/bin* directory:


```
serverStatus server_name
```

If the instance of the application server is not started, perform the instructions in “Starting the application server” on page 70.
 - For a Network Deployment or Extended Deployment, verify that Deployment Manager and Node Agents for the instances of the application server that will be monitored by the Data Collector are running.
3. On the Windows client, create a temporary directory as the installation working directory. For example, from a Windows command line:


```
mkdir %TEMP%\itcam_i_dc_install
```

4. Copy the DC installation package (iDC_itcam71.tar) to the installation working directory
 5. Extract the iDC_itcam71.tar file in the installation working directory. You can use archive management programs such as **7zip** to extract it.
 6. Prepare the response file for the installation and configuration tasks that you need to perform. Copy the dcInputs_Sample_os400.txt sample response file. See “Preparing a response file” on page 19. Save this file in the installation working directory as dcInputs.txt.
 7. Open a Windows console window and set the JAVA_HOME environment variable, making sure the java.exe program is accessible at %JAVA_HOME%\bin.
 8. Run the installer script from the Installer-DC\bin directory under the installation working directory. For example,


```
cd %TEMP%\itcam_i_dc_install\Installer-DC\bin
install-DC-os400.bat host_name user_name password
```

The *hostname* specifies the IBM i host name.
 The *user* specifies the IBM i user.
 The *password* specifies the password for the IBM i user.
- Note:** The installation writes log files into a subdirectory named `installdate_time.txt` in the installation working directory. The log files from the latest installation are also copied into the installation directory itself; their names are `install-dc-msg.log` and `install-dc-trc.log`.
9. If you have configured the Data Collector to monitor any application server instances, perform the additional post-configuration tasks for these instances, See “Additional configuration tasks” on page 23.

Installation of a new Data Collector maintenance level

If the Data Collector version 7.1 is already installed on an IBM i host, use the update installer supplied with a fix pack to install a new maintenance level. The old maintenance level is not overwritten when the new one is installed.

About this task

To install a new maintenance level, you need to acquire the fix pack package. Is it not the same as the full installation package.

Then perform the following procedure on the local IBM i computer.

Procedure

1. Log on to the computer on which the Data Collector is installed as a user with the proper permissions. (See “Permissions” on page 9.)
2. Copy the fix pack package to an IFS directory on the IBM i computer.
3. Enter the QSH environment by running the STRQSH command.
4. Create a temporary directory as the update working directory. For example,


```
mkdir /tmp/itcam_i_dc_update
```
5. Copy the fix pack package to the update working directory.
6. Extract this file in the installation working directory. For example,


```
cd /tmp/itcam_i_dc_update
/qopensys/usr/bin/tar xvf 7.1.0-TIV-ITCAMfAD_MP-FP0001_i.tar
```

7. Edit the `silentUpdate.properties` file. Set `product.location` to the directory where the Data Collector is installed. Do not make any other changes to the file.
8. Set the `JAVA_HOME` environment variable to point to the classic JDK, making sure the `java` program is accessible at `$JAVA_HOME/bin/`, for example:

```
export JAVA_HOME=/QIBM/ProdData/Java400/jdk14
```
9. Run the configuration script: ,

```
silentUpdate.sh -install
```
10. To update monitoring of application server instances to the new maintenance level, and to remove the old maintenance level if necessary, use the silent configuration utility. See “Silent configuration of the Data Collector.”

Silent configuration of the Data Collector

When the Data Collector is already installed, use the silent configuration utility to configure it for additional application server instances and to perform other configuration tasks.

About this task

You can perform the following installation and configuration tasks using the silent configuration utility:

- With the Data Collector already installed, configure it to monitor application server instances.
- Reconfigure monitoring of application server instances.
Attention: You must reconfigure monitoring of an application server instance after changing the Developer Kit for Java platform (virtual machine) type for this application server.
- Unconfigure the Data Collector from monitoring application server instances.
- Upgrade monitoring of application server instances that were monitored by the Data Collector for ITCAM for WebSphere version 6.1 or ITCAM For WebSphere Application Server version 7.0 or 7.1 (shipped with WebSphere Application Server).
- Change the Data Collector maintenance level for monitoring of application server instances. To do this, you need to install the maintenance level first; see “Installation of a new Data Collector maintenance level” on page 17.
- Remove unused Data Collector maintenance levels.

Run the silent configuration utility on the local IBM i computer.

Procedure

1. Log on to the computer on which the Data Collector is installed as a user with the proper permissions. (See “Permissions” on page 9.)
2. Enter the QSH environment by running the `STRQSH` command.
3. If you plan to configure or unconfigure the Data Collector for monitoring application server instances:
 - For a stand-alone deployment, verify that the instance of the application server that will be monitored by the Data Collector is running. For each instance, at a command prompt, enter the following command from the `AppServer_home/profiles/profile_name/bin` directory:

```
serverStatus server_name
```

If the instance of the application server is not started, perform the instructions in “Starting the application server” on page 70.

- For a Network Deployment or Extended Deployment, verify that Deployment Manager and Node Agents for the instances of the application server that will be monitored by the Data Collector are running.
4. Change to the `DC_home/config_dc` directory.
 5. Prepare the response file for the configuration task that you need to perform. In the response file, depending on the task, you might need to set the parameters for communication with the Managing Server and the Tivoli Enterprise Monitoring Agent, as well as the information on the monitored application server instances. Copy one of the files from the `Samples` subdirectory. See “Preparing a response file.” Save this file in the installation working directory as `dcInput.txt`.
 6. If the classic JDK is not installed in the `/QIBM/ProdData/Java400/jdk14` directory, set the `JAVA_HOME` environment variable to point to the classic JDK, making sure the `java` program is accessible at `$JAVA_HOME/bin/`, for example:

```
export JAVA_HOME=/QIBM/ProdData/Java400/jdk14
```

Tip: The default value is `/QIBM/ProdData/Java400/jdk14`, so if the classic JDK is in this directory, you can skip this step.
 7. Run the configuration script: ,

```
config-DC-os400.sh dcInput.txt
```
 8. If you have configured the Data Collector to monitor any application server instances, perform the additional post-configuration tasks for these instances, See “Additional configuration tasks” on page 23.

Preparing a response file

After unpacking the installation image, you need to prepare a response file. This file determines the installation and configuration tasks that the installer utility will perform.

The response file is a text file, containing parameter names and values in the format *parameter=value*, for example:

```
KERNEL_HOST01=servername.domain.com
```

Comment lines begin with a number sign (#). Do not use blank lines.

Any `\` character must be escaped as `\\`, and spaces must be prefixed with `\`, for example:

```
MS_AM_HOME=C:\\Progra~1\\ITCAM\\MS
```

In the file sections marked as “repeatable”, parameters are specific to an application server instance. For these parameters, use any unique value as a key, in the format *parameter.key=value*. If you are only configuring a single instance, or if the value of a parameter is the same for all the instances you are configuring, use the parameter name without the key. For example:

```
# instance names
WS_FULLNAME.instance1=cells/CellName/nodes/NodeName/servers/server1
WS_FULLNAME.instance2=cells/CellName/nodes/NodeName/servers/server1

# the WebSphere Application Server home directory will be the same
# for both instances
WS_HOME=/QIBM/UserData/WebSphere/AppServer/V7/Base/profiles/profile1
```

In the response file, depending on the task, you might need to set the installation directory, the parameters for communication with the Managing Server and the Tivoli Enterprise Monitoring Agent, as well as the information on the application server instances to be monitored.

Installing and configuring the Data Collector

To install the Data Collector on an IBM i host and, if necessary, configure it to monitor application server instances, create a copy of the `dcInputs_Sample_os400.txt` file. This file is located in the `Installer-DC/etc` subdirectory of the unpacked installation image. Save this copy as `dcinputs.txt` in the installation working directory, where you have unpacked the installation image.

Edit the file in the following way:

- Set `INSTALL=true`.
- Set the `ITCAM_home` value to the directory to which the Data Collector will be installed. (If a previous version of the Data Collector is installed on the IBM i host, use a different directory from that version).
- If you do not need to configure the Data Collector to monitor application server instances at this time, set `CONFIGURE_TYPE=none`. Otherwise, set `CONFIGURE_TYPE=configure`, and complete the other parameters in the file according to the comments provided in it.

Use the file with the installer utility; see “Silent installation and initial configuration of the Data Collector” on page 13.

Configuring an installed Data Collector to monitor application server instances

To configure the Data Collector to monitor application server instances, create a copy of the `dcInputs_Sample_os400.txt` file. This file is located in the `DC_home/config_dc/Samples` subdirectory. Save this copy as `DC_home/config_dc/dcInput.txt`.

Edit the file in the following way:

- Set `INSTALL=false`.
- Complete the other parameters in the file according to the comments provided in it.

Use the file with the configuration utility; see “Silent configuration of the Data Collector” on page 18.

Reconfiguring monitoring for application server instances

You can reconfigure monitoring for application server instances already monitored by the Data Collector. You can change Managing Server communication and Monitoring Agent communication parameters, and set the Developer Kit for Java platform (virtual machine) type for this application server.

To do this, create a copy of the `dcInputs_Reconfig_Sample_os400.txt` file. This file is located in the `DC_home/config_dc/Samples` subdirectory. Save this copy as `DC_home/config_dc/dcInput.txt`.

Edit the file and complete the parameters according to the comments provided in it.

Use the file with the configuration utility; see “Silent configuration of the Data Collector” on page 18.

Unconfiguring the Data Collector from monitoring application server instances

To unconfigure the Data Collector from monitoring application server instances, create a copy of the `dcInputs_Unconfigure_Sample_os400.txt` file. This file is located in the `DC_home/config_dc/Samples` subdirectory. Save this copy as `DC_home/config_dc/dcInput.txt`.

Edit the file in the following way:

- Edit the file and complete the parameters according to the comments provided in it.

Use the file with the configuration utility; see “Silent configuration of the Data Collector” on page 18.

Upgrade installation of the Data Collector

To install the Data Collector on an IBM i host when the Data Collector for ITCAM for WebSphere version 6.1 or ITCAM For WebSphere Application Server version 7.0 (shipped with WebSphere Application Server) is already installed, create a copy of the `dcInputs_Upgrade_Sample_os400.txt` file. This file is located in the `Installer-DC/etc` subdirectory of the unpacked installation image. Save this copy as `dcinputs.txt` in the installation working directory, where you have unpacked the installation image.

By default, this procedure will also upgrade monitoring of the application server instance that was configured automatically when the previous version of the Data Collector was installed. To upgrade other application server instances, you will need to provide information on them in the response file.

Important: The following cases require different procedures:

- To update the ITCAM Agent for WebSphere Applications Data Collector version 7.1 to a new maintenance level, use the maintenance level update procedure, see “Install a new maintenance level of the Data Collector” on page 22.
- To upgrade from ITCAM For WebSphere Application Server version 7.1, install the new Data Collector without configuring it (see “Installing and configuring the Data Collector” on page 20), and then use the silent configuration utility to upgrade monitoring from the ITCAM For WebSphere Application Server Data Collector (see “With an installed Data Collector, upgrading monitoring of application server instances from a previous version of the Data Collector” on page 22).

Edit the file in the following way:

- Set `INSTALL=true`.
- Set the `ITCAM_home` value to the directory to which the Data Collector will be installed. Do not use the same directory as the previous Data Collector version.
- Set the `DC_INSTALL_HOME` value to the directory to which the previous version of the Data Collector was installed.
- If you do **not** need to upgrade application server instance that was configured automatically when the Data Collector version 6.1 was installed, use the `WS_SKIP_UPGRADE` parameter.

- To upgrade monitoring of other application server instances, complete the other parameters in the file according to the comments provided in it.

Use the file with the installer utility; see “Silent installation and initial configuration of the Data Collector” on page 13.

With an installed Data Collector, upgrading monitoring of application server instances from a previous version of the Data Collector

If the new Data Collector is already installed, to upgrade monitoring of application server instances that were monitored by the Data Collector for ITCAM for WebSphere version 6.1 or ITCAM For WebSphere Application Server version 7.0 or 7.1 (shipped with WebSphere Application Server), create a copy of the `dcInputs_Upgrade_Sample_os400.txt` file. This file is located in the `DC_home/config_dc/Samples` subdirectory. Save this copy as `DC_home/config_dc/dcInput.txt`.

Important: Do not use this procedure to update the ITCAM Agent for WebSphere Applications Data Collector version 7.1 to a new maintenance level.

Edit the file in the following way:

- Set `INSTALL=false`.
- Set the `DC_INSTALL_HOME` value to the directory to which the previous version of the Data Collector was installed.
- Complete the other parameters in the file according to the comments provided in it.

Use the file with the configuration utility; see “Silent configuration of the Data Collector” on page 18.

Install a new maintenance level of the Data Collector

To install a new maintenance level of the Data Collector, when a 7.1 version of the ITCAM Agent for WebSphere Applications Data Collector is already installed, edit the `silentUpdate.properties` file, located in the directory of the unpacked fix pack image.

Set `product.location` to the directory where the Data Collector is installed.

Do not make any other changes to the file.

Use the file with the Update Installer; see “Installation of a new Data Collector maintenance level” on page 17.

Changing monitoring of application server instances to a different maintenance level of the Data Collector

If a new maintenance level for the Data Collector is already installed, to change monitoring of application server instances to the new maintenance level (or revert back to an older maintenance level), create a copy of the `dcInputs_Update_Sample_os400.txt` file. This file is located in the `DC_home/config_dc/Samples` subdirectory. Save this copy as `DC_home/config_dc/dcInput.txt`.

Edit the file in the following way:

- Set the *ITCAM_home* value to the directory to which the Data Collector was installed.
- Complete the other parameters in the file according to the comments provided in it.

Use the file with the configuration utility; see “Silent configuration of the Data Collector” on page 18.

Removing unused Data Collector maintenance levels

If a maintenance level for the Data Collector is no longer used, to remove it, create a copy of the *dcInputs_Remove_Unused_Sample_os400.txt* file. This file is located in the *DC_home/config_dc/Samples* subdirectory. Save this copy as *DC_home/config_dc/dcInput.txt*.

Important: You will not be able to remove a maintenance level if any application server instance is still monitored by the Data Collector at this level. Once no application server instance is monitored by a maintenance level, you can remove it but are not required to do so.

Edit the file in the following way:

- Set the *ITCAM_home* value to the directory to which the Data Collector was installed.
- Complete the other parameters in the file according to the comments provided in it.

Use the file with the configuration utility; see “Silent configuration of the Data Collector” on page 18.

Additional configuration tasks

Perform the instructions in each of the following sections, if applicable.

Setting up a secure connection to the Managing Server

If the Data Collector will communicate with ITCAM for Application Diagnostics Managing Server, you may need to set up a secure connection.

See Appendix A, “Setting up security,” on page 59 for more information on setting up a secure (SSL) connection between the Data Collector and the Managing Server.

IBM WebSphere Application Server 6.1: enabling MQ requests

If you have IBM WebSphere Application Server 6.1, perform the following procedure to enable MQ requests. (This is the data that is displayed in the Application Monitor user interface when selecting **Performance Analysis > Create Application Report > MQI**.)

1. Stop the application server that will be monitored by the Data Collector. See “Stopping the application server” on page 71.
2. Enter the *AppServer_home/profiles/profile_name/bin* directory for the application server that will be monitored by the Data Collector.
3. Run the *osgiCfgInit* script:
osgiCfgInit

IBM WebSphere Portal Server: increasing heap size and Transaction Service properties

If you have IBM WebSphere Portal Server, perform the following procedure to ensure successful communication with the application server:

1. Log into the IBM WebSphere Application Server administrative console.
2. Click **Server > Application Servers** and select the *server_name*.
3. In the **Configuration** tab, navigate to **Server Infrastructure > Java and Process Management > Process Definition > Additional Properties: Java Virtual Machine**.
4. Modify the initial heap size to be 512.
5. Modify the maximum heap size to be 1024.
6. Click **Server > Application Servers** and select the *server_name*.
7. Click **Container Services > Transaction Service**.
8. Change the total transaction lifetime timeout from 120 seconds to 12000 seconds.
9. Change the client inactivity timeout to 120 seconds.

