JENNIFER®
Enterprise Web Business Monitoring

JENNIFER.NET Installation Guide

JENNIFER'S KEY FEATURES

- Comprehensive Dashboard
- Real-time Resource/Service Monitoring
- Performance Problem Diagnosis and Resolution
- Application Tracing and Tuning
- Statistical Analysis and Reporting on the Performance Data
- Dynamic Profiling and StackTrace

JENNIFER.NET v4
Installation Guide

JS-0810-06

JENNIFER v4
Installation Guide

Written By **Kevin, Jung**
Published By **JENNIFERSOFT,Inc**
Edited By **Amie Song**
First Edition **Dec, 25th 2010**
Document version **4.5**
Document Number **JS-0810-06**
Reference Book **JENNIFER v4 User's Guide**

Table of Contents

Copyright

Trademarks

JENNIFER™ and JenniferSoft® are the trademarks of JenniferSoft,Inc in the United States, other countries, or both. All Rights Reserved under Copyright and intellectual property rights. As a condition of your use of the trademarks, you will not use the trademarks for any purpose that is unlawful or prohibited by these terms, conditions, and notices.

JenniferSoft® JENNIFER™

Other company, product, and service names may be trademarks or service marks of others. Any rights not expressly granted herein are reserved.

Upload, or otherwise make available, files that contain images, trademarks or other material protected by intellectual property laws, including, by way of example, and not as limitation, copyright or trademark laws (or by rights of privacy or publicity) unless you own or control the rights thereto or have received all necessary consent to do the same.

Sun, Sun Microsystems, Solaris, Java, JavaEE, EJB, GlassFish, JMX, JNI, JSP, JDBC, JNDI, and JVM are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

IBM, WebSphere, AIX, iSeries, z/OS, iSeries, DB2, CICS and CTG are trademarks of IBM Corporationin the United States, other countries, or both.

BEA, WebLogic, WLI, ALSB, TUXEDO, Jolt and WTC are trademarks of Oracle, Inc the United States, other countries, or both.

Microsoft, Window 2000, Vista, XP, and NT are trademarks of Microsoft Corporation in the United States, other countries, or both.

The following terms are trademarks of other companies:

Fujitsu® Interstage®
Hitachi® Cosminexus®
Sybase® EAServer®
Macromedia® JRun®
Tmaxsoft® JEUS® WebT®
Apache® Apache Derby® Jakarta Tomcat® DBCP®
Caucho Technology® Resin®
RedHat® JBoss®

Apache® JServ®
HP-UX® Itanium®
Sun Solaris®
Linux®
Compaq® True64®
Unipoint® J*Link®

Preface

This document comprehensively provides the detailed guidelines to install the JENNIFER.NET on the Microsoft IIS server and COM+ server.

People who should read this document include technical support engineers for JenniferSoft, customers who operates JENNIFER on a regular basis, as well as people who simply wish to understand the detailed functions of JENNIFER.

JENNIFER is an application performance management solution. We believe that JENNIFER can deliver special values and results to people who are in charge of performance tuning for capacity estimation and problem analysis, database performance tuning and the extraction of SQL queries from an application, as well as system operators for performance management and application developers for application performance improvements.

Before reading this document, it is essential for you to have an understanding of the basic environment for the Microsoft .NET Framework, be able to develop applications and analyze source code, and have sufficient knowledge regarding the installation, organization and operation of the IIS server. In addition, a basic knowledge of TCP/IP and UDP communication network protocols for inter-system communication is required, as well as a minimal understanding of TCP/UDP port setting for systems.

In addition, readers should fully understand the concept of a DB connection between the IIS and the database and have sufficient knowledge of the specific DB connections applied to the system.

Furthermore, readers should have some general understanding of the architecture of the Web Server and DB servers operating in a web-based system and fundamental terminologies for performance analysis, such as visitors, concurrent users, throughputs and response time.

This document does not include any content related to operation and management of JENNIFER v4. For specifics on operation and management, you are recommended to refer to the JENNIFER v4 Manual.

For further details of the application or specific case studies, you are recommended to refer to an additional guideline or technical notes provided on the JenniferSoft website.

Finally, I would like to thank all our customers and supporters, who have given positive feedback on JenniferSoft and its products prior to the release of JENNIFER v4 to the market, as well as the engineers at our partner companies who are fully devoted to providing technical support.

The team that wrote this book

Kevin Jung Senior Researcher of JenniferSoft R&D Institute

Based on the wide range of field experiences in .NET programming, Kevin Jung is mainly working on developing application performance monitoring solution, JENNIFER .NET and writing a technical documents overing .NET architecture.

In addition, Senior Manager Kyoung-sik Yoon(Sam) and Khalid of the JenniferSoft R&D team have also made significant contributions by verifying, proofreading and testing the content of this document.

In addition, Marketing Manager Amie Song from the overseas marketing division, helped to prepare and edit this guide as a coordinator.

Comments welcome

Your comments will be most beneficial to us!
Issues, concerns, and/or any other comments that you have regarding this book can be mailed to:
E-mail: manual@jennifersoft.com

Mail your comments to:

360 Fairview Way Milpitas, CA 95035

Chapter introduction

This document is organized in a certain sequence to help readers to understand the concepts. However, you may skip to the chapters that are most relevant to you if you wish, as each chapter is independent from each other. This document has the following chapters.

Chapter 1 **"Installing and Removing Agent for JENNIFER .NET"**

Chapter 2 **"Using the IISConfigHelper Application Program to Compose the IIS"**

Chapter 3 **"JENNIFER .NET Agent Composition : IIS**

Chapter 4 **"JENNIFER .NET Agent Installation and Removal: COM+**

Chapter 5 **"JENNIFER .NET Agent Composition : COM+**

Chapter 6 **“JENNIFER .NET Agent Installation and Unistallation: Independently Executable Application Programs and NT Services”**

About marks

The following marks are commonly used in this document. Before reading this document, it is essential for you to have an understanding of these marks.

Mark	Description
Arial (Bold)	Main font, method/class/file/directory name,
Body Box Courier	Code font Ex) `http_service_class = mysys.AServlet;mysys.BServlet`
Courier Bold	Code bold Ex) **`http_service_class = mysys.AServlet;mysys.BServlet`**
[Menu]-Bold	JENNIFER Dashboard menu Ex) **[Dashboard | JENNIFER Dashboard]**
[Main menu ⏐ Sub menu]	Main/Sub menu order
[Delimiter]	Delimiter Ex) [,]
Notice/Warning	Tipbox
Reference	Ex) [Transaction Data(32 page)]
Option name	**config_refresh_check_interval**

1

JENNIFER Server Installation

This chapter briefly describes about how to install or configure the JENNIFER server.

1.1. JENNIFER Server Specification

The following are the minimum hardware specifications for running the JENNIFER Server under the conditions : monitoring less than 10 of JENNIFER Agents with maximum 100 TPS(Transaction Per Second).

Table 1-1: JENNIFER Server Specification

Item	Recommended Specifications
CPU	Pentium 4 or higher(Intel)
Memory	More than 2GB
Hard Disk	More than 10GB - depending on the system size.
OS	Windows, Linux, AIX, HP, Solaris
JAVA VERSION	1.5 or higher
DB	Derby(Defalut), Oracle, DB2

The JENNIFER server operates on the most system installed JAVA and the above hardware specification can be changed depending on the system size and data amount.

1.2. JENNIFER Server Installation

1.2.1. Windows

1.2.1.1. JENNIFER Server Installation

The JENNIFER server runs based on Tomcat5.5. You can operates the JENNIFER server after setting the JAVA_HOME environment variables.

Table 1-2: System Environment Variables for Installation of the JENNIFER Server

Environment Variables	Values
Home directory for the JENNIFER server	/home/jennifer/server
[JAVA_HOME]	/usr/java15
JENNIFER server' s IP Address	192.168.0.139

1. Uncompress the Jennifer zip file that you downloaded in the JENNIFER installation directory(ex:c:\jennifer).
2. Execute cmd **[Start |Run | cmd input and execute]**
3. Check that the JAVA_HOME is set in the Windows environment variables.

```
C:\Java>echo %JAVA_HOME%
C:\Java\jdk16u13
```

4. If not, set JAVA_HOME environment variables in the **[Clicking the right mouse button on the MY Computer icon| Properties |Advanced | Environment Variables]**.
5. In the cmd, move to the JENNIFER server directory (ex:c:\jennifer\server\bin)and execute the startup.bat file. If you input the new environment variables, reexecute the cmd. You can execute the startup.bat directly in the Explorer.exe. When the JENNIFER server is started for the first time, the database table is created. For this reason, it may take a very long time depending on the server specifications.
6. Log-in.

If you enter the following URL, which consists of the port number and the domain address or the IP address of the server where the JENNIFER server is installed, the log-in screen will appear. All of the JENNIFER functions can be used after you log in.

The default port number for the JENNIFER server is 7900..

```
http://jennifer_server_ip:7900
```

If you want to log in, you need to enter your password and click the **[Log-in]** button. The administrator’ s ID is “admin,” and its password is also “admin.” After you log in for the first time, you are strongly recommended to change the password.

In addition, since the grid is implemented in flash files, Adobe Flash Player 9.0 or higher is requird.

1.3. JENNIFER Client Setting

The JENNIFER client is web-based, and therefore the user accesses the JENNIFER user interface through a web browser. Since the charts are implemented via Java applets, the web browser that is used must have the Java plug-in. In addition, since the grid is implemented in flash files, a flash player is required.

• Operating System

The JENNIFER client supports Microsoft windows XP and Vista. If you choose Windows classic theme on the windows XP, JENNIFER Black skin staly does not displayed properly. You can check the current theme on the [Properties] menu of the context menu by clicking the right mouse button.

Notice: Officially, the JENNIFER client does not support Linux or Mac OS. However, if your operating system can allow the installation of Firefox 3.0 or higher, with the Java plug-in 1.6.0_10 or higher, you should be able to use the JENNIFER client.

• Web Browser

Various web browsers are supported, including Microsoft IE 7.0 and 8.0, and Mozilla firefox 3.0.

Warning: IE 6.0 will not be supported. IE 6.0 does not fully meet the web standard inclusing CSS and not support PNG image file clearly. Therefore, JENNIFER does not support IE 6.0. We recommend you to use Firefox 3.0 unless you upgrade it into IE 7.0 or 8.0.

If you want to user JENNIFER, you should set the web browser in such that it can use cookies and Java scripts. Most of web browsers are basically set to use these functions.

• Java Plug-in

Since some aspects of the JENNIFER client are implemented via Java applets, you must install the Sun Java plug-in1.6.0_10 or higher.

Warning: JENNIFER 4.0 or higher does not support Microsoft Java VM.

• Installing and Uninstalling

To install or uninstall the Java plug-in, you should refer to the Sun Microsystems Java site at (http://java.sun.com/javase/downloads/index.jsp).

• Memory Setting

The default amount of maximum Java heap memory that Java applets can use is 96 MB(64 MB up to Java 1.6.0_6). You have to set the max/min value of the Java heap memory in order to use the JENNIFER client more stable. In most cases, the default setting is sufficient, but if there are too many agents or if the amount of processing is too great, then you need to set the maximum Java heap memory to a higher amount.

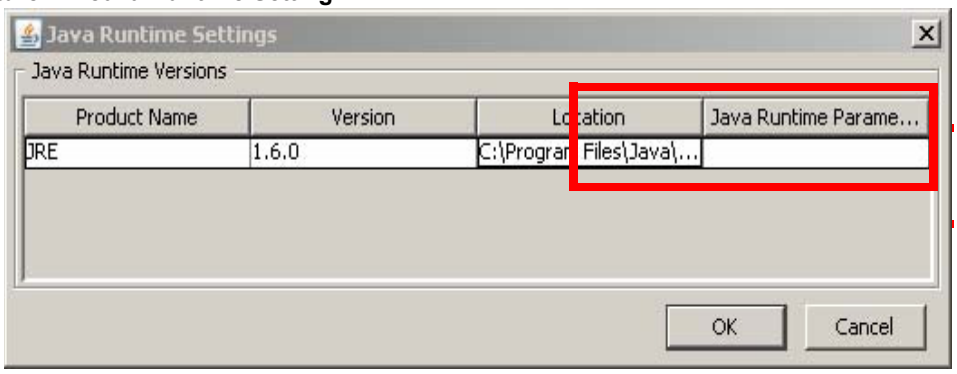
In Windows OS, you can set the Java heap memory amount of your Java plug-in as follows.

1. Move to the control panel.
2. Double-click on the Java icon in the control panel, and the Java control panel will appear. Click the Java tab.

Picture 1-1:Java Control Panel

3. If you click the **[View]** button in the Java applet runtime setting menu, the Java runtime setting screen will appear. If you set multiple Java, select the one that is to be used.

