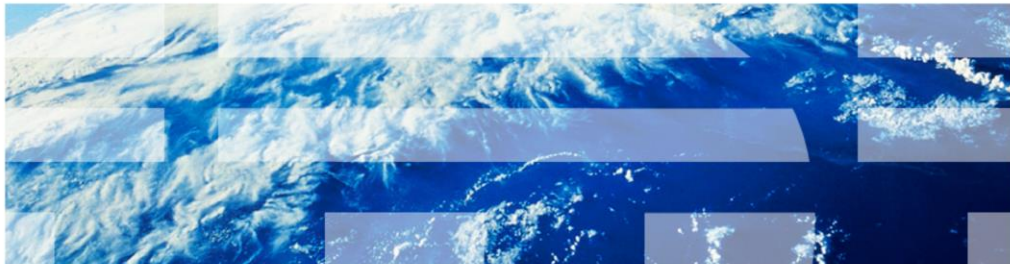


IBM Tivoli Application Dependency Discovery Manager 7.2

Troubleshooting Windows Discovery



© 2011 IBM Corporation

In this module, you learn about troubleshooting Windows® Discovery.

Assumption

You are familiar with Tivoli® Application Dependency Discovery Manager

Assumption.

An assumption for this module is that you are familiar with Tivoli Application Dependency Discovery Manager.

Objectives

When you complete this module, you can perform these tasks:

- Describe Windows Management Implementation (WMI)
- Describe Windows Discovery challenges
- Describe Windows Discovery troubleshooting steps
- Describe and use wmicmgmt, and use wmicdiag tools
- Describe and use wbemtest to query WMI classes, test connectivity, and run queries
- Describe and use testwmi, and use wmicexec tools
- Describe TADDMTTool calls and queries

Objectives.

When you complete this module, you can perform the following tasks:

- Describe WMI, Windows Discovery challenges, and troubleshooting steps.
- Describe and use wmicmgmt, and use wmicdiag tools.
- Describe and use wbemtest to query WMI classes, test connectivity, and run queries.
- Describe and use testwmi, and use wmicexec tools.
- Describe TADDMTTool calls and queries.

WMI overview

- Windows Management Instrumentation (WMI) is a core Windows management technology; which you can use for managing both local and remote computers
- You can use WMI for carrying out day-to-day management tasks with programming or scripting languages
- Examples of capabilities are as follows:
 - Starting a process on a remote computer
 - Scheduling a process to run at specific times on specific days
 - Restarting a computer remotely
 - Installing applications on a local or remote computer
 - Querying the Windows event logs on a local or remote computer
- In WMI, *instrumentation* means that WMI can get information about the internal state of computer systems as follows:
 - Instrumentation is analogous to dashboard instruments of cars retrieving and showing information about the state of the engine
 - WMI instruments by modeling objects, such as disks, processes, or other objects in Windows systems.
- You can find additional documentation by searching “about WMI” at the WMI troubleshooting and tips page at <http://technet.microsoft.com/en-us/library/ee692772>

WMI overview.

WMI is described, along with the location for additional WMI information.

Windows security considerations

- Potentially obscure and challenging technical area
- Mastered by Windows administrators
- Different in every environment
- Changing with every service pack
- Designed to prevent misused access in discovering information about the server

Note: Tivoli Application Dependency Discovery Manager is a system management tool designed to discover information about servers in your environment

Windows security.

Windows security and any application attempting remote access, including Tivoli Application Dependency Discovery Manager, requires special consideration.

Organizational challenges with Windows Discovery

- Tivoli Application Dependency Discovery Manager requires testing with a full local administrator user access on each server to be discovered
- Every customer has a different security policy and removes rights from the service account that Tivoli Application Dependency Discovery Manager is permitted to use for discovering
- The Tivoli Application Dependency Discovery Manager administrator and the Windows administrators are not always in the same organization, building, or city

Organization challenges with Windows Discovery.

Windows Discovery organization challenges are shown:

- The purpose of security is to prevent discovery.
- The Tivoli Application Dependency Discovery Manager administrator must work with Windows administrator supervision.

Steps to troubleshooting a Windows discovery (1 of 2)

1. Verify WMI problems
2. Verify access level/rights
3. Verify Tivoli Application Dependency Discovery Manager server configuration as follows:
 - Check that a Windows system been selected to be a gateway
 - Make sure the target and the gateway are in the scope
 - Check to make sure the access list has a full administrator account for the target
 - Know how your WMI discoveries are configured in the collation.properties file

Note: A common error is configuring for auto-deploying but not auto-restarting WMI, which causes the new provider to not load until WMI is manually restarted
4. Verify gateway as follows:
 - Confirm that the gateway is running.
 - If the gateway is not deployed or all the targets are not working, check that the Tivoli Application Dependency Discovery Manager server can connect to the gateway that uses SSH

Steps to troubleshooting a Windows discovery (1 of 2).

Verify WMI, access, server configuration, and gateway problems for Windows discovery.