Matching time zone settings with the Managing Server

If the Data Collector will communicate with ITCAM for Application Diagnostics Managing Server, ensure that the user time zone for the Data Collector is the same as the time zone setting for the Managing Server:

1. Edit the file `/QIBM/UserData/Java400/SystemDefault.Properties`. If the file does not exist create a new one.
2. Add or modify the setting `user.timezone` to the same time zone setting as the Managing Server.

Completing and verifying Data Collector configuration

To finish and verify configuration of the Data Collector for an application server instance, complete the following steps:

1. Restart the instance of the application server that will be monitored by the Data Collector. See “Restarting the application server” on page 69.
2. If you are using the IBM Tivoli Monitoring infrastructure, start a Tivoli Enterprise Portal client and verify that you can see monitored data for the application server instance.
3. If you are using the ITCAM for Application Diagnostics Managing Server infrastructure, access the Visualization Engine and verify that you can see monitored data for the application server instance.

Chapter 4. Uninstalling the Data Collector on IBM i

This section describes how to uninstall the Data Collector.

Perform the following procedure to uninstall the Data Collector:

1. Unconfigure the Data Collector from monitoring all application server instances. See “Silent installation and initial configuration of the Data Collector” on page 13.
2. Remove the files and service program:

CAUTION:

Because the Data Collector on IBM i uses the same service program to monitor all application server instances, do not remove the service program unless the Data Collector has been unconfigured from all application server instances.

- a. On the IBM i server, remove the Data Collector installation path in the Integrated File System (IFS):

```
rm -rf DC_home
```
- b. On the IBM i server, remove the service program. The following table lists the commands to delete the service programs:

Table 1. Removing service programs

Version 6.0 application servers	DLTSRVPGM AM710S14
Version 6.1 application servers	DLTSRVPGM AM710S15
Version 7 application servers	DLTSRVPGM AM710S16

Part 3. Advanced configuration of the Data Collector

Chapter 5. Customization and advanced configuration for the Data Collector

This section contains instructions for customizing your configuration of the Data Collector.

Perform the procedures in each of the following sections, if they apply.

Properties files for the Data Collector

Several properties files control Data Collector configuration and behavior.

These files, as well as other files used by the Data Collector, are located under the *DC_home* directory. This is the directory into which the Data Collector was installed (the *ITCAM_home* parameter in the silent installation response file, see “Silent installation and initial configuration of the Data Collector” on page 13).

For most common changes to this configuration, you will need to edit the Data Collector properties file and the Toolkit properties file. Several other properties files also exist.

The Data Collector properties file

The Data Collector properties file is automatically created by the installer, and is unique for every application server instance monitored by the Data Collector. Its name is *DC_home/runtime/appserver_version.node_name.server_name/*datacollector.properties.

However, to facilitate future upgrades, **do not change** this file.

Instead, add the settings you want to modify to the Data Collector custom properties file. This file is named *DC_home/runtime/app_server_version.node_name.server_name/custom/*datacollector_custom.properties ; this will override the values in the Data Collector properties file.

Note: If the *DC_home/runtime/app_server_version.node_name.server_name/custom/*datacollector_custom.properties file does not exist, create it when you need to make changes. You may also need to create the custom directory.

The Toolkit properties file

The Toolkit properties file is automatically created by the Data Collector at startup, using various input files. It is unique for every application server instance monitored by the Data Collector. Its name is *DC_home/runtime/appserver_version.node_name.server_name/toolkit.properties*.

Because this file is re-created at each Data Collector startup, **do not make any changes** to this file as they will be overwritten.

Instead, add the settings you want to modify to the Data Collector custom properties file. This file is named *DC_home/runtime/*

`app_server_version.node_name.server_name/custom/toolkit_custom.properties` ; this will override the values in the Toolkit properties file.

You may also set toolkit properties for all the application server instances monitored by this installation of the Data Collector. To do this, add the settings to the global toolkit custom properties file: `DC_home/runtime/custom/toolkit_global_custom.properties` . However, if a property is set in the instance specific `toolkit_custom.properties` file, it will override the value in the global file for this instance.

Note: If the `DC_home/runtime/app_server_version.node_name.server_name/custom/toolkit_custom.properties` or `DC_home/runtime/custom/toolkit_custom.properties` file does not exist, create it when you need to make changes. You may also need to create the custom directory.

Other properties files

The following properties files are unique for every application server instance monitored by the Data Collector:

- `DC_home/runtime/app_server_version.node_name.server_name/cynlogging.properties` defines the log file names and logging details for the Java portion of the Data Collector.
- `DC_home/runtime/app_server_version.node_name.server_name/cyncclog.properties` defines the log file names and logging details for the C++ portion of the Data Collector. This file has EBCDIC encoding.
- `DC_home/runtime/appserver_version.node_name.server_name/kwjdc.properties` defines communication with the monitoring agent, including the host name and port for the monitoring agent host.

Tuning Data Collector performance and monitoring scope

The Data Collector monitors the performance of the application server in several ways. This monitoring introduces a performance overhead. The scope and accuracy of the monitoring can vary; but, when more information is gathered, the performance overhead is increased.

The monitoring scope is broadly determined by the monitoring level, which the user can set as necessary.

In the Tivoli Enterprise Portal, the available monitoring levels are L1 and L2. In the Visualization Engine, monitoring levels (known as Monitoring On Demand, or MOD, levels) are L1, L2, and L3.

This level is set independently for IBM Tivoli Monitoring and the Managing Server. For example, the user may set monitoring level L1 for IBM Tivoli Monitoring from the Tivoli Enterprise Portal, and at the same time set MOD L2 in the Managing Server Visualization Engine. In this case, only L1 data will be available in the Tivoli Enterprise Portal, but L2 information will be displayed in the Visualization Engine.

You can also set the *sampling rate* for the Managing Server and Monitoring Agent independently. For the Managing Server, the sampling rate determines the percentage of monitored requests that are archived in the database; irrespective of the sampling rate, all data is sent to the Managing Server, so the resource usage of the Data Collector is not affected by it. The sampling rate is set separately for

every Data Collector installation. Having a low sampling rate does not prevent the user from seeing requests that have hung in-flight, nor does it prevent Managing Server traps and Tivoli Enterprise Portal situations from working on all requests. Generally, a 2% sampling rate is suggested for MOD L1 Tivoli monitoring data collection in a production environment where data is often stored for 15 to 30 days or more.

Note: For Managing Server data collection, the sampling rate does not apply to Custom Request and Nested Request monitoring.

You can also fine tune the Data Collector monitoring process using the properties files. This impacts the performance overhead, as well as the scope and accuracy of the monitoring. While the default configuration is broadly acceptable for common situations, you can use the properties files to reach the performance and monitoring that closely match the requirements of your environment.

Data Collector internal buffering and turbo mode settings

The following parameters affect buffering in communication between the Data Collector and ITCAM for Application Diagnostics Managing Server. In most cases, the default settings are appropriate. Do not change these parameters unless directed by IBM Software Support. These settings do not affect communication with the Monitoring Agent.

Internal Buffering settings

The following parameters in the Data Collector properties file (see “The Data Collector properties file” on page 29) control internal buffering in the Data Collector.

internal.memory.limit

The default value is 100 (MB). This property limits the amount of memory the Data Collector may use for all its buffering needs. Reducing this setting can lower the memory overhead introduced by the Data Collector; however, it can also increase the probability of buffer overflow at MOD L2 or L3 during periods of very high transaction volume. You can also reduce buffer load by limiting the monitoring scope for MOD L2 and L3, using the settings in the rest of this chapter, especially “Controlling instrumentation of application classes for lock analysis, memory leak analysis, and method profiling and tracing” on page 35.

internal.memory.accept.threshold

The default value is 2 (MB). Once the amount of memory specified in `internal.memory.limit` is reached, a buffer overflow state happens, and data will not be buffered. This property specifies the minimum amount of free memory to be reached before buffering is resumed.

internal.url.limit

The default value is 1000. This property controls the maximum URL length accepted by the Data Collector. If your URL length typically exceeds this value, it should be increased to avoid display truncation.

internal.sql.limit

The default value is 5000. This property controls the maximum SQL length accepted by the Data Collector. If your SQL statement length is typically greater than this value, the value should be increased to avoid display truncation.

internal.probe.event.queue.size.limit

The default value is 900000. This property controls the maximum size of the queue of events maintained by the Data Collector.

internal.probe.event.packet.size

The default value is 5000 Kbytes. Changing the default is not recommended. Valid values are 1 to 4000000 (or up to available process memory on the server). This property specifies the size of the Data Collector internal send buffer. The send buffer controls how much data the Data Collector can be sent to the Publish Server at a given time. In normal situations, this property does not have to be changed, as the default send buffer size is more than adequate.

Turbo Mode

If the Data Collector cannot send data to the Managing server fast enough, Data collector buffer space may be exhausted. The Data Collector may react in two ways, depending on whether *turbo mode* is enabled or disabled.

Turbo mode is controlled by the **dc.turbomode.enabled** setting in the Data Collector properties file (see "The Data Collector properties file" on page 29). Set it to true to enable turbo mode, or to false to disable it. By default, turbo mode is disabled.

If turbo mode is enabled, the Data Collector will enter turbo mode when three quarters of the maximum memory size (set in `internal.memory.limit`) is used. In turbo mode, the Data Collector raises the priority of the threads that send data to the Managing Server and the Monitoring Agent. Application threads remain at a lower priority, and may freeze until the buffer shortage is relieved. This ensures maximum reporting accuracy but can have a significant impact on application performance.

In turbo mode, the Data Collector also continues to monitor any existing requests, but any new incoming requests will not be monitored until the buffer shortage is relieved. The application will process them, and the Data Collector will notify the Managing Server of a "dropped" request.

If turbo mode is disabled, the Data Collector will drop any monitoring data in the event of a buffer shortage. Therefore, information in the Tivoli Enterprise Portal and Visualization Engine may be incomplete, but application performance will not be impacted.

Disable turbo mode on "mission critical" production systems, where any application thread freeze is not acceptable in any situation. However, a danger of buffer shortage (and, therefore, engagement of turbo mode) usually happens only in MOD L2 or L3; on production systems, users need to enable these levels only when investigating an issue. Turbo mode may be of use in such an investigation. So, if a significant performance impact in the event of an investigation is acceptable, you can enable turbo mode.

Enabling instrumentation and monitoring of RMI/IIOP requests between application servers

If the Data Collector will communicate with ITCAM for Application Diagnostics Managing Server, and two or more application servers are using Remote Method Invocation over Internet InterORB Protocol (RMI/IIOP), you need to enable

instrumentation and monitoring of RMI/IIOP requests in order to view composite requests (via correlation icons) in the Visualization Engine.

All the servers must be instrumented by Data Collectors connected to the same Managing Server.

For all the application servers, in the Toolkit custom properties file (see “The Toolkit properties file” on page 29), add or uncomment the following property:

```
org.omg.PortableInterceptor.ORBInitializerClass.com.ibm.tivoli.itcam.toolkit.  
ai.orbinterceptor.Initializer=true
```

Disabling various types of Byte Code Instrumentation for J2EE APIs

In Byte Code Instrumentation (BCI), the Data Collector intercepts method entry and exit calls for various types of Java APIs in order to create an execution flow of each application request. Some resources are used for the monitoring. You can tune the Data Collector so that some of the APIs are not monitored, reducing resource use. For some APIs, you can also disable collection of BCI information for MOD L1, reducing resource use for this level (typically enabled most of the time on production systems).

To disable BCI monitoring for J2EE APIs, set (or uncomment) the following properties in the toolkit custom properties file (see “The Toolkit properties file” on page 29). These properties are set to true by default. After changing the file, restart the application server instance.

Table 2. Adding lines to *toolkit_custom.properties*

Type of J2EE API	Line to add to <i>toolkit_custom.properties</i> file
Enterprise JavaBeans (EJB)	<code>com.ibm.tivoli.itcam.toolkit.ai.enableejb=false</code>
Java Connector Architecture (JCA)	<code>com.ibm.tivoli.itcam.toolkit.ai.enablejca=false</code>
Java Database Connectivity (JDBC)	<code>com.ibm.tivoli.itcam.toolkit.ai.enablejdbc=false</code>
Java Naming and Directory Interface (JNDI)	<code>com.ibm.tivoli.itcam.toolkit.ai.enablejndi=false</code>
Java Message Service (JMS)	<code>com.ibm.tivoli.itcam.toolkit.ai.enablejms=false</code>
Web containers for Servlets/JavaServer Pages (JSP)	<code>com.ibm.tivoli.itcam.dc.was.webcontainer=false</code>
HTTP session count tracking	<code>com.ibm.tivoli.itcam.toolkit.ai.enablesessioncount=false</code>
CICS® Transaction Gateway (CTG)	<code>com.ibm.tivoli.itcam.dc.ctg.enablectg=false</code>
IMS™	<code>com.ibm.tivoli.itcam.dc.mqi.enableims=false</code>
Java Data Objects (JDO)	<code>com.ibm.tivoli.itcam.dc.mqi.enablejdo=false</code>

Table 2. Adding lines to toolkit_custom.properties (continued)

Type of J2EE API	Line to add to toolkit_custom.properties file
Message Queue Interface (MQI)	com.ibm.tivoli.itcam.dc.mqi.enablemqi=false
Axis web service	com.ibm.tivoli.itcam.toolkit.ai.axis.enablewebservice=false
Remote Method Invocation (RMI)	am.ejb.rmilistener.enable=false
IBM WebSphere Application Server EJB container	com.ibm.tivoli.itcam.dc.was.enableEJBContainer=false
WebSphere Portal Server 5.1	com.ibm.tivoli.itcam.dc.was.wps51.enable.CaptureWPServlet=false com.ibm.tivoli.itcam.dc.was.wps51.enable.CaptureAuthorization=false com.ibm.tivoli.itcam.dc.was.wps51.enable.WPSDistributedMapCache=false com.ibm.tivoli.itcam.dc.was.wps51.enable.CaptureGatewayServlet=false com.ibm.tivoli.itcam.dc.was.wps51.enable.CaptureLogin=false com.ibm.tivoli.itcam.dc.was.wps51.enable.CaptureModelBuilding=false com.ibm.tivoli.itcam.dc.was.wps51.enable.CapturePageLoading=false com.ibm.tivoli.itcam.dc.was.wps51.enable.CapturePageRendering=false com.ibm.tivoli.itcam.dc.was.wps51.enable.CapturePortal=false com.ibm.tivoli.itcam.dc.was.wps51.enable.CapturePortalActionLegacy=false com.ibm.tivoli.itcam.dc.was.wps51.enable.CapturePortalActionStd=false
WebSphere Portal Server 6	com.ibm.tivoli.itcam.dc.was.wps6.enable.CaptureWPServlet=false com.ibm.tivoli.itcam.dc.was.wps6.enable.CaptureAuthorization=false com.ibm.tivoli.itcam.dc.was.wps6.enable.WPSDistributedMapCache=false com.ibm.tivoli.itcam.dc.was.wps6.enable.CaptureGatewayServlet=false com.ibm.tivoli.itcam.dc.was.wps6.enable.CaptureLogin=false com.ibm.tivoli.itcam.dc.was.wps6.enable.CaptureModelBuilding=false com.ibm.tivoli.itcam.dc.was.wps6.enable.CapturePageLoading=false com.ibm.tivoli.itcam.dc.was.wps6.enable.CapturePageRendering=false com.ibm.tivoli.itcam.dc.was.wps6.enable.CapturePortal=false com.ibm.tivoli.itcam.dc.was.wps6.enable.CapturePortalActionLegacy=false com.ibm.tivoli.itcam.dc.was.wps6.enable.CapturePortalActionStd=false com.ibm.tivoli.itcam.dc.was.wps6.enable.CaptureStandardPortal=false

For most API types, BCI information is only collected for MOD levels L2 and L3. If the MOD Level is set to L1, BCI data is not sent to IBM Tivoli Monitoring or the Managing Server, although the method entry and exit is still intercepted. (The MOD Level set for IBM Tivoli Monitoring does not affect the data sent to the Managing Server, and vice versa).