Picture 1-2:Java Runtime Setting



4. For instance, if you want to set the minimum and maximum Java heap memory to 100MB and 150MB, you should enter the following in the Java runtime variable column as follows:

```
-Xms100m -Xmx150m
```

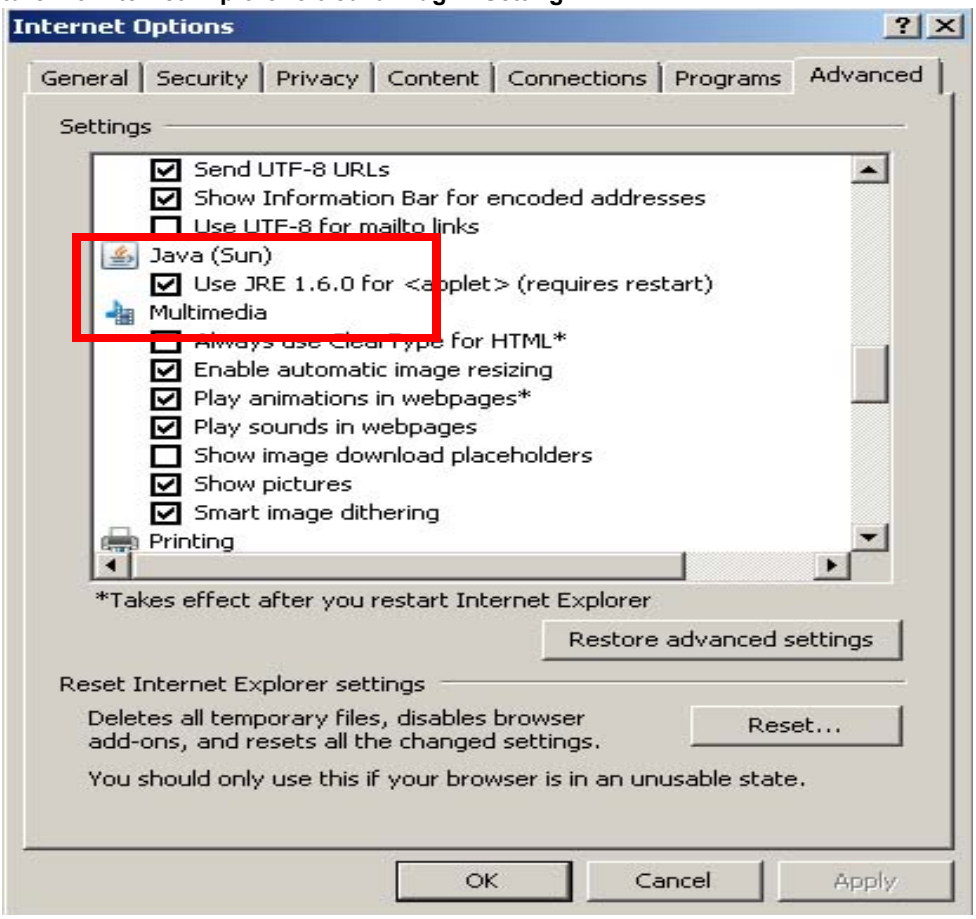
The maximum Java heap memory is affected by the computer specifications and environment. If you exceed the maximum setting, the web browser will be shut down or the following error message will appear:

Since there is no clear limit for the maximum permissible Java heap memory, a user must find the optimum setting for his or her needs through a process of trial and error.

• Web Browser Setting-Java Applet

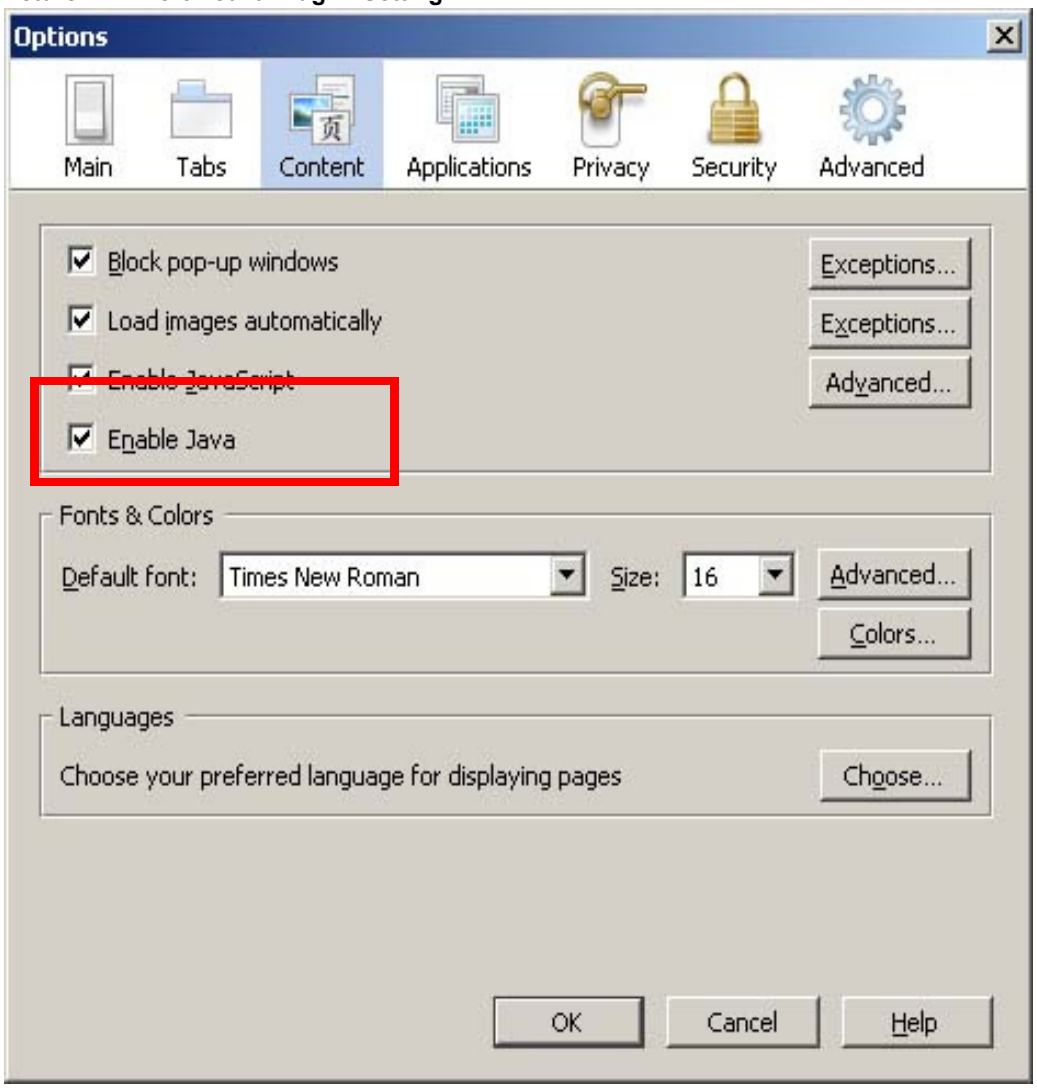
If you use Microsoft IE 7.0 or IE 8.0, then you should activate the Java plug-in by selecting [Tools | Internet options] from the menu. After clicking the [Advanced] tab in the internet options menu, scroll to the bottom of the list, and you will be able to see the Java plug-in information. Here, you should activate the Java plug-in.

Picture 1-3:Internet Explorer 8.0 Java Plug-in Setting



If you use Mozilla Firefox 3.0, you should activate the Java script by selecting [Tools |Internet Option| Security|Custom Level] from the menu. By default, this option should be activated.

Picture 1-4:Firefox Java Plug-in Setting



• Web Browser Setting-Java Script

If you use Microsoft's IE 7.0 or 8.0, you should activate the Java script by selecting [Tools |Internet Option| Security|Custom Level] from the menu. By default, this option should be activated.

• Flash Player

Some aspects of the JENNIFER client are implemented through flash files. This requires the installation of Adobe Flash Player 9.0 or higher.

1.3.0.1. Window Service Registration

1. Use the service.bat to register the JENNIFER server in the Window Service.

```
Execute cmd and move to the JENNIFER server
directory(c:\jennifer\server\bin)

Ex)
cd c:\jennifer\server\bin
service.bat install JenniferServer c:\jennifer\server\bin
C:\jennifer\server\bin>service install JenniferServer C:\Jennif
er\bin\jennifer4042\server\bin
Installing the service 'JenniferServer' ...
Using CATALINA_HOME:    C:\jennifer\server
Using CATALINA_BASE:    C:\jennifer\server
Using JAVA_HOME:        C:\Java\jdk16u13
Using JVM:              C:\Java\jdk16u13\jre\bin\server\jvm.dll
The service 'JenniferServer' has been installed.
Notice : Set the 3rd parameter to be selected by
[JENNIFER_SERVER_HOME]\bin.
```

2. Move to the **[Control pannel | Tool | Service]** menu and check that the JENNIFER server is registered in Window server and start JennfierServer.

2

Installing and Removing Agent for JENNIFER .NET

In this chapter, installation and removal procedure for Agent portion of JENNIFER Monitoring solution for .NET platform is covered.

2.1. JENNIFER .NET Agent Install Environment

List below indicates different operating environment that supports Agent for JENNIFER .NET version. .

Table 2-1: Operating Environment

Type	Supported
OS	Microsoft Windows Server 2003 32bit/64bit
OS	Microsoft Windows Server 2008 32bit/64bit
OS	Microsoft Windows Server 2008 R2

.NET Framework .NET Framework 2.0, 3.0, 3.5, 4.0 supported

Notice: In order to check the version of .NET Framework installed on the system, Go to ” Check installed .NET Framework version” .

2.2. Agent Installation

2.2.1. Install Environment

Information about a sample server environment is used in order to aid understanding of installation guide user. This installation guide uses value [Table 1] as bases for its installation process; user must proceed their installation using appropriate information from their own server environment.

Table 2-2: Default Configuration Environment

Item	Description
JENNIFER Agent Home Directory (Folder to copy install files)	C:\Jennifer
JENNIFER Sever IP Address	192.168.0.139

2.2.1.1. Pre-Installation

If the .NET application to be monitored is already in production, existence of .NET Framework can be assumed and JENNIFER Agent can be installed immediately, but if JENNIFER is to be installed on a new server, administrator must confirmed that .NET Framework is installed prior to JENNIFER Agent Installation.

Latest version of .NET Framework can be downloaded from following link..

```
NET Framework 3.5 SP1
http://www.microsoft.com/net/Download.aspx
```

Once installed, run automatic update to update it to its latest version.

2.2.1.2. Agent Installation

Install JENNIFER Agent for .NET in the server where application to be monitored is installed.

1. Insert the license key provided by JenniferSoft in "C:\Jennifer\agent.net\license.txt" file and save the file.

2. Run “C:\Jennifer\agent.net\install_Jennifer.bat” file with administrator privilege.

Notice: Administrator privilege by following instructions shown in this link. ” Run as administrator” .

2.2.1.3. Post Agent Installation

Once JENNIFER Agent has been installed, the next task would be different according to application which is going to be monitored.

2.2.1.4. Finish the Installation

If you run the application program after finishing the configuration, you will see the log file as following in the C:\Jennifer\agent.net\log folder.

```
Jennifer_[AppPool name]_[Agent Identification ID]_[date].log
```

For example, on 2010-March-8, AppPool name is “DefaultAppPool” and conf file's agent_pool value is “N10:9000” , then “jennifer_DefaultAppPool_N10_20100308.log” will should be generated.

In the log file, following information will be recorded.

```
20100111/203738:
-------------------------------------
Jennifer 1.0.0.b(2009-11-01) started.
agent_name=N10
jennifer.config=C:\Jennifer\agent.net\conf\app_pool.conf
agent_tcp_port=9000
udp_server_host=192.168.0.139
server_udp_runtime_port=6901
server_udp_listen_port=6902
server_udp_lwst_call_stack_port=6703
enable=True
license_filename=D:\JenniferSetup\agent.net\license.txt
logfile=D:\JenniferSetup\agent.net\log\jennifer_DefaultAppPool_N10.log
LWST enabled=True
enable_db_sql_trace=True
jennifer.current.directory=D:\JenniferSetup\agent.net
OS=Windows_NT
CLR=2.0.50727.4927
JENNIFER SysProf initialized!!
-------------------------------------
```

Confirm that address parameter for udp_server_host setting is pointing to JENNIFER Server and is “enable=True”.

2.3. Agent Removal

Remove .NET version from JENNIFER Agent from server and delete the installed files.

1. Run “C:/Jennifer/agent.net/uninstall_Jennifer.bat” file with Adminitrator privilege.

> **Notice:** For more information on administrator privilege, go to ” Run as administrator”.

2. If, the application program is running, stop it and delete the folder.

2.4. Agent Advance Setup

2.4.1. JENNIFER Server Configuration Change

In this section, we will cover how to change network port number which is used by JENNIFER Server and Agent.

2.4.1.1. Change Client Parameter

JENNIFER Server and JENNIFER Agent use HTTP protocol to communicate with one another. Default HTTP port is 7900 and port number for JENNIFER Server termination is 7999. In order to change the port number, edit the parameter in $JENNIFER_HOME/server/bin/catalina.sh(bat).

In case of UNIX or Linux, edit the parameter in catalina.sh as shown below.

```
JAVA_HOME="$JAVA_HOME"
if [ -z "${STARTUP_PORT}" ]
then
export STARTUP_PORT="7900"
fi
if [ -z "${SHUTDOWN_PORT}" ]
then
export SHUTDOWN_PORT="7999"
fi
```

In case of Window, edit the parameter in catalina.bat as shown below.

```
set JAVA_HOME=%JAVA_HOME%
if "%STARTUP_PORT%" == "" SET STARTUP_PORT=7900
if "%SHUTDOWN_PORT%" == "" SET SHUTDOWN_PORT=7999
```

Java Applet uses TCP port in order to send performance data from monitored application. The port number for TCP port is registered in server_tcp_port option and default port number is 6701.

```
server_tcp_port = 6701
```

2.4.2. Configuration for JENNIFER Agent

JENNIFER Agent sends performance data to JENNIFER Server using 3 different UDP Port. JENNIFER Agent uses JENNIFER Server's “server_udp_runtime_port” option to sent start and end time for all application transaction. the size of these data is very small and it is typically used for X-View Charting. The default port number is 6901.

```
server_udp_runtime_port = 6901
```

JENNIFER Agent uses JENNIFER Server's “server_udp_listen_port” UDP port to send server request rate, average response time, and other typical performance data every one second. The default port number for this is 6902. .

```
server_udp_listen_port = 6902
```

JENNIFER Agent uses JENNIFER Server's “server_udp_lwst_call_stack_port” UDP port to send X-View transaction profiling data every two seconds. The default port number of this is 6703.

```
server_udp_lwst_call_stack_port = 6703
```

Notice: When changing UDP port numbers as shown above, the information must be changed from JENNIFER Agent as well as JENNIFER Server.

When the JENNIFER Agent uses UDP method to send JENNIFER performance data to JENNIFER server, it may be necessary to assign a binding IP addresss. This is the value for IPEndPoint's first parameter of TCP Socket Programing [Socket instance].Bind(new IPEndPoint(address, port)). IF the hardware has more than two network card, use this option to select which card to accept performance data. If udp_server_host value as set as “0.0.0.0” value received from all network card is accepted.