For the following API types, BCI information (if it is not disabled completely, as shown above) is also collected at MOD L1. For performance reasons, you can also disable it only for L1 monitoring, while keeping it enabled for L2 and L3. To do this, add (or uncomment) the following lines in the toolkit custom properties file (see “The Toolkit properties file” on page 29):

Table 3. Modifying lines in toolkit_custom.properties

Type of J2EE API	Line to add to toolkit_custom.properties file
JCA	com.ibm.tivoli.itcam.toolkit.ai.jca.callback.unconditional=false
JDBC	com.ibm.tivoli.itcam.toolkit.ai.jdbc.callback.unconditional=false
JNDI	com.ibm.tivoli.itcam.toolkit.ai.jndi.callback.unconditional=false
JMS	com.ibm.tivoli.itcam.toolkit.ai.jms.callback.unconditional=false

Attention: Setting any of these properties to false may result in some missing data when MOD Level is switched from L1 to L2. For example, the Data Collector may not be able to determine the Data Source names for JDBC requests.

Controlling instrumentation of application classes for lock analysis, memory leak analysis, and method profiling and tracing

The Data Collector can use Byte Code Instrumentation (BCI) to collect lock analysis information, memory leak analysis information (at MOD L3), method profiling (at MOD L2) and application method entry and exit (at MOD L3). Instrumentation for this data is disabled by default, and you must enable it if the information is required. This information is only displayed in the Visualization Engine of ITCAM for Application Diagnostics Managing Server.

To enable BCI for lock analysis, Level 3 memory leak analysis, and Level 3 method entry and exit tracing, you must make adjustments to the toolkit custom properties file (see “The Toolkit properties file” on page 29).

Making these adjustments will activate the use of one or more configuration files in the *DC_home/itcamdc/etc* directory, which contain the default settings to control BCI. The configuration files and default settings are described in the following table:

Table 4. Byte Code Instrumentation configuration files

File name	Purpose	Default behavior
lock_analysis.xml	Defines application lock analysis BCI. Note: Specific behavior in each MOD level is determined by settings in the Data Collector properties file.	Lock acquire and release requests for all application classes are Byte-Code-Instrumented. Lock event, lock contention, and lock reporting information is provided in MOD L2 and MOD L3. (You can enable lock analysis for MOD L1 as well; see “Customizing lock analysis” on page 36).
memory_leak_diagnosis.xml	Defines application Memory Leak Diagnosis BCI.	Heap allocations for all classes instantiated by all application classes are Byte-Code-Instrumented. Leak Analysis data is collected at MOD L3.
method_entry_exit.xml	Defines application method entry and exit BCI.	All non-trivial methods for all application classes, subject to certain thresholds and limits, are Byte-Code-Instrumented. Method profiling data is collected at MOD L2; method entry and exit analysis data is collected at MOD L3.

If you want to enable one or more of the BCI features with the default settings, see “Enabling Byte Code Instrumentation features with default settings” on page 36.

If you want to customize the default settings and make more granular choices for what classes and methods to modify, see the following sections:

- “Customizing lock analysis” on page 36
- “Customizing memory leak diagnosis” on page 38
- “Customizing method profiling and method entry and exit tracing” on page 40

Attention: Enabling Byte Code Instrumentation for any of these features will slightly increase resource usage even on monitoring levels where no data is collected for them.

Enabling Byte Code Instrumentation features with default settings

Perform the following procedure to enable one or more of the BCI features (lock analysis, Level 3 memory leak analysis, and method profiling and entry and exit tracing,) with the default settings. Method profiling at MOD L2 and method entry and exit tracing at MOD L3 are enabled by the same properties.

1. In the toolkit custom properties file (see “The Toolkit properties file” on page 29), uncomment one or more of the following lines by removing the number sign (#) at the beginning of the line:

```
am.camtoolkit.gpe.customxml.lock=DC_home/itcamdc/etc/lock_analysis.xml
am.camtoolkit.gpe.customxml.leak=DC_home/itcamdc/etc/memory_leak_diagnosis.xml
am.camtoolkit.gpe.customxml.L3=DC_home/itcamdc/etc/method_entry_exit.xml
```

See Table 4 on page 35 for a description of the default behaviors when each of these configuration files is activated.

2. Set one or more of the following properties to true:

```
com.ibm.tivoli.itcam.toolkit.ai.enablelockanalysis=true
com.ibm.tivoli.itcam.toolkit.ai.enablememoryleakdiagnosis=true
com.ibm.tivoli.itcam.toolkit.ai.methodentryexittrace=true
```

Customizing lock analysis

By default, if lock analysis is enabled, lock acquire and release requests for all application classes are Byte-Code-Instrumented. With default settings, lock contention information is provided in MOD L2 and MOD L3. You may configure the Data Collector to modify the lock information provided on the different MOD levels, and to exclude some classes from BCI for lock analysis.

Configuring lock analysis information for MOD levels

The following properties in the Data Collector properties file (see “Properties files for the Data Collector” on page 29) control the lock analysis information provided by the Data Collector.

internal.lockanalysis.collect.Ln.lock.event

This property whether lock acquisition/release events are collected and passed to the Managing Server. (If the Managing Server is not used, this parameter is ignored). The variable *n* can represent MOD L1, L2 or L3. Possible values are true or false. In most cases, the recommended setting at all levels is false as there is little benefit in displaying lock acquisition events if they do not involve contention; lock contention events are collected separately. However, you may wish to enable lock event collection for some development tasks.

Example:

```
internal.lockanalysis.collect.L1.lock.event = true
```

internal.lockanalysis.collect.Ln.contend.events

This property controls whether lock contention events are collected and passed to the Managing Server. (If the Managing Server is not used, this parameter is ignored). The variable *n* can represent MOD L1, L2 or L3. Possible values are true, false or justone.

True indicates contention records are collected. For each lock acquisition request that results in contention, a pair of contention records are written

for each thread that acquired the lock ahead of the requesting thread. `False` indicates contention records are not written. `Justone` indicates contention records are written, however, a maximum of one pair of contention records are written for each lock acquisition request that encounters contention, regardless of how many threads actually acquired the lock prior to the requesting thread.

Setting this property to `true` enables you to determine whether a single thread is holding a lock for an excessive time, or if the problem is due to too many threads all attempting to acquire the same lock simultaneously.

The recommended setting at L1 is `false`. The recommended setting at L2 is `justone`, this enables you to collect just one pair of contention records for each lock acquisition that encountered contention. The recommended setting at L3 is `true`, in order to identify every thread that acquired the lock ahead of the requesting thread; this setting has a high performance cost, which is common for L3 monitoring, and the user should only enable L3 for a limited time to reduce performance impact.

Example:

```
internal.lockanalysis.collect.L2.contend.events = justone
```

internal.lockanalysis.collect.Ln.contention.inflight.reports

This parameter controls whether data is collected for the Lock Contention report, available in the Visualization Engine of the Managing Server. (If the Managing Server is not used, this parameter is ignored). The variable *n* can represent Mod L1, L2 or L3. Possible values are `true` or `false`. The recommended setting at L1 is `false`. The recommended setting at L2 and L3 is `true`.

Example:

```
internal.lockanalysis.collect.L3.contention.inflight.reports = true
```

Setting classes for lock analysis instrumentation

To set classes for lock analysis instrumentation, perform the following procedure:

1. Make a copy of the `DC_home/itcamdc/etc/lock_analysis.xml` in a temporary location. Open the copy in a text editor.
2. Modify the `lockingClasses` specification in the file.

This element defines the classes for which lock requests will be Byte-Code-Instrumented. By default, all lock requests in all application classes are selected. By modifying this tag, you can implement a more granular selection, although within a class all lock requests are Byte-Code-Instrumented. Multiple `lockingClasses` tags can be specified.

The `lockingClasses` element can include wildcard characters. The following section describes how the wildcard characters work:

- Asterisk (*) stands for zero or more occurrences of any character when used by itself. When embedded within a sequence of characters (for example, `java.*.String`), it matches zero or more occurrences of any character except the package separator (`.`).
- Two periods (..) can be used to specify all sub-packages. It matches any sequence of characters that starts and ends with the package separator (`.`). For example, `java..String` matches `java.lang.String` and `com.ibm..*` matches any declaration beginning with `com.ibm.`
- If the locking class name begins with an exclamation point (!), any classes matching the classes identified in the tag are specifically excluded from BCI

for lock analysis. This is useful for indicating that all classes are to be Byte-Code-Instrumented except for those classes that are specifically excluded.

In the following example, an application with a package name of `com.mycompany.myapp` has the following requirements:

- Only classes that begin with `Cus` or `Sup` should be Byte-Code-Instrumented for lock analysis.
- The `Supplier` class should not be Byte-Code-Instrumented for lock analysis.

The following example shows the contents of the customized `lock_analysis.xml` file that accomplishes this:

```
<aspect>
  <type>application</type>
  <name>com.ibm.tivoli.itcam.toolkit.ai.aspectj.appttrace.CaptureLock</name>
  <enabledProperty>
    com.ibm.tivoli.itcam.toolkit.ai.enablelockanalysis</enabledProperty>
  <defaultEnabled>true</defaultEnabled>
  <lockingClass>com.mycompany.myapp.Cus*</lockingClass>
  <lockingClass>com.mycompany.myapp.Sup*</lockingClass>
  <lockingClass>!com.mycompany.myapp.Supplier</lockingClass>
</aspect>
```

3. Complete one of the following steps:

- Save the file in the `DC_home/runtime/app_server_version.node_name.server_name/custom` directory, then complete the following steps:
 - a. In the toolkit custom properties file (see “The Toolkit properties file” on page 29), set the property `am.camtoolkit.gpe.customxml.lock` to the name (without path) of the file you modified in Step 2 on page 37.
 - b. In the same toolkit custom properties file, set the following property to true:
`com.ibm.tivoli.itcam.toolkit.ai.enablelockanalysis=true`
- Save the file in any directory on the monitored host, then complete the following steps:
 - a. In the toolkit custom properties file (see “The Toolkit properties file” on page 29), set the property `am.camtoolkit.gpe.customxml.lock` to the path and name for the file you modified in Step 2 on page 37.
 - b. In the same toolkit custom properties file, set the following property to true:
`com.ibm.tivoli.itcam.toolkit.ai.enablelockanalysis=true`

Customizing memory leak diagnosis

By default, if memory leak analysis is enabled, all application classes are Byte-Code-Instrumented for memory leak analysis and all information is collected in MOD L3. You may configure the Data Collector to exclude some classes from BCI for memory leak analysis.

Perform the following procedure to set classes for Memory Leak Diagnosis:

1. Make a copy of the `DC_home/itcamdc/etc/memory_leak_diagnosis.xml` file in a temporary location. Open the copy in a text editor.
2. Modify the parameters in the file. The following table describes the tags you can modify:

Table 5. Parameters for the memory leak diagnosis configuration file

Tag name	Description
heapAllocationTarget	Defines the allocating and allocated classes for which heap allocations will be Byte-Code-Instrumented. By default, all allocating and allocated classes are selected. By modifying the allocatingClassName and allocatedClassName tags within the heapAllocationTarget tag, you can implement a more granular selection. Each heapAllocationTarget tag must contain exactly one allocatingClassName tag, and one or more allocatedClassName tags. Multiple heapAllocationTarget tags can be specified.
allocatingClassName	Identifies the name of a class or classes to be modified. Each heapAllocationTarget tag must contain exactly one allocatingClassName tag.
allocatedClassName	Identifies the specific heap allocation requests within the class or classes identified by the allocatingClassName tag that are to be Byte-Code-Instrumented. Each heapAllocationTarget tag must contain one or more allocatedClassName tags.

Both allocatingClassName and allocatedClassName tags can include wildcard characters. The following is a summary of how the wildcard characters work:

- Asterisk (*) stands for zero or more occurrences of any character when used by itself. When embedded within a sequence of characters (for example, java.*.String), it matches zero or more occurrences of any character except the package separator (.).
- Two periods (..) can be used to specify all sub-packages. It matches any sequence of characters that starts and ends with the package separator (.). For example, java..String matches java.lang.String and com.ibm..* matches any declaration beginning with com.ibm.
- If the allocated class name begins with an exclamation point (!), any heap allocations for classes that match the allocated class name are specifically excluded from BCI for Memory Leak Diagnosis. This is useful for indicating that all heap allocations within a class or group of classes are to be Byte-Code-Instrumented except for those allocations that are specifically excluded.

For example, an application with a package name of com.mycompany.myapp has the following requirements:

- Within the Customer class, all heap allocations should be Byte-Code-Instrumented.
- Within the Supplier class, all heap allocations should be Byte-Code-Instrumented except for allocations for classes beginning with java.lang.String.

The following example shows the contents of the customized memory_leak_diagnosis.xml file that accomplishes this:

```
<aspect>
  <type>application</type>
  <name>com.ibm.tivoli.itcam.toolkit.ai.aspectj.appttrace.CaptureHeap</name>
  <enabledProperty>
    com.ibm.tivoli.itcam.toolkit.ai.enablememoryleakdiagnosis</enabledProperty>
  <defaultEnabled>true</defaultEnabled>
  <!-- Modify the heapAllocationTarget tag to select or deselect the allocating and
    allocated classes for Memory Leak Diagnosis -->
  <heapAllocationTarget>
    <allocatingClassName>
      com.mycompany.myapp.Customer</allocatingClassName>
    <allocatedClassName>*</allocatedClassName>
  </heapAllocationTarget>
</heapAllocationTarget>
```

```

<allocatingClassName>
    com.mycompany.myapp.Supplier</allocatingClassName>
    <allocatedClassName>!java.lang.String*</allocatedClassName>
</heapAllocationTarget>
</aspect>

```

3. Complete one of the following steps:

- Save the file in the *DC_home/runtime/app_server_version.node_name.server_name/custom* directory, then complete the following steps:
 - a. In the toolkit custom properties file (see “The Toolkit properties file” on page 29), set the property `am.camtoolkit.gpe.customxml.leak` to the name (without path) of the file you modified in Step 2 on page 38.
 - b. In the same toolkit custom properties file, set the following property to true:


```
com.ibm.tivoli.itcam.toolkit.ai.enablememoryleakdiagnosis=true
```
- Save the file in any directory on the monitored host, then complete the following steps:
 - a. In the toolkit custom properties file (see “The Toolkit properties file” on page 29), set the property `am.camtoolkit.gpe.customxml.leak` to the path and name for the file you modified in Step 2 on page 38.
 - b. In the same toolkit custom properties file, sets the following property to true:


```
com.ibm.tivoli.itcam.toolkit.ai.enablememoryleakdiagnosis=true
```

Customizing method profiling and method entry and exit tracing

Method profiling and method entry and exit tracing are enabled together and use the same call interceptions. Method profiling is performed at MOD L2, and method entry and exit tracing is performed at MOD L3. You may configure the Data Collector to change the thresholds and limits for method profiling, and to exclude some classes and methods for method entry and exit tracing.

Customizing thresholds for Level 2 method profiling

The Data Collector will only instrument method profiling data when the method exceeds certain thresholds of CPU time and real (“wall clock”) time usage. There are also limits on the total number of methods, stack size, and running thread size. You can customize the thresholds and limits.

The following properties in the Data Collector properties file (see “The Data Collector properties file” on page 29) control the thresholds and limits for method profiling.

am.mp.cpuThreshold

The default is 30 milliseconds. Only the methods which take at least the minimum amount of CPU time specified in this property are captured for method profiling data. This avoids unnecessary clutter. Generally, methods with greater than the value specified in this property are considered useful. Customers can reduce or increase this value if needed.

am.mp.clockThreshold

The default is 30 milliseconds. Only the methods which take at least the minimum amount of wall clock time specified in this property are captured for method profiling data. This avoids unnecessary clutter. Generally, methods with greater than the value specified in this property are considered useful. Customers can reduce or increase this value if needed.

am.mp.leagueTableSize

The default is 1000. This is the maximum number of methods that are monitored for method profiling data. Customers can reduce or increase this value if needed. Decreasing it will help in reducing memory requirements.

am.mp.methodStackSize

The default is 100. This is the maximum stack size of any running thread that is recorded in method profiling.

Setting classes and methods for Level 3 method entry and exit tracing

By default, method entry and exit tracing on MOD L3 is performed for all classes and methods. To set specific classes and methods for method entry and exit analysis, perform the following procedure:

1. Make a copy of the *DC_home/itcamdc/etc/method_entry_exit.xml* file in a temporary location. Open the copy in a text editor.
2. Modify the parameters in the file. The following table describes the parameters you can modify:

Table 6. Parameters for the Level 3 method entry and exit analysis configuration file

Tag name	Description
methodSelection	Defines the classes and methods to be modified. By default, all classes and methods are selected. By modifying the <code>className</code> and <code>methodName</code> tags within the <code>methodSelection</code> tag, you can implement a more granular selection. Each <code>methodSelection</code> tag must contain exactly one <code>className</code> tag, and one or more <code>methodName</code> tags. Multiple <code>methodSelection</code> tags can be specified.
className	Identifies the name of a class or classes to be modified. Each <code>methodSelection</code> tag must contain exactly one <code>className</code> tag.
methodName	Identifies a method or method within the class or classes identified by the <code>className</code> tag to be modified for entry/exit tracing. Each <code>methodSelection</code> tag must contain one or more <code>methodName</code> tags.

Both `className` and `methodName` tags can include wildcard characters. The following section describes how the wildcard characters works:

- Asterisk (*) stands for zero or more occurrences of any character when used by itself. When embedded within a sequence of characters (for example, `java.*.String`), it matches zero or more occurrences of any character except the package separator (`.`).
- Two periods (..) can be used to specify all sub-packages. It matches any sequence of characters that starts and ends with the package separator (`.`). For example, `java..String` matches `java.lang.String` and `com.ibm..*` matches any declaration beginning with `com.ibm`.
- If the method name begins with an exclamation point (!), any methods that match the method name are specifically excluded from BCI for entry and exit tracing. This is useful for indicating that all methods within a class or group of classes are to be Byte-Code-Instrumented except for those methods that are specifically excluded.

For example, an application with a package name of `com.mycompany.myapp` has the following requirements:

- Within the `Customer` class, all methods should be Byte-Code-Instrumented.
- Within the `Supplier` class, all methods should be Byte-Code-Instrumented except for those methods beginning with the `get` or `set`.

The following example shows the contents of the customized `method_entry_exit.xml` file that accomplishes this:

```
<aspect>
  <type>application</type>
  <name>com.ibm.tivoli.itcam.toolkit.ai.aspectj.apprace.EntryExitAspect</name>
  <enabledProperty>
    com.ibm.tivoli.itcam.toolkit.ai.methodentryexittrace</enabledProperty>
  <defaultEnabled>true</defaultEnabled>
  <methodSelection>
    <className>com.mycompany.myapp.Customer</className>
    <methodName>*</methodName>
  </methodSelection>
  <methodSelection>
    <className>com.mycompany.myapp.Supplier</className>
    <methodName>!get*</methodName>
    <methodName>!set*</methodName>
  </methodSelection>
</aspect>
```

3. Complete one of the following steps:

- Save the file in the `DC_home/runtime/app_server_version.node_name.server_name/custom` directory, then complete the following steps:
 - a. In the toolkit custom properties file (see “The Toolkit properties file” on page 29), set the property `am.camtoolkit.gpe.customxml.L3` to the name (without path) of the file you modified in Step 2 on page 41.
 - b. In the same toolkit custom properties file, set the following property to true:


```
com.ibm.tivoli.itcam.toolkit.ai.methodentryexittrace=true
```
- Save the file in any directory on the monitored host, and then complete the following steps:
 - a. In the toolkit custom properties file (see “The Toolkit properties file” on page 29), set the property `am.camtoolkit.gpe.customxml.L3` to the path and name for the file you modified in Step 2 on page 41.
 - b. Set the following property to true:


```
com.ibm.tivoli.itcam.toolkit.ai.methodentryexittrace=true
```

Defining custom requests

By default, only certain types of Java classes and methods are monitored as requests by the Data Collector. Servlets, JSPs, EJB business methods, and certain standard J2EE APIs are recognized as requests. You can designate additional classes and methods as *custom requests*.

For example, ITCAM will not recognize Struts Action classes as requests by default, however you may set up custom request definitions and cause the Actions to be recognized as Nested Requests.

Perform the following procedure to enable monitoring of custom requests and designate one or more methods as custom requests:

1. Make a copy of the `DC_home/itcamdc/etc/custom_requests.xml` file in a temporary location. Open the copy in a text editor.
2. Modify the parameters in the file. The following table describes the parameters you can modify:

Table 7. Parameters for the custom requests configuration file

Tag name	Description
edgeRequest	Identifies one or more application methods that are to be Byte-Code-Instrumented for custom request processing. By modifying the requestName, Matches, type, and methodName tags within the edgeRequest tag, you can customize the selection. Each edgeRequest tag must contain exactly one methodName tag, and one or more Matches tags. Multiple edgeRequest tags can be specified.
requestName	Defines a unique name for this request. The request name is displayed to the user when the method entry and exit is traced.
Matches	Identifies a class or classes that contain the methods that are to be Byte-Code-Instrumented for custom request processing. Multiple Matches tags can be present within a single edgeRequest tag.
type	Indicates whether or not a class must be a system or application class in order to match the edgeRequest tag.
methodName	Identifies the names of the methods within one of the classes identified by the Matches tag that are to be Byte-Code-Instrumented for custom request processing. Exactly one methodName tag can be specified in each edgeRequest tag.

The Matches and methodName tags can include wildcard characters. The following section describes how the wildcard characters works:

- Asterisk (*) stands for zero or more occurrences of any character when used by itself. When embedded within a sequence of characters (for example, java.*.String), it matches zero or more occurrences of any character except the package separator (.).
- Two periods (..) can be used to specify all sub-packages. It matches any sequence of characters that starts and ends with the package separator (.). For example, java..String matches java.lang.String and com.ibm..* matches any declaration beginning with com.ibm.

For example, an application with a package name of com.mycompany.myapp has the following requirements:

- Within the Customer class, the creditCheck() method should be treated as a custom request called CreditCheck.
- Within the Supplier class, the inventoryCheck() method should be treated as a custom request called SupplyCheck.

The following example shows the contents of the customized custom_requests.xml file that accomplishes this:

```
<customEdgeRequests>
  <edgeRequest>
    <requestName>CreditCheck</requestName>
    <Matches>com.mycompany.myapp.Customer</Matches>
    <type>application</type>
    <methodName>creditCheck</methodName>
  </edgeRequest>
  <edgeRequest>
    <requestName>SupplyCheck</requestName>
    <Matches>com.mycompany.myapp.Supplier</Matches>
    <type>application</type>
    <methodName>inventoryCheck</methodName>
  </edgeRequest>
</customEdgeRequests>
```

3. Complete one of the following steps:

- Save the file in the *DC_home/runtime/app_server_version.node_name.server_name/custom* directory. Then, in the

toolkit custom properties file (see “The Toolkit properties file” on page 29), set the property `am.camtoolkit.gpe.customxml.custom` to the name (without path) of the file you modified in Step 2 on page 42.

- Save the file in any directory on your computer. Then, in the toolkit custom properties file (see “The Toolkit properties file” on page 29), set the property `am.camtoolkit.gpe.customxml.custom` to the path and name for the file you modified in Step 2 on page 42.

Enabling Asynchronous Bean request monitoring

If your applications use asynchronous bean requests, and the requests are not displayed in the Tivoli Enterprise Portal or the Visualization Engine, you will need to enable Asynchronous Bean request monitoring using the Toolkit custom properties file..

First, check whether custom requests have been defined on the Data Collector (see “Defining custom requests” on page 42). Open the toolkit custom properties file and, if it exists, the global toolkit custom properties file (see “The Toolkit properties file” on page 29). Check both of these files for the following property (not commented out):

```
am.camtoolkit.gpe.customxml.custom=xml_filename
```

If this property exists, custom requests have been defined. In this case, edit the XML file named in the property. If both the instance specific toolkit custom properties file and the global toolkit custom properties file have this property, use the filename from the instance specific file. If the filename does not have a path, the file is located in the `DC_home/runtime/app_server_version.node_name.server_name/custom` directory. Find the tag `</customEdgeRequests>`, and add the following text immediately before this line:

```
<edgeRequest>
  <requestName>AsyncWorkBean</requestName>
  <Implements>com.ibm.websphere.asyncbeans.Work</Implements>
  <type>application</type>
  <methodName>run</methodName>
</edgeRequest>
<edgeRequest>
  <requestName>AsyncTimerBean</requestName>
  <Implements>commonj.timers.TimerListener</Implements>
  <type>application</type>
  <methodName>timerExpired</methodName>
</edgeRequest>
<edgeRequest>
  <requestName>AsyncAlarmBean</requestName>
  <Implements>com.ibm.websphere.asyncbeans.AlarmListener</Implements>
  <type>application</type>
  <methodName>fired</methodName>
</edgeRequest>
```

If the property does not exist (or is commented out), custom requests have not been defined. In this case, perform the following procedure:

1. create a new file: `DC_home/runtime/app_server_version.node_name.server_name/custom/custom_requests_async.xml`, with the following text:

```
<gpe>
  <bci>
    <customEdgeRequests>
      <edgeRequest>
        <requestName>AsyncWorkBean</requestName>
        <Implements>com.ibm.websphere.asyncbeans.Work</Implements>
```

```

        <type>application</type>
        <methodName>run</methodName>
    </edgeRequest>
    <edgeRequest>
        <requestName>AsyncTimerBean</requestName>
        <implements>commonj.timers.TimerListener</implements>
        <type>application</type>
        <methodName>timerExpired</methodName>
    </edgeRequest>
    <edgeRequest>
        <requestName>AsyncAlarmBean</requestName>
        <implements>com.ibm.websphere.asynchbeans.AlarmListener</implements>
        <type>application</type>
        <methodName>fired</methodName>
    </edgeRequest>
    </customEdgeRequests>
</bci>
</gpe>

```

2. In the toolkit custom properties file (see “The Toolkit properties file” on page 29), set the property:

```
am.camtoolkit.gpe.customxml.custom=custom_requests_async.xml
```

After performing the changes described in this section, restart the application server instance monitored by the Data Collector.

Customizing monitoring of custom MBeans

By default, the Data Collector will monitor WebSphere Application Server MBeans. If the environment includes custom MBeans, you may configure the Data Collector to monitor some of them (according to specific definitions), or all of them.

Configuring data collection for specific custom MBeans

You may define specific MBeans. Data collection will be enabled for these MBeans only.

Note: Data collection for the J2EE Domain MBean is not available in an IBM WebSphere Application Server environment.

Perform the following procedure to customize the generic configuration for JMX data collection:

1. Make a copy of the `DC_home/itcamdc/etc/was/app_server_version/mbeanconfig_app_server_version.xml` file in a temporary location. Open the copy in a text editor.
2. Modify the parameters to fit your custom MBean. The following table describes the parameters you can modify:

Table 8. Parameters for the JMX MBean configuration file

Element	Nested within	Description
Version	DomainList	Defines the version of the application server
Name	Domain	Defines a domain. If the asterisk (*) is defined, all MBeans that match the query <code>ObjectName</code> will be returned; otherwise, only the MBeans that belong to this domain name will be returned.
Description	Domain	Describes the domain. This can be any text string.
MBean	Domain	Defines the MBeans to be collected.

Table 8. Parameters for the JMX MBean configuration file (continued)

Element	Nested within	Description
ObjectName	MBean	Defines the MBean object name for collection. If the MBean element is used within an Attr element (which indicates the embedded MBean), then the object name is any symbolic name, such as \$ATTRIBUTE_VALUE. This symbolic name will be replaced with the actual object name internally.
Category	MBean	Defines a unique key for the MBean. Each MBean must have a unique key, which is used in the JMXAcquireAttribute to get the MBean attributes.
RetrieveAllAttrs	MBean	A value of true indicates that all the attributes for the MBean must be collected. If you set this to false, you must define the attributes in the Attr element.
Attr	MBean	Defines the attributes to be collected. There may be multiple Attr elements, defining multiple attributes.
Name	Attr	The attribute name.
MappedKey	Attr	Defines a unique key for the attribute. Each attribute must have a unique key, which is used in the JMXAcquireAttribute to get the specific attribute.
MBean	Attr	Defines the embedded MBean within this attribute. This tag is used when an attribute has an embedded MBean, which points to another MBean with the object name.
JavaBean	Attr	Defines the embedded MBean within this attribute. This tag is used when an attribute has an embedded MBean, which refers to another java object. The references java object has the elements of a JavaBean (setter, getter).
TargetType	Attr	Defines the type of the attribute. This is usually specified for the JavaBean type to determine the attribute type.

3. Complete one of the following steps:

- Save the file in the `DC_home/runtime/app_server_version.node_name.server_name/custom` directory. Then, in the toolkit custom properties file (see “The Toolkit properties file” on page 29), set the property `am.camtoolkit.jmxe.custom` to the name (without path) of the file you modified in Step 2 on page 45.
- Save the file in any directory on the monitored host. Then, in the toolkit custom properties file (see “The Toolkit properties file” on page 29), set the property `am.camtoolkit.jmxe.custom` to the path and name for the file you modified in Step 2 on page 45.

Enabling and customizing data collection for all custom MBeans

You may enable data collection for all custom MBeans and customize the way they are identified to the user.

To enable it, set the following property in the Toolkit properties file (see “The Toolkit properties file” on page 29):

```
am.getAllmbeans=y
```

This property is in effect only if the custom MBeans property is commented out in the Toolkit properties file, as shown in the following example:

```
# Uncomment the line to enable custom mbeans
#am.camtoolkit.jmxe.custom=
C:/PROGRA~1/IBM/itcam/WEBSPH~1/DC/itcamdc/etc/custom_mbeanconfig.xml
```

Tip: Check that the `am.camtoolkit.jmxe.custom` property is also not present in the Toolkit global properties file (see “The Toolkit properties file” on page 29).

The following properties provide additional options for display of the MBean data:

```
am.jmxkeyword=type_identifier
am.jmxusecanonical=y
am.jmxtruncate=n
am.jmxlength=30
```

The properties and their definitions are:

am.getallmbeans

Set this property to `y` to enable data collection for all MBeans. If it is set to `n`, data collection for all MBeans is disabled. If the custom MBeans property (`am.camtoolkit.jmxe.custom`) is set, `am.getallmbeans` has no effect.

am.jmxkeyword

The Visualization Engine presents data on monitored MBeans by organizing them into categories. The category name is formed from the Domain and type keywords in the MBeans Object Name. If the type keyword does not exist, the name keyword is used to create the category. If the name keyword does not exist, then the object name is used as the category. If this default behavior does not provide enough granularity to distinguish MBean categories, you can use the `am.jmxkeyword` property to define more keywords to be included in the category name.

For example, if you specify `am.jmxkeyword=identifier`, then the value of the identifier keyword from the object name is included in the category name, in addition to value of the type keyword. More than one keyword can be specified in the property. The keywords must be separated by a comma (,).

am.jmxusecanonical

If you need to see all of the keywords from the object name in the category, assign the `am.jmxusecanonical=y` property. This setting will result in including all keyword values for the category name, separated by an underscore (_) character.

am.jmxtruncate

In some cases, especially if `am.jmxusecanonical=y`, the category name can be quite long. By default, the Data Collector will truncate the category name to the length specified by the `am.jmxlength` property, or to 30 characters if the `am.jmxlength` property is not specified. If you do not want the category name to be truncated, specify the `am.jmxtruncate=n` property.

am.jmxlength

This property specifies the maximum length of the category name. The default is 30. This property is ignored if the `am.jmxtruncate=n` property is specified.