2.4.2.1. Agent TCP Connection Option

TCP Connection from JENNIFER Server to JENNIFER Agent has TIME OUT option. Use the below setting to configure is default(ms). .

```
agent_tcp_io_timeout=5000
agent_tcp_connect_timeout=3000
```

Notice: If there is a network problem which causes TCP connection to time out, error logs will be generated in JENNIFER Server. If this is the case, I recommend tuning the network connection before increasing TCP connection time out setting.

2.4.2.2. Firewall configuration for JENNIFER Agent

If there exists Firewall between JENNIFER Server and JENNIFER Agent or JENNIFER Client, firewall setting must be configured so that the port numbers are not blocked in firewall.

1. From User's Computer to JENNIFER server, HTTP port 7900 and TCP port 6701 must be accessible.
2. From JENNIFER Agent to JENNIFER Server, UDP port 6901, 6902, 6703 must be accessible.
3. From JENNIFER Server to JENNIFER Agents, TCP port 7750 must be accessible.

Notice: UDP For changes to UDP network and firewall, refer to JENNIFER User's Guide.

2.5. Troubleshooting

In this section, we discuss common problems encountered during JENNIFER Agent installation and how to resolve them.

2.5.1. License Error

Q: When I entered the license key received, I get "Trial license key is not available" error.

A:Common cause of this error is that the start date of the license has not yet come to past. Also, if the clock in the server where JENNIFER Agent is different JENNIFER Server, you may also encounter this error.

2.5.2. Dashboard Error

Q: I am getting "Recent DB Connection count" is zero(0) in the .NET 4.0 web site.

A: In .NET 4.0, youcan see the exact performance count when the following configuration should be included in the web.config file.

```
<system.diagnostics>
   <switches>
     <add name="ConnectionPoolPerformanceCounterDetail" value="4"/>
   </switches>
 </system.diagnostics>
```

Q: I am using Windows 2008 R2 and Oracle DB, and I am getting “Recent DB Connection count” is zero(0).

A: In Oracle ODAC(Oracle Data Access Component), DB Connection count may not be provided. from Window, “Administrative Tool”, run “Performance Counter” and check the Oracle connection information.

Q: DB connection is shown as 0.?

A: In the DB Connection Setting, if value “Pooling=false”, DB Connection count is not provided to JENNIFER.

3

Using the ACT(Agent Configuration Tool) Application Program

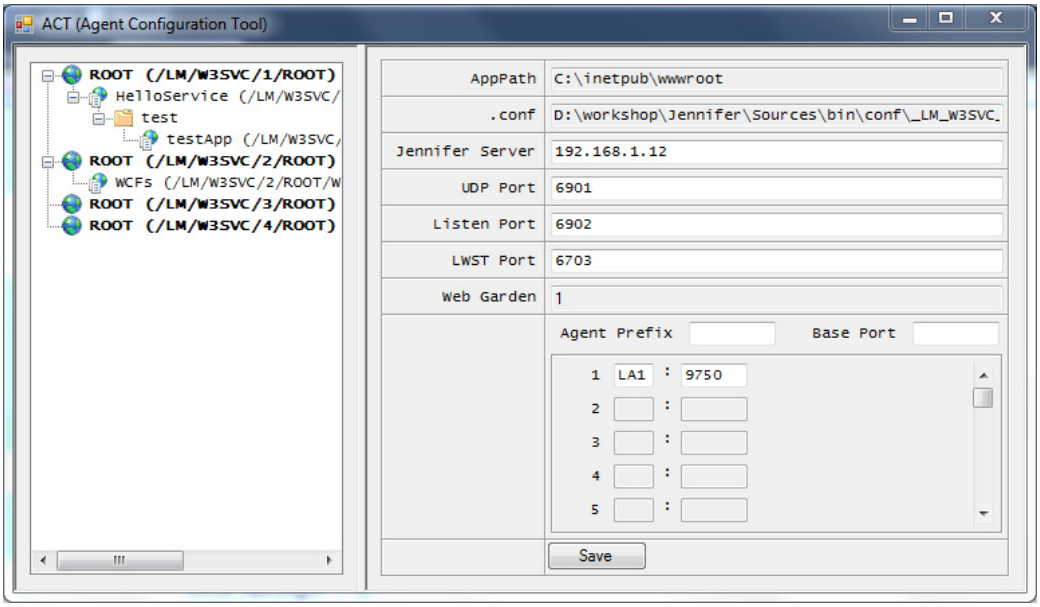
“jennifer.version” Since the “4.5.0(2010-10-29.1)” version, “Jennifer.version: has been the newly supported composition method and the disadvantages and the side effects of the existing installation method requiring the changes in web.config were successfully removed. The existing installation method is still supported but not recommended for use.

Notice: Depending on the version of the .NET Framework installed, additional exe files are provided. (.NET 2.0: IISConfigHelper.Clr20.exe, .NET 4.0: IISConfigHelper.Clr40.exe)

3.1. Web Application Composition

3.1.1. Execution

Execute the [Jennifer Agent Installation Folder]\utility\IISConfigHelper.Clr[xx].exe file.



The following screen illustrates four web sites organized and a sample of web application composition such as HelloService, testApp, and WCFs in the below.

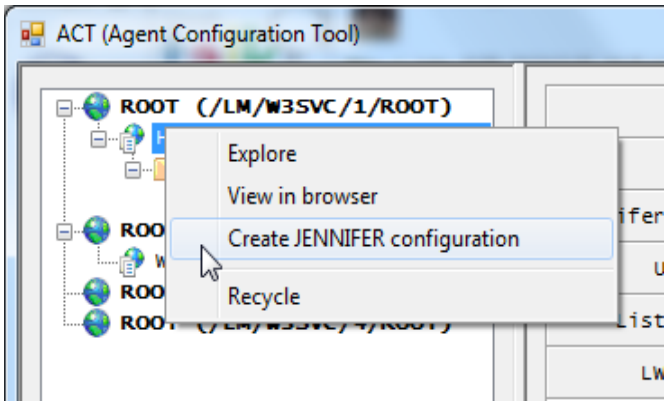
3.1.2. Composition

The IISConfigHelper application program can designate or cancel a web application that needs to be monitored and recycle the web sites.

3.1.2.1. Setting the Web Application Program that You Want to Monitor

1. Create the conf file.

Select the web application program that you want to monitor from the tree and then right click with your mouse button to launch the menu. In the menu, execute “Create JENNIFER configuration”.



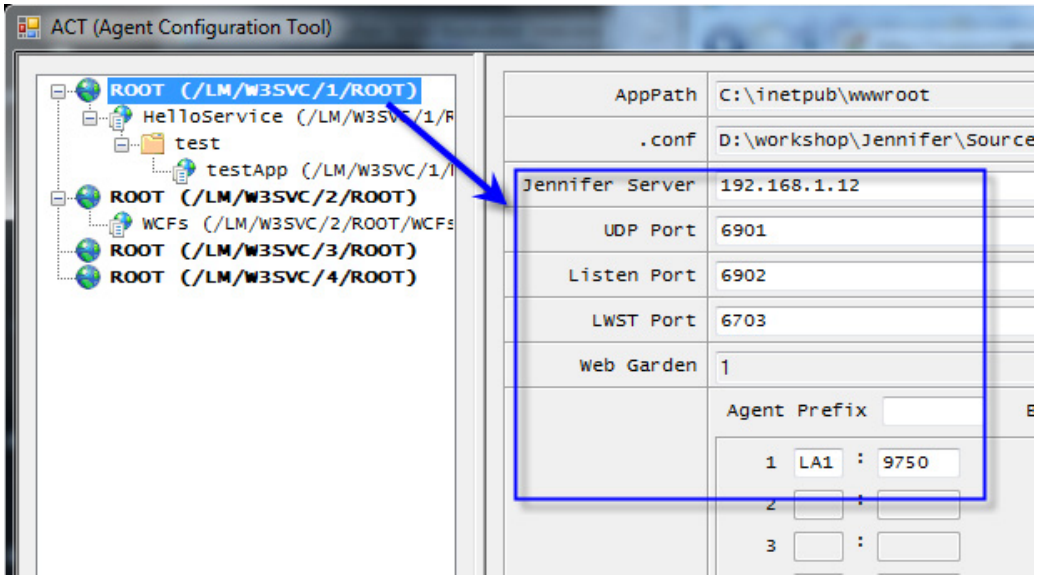
After execution, the .conf file that can perform Jennifer monitoring on the target application program will be created in the ” [Jennifer Agent Installation Folder]\conf” folder.

Notice: The newly created conf file is a copy of the app_pool.conf file that is provided in the basic package when you install Jennifer, so you must have this app_pool.conf file.

2. conf Setting

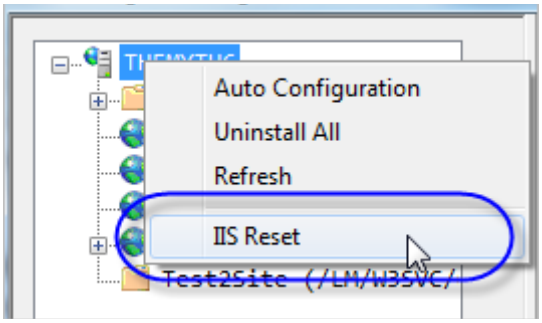
In the newly created conf file, you must configure “udp_server_host”, “server_udp_runtime_port”, “server_udp_listen_port”, “server_udp_lwst_call_stack_port” & “Agent ID” that represents the web application program.

For this configuration, you can select the right panel in the IISConfigHelper tool and save the changes you made.



3. Rexecute w3wp.exe.

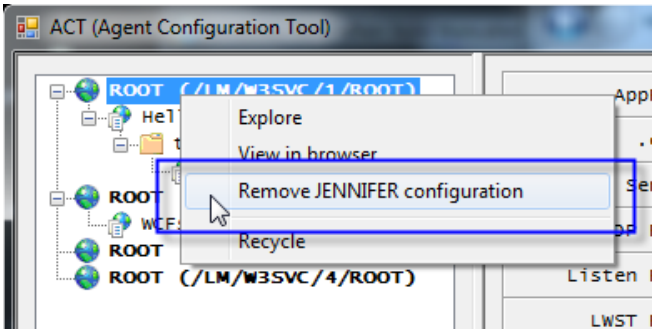
Select the web application program from the tree so that the configuration can be reflected in the system and right click with your mouse button to launch the menu and execute “IIS Reset” .



3.1.3. Cancel Monitoring

1. Create the conf File.

Select the web application program that you want to exclude from monitoring in the tree and right click with your mouse button and execute “Remove JENNIFER configuration” .



2. Rexecute w3wp.exe.]

Select the web application program from the tree so that the configuration can be reflected in the system and right click with your mouse button to launch the menu and execute “Recycle” .

4

JENNIFER.NET Agent Installation and Removal: COM+

In this chapter, we will describe the method for installing the .NET version of Jennifer Agent in a ‘server activation’ type of COM+ package.

4.1. Agent Installation

4.1.1. Installation Environment

The following table shows the environment information of the server where the Jennifer agent will be installed in order to help understanding the Jennifer agent installation procedures. This installation guideline is explained based on the environment values in [Table 1] and in actual works, you should proceed with installation works in compliance with the user server environment information.

Table 4-1: Default Composition Environment

Classification	Description
JENNIFER Agent Home Directory(folder copied from the installation disk)	C:\Jennifer
JENNIFER Server IP Address	192.168.0.139

4.1.2. Agent Installation

Install the .NET version of Jennifer Agent in the server where the COM+ components are installed.

1. Enter the license key provided by JenniferSoft in the "C:\Jennifer\agent.net\license.txt" file and save the file.
2. Execute the C:\Jennifer\agent.net\install_Jennifer.bat file with an administrator's privilege.

Notice: For more details on how to execute with an administrator's privilege on each OS, refer to the content in " execute with an administrator's privilege" .

3. Create application.config, application.manifest.

For one COM+ component, you need to create one arbitrary folder. For instance, if you want to monitor two COM+ components called "MyComponentServer" and "SecondEntLibServerApp" , then you need to create two folders as shown in the following example.

```
d:\manifest\mycomapp
d:\manifest\secondapp
```

In both of the newly created folders, you need to copy the "application.config" , "application.manifest" files from the " [Jennifer Home Folder]\agent.net\sample\complus" folder. Among these two copied files, the "application.manifest" file can be used without modifications, but for the

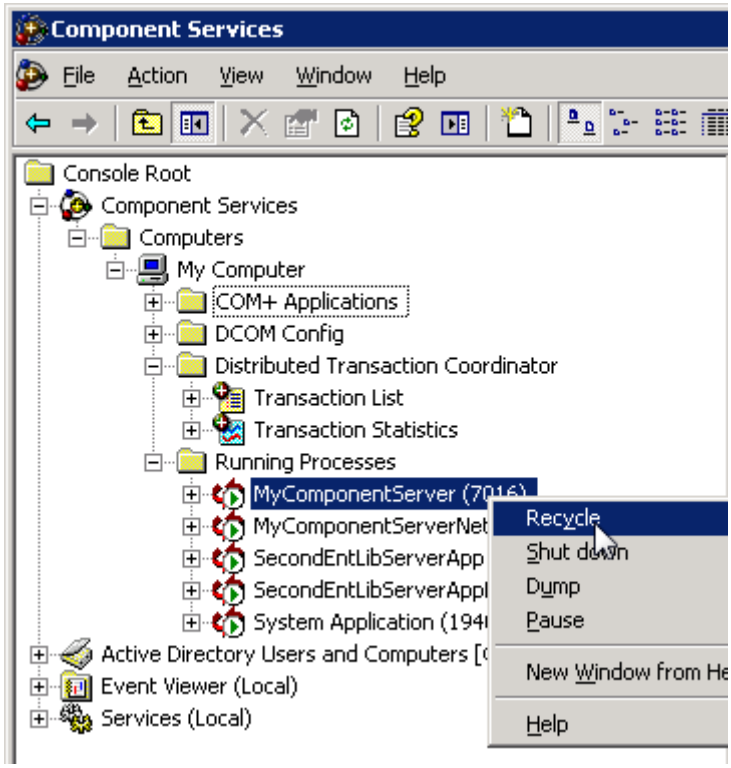
“application.config” file, you need to set the Jennifer.FileName value to the config file name created in [Step 1] just like the way you edited the web.config file.

```
<?xml version="1.0" encoding="UTF-8"?>


<configuration>
  <appSettings>
    <add key="Jennifer.FileName" value="default_pool.conf" />
  </appSettings>
</configuration>
```

4. Rexecute the Components.

As shown below, in the COM+ component service manager, you need to recycle each individual process so that the Jennifer .NET setting can be reflected in the dilhost.exe files that are currently running.