Modifying Performance Monitoring Infrastructure settings

If the Data Collector communicates with ITCAM for Application Diagnostics Managing Server, the level of instrumentation for Performance Monitoring Infrastructure (PMI) is determined by the current MOD level set in the

Visualization Engine. You can customize the PMI level for each MOD level. The collection level set in Tivoli Enterprise Portal does not affect the PMI level.

By default, the following PMI setting will be enabled at each Managing Server MOD level:

Table 9. Default Performance Monitoring Infrastructure instrumentation settings

Monitoring (MOD) level	PMI setting
1	Basic
2	Extended
3	All

To customize these settings, perform the following procedures in the Data Collector custom properties file (see “The Data Collector properties file” on page 29):

Table 10. Procedures to customize instrumentation of the Performance Monitoring Infrastructure

Type of customization	Procedure
Change the PMI setting that will be set for a particular Managing Server monitoring (MOD) level.	Add or uncomment one or more of the following lines and give it a different setting. The possible values are none, basic, extended, or all: <pre>am.was6pmi.settings.1=basic am.was6pmi.settings.2=extended am.was6pmi.settings.3=all</pre>
Perform fine-grained customization of the instrumentation for a particular PMI module at a particular monitoring level. See the following Web site for a description of the numeric IDs that you will need when customizing PMI instrumentation at this detailed level: http://www.ibm.com/support/docview.wss?uid=swg21221308	Add a line to set fine-grained customization for a particular module at a particular monitoring level. It has the format <code>module_type=number1,number2,...</code> , for example: <pre>am.was6custompmi.settings.1=beanmodule=1,2,3,4,5,6,7,8,9,10,14,15,19,20,21,22,23,24,25,28,29,30,31,32,33,34</pre> Use * to monitor all IDs in the module, or none to monitor none: <pre>am.was6custompmi.settings.3=beanModule=* am.was6custompmi.settings.3=webAppModule=none</pre>

Attention: The `am.was6pmi.*` property names are also valid for monitoring Version 7 application servers.

If you do not want the level of instrumentation for PMI to change as the Managing Server MOD level changes, add the following line to the Data Collector custom properties file (see “The Data Collector properties file” on page 29):

```
am.pmi.settings.nochange=true
```

Enabling Performance Monitoring Infrastructure settings for the Service Integration Bus

You can configure the Data Collector to collect Service Integration Bus (SIB) Performance Monitoring Infrastructure (PMI) data.

To do this, perform the following procedure:

1. Add the following lines to the Data Collector custom properties file (see “The Data Collector properties file” on page 29):

```
am.was6custompmi.settings.1=SIB Service=*
am.was6custompmi.settings.2=SIB Service=*
am.was6custompmi.settings.3=SIB Service=*
```

These lines set custom PMI settings for Level 1, Level 2 and Level 3 monitoring levels respectively.

2. Restart the instance of the application server that is being monitored by the Data Collector. See “Restarting the application server” on page 69.

Enabling and disabling instrumentation of Web Services as new request types

By default, the Data Collector monitors JAX-RPC 1.1 and Axis 1.x Web Services. To enable monitoring of JAX-WS Web Services, you need to perform additional steps. You can also disable monitoring of all Web Services.

To enable instrumentation of JAX-WS Web Services, you need to deploy the JAX-WS handler for the Data Collector. To do this, perform the following procedure on every application server that is a requester (client) or provider (server) of JAX-WS Web Services, and is monitored by the Data Collector:

1. Copy the file `DC_home/itcamdc/lib/com.ibm.tivoli.itcam.dc.jaxws-handlers.jar` into the `AppServer_home/plugins` directory.
2. Change to the `AppServer_home` directory, and run the following command:
 - on Windows, `osgiCfgInit.bat`
 - on Linux or UNIX systems, `./osgiCfgInit.sh`
3. Restart the application server instances monitored by the Data Collector.

To disable instrumentation of Web Services, set the following property in the Data Collector custom properties file (see “The Data Collector properties file” on page 29):

```
ws.instrument=false
```

Important: To enable Web Services composite request monitoring and correlation in the Visualization Engine and in ITCAM for Transactions, you need to monitor both the Web services requester (client) and the Web services provider (server) using ITCAM Agent for WebSphere Applications Data Collectors, and the Data Collectors must be connected to the same Managing Server.

Enabling and disabling memory monitoring

The Data Collector can monitor native memory usage and save results to a log file. This capability is disabled by default.

If you enable Data Collector memory monitoring, the Data Collector will save memory usage statistics to the trace log file (`trace-dc-native.log`). For the location of the Data Collector trace log file, see *IBM Tivoli Composite Application Manager for Application Diagnostics Troubleshooting Guide*.

The statistics reflect Data Collector memory consumption on the native side. The Java side memory consumption is not reflected in the logged numbers.

To enable memory monitoring, set the following property in the Data Collector custom properties file (see “The Data Collector properties file” on page 29):

```
log.statistics=true
```

By default, the statistics are logged once every 30 seconds. You can set a different period, in milliseconds, in the `log.statistics.frequency` property in the Data

Collector custom properties file. For example, to log memory usage statistics once every 10 seconds, use the following setting:

```
log.statistics.frequency=10000
```

To disable memory monitoring, set the following property in the Data Collector custom properties file:

```
log.statistics=false
```

Changing Managing Server communication configuration

If you want to enable or disable communication with the Managing Server, or to change the Managing Server for your Data Collector, configure the Data Collector for the application server instance again. See “Silent installation and initial configuration of the Data Collector” on page 13.

Steps to perform if the IP address of the application server host is to be changed

If the IP address of the application server host is to be changed, perform the following procedure:

1. Unconfigure the Data Collector for this application server instance. See “Silent installation and initial configuration of the Data Collector” on page 13.
2. Stop the instance of the application server that is being monitored by the Data Collector. See “Stopping the application server” on page 71.
3. Perform the IP address change at the operating system and network level.
4. Start the instance of the application server that is being monitored by the Data Collector. See “Starting the application server” on page 70.
5. Configure the Data Collector again for this application server instance. See “Silent installation and initial configuration of the Data Collector” on page 13.

Steps to support changing application the type of virtual machine for the Java platform

You can use one of the following types of virtual machine for the Java platform with WebSphere Application Server: classic, standard 32-bit, and standard 64-bit. When the Data Collector is configured to monitor an application server instance, it determines the type of virtual machine automatically. If you change the type of the virtual machine later, you need to adjust Data Collector configuration.

If you change the virtual machine type for the application server and do not configure the Data Collector for it, the application server will fail to start.

You can reconfigure the Data Collector after changing the application server virtual machine type; see “Silent configuration of the Data Collector” on page 18. Use the `dcInputs_Reconfig_Sample_os400.txt` response file template.

Tip: For a stand alone application server, as it will not be running, set the parameter `WS_CONNECT_TYPE.was_alias=NONE` in the reconfiguration response file. For an application server in Network Deployment or Extended Deployment, set the connection parameters normally, as the Deployment Manager is not affected.

Alternatively, to avoid any downtime of the application server, you can reconfigure the Data Collector before changing the virtual machine type for the application

server. In this case, set the parameter JVM_TYPE to the new value (classic, std32, or std64) in the reconfiguration response file.

Moving the Data Collector to a different host computer

If the Data Collector communicates to ITCAM for Application Diagnostics Managing Server, you may move it to a different host computer while maintaining the same Managing Server identity (Probe ID and Controller ID). The Managing Server will see the new host as the continuation of the old, preserving history, analysis, etc.

The following section describes some prerequisites for moving the Data Collector to a different host computer while keeping the same Probe ID and Controller ID:

- Host A and host B have the same configuration at the operating system level.
- You must move the same version of the Data Collector from host A to host B.

To maintain the Probe ID and Controller ID when moving to another physical host, perform the following procedure:

1. On host A, stop the instance of the application server that is being monitored by the Data Collector. See “Stopping the application server” on page 71.
2. On host B, install the Data Collector and configure it using the Visualization Engine (Application Monitor) user interface. Configuring the Data Collector will generate the *DC_home/runtime/appserver_version.node_name.server_name/id* file and other Data Collector runtime property files.
3. Using the Visualization Engine (Application Monitor) user interface, unconfigure the Data Collector on host B. This step deletes all information about this Data Collector from the ITCAM for Application Diagnostics database. Do not unconfigure the Data Collector using the Configuration tool.
4. On host B, stop the instance of the application server that is being monitored by the Data Collector. See “Stopping the application server” on page 71.
5. Copy the contents of the *DC_home/runtime/appserver_version.node_name.server_name/id* file on host A to the *DC_home/runtime/appserver_version.node_name.server_name/id* file on host B.
6. On host B, save the *DC_home/runtime/appserver_version.node_name.server_name/id* file.
7. On host B, start the instance of the application server that is being monitored by the Data Collector. See “Starting the application server” on page 70.

The Data Collector on host B assumes the identity of the Data Collector on host A and is configured by the Managing Server with the runtime configuration of the Data Collector on host A. This does not affect monitoring in Tivoli Enterprise Portal.

Configuring a Data Collector for multiple network interfaces

If the application server host has multiple IP addresses at the time of Data Collector configuration, the Configuration tool will set the preferred IP address for communication with the Managing Server. If more than one IP address is added later, set the preferred IP address manually, as described in this section.

If the Data Collector needs to expose a specific IP to the Managing Server, complete one of the following steps:

1. In the Data Collector custom properties file (see “The Data Collector properties file” on page 29), set the `am.socket.exportip` and `am.socket.bindip` properties to the IP address to be exposed.
2. In the file `DC_home/runtime/appserver_version.node_name.server_name/dc.java.properties`, set the `appserver.rmi.host` property to the IP address to be exposed.
3. Make sure that the Managing server can access the required IP address of the Data Collector (You can verify this by doing a ping).

Customizing RMI garbage collection interval

If the Data Collector communicates with ITCAM for Application Diagnostics Managing Server, it uses RMI over TCP/IP for this communication. One effect of using RMI is that garbage collection occurs every minute. On IBM J9 JVM (std32 or std64), if you don't want this to happen, you can specify the garbage collection interval explicitly to a preferred interval.

The Data Collector communicates with the Managing Server using TCP/IP sockets and RMI. One effect of using RMI is that garbage collection occurs every minute. If you don't want this to happen, on IBM J9 JVM (std32 or std64), you can specify the garbage collection interval explicitly to a preferred interval by specifying the parameters in the **Generic JVM arguments** field. These parameters must be implemented as a pair.

To do this, complete the following steps:

1. Log into the IBM WebSphere Application Server administrative console.
2. Click **Server > Application Servers** and select the `server_name`.
3. In the **Configuration** tab, navigate to **Server Infrastructure > Java and Process Management > Process Definition > Additional Properties: Java Virtual Machine**.
4. In the **Generic JVM arguments** field, append the following parameters if such parameters don't exist, or update their values if they already exist.

```
-Dsun.rmi.dgc.client.gcInterval=3600000 -Dsun.rmi.dgc.server.gcInterval=3600000
```

Note: These values require a dash (-) in front of each parameter, and a single space between parameters. You must specify both parameters if you specify them at all. The value is in milliseconds; 3,600,000 represents one hour.

Customizing CICS transaction correlation

CICS is a transaction framework, primarily used to run mature applications. To communicate with CICS, Java applications can use the CICS Transaction Gateway (CTG).

If CICS translation correlation is enabled, the Data Collector callback code will add composite tracking data, called Global Publish Server (GPS) tokens, into the communications area (COMMAREA) used to carry transaction request data to CICS. This data can be used by ITCAM for Transactions, which instruments the CICS transaction framework. ITCAM for Transactions will correlate every CICS transaction with the corresponding CTG call using the GPS token. The user can then view a detailed breakdown of transaction response time in the ITCAM Visualization Engine.

However, the presence of the GPS token in COMMAREA may not always be desirable. If ITCAM for CICS Data Collector or ITCAM for CICS Client is not installed on the CICS server, the GPS token might reach the server application, which may (in some cases) not process it correctly. For this reason, transaction correlation is disabled by default.

You can enable GPS tokens for specific transactions based on CTG gateway address or protocol; by CICS system; by CICS program or by the CICS transaction ID. Enable correlation with CICS systems that have the ITCAM for CICS Data Collector installed, configured, and enabled. To do this, edit the file *DC_home/runtime/app_server_version.node_name.server_name/custom/ctg.filters* . This file can contain any number of lines with the following syntax:

```
Type=E|I[,Gateway=<CTG URL>][,Server=<CICS Server>][,Program=<CICS Program>]
[,Transid=<Mirror tran ID>]
```

Each line defines a filter, which disables or enables GPS tokens for some transactions.

The Type parameter is mandatory for each line. A value of "E" sets up an Exclude filter; transactions matching it will not have a GPS token inserted into the COMMAREA. "I" denotes an Include filter; any transactions matching an include filter will have a GPS token, overriding any Exclude filter applying to them.

All other parameters are optional, but at least one of them must be present on every line. To match a filter, a transaction must match all of the parameters set on the line:

- Gateway is any part of the CTG URL, including the protocol, host name and/or port
- Server is the host name of the CICS server (this may be different from the CTG host name)
- Program is the CICS program name (a field in a CICS transaction request)
- Transid is the CICS Mirror Transaction ID. Except Multi Regional Operation (MRO) CICS/CTG environments, this parameter is of little use as all CTG transactions will have the same Mirror Transaction ID

For example, to disable addition of GPS tokens to the COMMAREA of all transactions routed through the local protocol, add the following line to *DC_home/runtime/app_server_version.node_name.server_name/custom/ctg.filters*:

```
Type=E,Gateway=local://*
```

To disable addition of GPS tokens to transactions for programs starting 'CYN\$' to be run on the CICS3101 server, but enable them for transactions for the CYN\$ECI2 program on the same server, use the following lines:

```
Type=E,Program=CYN$*,Server=CICS3101
Type=I,Program=CYN$ECI2,Server=CICS3101
```

The default configuration is to disable all correlation through the following line:

```
Type=E,Gateway=*
```

Modifying the garbage collection log path

The Data Collector configuration set the path for the garbage collection log file (itcam_dc_gclog.log or native_stderr.log) to *AppServer_home/profiles/profile_name/logs/server_instance_name*. For example, C:\Program~1\IBM\WebSphere\AppServer\profiles\AppSrv01\logs\server1. On IBM J9 JVM (std32 or std64), if you want to change the location or name of this log file, perform the following procedure:

1. In the *DC_home/runtime/app_server_version.node_name.server_name/kwjdc.properties* file, make the following modification:

Change the following parameter to point to the new garbage collection log file location:

```
TEMAGCCollector.gclog.path=gc_logfile_path_and_name
```

Attention: Do not use variables or templates in this value.

You can also optionally limit the size of the Garbage Collector logs. To do this you need to set the parameter to the following value:

```
TEMAGCCollector.gclog.path=gc_logfile_path_and_name, x, y
```

Where *x* and *y* are numbers. In this case, the logging will be performed to *x* files in rotation; information for *y* garbage collection cycles will be sent to one file before switching to the next file.

2. Log into the IBM WebSphere Application Server administrative console for the instance of the application server for the Data Collector installed on the RMI server.
3. Click **Server > Application Servers** and select the *server_name*.
4. In the **Configuration** tab, navigate to **Server Infrastructure > Java and Process Management > Process Definition > Additional Properties: Java Virtual Machine**.
5. In the **Generic JVM arguments** field, change the following parameters to point to the new garbage collection log file location:

Table 11. JVM options for garbage collection logging

JDK version	Parameter
IBM 1.4 and 1.5	-verbosegc -Xverbosegclog:\${SERVER_LOG_ROOT}/itcam_gc.log,5,3000
Sun and HP 1.4 and 1.5	-Xloggc:gc_logfile_path_and_name -XX:+PrintGCTimeStamps

Make sure the *gc_logfile_path_and_name* matches the value you specified in Step 1.

6. Click **Apply**.
7. In the Messages dialog box, click **Save**.
8. In the Save to Master Configuration dialog box, complete the following steps:
 - If you are under a Network Deployment environment, be sure the check box **Synchronize changes with Nodes** is selected and then click **Save**.
 - If you are not under a Network Deployment environment, click **Save**.
9. Restart the instance of the application server that is being monitored by the Data Collector. See “Restarting the application server” on page 69.

Overriding the Data Collector autoconfiguration

By default, if the Data Collector communicates with ITCAM for Application Diagnostics Managing Server, it will be automatically configured by the Managing Server with the default configuration profile at the time of first connection. You may disable automatic configuration or select a different profile. These settings only take effect if you perform them before the Data Collector connects to the Managing Server for the first time.

To disable automatic configuration of the Data Collector in the Managing Server, set the following property in the Data Collector custom properties file (see “The Data Collector properties file” on page 29):

```
dc.autoconfigure=false
```

To change the profile name for automatic configuration of the Data Collector in the Managing Server, set the following property in the Data Collector custom properties file (see “The Data Collector properties file” on page 29):

```
dc.autoconfigure.configname=config_name
```

Note: If the Data Collector has already been configured by the Managing Servers, changing these settings will not have any effect.

To configure or unconfigure a Data Collector from the Managing Server, or to change the Data Collector configuration profile, use the Visualization Engine. From the top navigation, select **Administration > Server Management > Data Collector Configuration**. For more information on Data Collector configuration by the Managing Server, see the Visualization Engine online help.

Properties for communication with a Deployment Manager

The following properties define Data Collector communication with the Deployment Manager in a Network Deployment or Extended Deployment. They are normally set by the configuration utility.

The properties are in the Data Collector properties file (see “The Data Collector properties file” on page 29).

deploymentmgr.rmi.port

Defines the port for RMI communication to the Deployment Manager.

Example:

```
deploymentmgr.rmi.port=Deployment_Manager_RMI_(bootstrap)_port
```

deploymentmgr.rmi.host

Defines the host name or IP address for RMI communication to the Deployment Manager.

Example:

```
deploymentmgr.rmi.host=dmgr.domain.com
```

Part 4. Appendixes

Appendix A. Setting up security

Setting up optional security for ITCAM for Application Diagnostics is described in this chapter.

For information on optional security for ITCAM for z/OS refer to the *IBM Tivoli Composite Application Manager: Agent for WebSphere Applications Installation and Configuration Guide for z/OS*

Perform the procedures in each of the following sections, if they apply.

Node Authentication

Node Authentication is the technique used to ensure that the managing server and data collectors communicate with each other in a secure manner. In Node Authentication related configuration, the Kernel, Data Collectors or Port Consolidator operate in secure mode either individually or in combination. The configuration changes are common for all the modes except that a particular component can be made to operate in a different mode by changing the property `security.enabled` on that particular component. You can use the following combinations:

- Managing server in secure mode and the data collector in non secure mode.
- Data collector in secure mode and the managing server in non secure mode.
- Managing server and data collector in secure or non secure mode.

Script to run if your SSL certificates have expired

All SSL certificates have an expiration time. For some certificates, the expiration time is 4 years, after which the product will not function if you have enabled Node Authentication and SSL. If this is the case, to increase the expiration time, perform the procedure at “Script to run if your SSL certificates have expired” on page 66.

Node Authentication on the Managing Server

The following procedures are Node Authentication related configuration that occurs on the Managing Server component.

Kernel-related changes

In the managing server in the `$MSHOME/bin` directory there is `setenv.sh` file that is shared by all managing server components. All changes made to the `setenv.sh` file apply to all managing server components. All the managing server components initialize their respective security modules based on the properties in this `setenv.sh` file. The installer configures all managing server components with security enabled configuration by default with the exception of kernel-related changes which are enabled by changing the `.kl1` and `.kl2` property files on the managing server.

In the Kernel properties file (`MS_home/etc/kl1.properties`) complete the following steps:

1. To enable a Kernel to operate in secure mode, set the following property:
`security.enabled=true`

2. If you have a multiple Network Interface Card (NIC) environment or are upgrading the Managing Server from version 6.0 to version 7.1.0.2, in the Kernel properties file (*MS_home/etc/kl1.properties*), set `codebase.security.enabled=false`.
If you have more than one instance of the Kernel, set `codebase.security.enabled=false` in *kl2.properties* as well.
3. Restart the Managing Server. See *IBM Tivoli Composite Application Manager for Application Diagnostics Managing Server Installation Guide*.

Data Collector custom properties file changes

The following procedure is Node Authentication related configuration that occurs by modifying the `datacollector_custom.properties` file.

Enabling the Data Collector to operate in secure mode

In the Data Collector custom properties file (*DC_home/runtime/app_server_version.node_name.server_name/custom/datacollector_custom.properties*) complete the following steps:

1. Set `security.enabled=true`
2. Restart the application server.

Keystore management and populating certificates

You do not have to use the following commands unless you want to create unique certificates with a new storepass and keypass. You can run keystore management on the managing server and the data collector. These commands will populate a new store with those certificates.

To run the `keytool` commands, you must be in the `java/bin` directory or have `keytool` in your `PATH`. This is the command with the necessary parameters:

```
keytool -genkey -alias alias_name -keyalg RSA -keysize 1024 -sigalg MD5withRSA
-validity 2000 -keypass keypass -keystore ./storename -storepass storepass -dname
"cn=cyaneamgmt, OU=CyaneaComp, O=Cyanea, L=Oakland, ST=CA, C=US"
```

Use the following details to create all the necessary stores and certificates:

```
keytool -genkey -alias mgmttmgmt -keyalg RSA -keysize 1024 -sigalg MD5withRSA
-validity 2000 -keypass oakland1 -keystore ./CyaneaMgmtStore
-storepass oakland2 -dname "cn=cyaneamgmt, OU=CyaneaComp, O=Cyanea, L=Oakland,
ST=CA, C=US"

keytool -genkey -alias dctomgmt -keyalg RSA -keysize 1024 -sigalg MD5withRSA
-validity 2000 -keypass oakland1 -keystore ./CyaneaMgmtStore -storepass oakland2
-dname "cn=cyaneadc, OU=CyaneaComp, O=Cyanea, L=Oakland, ST=CA, C=US"

keytool -genkey -alias proxytomgmt -keyalg RSA -keysize 1024 -sigalg MD5withRSA
-validity 2000 -keypass oakland1 -keystore ./CyaneaMgmtStore -storepass oakland2
-dname "cn=cyaneaproxy, OU=CyaneaComp, O=Cyanea, L=Oakland, ST=CA, C=US"

keytool -genkey -alias proxytodc -keyalg RSA -keysize 1024 -sigalg MD5withRSA
-validity 2000 -keypass oakland1 -keystore ./CyaneaDCStore -storepass oakland2
-dname "cn=cyaneaproxy, OU=CyaneaComp, O=Cyanea, L=Oakland, ST=CA, C=US"

keytool -genkey -alias mgmttodc -keyalg RSA -keysize 1024 -sigalg MD5withRSA
-validity 2000 -keypass oakland1 -keystore ./CyaneaDCStore
-storepass oakland2 -dname "cn=cyaneamgmt, OU=CyaneaComp, O=Cyanea,
L=Oakland, ST=CA, C=US"

keytool -genkey -alias mgmttoproxy -keyalg RSA -keysize 1024 -sigalg MD5withRSA
-validity 2000 -keypass oakland1 -keystore ./CyaneaProxyStore -storepass oakland2
-dname "cn=cyaneamgmt, OU=CyaneaComp, O=Cyanea, L=Oakland, ST=CA, C=US"
```

```
keytool -genkey -alias dctoproxy -keyalg RSA -keysize 1024 -sigalg MD5withRSA
-validity 2000 -keypass oakland1 -keystore ./CyaneaProxyStore
-storepass oakland2 -dname "cn=cyaneadc, OU=CyaneaComp, O=Cyanea, L=Oakland,
ST=CA, C=US"
```

Extracting Certificates:

When you have created the three 3 Stores, extract the certificates by completing the following steps:

1. Extract all certificates from CyaneaMgmtStore by running the following commands:

```
keytool -export -alias mgmttomgmt -keypass oakland1 -keystore ./CyaneaMgmtStore
-storepass oakland2 -file mgmttomgmt.cer
keytool -export -alias dctomgmt -keypass oakland1 -keystore ./CyaneaMgmtStore
-storepass oakland2 -file dctomgmt.cer
keytool -export -alias proxytomgmt -keypass oakland1 -keystore ./CyaneaMgmtStore
-storepass oakland2 -file proxytomgmt.cer
```
2. Extract all certificates from CyaneaDCStore by running the following commands:

```
keytool -export -alias proxytodc -keypass oakland1 -keystore ./CyaneaDCStore
-storepass oakland2 -file proxytodc.cer
keytool -export -alias mgmttodc -keypass oakland1 -keystore ./CyaneaDCStore
-storepass oakland2 -file mgmttodc.cer
```

When you have extracted your files, copy the following certificates and Stores to the following locations:

Configuring components to use new keystores and certificates

Configure components to use new keystores and certificates:

1. Modify *MS_home/bin/setenv.sh*. At the end of the script you will need to modify the following lines with the new keystore name, storepass, and keypass:

```
KEYSTR_LOC=MS_home/etc/IBMMSSStore
KEYSTR_PASS=oakland2
KEYSTR_KEYPASS=oakland1
```
2. Modify the Visualization Engine (Application Monitor) user interface with the new keystore name, storepass and keypass. Perform the following procedure:
 - a. Start the Managing Server.
 - b. Log into the IBM WebSphere Application Server administrative console.
 - c. Click **Server > Application Servers** and select the *server_name*.
 - d. In the **Configuration** tab, navigate to **Server Infrastructure: Java and Process Management > Process Definition > Additional Properties: Java Virtual Machine > Additional Properties: Custom Properties**.
 - e. For the following name and value pairs, click **New**, enter the Name and Value, and click **Apply**:
 - 1) Set the path of the certificate to use when security is enabled for the Visualization Engine (Application Monitor) user interface:

```
certificate.path=MS_home/etc/mgmttomgmt.cer
```
 - 2) Set the keystore location of the Managing Server:

```
keystore.location=MS_home/etc/CyaneaMgmtStore
```
 - 3) Set the keystore password of Managing Server:

```
keystore.storepass=oakland2
```
 - 4) Set the keystore key password of Managing Server:

```
keystore.keypass=oakland1
```

- 5) Set the user ID passed to the other end for authentication:
`nodeauth.userid=cyaneaamgmt`
- f. Restart the application server.
3. Modify `DC_home/runtime/app_server_version.node_name.server_name/custom/datacollector_custom.properties` file with the new storename, storepass and keypass.
 - a. Stop the instance of the application server that is being monitored by the Data Collector.
 - b. Go to `DC_home/runtime/app_server_version.node_name.server_name/custom/datacollector_custom.properties`.
 - c. Set the following property definitions:

Note: All the following properties are set during the installation or at configuration time. By default you do not need to do anything. You only need to change the following properties if you have changed items that the following properties are referring to. All the keywords in angle (< >) brackets need to be replaced by the appropriate value.

 - The path of the certificate to use when communicating with the data collector. This is only needed when the data collector is operating in secure mode. The delimiter must be a semicolon on all platforms
`certificate.path=<AM_HOME>/etc/dctomgmt.cer;<AM_HOME>/etc/dctoproxy.cer.`
 - The keystore location of the data collector
`keystore.location=@{AM_HOME}/etc/CyaneaDCStore.`
 - The keystore password of data collector server
`keystore.storepass=oakland94612.`
 - The keystore key password of data collector server
`keystore.keypass=oakland94612.`
 - d. Start the instance of the application server that is monitored by the data collector for the property changes to take effect.
4. Restart the Managing Server to implement the changes made to the Managing Server and Data Collector. See *IBM Tivoli Composite Application Manager for Application Diagnostics Managing Server Installation Guide*.

Secure Socket Layer communications

On distributed platforms, ITCAM for Application Diagnostics uses the SSL security protocol for integrity and confidentiality. You have the option of configuring all monitoring components to utilize SSL for communications. The following steps describe a sample HTTP-based SSL transaction using server-side certificates:

1. The client requests a secure session with the server.
2. The server provides a certificate, its public key, and a list of its ciphers to the client.
3. The client uses the certificate to authenticate the server (verify that the server is who it claims to be).
4. The client picks the strongest common cipher and uses the server's public key to encrypt a newly-generated session key.
5. The server decrypts the session key with its private key.
6. From this point forward, the client and server use the session key to encrypt all messages.

The monitoring software uses the Java Secure Sockets Extensions (JSSE) API to create SSL sockets in Java applications.

Note: If you performed an embedded installation of the IBM WebSphere Application Server with the Managing Server, use the IBM WebSphere Application Server default key. For more information on IBM WebSphere Application Server default keys, refer to the IBM WebSphere Application Server documentation.

This section describes how to customize the default settings for SSL authentication in ITCAM for Application Diagnostics.

Password encryption and Kernel property file encryption

The `amcrypto.sh` script comes with the Managing Server and is present in `MS_home/bin` to encrypt the passwords related to Node Authentication and SSL.

Password encryption

To encrypt a password, complete the following steps:

1. Enter:

```
amcrypto.sh -encrypt password
```

The password is written to stdout.
2. Copy this encrypted password and place it in the appropriate config files.
Currently password encryption is supported only for the following property values on both the Managing Server and Data Collectors:
 - KEYSTR_PASS and KEYSTR_KEYPASS in `MS_home/bin/setenv.sh`
 - JDBC_PASSWORD in `MS_home/bin/setenv.sh`. for full instructions for changing the Java Database Connectivity (JDBC) user ID and password for the database Schema user.
 - keystore.storepass, keystore.keypass using the same method mentioned in the Step 2 on page 61.
 - keystore.storepass and keystore.keypass in `DC_home/runtime/app_server_version.node_name.server_name/custom/datacollector_custom.properties` file.
3. Restart the Managing Server to activate the password encryption changes:
 - a. If it is not already stopped, stop the Managing Server.
 - b. Start the Managing Server.
4. Restart the VE application server.

Properties file encryption

Complete the following steps:

1. To encrypt a properties file, use:

```
amcrypto.sh -encryptPropertyFile file
```

The *file* is `kl1.properties` or `kl2.properties` in `MS_home/etc`. This command encrypts the given input file and stores it in a file with different name. The user can back up the existing properties file and have it replaced by the encrypted file for more security.
2. To decrypt a properties file, use:

```
amcrypto.sh -decryptPropertyFile file
```

The *file* is `kl1.properties` or `kl2.properties` in `MS_home/etc`. This command decrypts the given file and writes the decrypted file to another file with a different name.
3. Restart the Managing Server to activate the changes:

- a. If it is not already stopped, stop the Managing Server.
- b. Start the Managing Server.

Enabling Secure Socket Layer at the Data Collector level

To enable SSL, enable Node Authentication first (See “Node Authentication” on page 59). SSL works only with Node Authentication enabled.

Configuration with default options involves setting one property to true to operate the Data Collector in SSL mode:

1. In the `DC_home/runtime/app_server_version.node_name.server_name/custom/datacollector_custom.properties` file, set the following property to true by removing the comment symbol (#) in front of the property definition (by default, this property is commented out).

```
comm.use.ssl.dc=true
```

2. Restart the application server.

Note: On the managing server only the Kernel-related changes need to be enabled other managing server components are enabled automatically.

Verifying secure communications

To verify SSL is properly configured, look for the message labeled CYND4051I in one of the following files:

Table 12. Location of the CYND4051I message

Windows	C:\Program Files\IBM\tivoli\common\CYN\logs\node_name.server_name\java_msg_log_file. For example: C:\Program Files\IBM\tivoli\common\CYN\logs\IBMNode01.server1/msg-dc-Ext.log
UNIX and Linux	/var/ibm/tivoli/common/CYN/logs/node_name.server_name/java_msg_log_file. For example: /var/ibm/tivoli/common/CYN/logs/IBMNode01.server1/msg-dc-Ext.log
z/OS	[ITCAM_CONFIG]/runtime/wasXX.node.server/logs/CYN/logs
IBM i	/QIBM/UserData/tivoli/common/CYN/logs/node_name.server_name/java_msg_log_file. For example: /QIBM/UserData/tivoli/common/CYN/logs/IBMNode01.server1/msg-dc-Ext.log

That message includes the text Join Proxy Server and Kernel successfully.