Notice: The “System Application” components are the service registered by the system, so you do not need to execute the “Recycle” command on them.

4.2. Confirm Installation

Access the application registered in COM+ and log and entry will be generated in the log file located at C:\Jennifer\agent.net\log. Confirm that log is being generated.

```
Jennifer_[Agent Unique ID]_[Date].log
```

For example, on 2010-March-8, AppPool name is “DefaultAppPool” and conf file's agent_pool value is “N10:9000”, then “jennifer_DefaultAppPool_N10_20100308.log” will should be generated. In the log file, following information will be recorded.

```
20100111/203738:
---------------------------------------
Jennifer 1.0.0.b(2009-11-01) started.
agent_name=N10
jennifer.config=C:\Jennifer\agent.net\conf\default_pool.conf
agent_tcp_port=9000
udp_server_host=192.168.0.139
server_udp_runtime_port=6901
server_udp_listen_port=6902
server_udp_lwst_call_stack_port=6703
enable=True
license_filename=D:\JenniferSetup\agent.net\license.txt
logfile=D:\JenniferSetup\agent.net\log\jennifer_N10_20100308.log
LWST enabled=True
enable_db_sql_trace=True
jennifer.current.directory=D:\JenniferSetup\agent.net
OS=Windows_NT
CLR=2.0.50727.4927
JENNIFER SysProf initialized!!
---------------------------------------
```

4.3. Agent Removal

Remove .NET version fo JENNIFER Agent from server and delete the installed files.

1. Run “C:/Jennifer/agent.net/uninstall_Jennifer.bat” with Adminitrator privilege.

Notice: For more information on administrator privilege, go to ”Run as Administrator”.”Run as Administrator”.

2. If any of the COM+ system application is currently running, use “recycle” command to the running instance in “COM+ Application Management”.

5

JENNIFER .NET Agent Composition : COM+

Since the Jennifer “4.5.0(2010-10-29.2)” version, the configuration method for monitoring the COM+ application programs have been much more simplified compared to the Previous Version. In the previous versions, we had to create ” .config” , ” .manifest” and their folders to set them in the “Application Root Directory” which was very complicated. In addition, during the new registration process of a COM+ object, the “Application Root Directory” setting was initialized mistakenly, so that we must reconfigure them each time of registration. In the new environment configuration method, all these problems were completely removed.

5.1. Agent Composition

5.1.1. Organization Environment

The following table shows the server environment where the Jennifer Agent is installed in order to help understanding the installation procedure. This installation guideline is described based on the environment values shown in [Table 1] and in the actual works, you must proceed with installation in compliance with the server environment information.

Table 5-1: Default Composition Environment

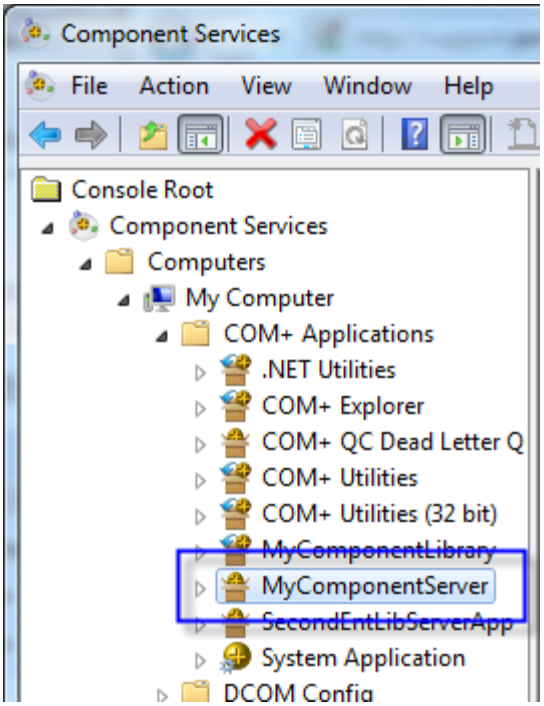
Classification	Description
Jennifer Agent home directory(folder copied from the installation disk)	C:\Jennifer
Monitoring target COM+ application program's name A	MyComponentServer

5.1.2. COM+ Application Program Name's conf File Composition

After the Jennifer Agent is installed in your system, you must create a conf file corresponding to the name of the COM+ components that you want to monitor.

1. Check the name of the COM+ application program

In the "component service)", check the name of the COM+ application program that you want to monitor. The below screen indicates that the name is "MyComponentServer".



2. Create a conf file

The following rule is applied when you create a conf file that contains the composition of monitoring that will be connected to COM+.

```
[Jennifer server installation folder]\conf\complus.[COM+ application program's name].conf
```

Therefore, the following should apply to this example.

```
C:\Jennifer\conf\complus.MyComponentServer.conf
```

Although the same composition rules(udp_server_host,…) apply to the rest of internal conf files, the number of agent_pool should be determined by the pool size configured in the application pooling of the COM+ server application program.

Notice: For more details on how to find the number of application pools in the COM+ components, please refer to " COM+ Application Pool Check."

3. Rexecute the components.

In the COM+ component service, execute the recycle menu for individual processes so that Jennifer .NET is applied to the dllhost.exe files that are being executed.

5.2. Agent Composition Removal

If you want to quit monitoring only while maintaining the same product installation, then you must delete the " [Jennifer Server Installation Folder]\conf\complus.[COM+ Application Program Name].conf" file that is connected to the application program. (ex: "C:\Jennifer\conf\complus.MyComponentServer.conf")

After that, if there is some COM+ components being serviced, then you should execute the recycle commands for each instance being executed in the "COM+ component service" to cancel the state of monitoring.

6

JENNIFER .NET Agent Installation and Unistallation:

6.1. JENNIFER .NET Agent Installation and Unistallation: Independently Executable Application Programs and NT Services

In this chapter, we will describe the methods for monitoring other application programs than IIS/COM+ with the Jennifer .Net Agents.

6.1.1. Agent Installation

6.1.1.1. Installation Environment

The following table shows the server environment where the Jennifer Agent is installed in order to help understanding the installation procedure. This installation guideline is described based on the environment values shown in [Table 1] and in the actual works, you must proceed with installation in compliance with the server environment information.

Table 6-1: Sample Environment Setting

Classification	Description
Jennifer Agent home directory(folder copied from the installation disk)	C:\Jennifer

6.1.2. Agent Installation

Install the Jennifer .NET Agent in the server where the monitoring target application program is executed.

1. Enter the license key provided by JenniferSoft in the "C:\Jennifer\agent.net\license.txt" file and save the file.
2. Execute C:\Jennifer\agent.net\install_Jennifer.bat with an administrator's privilege.

Notice: For more details on how to execute with an administrator's privilege on each OS, refer to the content in " execute with an administrator' s privilege" .

6.1.3. Post Installation

After the Jennifer Agent is installed in your system, designate .conf files in one to one relationship to the application programs which are the targets of monitoring.

1. conf File Setting

Copy the app_pool.conf file that has the basic configurations for the "C:\Jennifer\agent.net\conf" folder. In the (ex, myapp_pool.conf) copied conf file, designate the Jennifer server address explicitly.

Enter the address of your computer where the Jennifer Server is installed in udp_server_host and configure the unique ID and the port number properly in agent_pool.

2. Application Program Name].exe.config created(or changed).

For instance, if the monitoring target application program has the following path,

```
C:\Program Files\MyApp\Myapp.exe
```

Then create the following file in the same folder. (If there already exists one, just use it.),

```
C:\Program Files\MyApp\Myapp.exe.config
```

Open the config file and in the appSettings setting, add the following “Jennifer.FileName” entity and set the conf file name that you already made.

```
<?xml version="1.0" encoding="UTF-8"?>

<configuration>
  <appSettings>
    <add key="Jennifer.FileName" value="myapp_pool.conf" />
  </appSettings>
</configuration>
```

3. Environment Variable Setting

Your environment variable setting method will vary depending on whether it is a general independent executable application program or an NT service application program.

In case of an independent executable application program, using the following batch file, you should set the environment variables prior to execution.

```
REM ===== startapp.bat =====
SET COR_ENABLE_PROFILING=1
SET COR_PROFILER={FF8C2B6C-DBB5-4AED-9876-2CED6FFAF4C9}

Myapp.exe
```

To the contrary, if you want to monitor an “NT service application program,” then since you can’ t create a batch file to set the environment variables as shown above, you need to use an additional registry key to manually configure the environment variables in COR_ENABLE_PROFILING, COR_PROFILER .

The following path is applied to the registry key.

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\[Target NT
Service Name]
```

For instance, if the service name is assumed to be “RemoteTestService” , the following path should apply. (If you do not know the NT service name, then please refer to ” NT service name search method” .)

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\RemoteTestService
```

You should add the following values to the above registry key.

```
Name: Environment
Format: REG_MULTI_SZ
Value: COR_ENABLE_PROFILING=1
    COR_PROFILER={FF8C2B6C-DBB5-4AED-9876-2CED6FFAF4C9}
```

4. Restart the Application Program.

Restart the monitoring target application program so that the newly configured environment variables can be applied to it.

6.2. Check Installation

After successfully finishing compositions and executing the application program, wait until the method called from the service is executed. Then the following log file will be created in the C:\Jennifer\agent.net\log folder. (For instance, if the application program provides the WCF service, then the WCF service should be used on the client side.)

```
Jennifer_[Agent ID]_[Date].log
```

For instance, if the agent_pool value was set to N10:9000 on March 8, then the “jennifer_N10_20100308.log” file will be created.

Examining the content of the log file, we have,

```
20100111/203738:
-------------------------------------
Jennifer 1.0.0.b(2009-11-01) started.
agent_name=N10
jennifer.config=C:\Jennifer\agent.net\conf\default_pool.conf
agent_tcp_port=9000
udp_server_host=192.168.0.139
server_udp_runtime_port=6901
server_udp_listen_port=6902
server_udp_lwst_call_stack_port=6703
enable=True
license_filename=D:\JenniferSetup\agent.net\license.txt
logfile=D:\JenniferSetup\agent.net\log\jennifer_N10_20100308.log
LWST enabled=True
enable_db_sql_trace=True
jennifer.current.directory=D:\JenniferSetup\agent.net
OS=Windows_NT
CLR=2.0.50727.4927
JENNIFER SysProf initialized!!
-------------------------------------
```

6.3. Agent Removal

Remove the Jennifer .NET Agent from the server and delete the file.

1. Execute the “C:/Jennifer/agent.net/uninstall_Jennifer.bat” file with an administrator’ s privilege.

Notice: For more information on administrator privilege, go to ” Run as Administrator”. ” Run as Administrator” .

2. If there is a process running, then restart it.

7

.NET Batch Process Monitoring

This chapter will describe the method for monitoring a batch job by using the Jennifer Agent(.Net version).

Notice: For the concepts of batch monitoring implemented in Jennifer and installation in Java, please refer to the document, ” Batch JOB Monitoring” .

7.1. MasterAgent Installation and Execution

1. conf file configuration

Copy the ” [agent installation folder]\conf\app_pool.conf” file and create the “batchjob_master.conf” file. Since you need to make a connection to the Jennifer Server, open the batchjob_master.conf file in your memo pad and set the Jennifer Server address.

```
udp_server_host = [Jennifer Server Address]
```

2. Master Agent Execution

The master agent execution file is installed altogether with JENNIFER, so no additional installation work is necessary. Simply, execute the module provided in the below path.

```
.NET 1.1: [Agent Installation Folder]\bin\MasterAgent.Clr10.exe
.NET 2.0 or higher: [Agent Installation Folder]\bin\MasterAgent.exe
```

7.2. SubAgent Installation and Execution

A sub agent is the target of monitoring. It is the batch process(exe) where the Jennifer Agent is activated.

1. conf file setting

Copy the config file for the master agent where your sub agent will be subordinated to and create the new conf file. To imply that it is a sub agent, add the following.

```
[Ex: youragent.conf]


SUB_AGENT = true
```

2. Connecting the config file to a batch process

Assuming that the following execution file path is used for the batch process to be monitored,

```
C:\BatchJobs\DailyWorker.exe
```

Connect the ” .config” file to the process name and create a new file in the conf folder.

```
[Agent Installation Folder]\conf\DailyWorker.exe.config
```

The file content is shown below. Designate the conf(ex. Youragent.conf) that was made for the sub agent previously.

```
<?xml version="1.0" encoding="UTF-8"?>
<configuration>
    <appSettings>
        <add key="Jennifer.FileName" value="youragent.conf" />
    </appSettings>
</configuration>
```

3. User Defined Method Profiling Configuration

In general, in a batch process, the user defined method is regarded as the unit of transactions. So, according to the method indicated in ” .Net method profiling” , you need to make an additional ” [txserver]” section.

In addition, in case of a batch process, you must define the ” [batchjob]” section and define the main function for the application program.