Only the CommandAgent port uses SSL. Other ports opened by the Data Collector (the ProbeController port and the Data Collector - Publish Server port do not use SSL. Therefore, when SSL is enabled, only the data on the channels connected to the CommandAgent port is encrypted.

All the data processed on the CommandAgent channel is encrypted when SSL is enabled. The data can be classified as follows:

Table 13. Classification of the data processed on the CommandAgent channel

Classification	Data
Command and control data	Configuring and unconfiguring the Data Collector
User actions related to threads	- Starting and stopping JVM threads - Changing thread priorities - Getting thread priorities and thread status - Requesting drill down information to see cookies, etc ... - Generating thread dumps - Getting thread stack traces
System information	- information - Operating system platform information - JVM information
Application information	- All the applications installed on the monitored - Application binaries and location information - Thread pool information related to JMS, JCA, JTA, Servlet, EJB, etc ... - Data source information
Performance data	All Performance Monitoring Infrastructure data
Transport data	- ORB data - SOAP ports
Memory Information	- Obtaining JVM Heap Snapshot data - Performing memory leak analysis - Performing heap dump

Privacy filtering

The following procedures describe how to enable and verify privacy filtering.

Enabling privacy filtering

Privacy filtering is used to filter out SQL, cookie, and HTTP request query strings and other private data, for example drivers license numbers. When this property is set to true, this data is not collected by the Data Collector.

1. Stop the instance of application server that is being monitored by the Data Collector.
2. Go to `DC_home/runtime/app_server_version.node_name.server_name/custom/datacollector_custom.properties`.
3. Set the following property definition:
`secure.filter.on=true`
4. Start the instance of application server that is being monitored by the Data Collector.

Verifying privacy filtering

The following statement is printed out to the Data Collector log when privacy filtering is properly configured:

Privacy Filter is On. Http Request Query String, SQL String and Http Cookie data is not trasmitted.

The log file is trace-dc.log.

Script to run if your SSL certificates have expired

All SSL certificates have an expiration time. For some certificates, the expiration time is 4 years, after which the product will not function if you have enabled Node Authentication and SSL. If this is the case, to increase the expiration time, perform the following procedure:

1. Open the script located at *MS_home/bin/security_cert.sh* with a text editor. This is the content of the script:

```
#!/bin/sh

# (C) Copyright IBM Corp. 2005 All Rights Reserved.
#
# US Government Users Restricted Rights - Use, duplication or
# disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
#

# Note: This script requires $JDK_HOME to be defined and it requires
# JDK_HOME/bin/keytool to be present. This keytool is available in FULL JDK
# versions and may not be available in JRE versions of the install

# PLEASE DEFINE JDK HOME

JDK_HOME=/opt/IBM/WebSphere/AppServer6/java

PATH=${JDK_HOME}/bin:$PATH

# This script generates ALL the certificates and certificate stores required for
# ITCAMfWAS Product (DC/MS/Port Consolidator). Currently it populates
# certificates with validity of 7000 days. If you feel its too high replace
# validity period to a lower number according to your needs. Please Note: once
# limit is reached, Product will stop working when NodeAuthentication/SSL is ON
# Its your responsibility to re-generate the certificates and stores.
# Please replace ALL the certificates at DC, MS and PortCosolidator level.
# Partial replacement will NOT work

keytool -genkey -alias mgmttomgmt -keyalg RSA -keysize 1024 -sigalg MD5withRSA -validity 7000
-keypass cyanea94612 -keystore ./CyaneaMgmtStore -storepass cyanea94612 -dname
"cn=cyaneamgmt, OU=CyaneaComp, O=Cyanea, L=Oakland, ST=CA, C=US"

keytool -genkey -alias dctomgmt -keyalg RSA -keysize 1024 -sigalg MD5withRSA -validity 7000
-keypass cyanea94612 -keystore ./CyaneaMgmtStore -storepass cyanea94612 -dname
"cn=cyaneadc, OU=CyaneaComp, O=Cyanea, L=Oakland, ST=CA, C=US"

keytool -genkey -alias proxytomgmt -keyalg RSA -keysize 1024 -sigalg MD5withRSA
-validity 7000 -keypass cyanea94612 -keystore ./CyaneaMgmtStore -storepass cyanea94612
-dname "cn=cyaneaproxy, OU=CyaneaComp, O=Cyanea, L=Oakland, ST=CA, C=US"

keytool -genkey -alias proxytodc -keyalg RSA -keysize 1024 -sigalg MD5withRSA
-validity 7000 -keypass oakland94612 -keystore ./CyaneaDCStore -storepass oakland94612
-dname "cn=cyaneaproxy, OU=CyaneaComp, O=Cyanea, L=Oakland, ST=CA, C=US"

keytool -genkey -alias mgmttodc -keyalg RSA -keysize 1024 -sigalg MD5withRSA
-validity 7000 -keypass oakland94612 -keystore ./CyaneaDCStore -storepass oakland94612
-dname "cn=cyaneamgmt, OU=CyaneaComp, O=Cyanea, L=Oakland, ST=CA, C=US"

keytool -genkey -alias mgmttproxy -keyalg RSA -keysize 1024 -sigalg MD5withRSA
-validity 7000 -keypass oakland94612 -keystore ./CyaneaProxyStore -storepass oakland94612
-dname "cn=cyaneamgmt, OU=CyaneaComp, O=Cyanea, L=Oakland, ST=CA, C=US"

keytool -genkey -alias dctoproxy -keyalg RSA -keysize 1024 -sigalg MD5withRSA
-validity 7000 -keypass oakland94612 -keystore ./CyaneaProxyStore -storepass oakland94612
-dname "cn=cyaneadc, OU=CyaneaComp, O=Cyanea, L=Oakland, ST=CA, C=US"

keytool -export -alias mgmttomgmt -keypass cyanea94612 -keystore ./CyaneaMgmtStore
```

```

-storepass cyanea94612 -file mgmttomgmt.cer

keytool -export -alias dctomgmt -keystore ./CyaneaMgmtStore -storepass cyanea94612 -file dctomgmt.cer

keytool -export -alias proxytomgmt -keystore ./CyaneaMgmtStore -storepass cyanea94612 -file proxytomgmt.cer

keytool -export -alias proxytodc -keystore ./CyaneaDCStore -storepass oakland94612 -file proxytodc.cer

keytool -export -alias mgmttodc -keystore ./CyaneaDCStore -storepass oakland94612 -file mgmttodc.cer

keytool -export -alias mgmttoproxy -keystore ./CyaneaProxyStore -storepass oakland94612 -file mgmttoproxy.cer

keytool -export -alias dctoproxy -keystore ./CyaneaProxyStore -storepass oakland94612 -file dctoproxy.cer

cp ./CyaneaMgmtStore ./CyaneaMgmtStore_Comm
cp ./CyaneaDCStore ./CyaneaDCStore_Comm
cp ./CyaneaProxyStore ./CyaneaProxyStore_Comm

keytool -keystore ./CyaneaMgmtStore_Comm -storepass cyanea94612 -import -alias mgmttodc -file ./mgmttodc.cer

keytool -keystore ./CyaneaMgmtStore_Comm -storepass cyanea94612 -import -alias mgmttoproxy -file ./mgmttoproxy.cer

keytool -keystore ./CyaneaDCStore_Comm -storepass oakland94612 -import -alias dctomgmt -file ./dctomgmt.cer

keytool -keystore ./CyaneaDCStore_Comm -storepass oakland94612 -import -alias dctoproxy -file ./dctoproxy.cer

keytool -keystore ./CyaneaProxyStore_Comm -storepass oakland94612 -import -alias proxytodc -file ./proxytodc.cer

keytool -keystore ./CyaneaProxyStore_Comm -storepass oakland94612 -import -alias proxytomgmt -file ./proxytomgmt.cer

```

2. Specify the path for the location of the Java home directory for the JDK_HOME parameter. For example,
JDK_HOME=D:\IBM\AppServer\java
3. If the increase in expiration time to 20 years (7000 days) is too much, modify the script. Change the value of -validity 7000 to a lower number of days, in all instances it occurs in the script. For example, change all instances of -validity 7000 to -validity 3500.
4. Save the changes and run the script.

Settings for the Data Collector if Java 2 security is enabled

By default, Data Collector configuration enables Java 2 security on the application server, and sets a permissive policy. This policy ensures that the Data Collector can run properly, and provides no other security protection. If you need a more restrictive policy, perform the following procedure to ensure that the policy becomes active and the Data Collector can still work properly.

The Data Collector sets the Java security policy file location for all monitored application server instances (java.security.policy system property) to *DC_home/itcamdc/etc/datacollector.policy*. You must edit this file in the following way:

- Remove all existing content.
- Copy the sample security policy for the Data Collector from the file `DC_home/itcamdc/etc/datacollector.security.policy`.
- If ITCAM for Transactions is installed on the server, add a grant statement for the ITCAM for Transactions code base to the security policy file. Follow the model for the grant statements provided in the sample `datacollector.security.policy` file, but use the ITCAM for Transactions installation root directory in the `codeBase` statement.
- Add your required security policy settings.

Save the file, and create a backup copy.

Attention: Each time you configure or reconfigure the Data Collector for an application server instance, the file `DC_home/itcamdc/etc/datacollector.policy` might be overwritten. To ensure that your security policy remains active, restore this file from the backup copy after configuring or reconfiguring the Data Collector for any application server instance.

Appendix B. Starting and stopping the monitoring environment

This chapter contains procedures for starting and stopping ITCAM for Application Diagnostics's various components, databases, and application servers.

Disabling and re-enabling a Data Collector

If you need to disable a Data Collector without unconfiguring or uninstalling it, perform the following procedure:

1. Log into the IBM WebSphere Application Server administrative console.
2. Click **Server > Application Servers** and select the *server_name*.
3. Navigate to **Process Definition > Java Virtual Machine > Custom Properties**.
4. Find a property with the name ITCAM_DC_ENABLE. If this property is not present, add it.
5. Set the value of this property to false.
6. Click **OK** or **Apply**. Then click **Save**.
7. Restart the application server (see “Restarting the application server”)

To re-enable a Data Collector that was disabled in this way, perform the following procedure:

1. Log into the IBM WebSphere Application Server administrative console.
2. Click **Server > Application Servers** and select the *server_name*.
3. Navigate to **Process Definition > Java Virtual Machine > Custom Properties**.
4. Find the property with the name ITCAM_DC_ENABLE.
5. Set the value of this property to true.
6. Click **OK** or **Apply**. Then click **Save**.
7. Restart the application server (see “Restarting the application server”)

Restarting the application server

There are separate procedures for restarting the application server in Network Deployment and non-Network Deployment environments.

Restarting the application server in a non-Network Deployment environment

To restart the application server, enter the following commands:

```
cd AppServer_home/profiles/profile_name/bin
stopServer server_name [options]
startServer server_name
```

The *server_name* is the name of the configuration directory of the server you want to restart. The default is server1.

The *profile_name* specifies the profile name. The default is default.

If WebSphere Global Security is enabled, add the following options to every command:

- The `-username name` or `-user name` option specifies the user name for authentication if security is enabled in the server.
- The `-password password` option specifies the password for authentication if security is enabled in the server.

Attention: If you are running in a secure environment but have not provided a user ID and password, you will receive an error message.

Restarting the application server in a Network Deployment environment

Complete the following steps to restart the application server:

1. Change to the `AppServer_home/bin` directory.
2. Stop all servers on the node, and the node itself. Run the `stopNode -stopservers` command.
3. Stop the Deployment Manager process. Run the `stopManager` command.
4. Start the Deployment Manager process. Run the `startManager` command.
5. Start the node. Run the `startNode` command.
6. For each application server on the node, start the application server using the procedure in “Starting the application server in a non-Network Deployment environment.”

On Linux or UNIX systems, add `./` before every command to run it.

If WebSphere Global Security is enabled, add the following options to every command:

- The `-username name` or `-user name` option specifies the user name for authentication if security is enabled in the server.
- The `-password password` option specifies the password for authentication if security is enabled in the server.

Attention: If you are running in a secure environment but have not provided a user ID and password, you will receive an error message.

Starting the application server

There are separate procedures for starting the application server in Network Deployment and non-Network Deployment environments.

Starting the application server in a non-Network Deployment environment

To start the application server, enter the following commands:

```
cd AppServer_home/profiles/profile_name/bin
startServer server_name
```

The `server_name` is the name of the configuration directory of the server you want to start. The default is `server1`.

The `profile_name` specifies the profile name for version 6 application servers. The default is `default`.

If WebSphere Global Security is enabled, add the following options to every command:

- The `-username name` or `-user name` option specifies the user name for authentication if security is enabled in the server.
- The `-password password` option specifies the password for authentication if security is enabled in the server.

Attention: If you are running in a secure environment but have not provided a user ID and password, you will receive an error message.

Starting the application server in a Network Deployment environment

Complete the following steps to start the application server:

1. Change to the `AppServer_home/bin` directory.
2. Start the Deployment Manager process. Run the `startManager` command.
3. Start the node. Run the `startNode` command.
4. For each application server on the node, start the application server using the procedure in “Starting the application server in a non-Network Deployment environment” on page 70.

On Linux or UNIX systems, add `./` before every command to run it.

If WebSphere Global Security is enabled, add the following options to every command:

- The `-username name` or `-user name` option specifies the user name for authentication if security is enabled in the server.
- The `-password password` option specifies the password for authentication if security is enabled in the server.

Attention: If you are running in a secure environment but have not provided a user ID and password, you will receive an error message.

Stopping the application server

There are separate procedures for stopping the application server in Network Deployment and non-Network Deployment environments.

Stopping the application server in a non-Network Deployment environment

To stop the application server, enter the following commands:

```
cd AppServer_home/profiles/profile_name/bin
stopServer server_name [options]
```

The `server_name` is the name of the configuration directory of the server you want to stop. The default is `server1`.

The `profile_name` specifies the profile name for version 6 application servers. The default is `default`.

If WebSphere Global Security is enabled, add the following options to every command:

- The `-username name` or `-user name` option specifies the user name for authentication if security is enabled in the server.

- The `-password password` option specifies the password for authentication if security is enabled in the server.

Attention: If you are running in a secure environment but have not provided a user ID and password, you will receive an error message.

Stopping the application server in a Network Deployment environment

Complete following steps to stop the application server:

1. Change to the `AppServer_home/bin` directory.
2. Stop all servers on the node, and the node itself. Run the `stopNode -stopservers` command
3. Stop the Deployment Manager process. Run the `stopManager` command.

On Linux or UNIX systems, add `./` before every command to run it.

If WebSphere Global Security is enabled, add the following options to every command:

- The `-username name` or `-user name` option specifies the user name for authentication if security is enabled in the server.
- The `-password password` option specifies the password for authentication if security is enabled in the server.

Attention: If you are running in a secure environment but have not provided a user ID and password, you will receive an error message.

Appendix C. Using regular expressions

Regular expressions are sets of symbols and characters that are used to match patterns of text. You can use regular expressions to search specific IP addresses across your Web environment. Regular expressions also enable you to search a simple, fixed URI or a complex URI pattern that matches one or more groups of transactions.

Regular expression library

An extensive library of regular expression characters and operators is available for your URI filters and IP address specifications. The International Components for Unicode (ICU) open-source development project provides this library for your use. The next section provides the most frequently used expressions for this product. However, you can refer to the following Web page for a full description of the ICU regular expression library and an explanation of how to use the characters and operators for complex expressions: <http://oss.software.ibm.com/icu/userguide/regex.html>

Frequently used regular expressions

The following list highlights characters and operators most frequently used in regular expressions:

**** Quotes the character that follows it, which treats that character as a literal character or operator (not a regular expression). When you want the following characters to be treated as literal, you must precede them with a backslash:

`* ? + [ ( ) { } ^ $ | \ . /`

In other words, use a backslash followed by a forward slash (`\`/`/`) to include a forward slash in a URI filter. Use a backslash followed by a period (`\`.) to include a period in a URI filter.

Example: to specify the URI pattern `http://www.ibm.com/`, use the following regular expression:

`http:\\www\\.ibm\\.com\\`

To specify all URIs that begin with `http://www.ibm.com/`, use the following regular expression:

`http:\\www\\.ibm\\.com\\.*`

. Matches any one character.

Example: to match both `ibm2` and `ibm3` within a string, use `ibm.` such as in the following example: `http:\\www\\.ibm\\.com\\`

(?: ...)

Non-capturing parentheses. Groups the included pattern, but does not provide capturing of matching text. Somewhat more efficient than capturing parentheses.

Example: you can use the non-capturing parenthesis to group expressions to form more complicated regular expressions. To match a URI that starts

with one of the following URLs: `http://www.ibm.com/marketing/` or `http://www.ibm.com/sales/`, you would do a grouping with a pipe sign (`|`) (represents *or*):

`http://www.ibm.com/(? :marketing)|(? :sales)/`

- * Matches the preceding element zero or more times. You must quote this character.

Example: the expression, `ca*t`, matches `cat`, `caat`, `ct`, and `caaaaat`. The term `cabt`, would not return as a match.

Specifying exclusions with the bang (!) operator (Quality of Service listening policies only)

Note: This section applies to the entry of URI and client IP filters for Quality of Service listening policies only.

You can use an exclamation point (!), also called the *bang* operator, to filter out transactions that might match the regular expressions already entered, but that are not to be considered valid transactions for this listening policy. These exclusions are considered negative filters. You can enter these exclusions as additional URI or client IP filters. The formatting of these additional filters is as follows:

URI Filter Exclusions

Use only fixed strings. For example, you can use the following strings:

```
!http://www.ibm.com/  
!http://www.ibm.com/hr/index.html  
!http://www.ibm.com/it/errorpage.html
```

Client IP Exclusions

The following are valid:

```
!*24.45.46  
!12.*45.56  
!12.24.*56  
!12.24.45.*  
!12.24.45.56
```

You can replace any "octet" (there are four in an IP address: `octet . octet . octet . octet`) with a wildcard (*). Note that this is not the regular expression wildcard (.) from the positive filters.

Appendix D. Support information

This chapter describes options for obtaining support for IBM products.

Searching knowledge bases

If you have a problem with your IBM software, you want it resolved quickly. Begin by searching the available knowledge bases to determine whether the resolution to your problem is already documented.

Finding Release Notes

You can find Release Note information online by viewing IBM Technotes. Technotes replace the Release Notes[®] manual for this product. *Technotes* are short documents that cover a single topic. You can search the Technote collection for common problems and solutions, as well as known limitations and workarounds. Technotes are continuously updated to provide current product information.

The following two procedures describe how to view Technotes and subscribe to have future Technotes e-mailed to you. Alternatively, you can watch demos of these procedures at the following Web site:

<http://www.ibm.com/software/support/sitetours.html>

Viewing Technotes

Perform the following actions to access Technotes for this product:

1. Launch the IBM Software Support Web site: <http://www.ibm.com/software/support>
2. Follow the instructions on the screen to search for the Technotes related to the issue encountered.

Subscribing to new Technotes

You can subscribe to an RSS feed of the product support page or subscribe to receive e-mail notification about product tips and newly published fixes through My support. To subscribe to an RSS news feed of the product support page, click the orange **RSS** button under the **Stay up to date** pane.

My Support is a personalized portal that enables you to:

- Specify the products for which you want to receive notifications
- Create a personalized page that provides product information for the products you use
- Choose from flashes, downloads, and Technotes
- Receive an e-mail update in your inbox

Perform the following actions to subscribe to My support e-mails:

1. Launch an IBM support Web site such as the following site:
<http://www.ibm.com/support/>
2. Click **My support** in the upper-right section of the page.
3. If you have not yet registered, click **register** in the upper-right corner of the support page to create your user ID and password.
4. Sign in to **My support**.

5. On the My support page, click **Add products**.
6. Make the following selections from the lists to add this product to your personal page:
 - a. Software
 - b. Systems Management
 - c. Application Performance & Availability
7. Click **Add products**.
8. Click **Subscribe to email**.
9. Set your preferences to specify the information you want in your e-mails.
10. Click **Update**.
11. Click **Submit**.

Searching the information center on your local system or network

IBM provides extensive documentation that can be installed on your local computer or on an intranet server. You can use the search function of this information center to query conceptual information, instructions for completing tasks, reference information, and support documents.

Searching the Internet

If you cannot find an answer to your question in the information center, search the Internet for the latest, most complete information that might help you resolve your problem. To search multiple Internet resources for your product, expand the product folder in the navigation frame to the left and select **Web search**. From this topic, you can search a variety of resources including:

- IBM technotes
- IBM downloads
- IBM Redbooks®
- IBM DeveloperWorks
- Forums and newsgroups
- Google

Obtaining fixes

A product fix might be available to resolve your problem. To determine what fixes are available for your IBM software product, follow these steps:

1. Go to the IBM Software Support Web site at <http://www.ibm.com/software/support>.
2. Click **Downloads** in the **Software** section.
3. Under the **Updates, drivers, and fixes** section, select **Fixes, fixpacks and utilities**.
4. Navigate to ITCAM for WebSphere or ITCAM for J2EE to obtain a list of available fixes.

For more information about the types of fixes that are available, see the *IBM Software Support Handbook* at <http://techsupport.services.ibm.com/guides/handbook.html>.

Receiving Weekly support updates

To receive e-mail notifications about software support news and updates, follow these steps:

1. Go to the IBM Software Support Web site at <http://www.ibm.com/software/support>.
2. On the right hand side, click **My Notifications**.
3. If you have already registered for **My Notifications**, login. If you have not registered, click **register now**. Complete the registration form using your e-mail address as your IBM ID. When you have logged in, the **My notifications for IBM technical support** home page is displayed.
4. Select the **Subscribe** tab.
5. Under the **Software** list, select **Tivoli**.
6. Select **Tivoli Composite Application Manager for J2EE** and/or **Tivoli Composite Application Manager for WebSphere**. Click **Continue**.
7. In the **Options** section, enter a folder name, update notifications will be saved in this folder.
8. In the **Notify me by** section, choose if you want to be notified of updates daily or weekly.
9. In the **Notify me by** section, choose if you want to receive notifications in plain text or html.
10. In the **Document Types** section, customize the types of information you want to be updated on, for example, white papers, drivers etc. Click **Submit**.

If you experience problems with the **My Notifications** feature, you can obtain help in one of the following ways:

Online

Send an e-mail message to erchelp@ca.ibm.com, describing your problem.

By phone

Call 1-800-IBM-4You (1-800-426-4968).

Contacting IBM Software Support

IBM Software Support provides assistance with product defects.

Before contacting IBM Software Support, your company must have an active IBM software maintenance contract, and you must be authorized to submit problems to IBM. The type of software maintenance contract that you need depends on the type of product you have:

- For IBM distributed software products (including, but not limited to, IBM Tivoli, IBM Lotus®, and IBM Rational® products, as well as IBM DB2 and IBM WebSphere Application Server products that run on Windows, UNIX, or Linux operating systems), enroll in IBM Passport Advantage® in one of the following ways:
 - **Online:** go to the IBM Passport Advantage Web page (<http://www.ibm.com/software/passportadvantage>) and click **How to Enroll**
 - **By phone:** for the phone number to call in your country, go to the IBM Software Support Web site (<http://www14.software.ibm.com/webapp/set2/sas/f/handbook/home.html>) and click the name of your geographic region.
- For IBM eServer™ software products (including, but not limited to, IBM DB2 and IBM WebSphere Application Server products that run in IBM zSeries®, IBM

pSeries®, and IBM iSeries® environments), you can purchase a software maintenance agreement by working directly with an IBM sales representative or an IBM Business Partner. For more information about support for eServer software products, go to the IBM Technical Support Advantage Web page (<http://www.ibm.com/servers/eserver/techsupport.html>).

If you are not sure what type of software maintenance contract you need, call 1-800-IBMSERV (1-800-426-7378) in the United States or, from other countries, go to the contacts page of the IBM Software Support Handbook on the Web (<http://www14.software.ibm.com/webapp/set2/sas/f/handbook/home.html>) and click the name of your geographic region for phone numbers of people who provide support for your location.

Follow the steps in this topic to contact IBM Software Support:

1. Determine the business impact of your problem.
2. Describe your problem and gather background information.
3. Submit your problem to IBM Software Support.

Determining the business impact of your problem

When you report a problem to IBM, you are asked to supply a severity level. Therefore, you need to understand and assess the business impact of the problem you are reporting. Use the following criteria:

Table 14. Criteria for assessing the business impact of your problem

Severity 1	Critical business impact: you are unable to use the program, resulting in a critical impact on operations. This condition requires an immediate solution.
Severity 2	Significant business impact: the program is usable but is severely limited.
Severity 3	Some business impact: the program is usable with less significant features (not critical to operations) unavailable.
Severity 4	Minimal business impact: the problem causes little impact on operations, or a reasonable circumvention to the problem has been implemented.

Describing your problem and gathering background information

When explaining a problem to IBM, be as specific as possible. Include all relevant background information so that IBM Software Support specialists can help you solve the problem efficiently. To save time, know the answers to these questions:

- What software versions were you running when the problem occurred?
- Do you have logs, traces, and messages that are related to the problem symptoms? IBM Software Support is likely to ask for this information.
- Can the problem be re-created? If so, what steps led to the failure?
- Have any changes been made to the system? (For example, hardware, operating system, networking software, and so on.)
- Are you currently using a workaround for this problem? If so, please be prepared to explain it when you report the problem.

Submitting your problem to IBM Software Support

You can submit your problem in one of two ways:

- **Online:** go to the "Submit and track problems" page on the IBM Software Support site (<http://www.ibm.com/software/support/probsub.html>). Enter your information into the appropriate problem submission tool.
- Do you have logs, traces, and messages that are related to the problem symptoms? IBM Software Support is likely to ask for this information.
- Can the problem be re-created? If so, what steps led to the failure?
- Have any changes been made to the system? (For example, hardware, operating system, networking software, and so on.)
- Are you currently using a workaround for this problem? If so, please be prepared to explain it when you report the problem.

If the problem you submit is for a software defect or for missing or inaccurate documentation, IBM Software Support creates an Authorized Program Analysis Report (APAR). The APAR describes the problem in detail. Whenever possible, IBM Software Support provides a workaround for you to implement until the APAR is resolved and a fix is delivered. IBM publishes resolved APARs on the IBM product support Web pages daily, so that other users who experience the same problem can benefit from the same resolutions.

For more information about problem resolution, see Searching knowledge bases and Obtaining fixes.

Tivoli Support Technical Exchange

You can become a participant in the new Tivoli Support Technical Exchange, where you can expand your technical understanding of your current Tivoli products in a convenient format hosted by Tivoli support engineers. This program provides support discussions about product information, troubleshooting tips, common issues, problem solving resources and other topics. As Exchange leaders, Tivoli engineers provide subject matter expert direction and value. Participating in the Exchange helps you manage your Tivoli products with increased effectiveness.

What do you do to participate? Review the schedule of Exchange sessions. Find a topic of interest and select **register**. Provide your name, phone number, company name, number of attendees, the Exchange Topic and IBM Customer number. You will be invited to attend a 1-hour to 2-hour conference call where the information is presented. The new Tivoli Support Technical Exchange can help with the following areas:

- Increased product knowledge
- Ways to avoid common pitfalls
- Support recommendations
- Proactive customer support
- Helpful hints and tips
- Knowledge transfer
- Expansion of your knowledge base

For more information or to suggest a future Exchange session, contact Support Technical Exchange (xchange@us.ibm.com). To learn more, visit the following Web site: http://www.ibm.com/software/sysmgmt/products/support/supp_tech_exch.html

Appendix E. 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 through 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>

Index

Special characters

-Xrun parameters, removing 9

A

accessibility xii, 81
application server
 restarting 69
 starting 70
 stopping 71
AppServer_home xiii
Asynchronous Bean requests 44
autoconfiguration 55

B

books ix
Byte Code Instrumentation, disabling
 types of 33

C

CICS 52
classic JVM 50
configuring 13
conventions
 typeface xiii
CTG 52
custom MBeans 45, 46
custom requests 42
customer support 77

D

Data Collector 5
 configuring 13
 disabling 69
 installing 13
 moving to a different host
 computer 51
 permissions 9
Data Collector buffering 31
Data Collector properties 29
datacollector_custom.properties 29
datacollector.properties 29
DC_home xiii
directories, variables for xiii
disabling Data Collectors 69

E

encryption 63

F

firewall 10
fixes, obtaining 76

G

garbage collection
 interval 52
garbage connection
 log path 54

H

heap size
 increasing 10, 24

I

information centers, searching to find
 software problem resolution 76
installation directory 9
installing 13
instrumentation 35
Internet, searching to find software
 problem resolution 76
IP address 50
ISA xii

J

JVM 50

K

keystore management 60
knowledge bases, searching to find
 software problem resolution 75

L

lock analysis 35, 36

M

Managing Server
 changing 50
 matching time zone settings with 24
manuals ix
MBeans 45
memory leak analysis 38
memory leak diagnosis 35
memory monitoring 49
method entry and exit analysis 35, 40
method tracing 35, 40
Monitoring Agent 10
MQ requests, enabling 23

N

NATs 51
Network Deployment 55
network interfaces 51

Node Authentication 59
 Data Collector 60
 Managing Server 59

O

ordering publications xi

P

Performance Monitoring Infrastructure
 customizing 48
 Service Integration Bus 48
permissions
 Data Collector 9
physical disability 81
populating certificates 60
privacy filtering 65
problem determination
 describing problem for IBM Software
 Support 78
 determining business impact for IBM
 Software Support 78
 submitting problem to IBM Software
 Support 78
properties files 29
publications ix
 ordering xi

Q

Quality of Service
 using regular expressions 74

R

reader requirements ix
regular expressions 73
 bang (!) operator 74
 frequently used 73
 library 73
 Quality of Service 74
Release Notes 75
requirements for readers ix
response file 19
restarting application servers 69
RMI/IIOP requests, enabling
 instrumentation 33

S

secure communications, verifying 64
security 59
Service Integration Bus
 Performance Monitoring Infrastructure
 settings 48
silent installation
 IBM i
 from local IBM i computer 14

- silent installation (*continued*)
 - IBM i (*continued*)
 - using remote attached Linux client 15
 - using remote attached Windows client 16
- Software Support
 - contacting 77
 - describing problem for IBM Software Support 78
 - determining business impact for IBM Software Support 78
 - receiving weekly updates 77
 - submitting problem to IBM Software Support 78
- SSL 62
 - Data Collector 64
- starting
 - application servers 70
- stopping
 - application servers 71

T

- Technotes
 - e-mail subscriptions 75
 - viewing 75
- The IBM Support Assistant xii
- time zone settings, matching with Managing Server 24
- Tivoli Enterprise Portal
 - benefits 3
- Tivoli Support Technical Exchange 79
- toolkit properties 29
- toolkit_custom.properties 29
- toolkit_global_custom.properties 29
- toolkit.properties 29
- Transaction Service properties,
 - increasing 24
- Turbo Mode 32
- typeface conventions xiii

U

- uninstalling 25

V

- variables for directories xiii
- virtual machine 50

W

- Web Services 49

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at “Copyright and trademark information” at www.ibm.com/legal/copytrade.shtml.

Adobe is either a registered trademark or a trademark of Adobe Systems Incorporated in the United States, other countries, or both.

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

Windows is a trademark of Microsoft Corporation in the United States, other countries, or both.

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



Java and all Java-based trademarks are trademarks or registered trademarks of Oracle and/or its affiliates.

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

Notices

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

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

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785 U.S.A.

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

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
1623-14, Shimotsuruma, Yamato-shi
Kanagawa 242-8502 Japan

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

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

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

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

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

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

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

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

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

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

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

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

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

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

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to

IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM's application programming interfaces.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. _enter the year or years_. All rights reserved.

If you are viewing this information in softcopy form, the photographs and color illustrations might not be displayed.



Printed in USA

SC27-2826-00