For instance, if the user code is defined as follows,

```
namespace BatchJob1
{
    class Program
    {
        static void Main(string[] args)
        {
        }

        void DoBatchJob()
        {
        }
    }
}
```

If DoBatchJob is the method executing the actual batch job and it is the target of monitoring by txserver, then in the profiler.ini file, you must set the configuration as follows.

```
[batchjob.process name]
BatchJob1.Program.Main=1

[txserver.process name]
BatchJob1.Program.DoBatchJob=1
```

4. Jennifer Agent Environment Variable Setting

For profiling of the batch process, you must set the following environment variables prior to execution of the process.

```
SET COR_ENABLE_PROFILING=1
SET COR_PROFILER={FF8C2B6C-DBB5-4AED-9876-2CED6FFAF4C9}
```

An environment variable can be set in various methods. In case of a .bat file or NT service, you can use the registry as described in ” Environment Variable Setting” .

But if the two methods can’t help you, you can then use a global environment variable. If it is registered as a system variable, all the .net processes executed in the OS are involved with Jennifer Agent profiling works. To resolve this problem, Jennifer .NET recognizes an additional environment variable option called “JENNIFER_PROFILE”. Therefore, if the user’ s batch process name is “DailyWorker.exe” , then you can register the following environment variables.

Environment variable name: JENNIFER_PROFILE

Value: dailyworker.exe

Notice: The process name should be lower case letters.

If there are multiple batch processes, you can use a semi colon as a separator to designate it as follows.

```
Environment variable name: JENNIFER_PROFILE
Value: dailyworker.exe;weeklyworker.exe;monthlyworker.exe
```

After setting, execute the batch process to start normal monitoring.

7.3. Agent Monitoring Cancel / Removal

Once the Jennifer Agent is installed, if you want to cancel batch process monitoring only, then you must set , in your environment variable, “COR_ENABLE_PROFILING” to “0”. (Or if it is set to “JENNIFER_PROFILE”, omit the process name or delete the JENNIFER_PROFILE environment variable itself.)

If you want to remove the Jennifer Agent as well, then terminate all the batch processes being monitoring and execute the “[agent installation folder]/uninstall_Jennifer.bat” with the privilege of an administrator and manually delete the installation folder.

8

JENNIFER .NET Agent: How to Monitor Oracle DB Calls

There are the following two ways to connect to the Oracle DB when using a MS .NET application program.

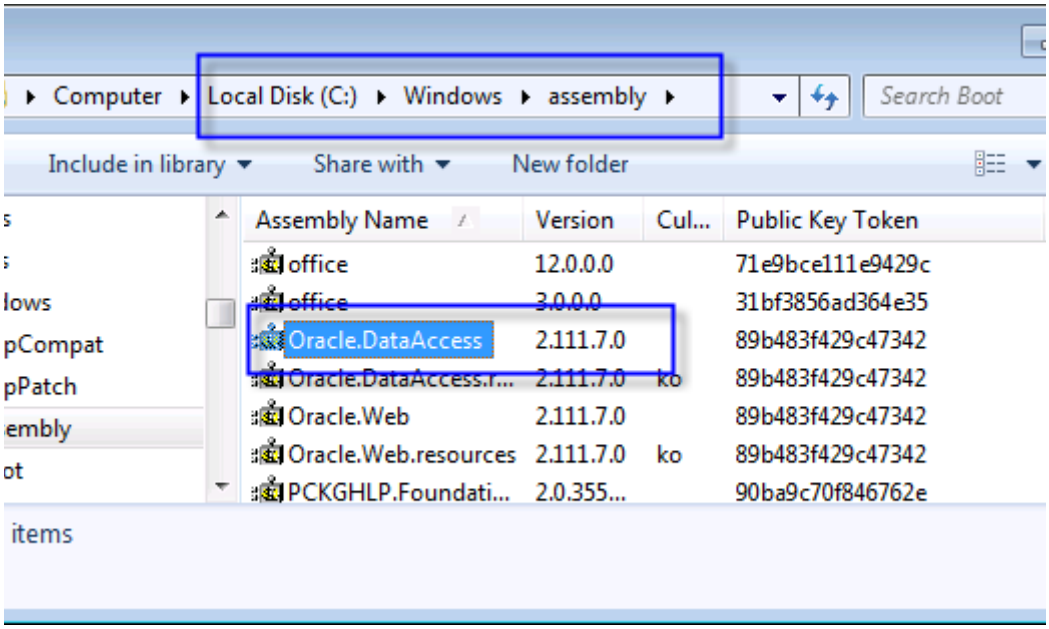
- System.Data.OracleClient (provided by MS)
- Oracle.DataAccess.Client (provided by Oracle)

JENNIFER .NET can properly monitor an application program that uses “System.Data.OracleClient”, but if it uses “Oracle.DataAccess.Client” then only some versions are supported. Thus it may required additional configurations. (For your information, MS displays the System.Data.OracleClient functions as [depreciated] since .NET 4.0, and recommends you to use the DB Prover that is supplied by partner companies. – For instance, the abovementioned Oracle.DataAccess.Client).

Therefore, while the application program is using “Oracle.DataAccess.Client”, if you can’t monitor it after installing the Jennifer .NET, then you can use the following method to resolve the problem.

1. Check the Oracle.DataAccess.Client version being used]

To check the version of ODP.NET(Oracle Data Provider for .NET) that the application program is using, then look at the Oracle.DataAccess item in GAC(Global Assembly Cache). The following version number will be indicated.



2. Explicitly show the version number in the Profiler.odpnet.conf file.

To clearly show the Oracle version number, there is the Profiler.odpnet.conf file in the [Jennifer .NET Installation Folder]\agent.net folder. The following content will be shown in this file if you open it with a wordpad.

```
2.111.7.20
2.111.6.0
2.102.3.2
2.102.2.20
# 10.2.0.100
```

Using GAC, add the confirmed version number, “2.111.7.0” as shown in the following.

```
2.111.7.20
2.111.6.0
2.102.3.2
2.102.2.20
# 10.2.0.100
2.111.7.0
```

Save the changes and restart the monitoring target application program. You can see that monitoring will take place successfully.

9

Reference : .NET Framework

This chapter describes the environment setting that requires additional explanations as the Jennifer .NET Agent is supported.

9.1. Window Environment

9.1.1. Execution with an Administrator’ s Privilege

When running a program, the methods for obtaining an administrator’ s privilege and executing it vary depending on whether it is Windows Server 2003 or Windows Server 2008. Separate explanations will be provided for each case.

9.1.1.1. Windows Server 2003

The following two methods are provided.

- Log into a system with an administrator’ s privilege and execute the following program.

- In the file browser, right click with your mouse on the program and select “Run as …” in the menu. Enter an administrator’ s account and execute it. For instance, in case of a batch file(.bat), since the “Run as…” menu is not available, you need to execute “Start” / “All Programs” / “Sub Programs” / “Command Prompt” with an administrator’ s account and execute a batch file in the command line.

9.1.1.2. Windows Server 2008, Windows Server 2008 R2

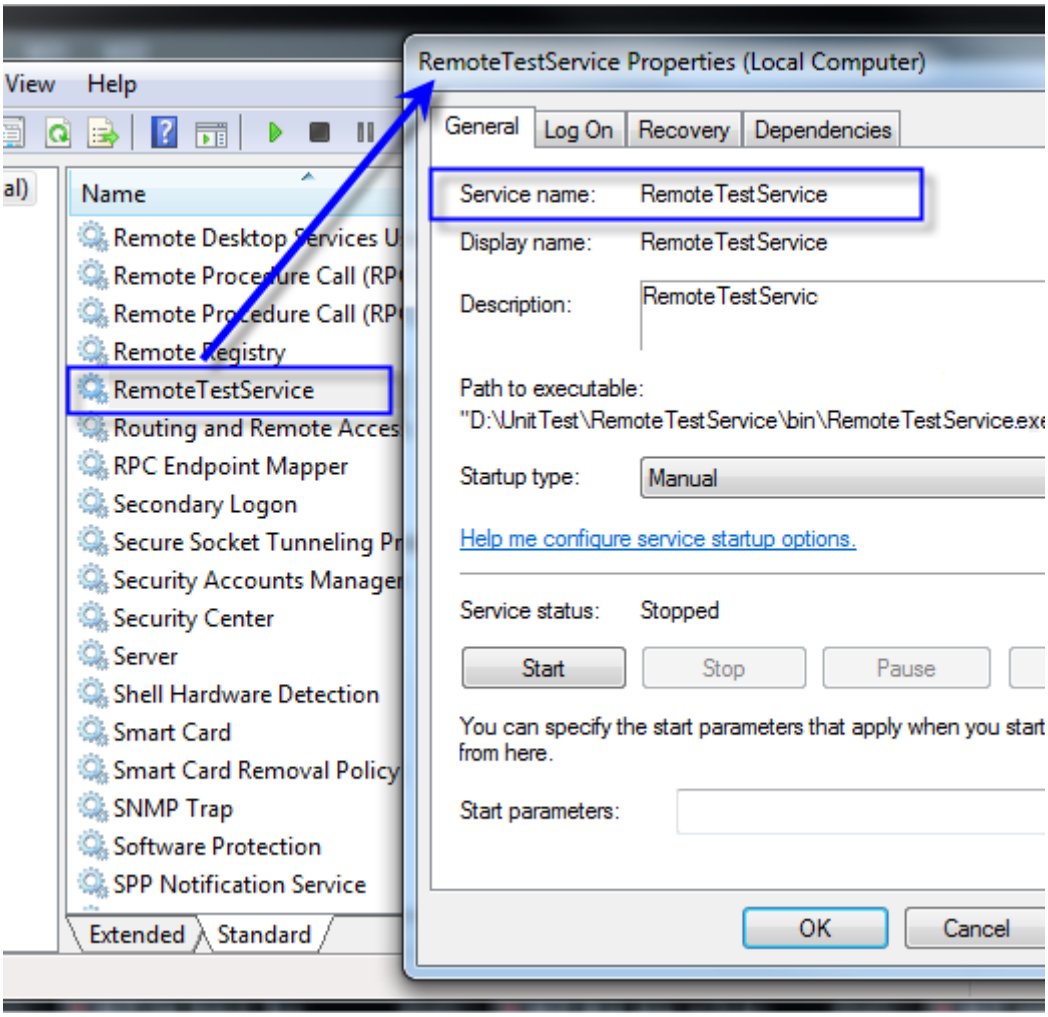
Since Windows Server 2008, the “UAC(User Access Control)” function is applied to an account other than an “administrator” account, you need to choose a proper method depending on the situation.

- If you are logged in with an administrator account: All the executed application programs are basically executed with an administrator’ s privilege.
- If you are logged in with an account belonging to the “administrator’ s group” : Right click with your mouse button on the application program that you want to run in the file browser and then select the ‘execute with an administrator’ s privilege’ menu. The “User Access Control” window will appear. You can select the OK button to execute it with an administrator’ s privilege.
- If you are logged in with a general user account other than the “administrator group” : Right click with your mouse button on the application program that you want to run in the file browser and then select the ‘execute with an administrator’ s privilege’ menu. Now enter the user account information that belongs to the administrator group.

9.2. How to Find an NT Service Name

Execute the “Start” / “Administrative Tools” / “Service” menu and open the property window for the target NT service. For instance, the following example shows

how to check the service name in the property window for the sample NT service, “RemoteTestService” .



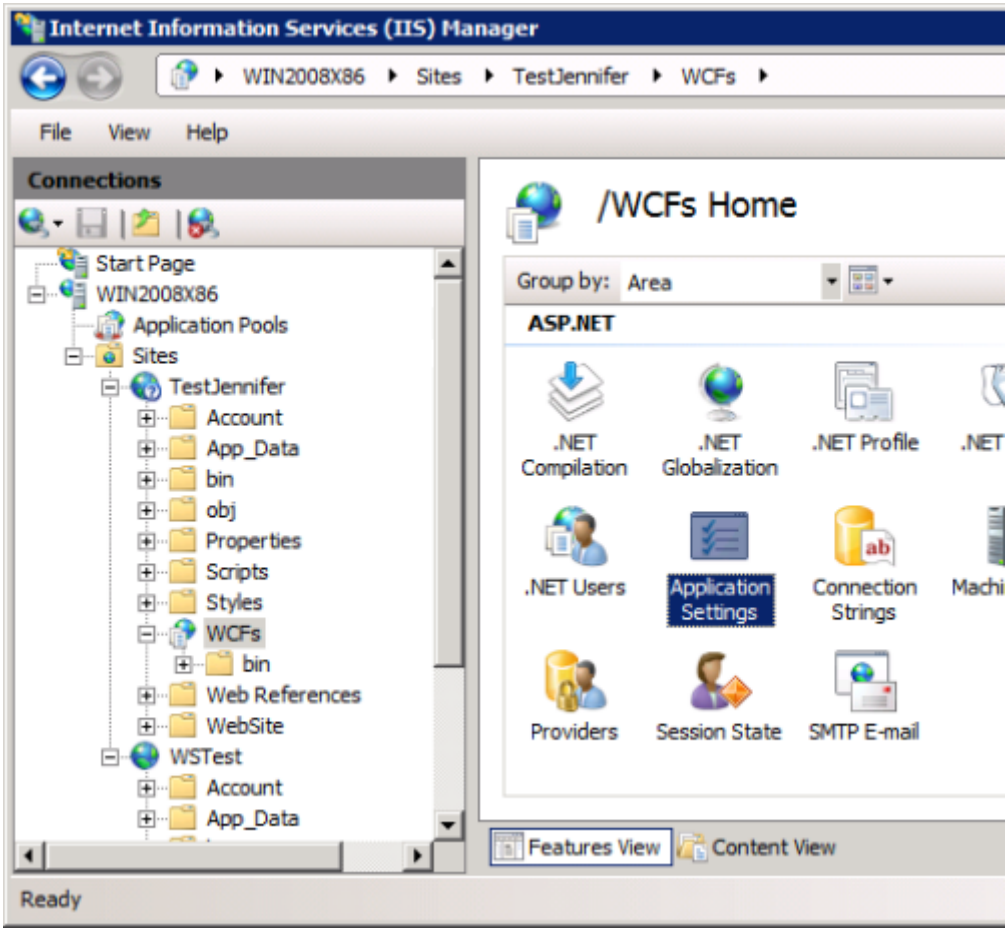
In this sample, you can easily see that the NT service name is “RemoteTestService” .

9.2.1. IIS Environment

9.2.1.1. How to Designate a conf file to web.config in Windws

Server 2008

Run the IIS 7(IIS 7.5 for R2) manager, and select a virtual application program or web site that you want to monitor. In the “Features View” menu, double click on the “Application Settings” icon.



In the right side “Actions” menu, press the “Add…” button and then enter the following value.

```
Name: Jennifer.FileName
Value: [conf file name]
```

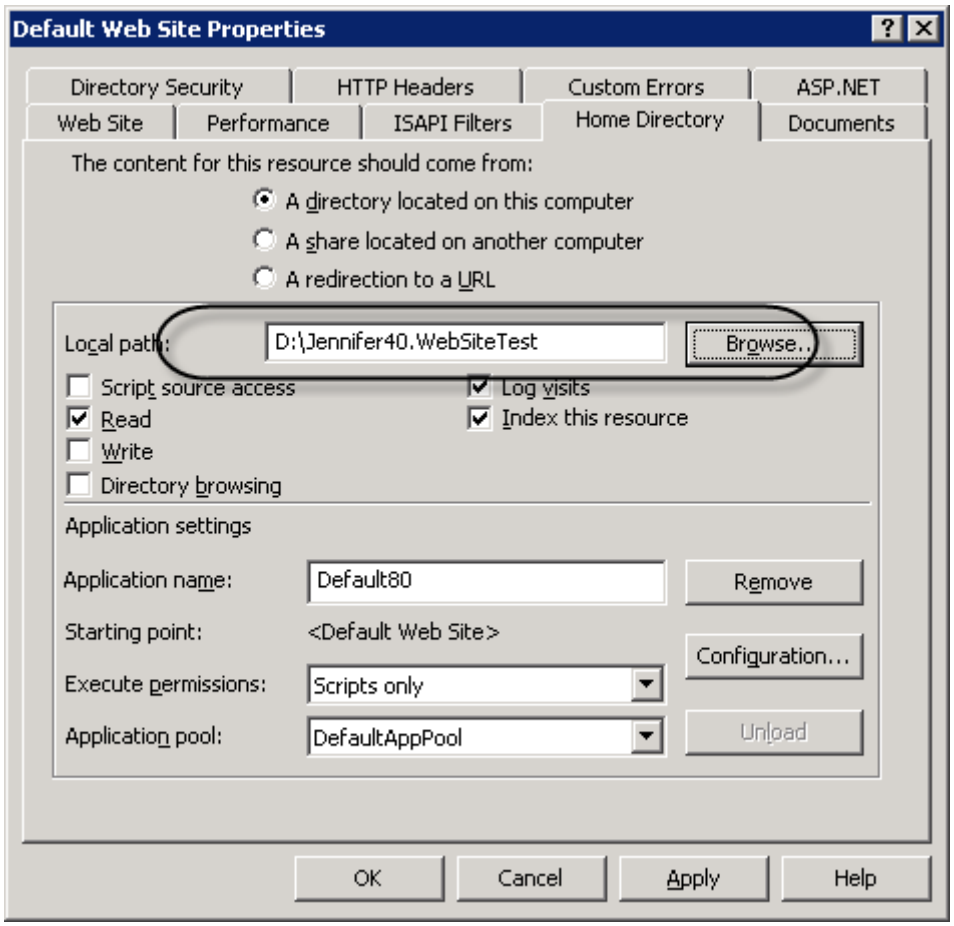
If the configured conf file name is app_pool.conf, then you can enter it as follows.

```
Name: Jennifer.FileName
Value: app_pool.conf
```

9.3. How to Check the Location of web.config in the Application Program

9.3.1. Checking in IIS 6

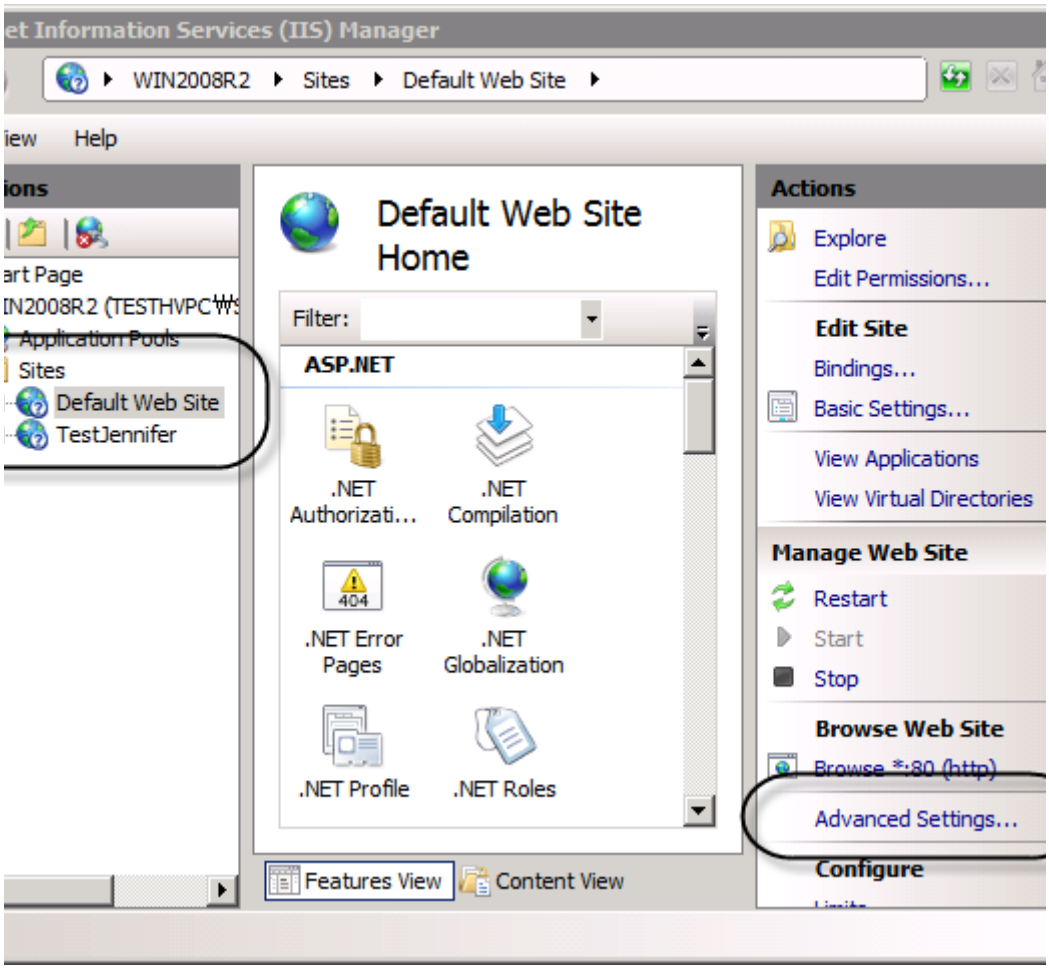
Run the IIS manager and launch the property window for a virtual application program or a website that you want to monitor. Select the “Home Directory” tab.



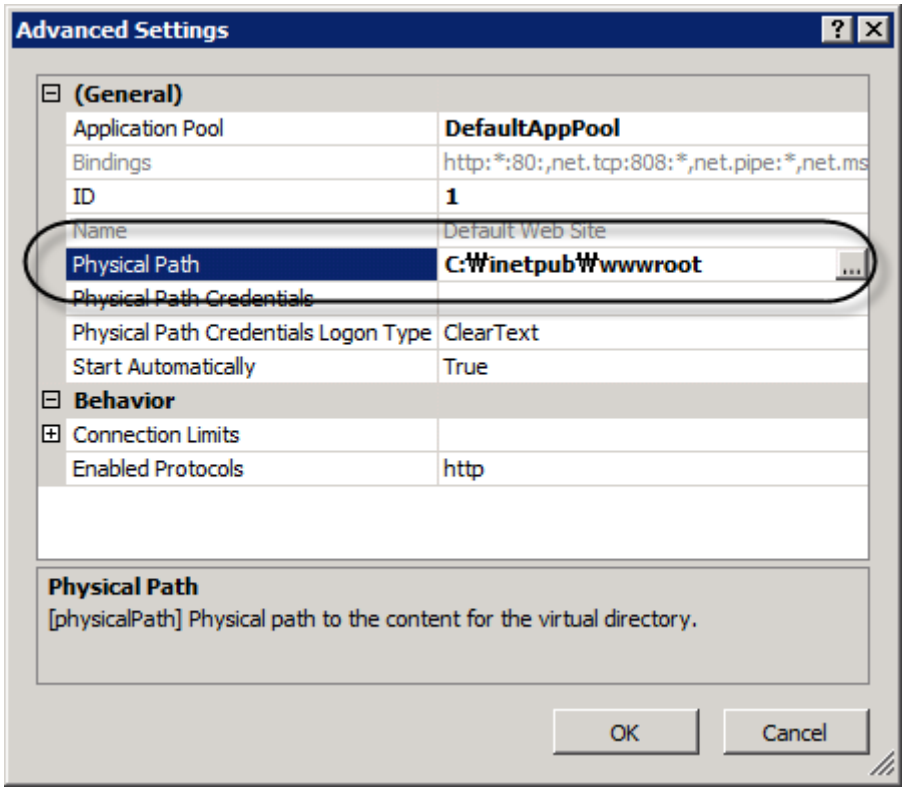
In the above example, you can see that the installation folder for the application program is set to “D:\Jennifer40.WebSiteTest” which has the web.config file.

9.3.2. Checking in IIS 7 or Higher

Run the IIS manager and launch the property window for a virtual application program or a website that you want to monitor. Then press the “Advanced Settings…” link.



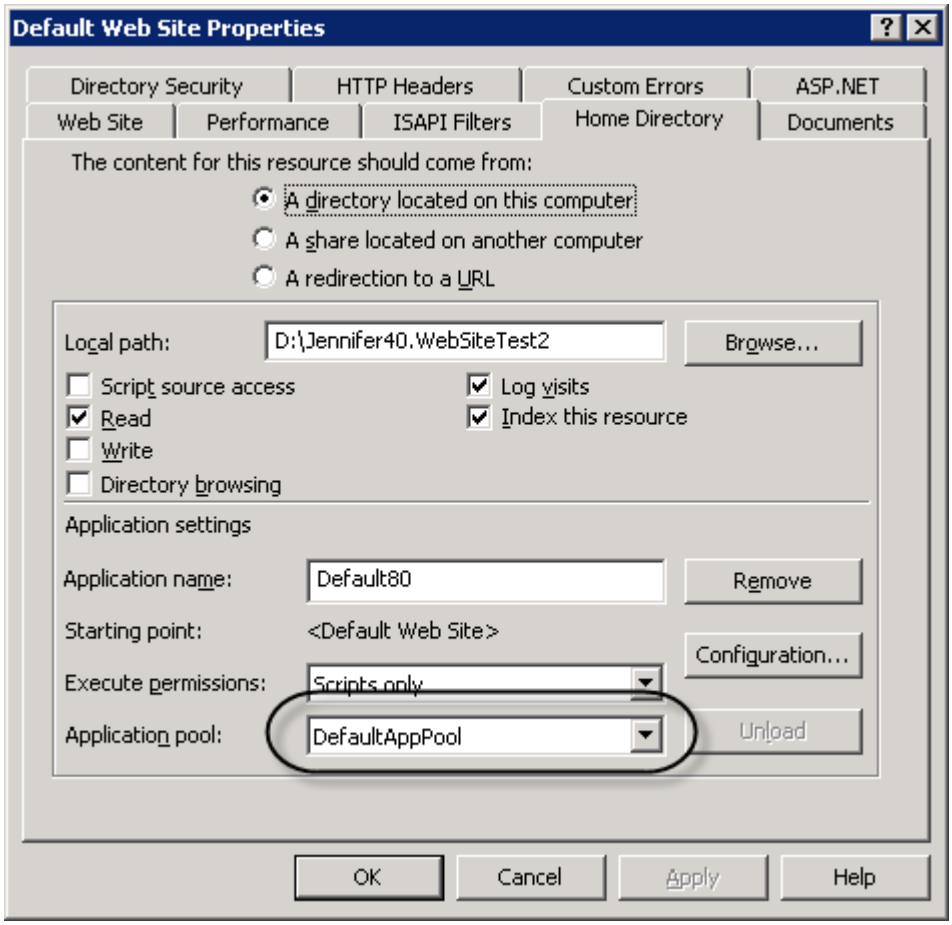
You can see in the dialogue box that the following application program is installed in the folder. In this example, you can see that the web.config file is located in the “C:\inetpub\wwwroot” folder.



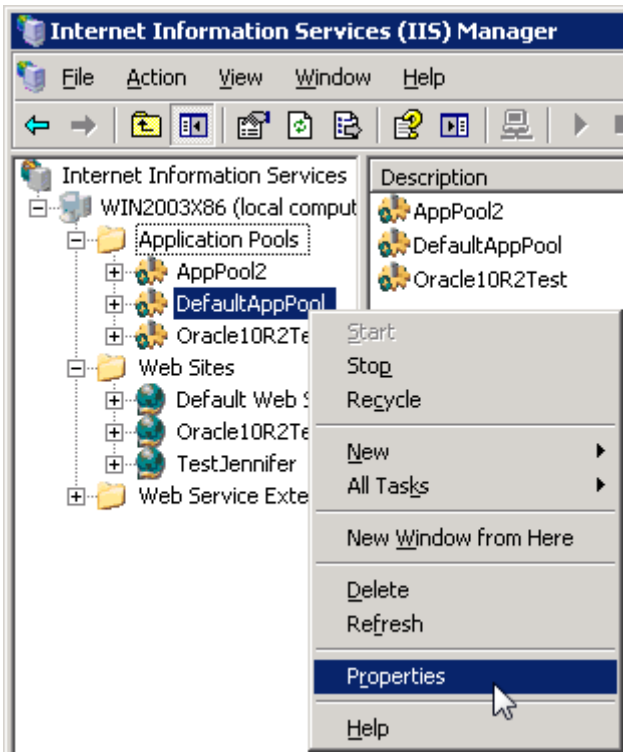
9.4. How to Check the Number of Web Gardens

9.4.1. Checking in IIS 6

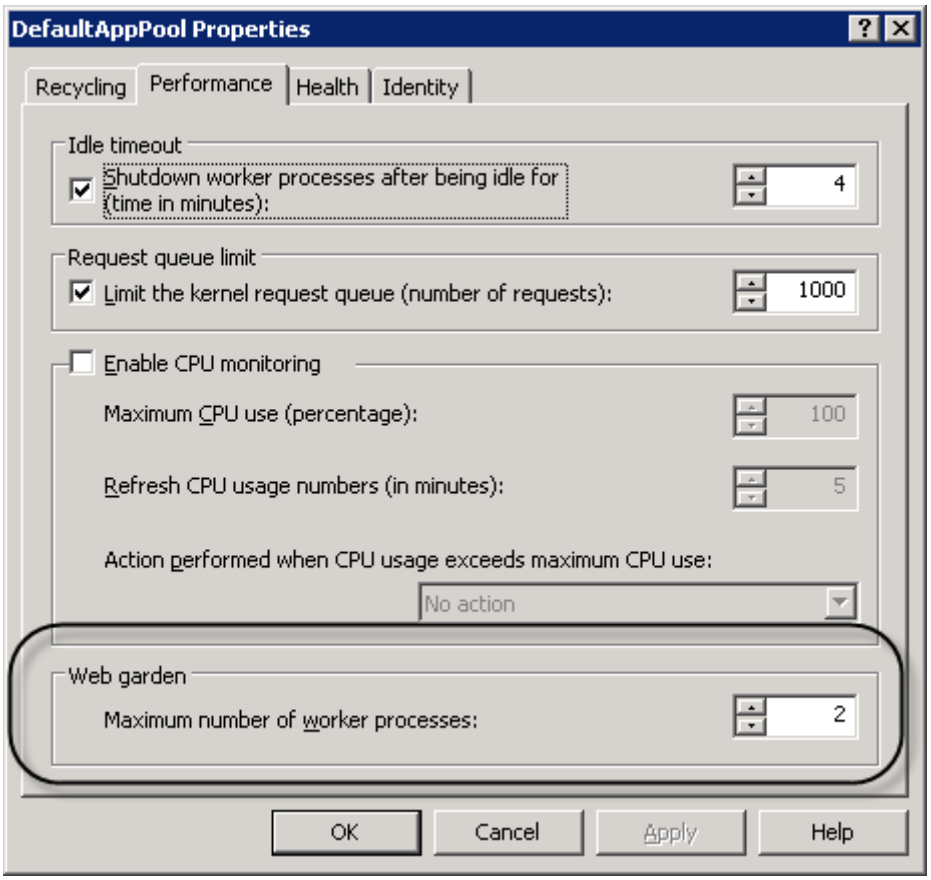
Run the IIS manager and launch the property window for a virtual application program or a website that you want to monitor. Select the “Home Directory” tab.



In this example, you can see that the application program’ s “Application Pool” is set to “DefaultAppPool”. Now in the IIS manager’ s “Application Pools” node, select a node corresponding to the name of “DefaultAppPool” and launch a property window.

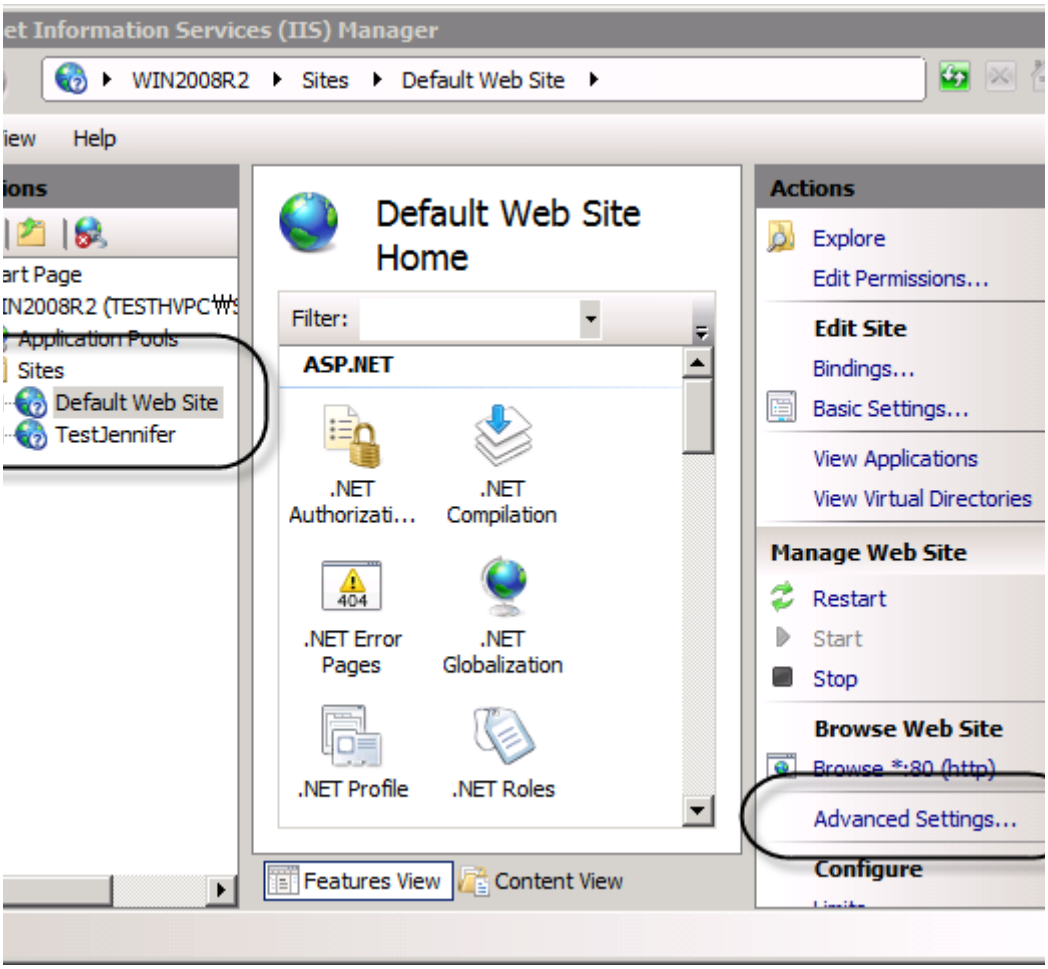


Finally, select the “Performance” tab. As shown below, you can check the number of “Web Gardens” . (default: 1)

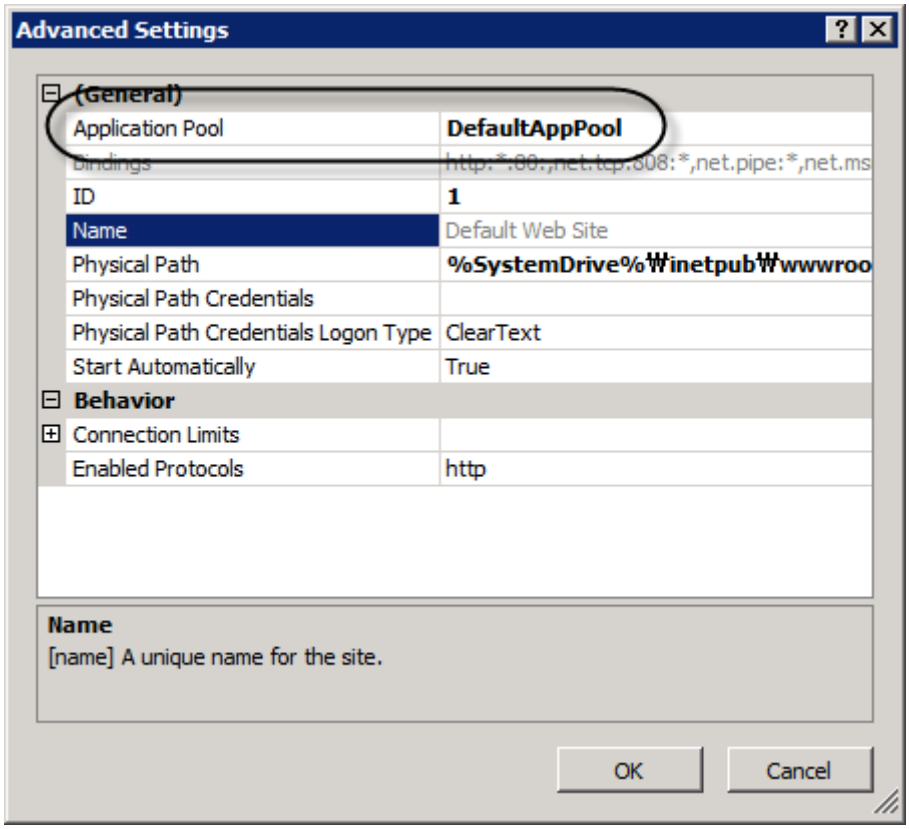


9.4.2. Checking in IIS 7 or Higher

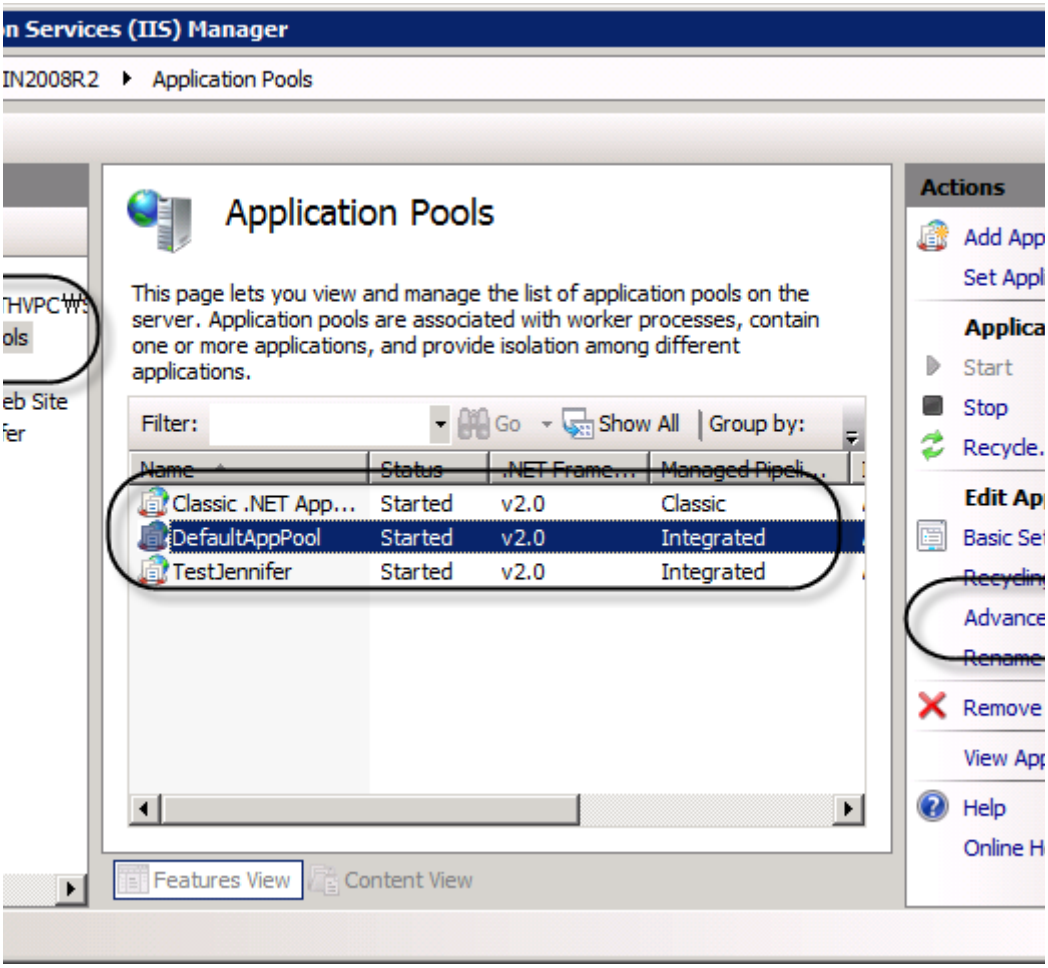
Run the IIS manager and launch the property window for a virtual application program or a website that you want to monitor. Then press the “Advanced Settings…” link.



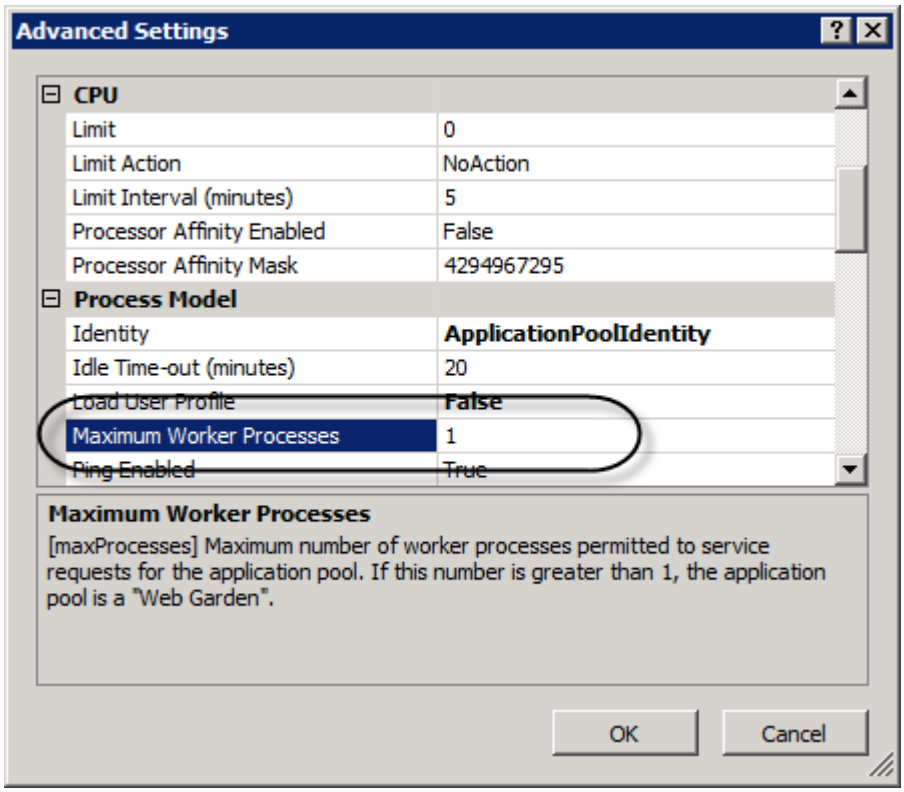
As shown below, in the dialogue box, you can check the “Application Pool” where the application program is activated.



In the IIS manger’ s “Application Pools” node, select the name of “DefaultAppPool” that you checked in the above example and press the “Advanced Settings…” link.

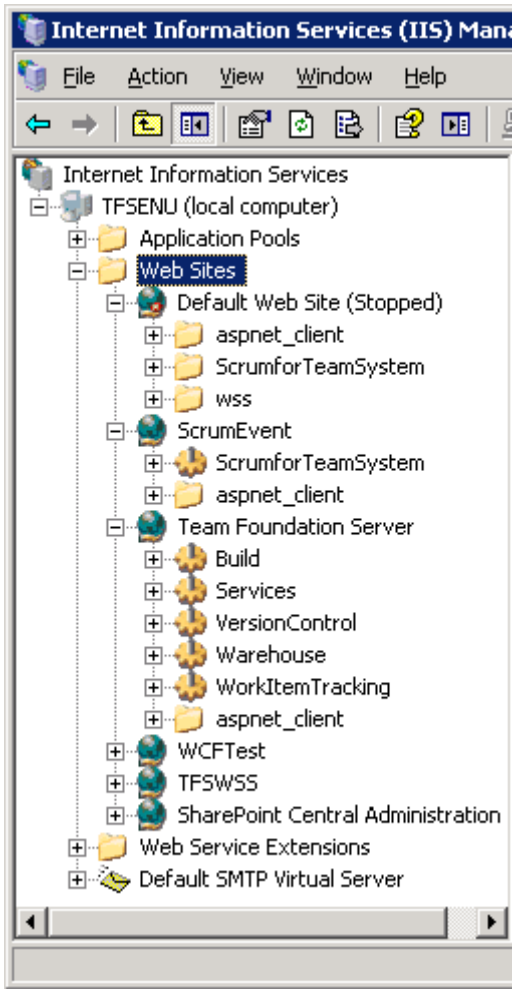


Then, as shown in the below screen, you can check the number of “Maximum Worker Processes” which is the web garden value that is designated in IIS 6.



9.5. Unit of Application Programs

In IIS, an application program is a web site and a virtual application program below it. For instance, you can use the following IIS environment as an example,



There is a total of 11 “application programs” that can be classified as follows.

Websites

- Default Web Site
- ScrumEvent
- Team Foundation Server
- WCFTest
- TFSWSS
- SharePoint Central Administration

Virtual Application Program

- ScrumforTeamSystem

- Build
- Services
- VersionControl
- Warehouse
- WorkItemTracking

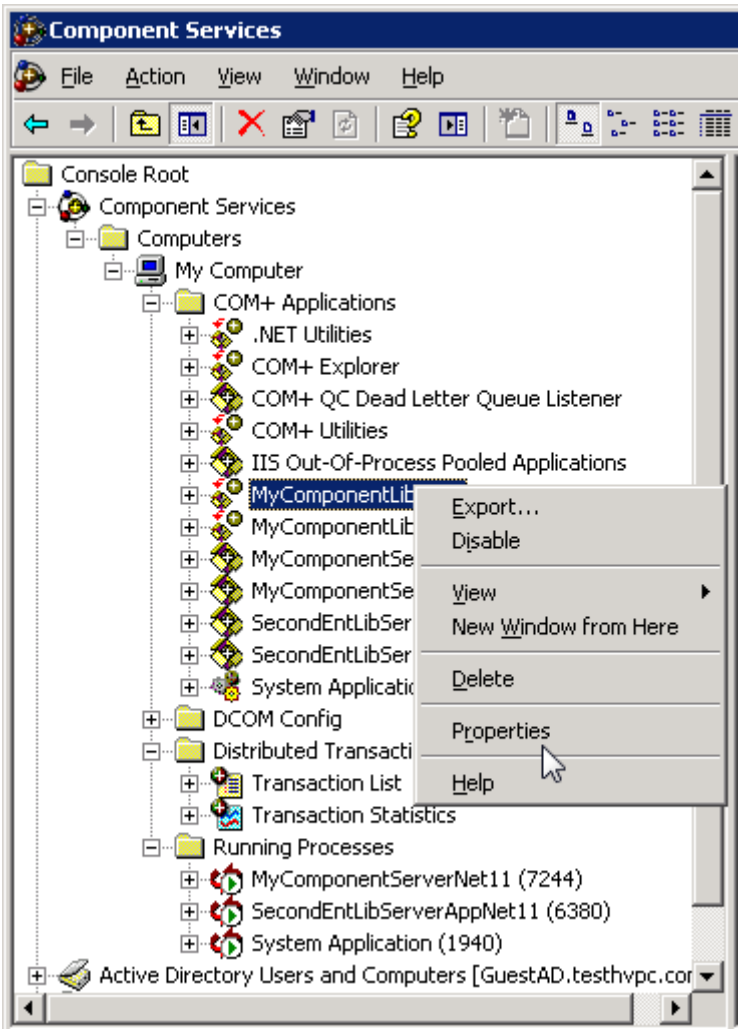
9.6. COM+ Component Service

9.6.1. Checking the COM+ Application Pool

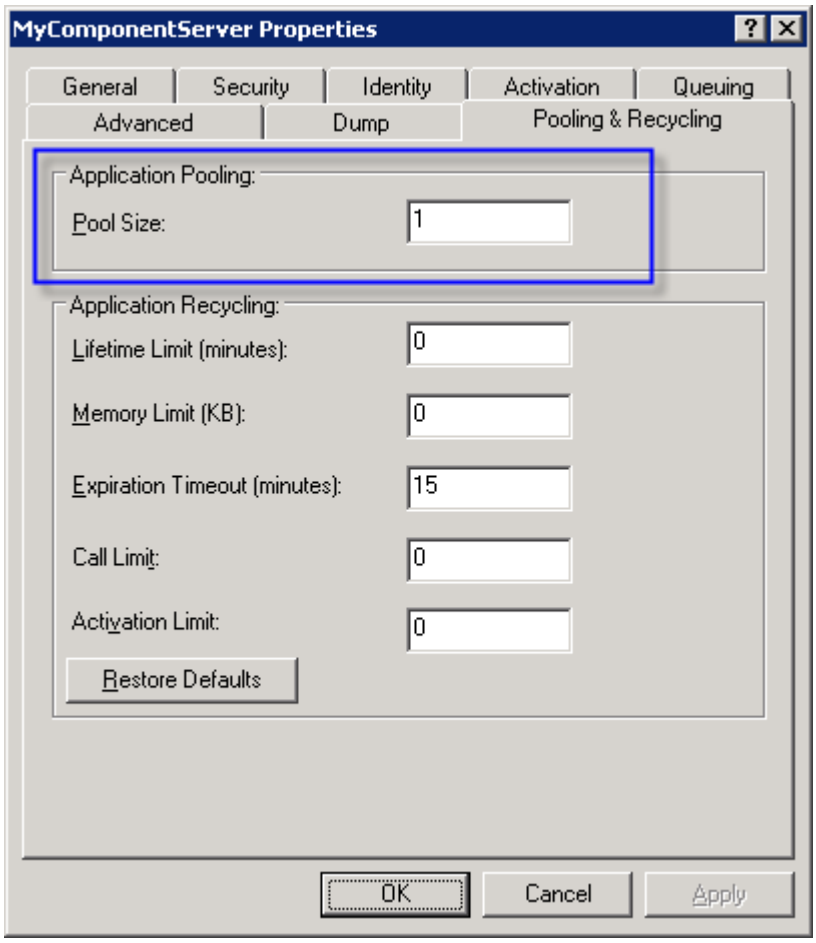
A server activation type of COM+ service, similarly to the IIS’ s Web Garden, uses the application pool setting for multiple execution of the processes hosting the same COM+ component. To check the number of configured pools, you can follow the next procedures.

1. Execute “Start”/ “Administrative Tools”/ “Component Services”.

2. As shown below, select a desirable component and right click with your mouse button to select the property menu.



3. In the property window, select the “Pooling & Recycling” tab to check the following “Pool Size” menu.



9.7. .NET Framework Environment

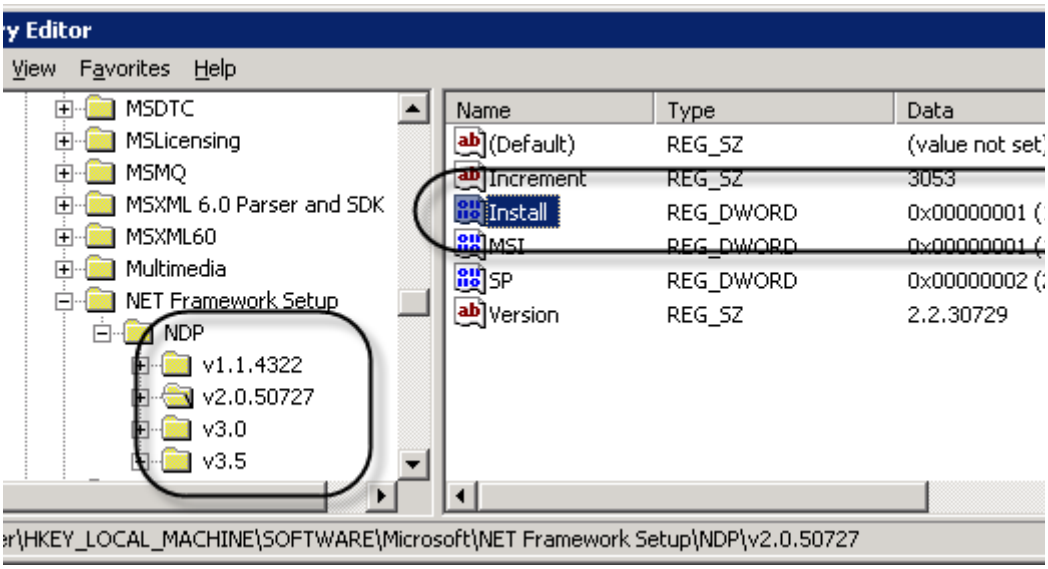
9.7.1. Checking the Version of the Installed .NET Framework

Based on the .NET framework version 2.0, various DLL corresponding to the WCF, WF and WPF functions are added to establish version 3.0. Also, LINQ is added to version 3.5. Therefore, the fact that version 3.5 is installed in the system means that versions 3.0 and 2.0 are also installed. Likewise, the fact that version 3.0 is installed in the system means that version 2.0 is basically installed.

You can use the following registry keys to check the version of the installed .NET framework.

```
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\NET Framework Setup\NDP
```

The following figure shows the fact that versions 1.1, 2.0, 3.0 and 3.5 are installed in Windows 2003 OS.



In this case, you must check that the value of “Install” (REG_DWROD) that belongs to the registry key is 1.

9.7.2. Checking the Version of 32 Bit .NET Framework Installed in a 64 bit OS

Basically, the registry path “HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\NET Framework Setup\NDP” described in checking the installed .NET framework version indicates whether a 32 bit .NET framework is installed for your 32 bit OS and whether a 64 bit .NET framework is installed for your 64 bit OS.

Additionally, in case of a 64 bit OS, you can also install a 32 bit .NET framework. To check this, you must check the registry path that contains “Wow6432Node” as shown below.

```
HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Microsoft\NET Framework
Setup\NDP
```

Powerful JENNIFER .NET

Application Performance Management (APM)

Application Performance Management (APM) is a process of developing a progressive system that effectively monitors the availability of enterprise applications, identifies and resolves the application performance problems, and predicts/prevents future application performance problems, considering the limit and potential of the resources available to an organization's IT infrastructure and its supporting systems. Unlike the traditional system management (SMS and NMS), APM enhances the organization's ability to manage the enterprise application services and resolve application performance problems, resulting in reduced Total Cost of Ownership (TCO) and enhanced customer service.

JENNIFER

JENNIFER is the Application Performance Management (APM) solution developed by JenniferSoft Inc. JENNIFER provides total application performance management and operational support services for enterprise web system, performing tasks such as real-time resource and service monitoring, immediate performance problem diagnosis, and effective performance problem resolution. JENNIFER is the premier APM solution in Korea.

Dynamic Profiling

JENNIFER can register additional package, class, method and/or activate/deactivate transaction-profiling without restarting the web application server.

Dynamic StackTrace

The traditional method for extracting Java Full StackTrace is intentionally causing an exception/error for an application resource and outputting it onto the stacktrace; JENNIFER can register a class/method during operation, allowing dynamic full stacktrace of additional class/method without changing the application source code.

Monitoring per Domain

In a large scale of enterprise environment, many different business systems may exist, triggering a need for a solution that individually monitors each business system under integrated one view. JENNIFER provides performance management capability per domain that allows the user to allocate multiple business systems into different domain and manage each system under one umbrella.

Extended Monitoring Adaptor

JENNIFER extracts the performance data from web application server and communication between WAS and other system devices. JENNIFER features Extended Monitoring Adaptors Functionality (EMAF) that allows performance data from other system devices to be extracted and inputted into JENNIFER for analysis and reporting.

JENNIFER®
Enterprise Web Business Monitoring

www.jennifersoft.com

Starvalley 1104, 60-11 Gasan-dong, Geumcheon-gu, Seoul, Korea.

TEL. +82.2.2027.0391~9 **FAX.** +82.2.2027.0390 **E-mail.** sales@jennifersoft.com