Steps to troubleshooting a Windows discovery (2 of 2)

5. Verify communication between the gateway and target

Use `wmimgmt.msc` to check communications between the gateway and the target (Make sure you run the `.msc` file from the gateway at the target and are logged in as the discovery user.)

- Use `nslookup` to check naming in both directions
- Ensure gateway and target can ping each other

6. Verify target

- Confirm the correct version of .NET
- Verify `%SystemRoot%\System32\wbem` for following provider files:
 - `TaddmWmi.dll`
 - `TaddmWmi.mof`
 - `TaddmWmi.exe`
 - `TaddmWmi.pdb`
- If the provider does not deploy, run “net share” and make sure `c$` or `admin$` are shared
- Run `wmidiag` and look for errors in the `.log` file
- Run `net start` and check that WMI, DCOM, and RPC services are running
- Ensure the gateway can ping the target

7. Verify problem by using a specific Tivoli Application Dependency Discovery Manager call

Steps to troubleshooting a Windows discovery (2 of 2).

Verify communications and target problems for Windows discovery.

Troubleshooting tips

- Adjusting the Windows or network environment can resolve most Windows discoveries
- Failure of wmingmt, nslookup, ping, or wmiidag indicates that the problem is not because of Tivoli Application Dependency Discovery Manager. Notify the Windows and network administrators.
- Using the Tivoli Application Dependency Discovery Manager Windows document at this website can guide you through testing:
<http://www-304.ibm.com/support/docview.wss?uid=swg21426185>
- Using Tivoli Application Dependency Discovery Manager Discovery of Windows Targets without an Anchor (<http://www-304.ibm.com/support/docview.wss?uid=swg21295127>) does not work well for large numbers of targets because the registry of each target requires modification to open ports

Troubleshooting tips.

Additional troubleshooting information is listed.

wmimgmt.msc tool

Run wmimgmt.msc

1. Use the Microsoft WMI Control Console (wmimgmt.msc) on the gateway to connect to the failing target
2. Log in as the user that you specified in the access list for Windows discoveries. When you are logged in, you can select Properties from the Action list and show system information from the target.
 - If wmimgmt.msc runs successfully, the connection between the gateway and the target is good. The discovery account was logged into and WMI responded.
 - If wmimgmt.msc fails, request that your Windows administrator investigate the WMI problem.
 - Some possible failure causes are as follows:
 - NET is not working correctly, wrong version, or missing
 - Remote Process Calls service is not working
 - The account you are using might be invalid or inactive

wmimgmt.msc tool.

Steps for running the wmimgmt.msc tool are shown.

WMIDiag tool from Microsoft

Verify access to the computer you are discovering.

- You can use the Microsoft WMIDiag tool at this website for finding all access level and WMI problems

<http://www.microsoft.com/download/en/details.aspx?displaylang=en&id=7684>

- The tool has the following capabilities:
 - Querying the common WMI classes
 - Reporting if the user has the appropriate rights
 - Reporting if any of the classes are broken
 - Checking the registry
 - Reporting if WMI is running correctly
 - Checking if providers are loaded

WMIDiag tool from Microsoft.

WMIDiag tool installation process is shown.

- You must download and install an executable file on the target system.
- Windows 2008 and Windows 7 do not support WMIDiag.
- Functionality can differ for different versions of Windows.

Using WMIDiag

Run WMIDiag: cscript wmidiaq.vbs

1. Running WMIDiag produces these three files, by default, in the %TEMP% directory:
 - A .log file containing a verbose dialog of the WMIDiag tool activity
 - A .txt file containing a summarized report with warnings and errors for possible investigation
 - A .csv file containing statistics that can be used for measuring trends in WMI issues over time
2. Example error message is as follows:
ERROR: Actual trustee 'NT AUTHORITY\NETWORK SERVICE' DOES NOT match corresponding expected trustee rights (Actual->Default)

A website with WMIDiag information is at the following website:

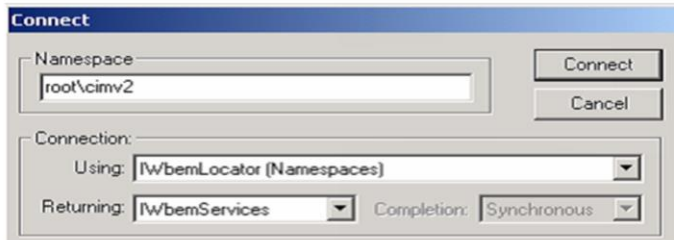
<http://www.windowsitpro.com/article/articleid/100845/resolve-wmi-problems-quickly-with-wmidiaq.html>

Using WMIDiag.

Running WMIDiag, and troubleshooting information of wmidiaq problems are shown.

WMI tests using wbemtest

On the target server, you can use wbemtest to query WMI classes



WMI test using wbemtest.

You can use wbemtest to query WMI classes on the target server. Tivoli Application Dependency Discovery Manager Discovery uses the following classes: Win32_Process, Win32_OperatingSystem, Win32_WMISetting and Win32_ComputerSystem.

WMI connectivity tests using wbemtest

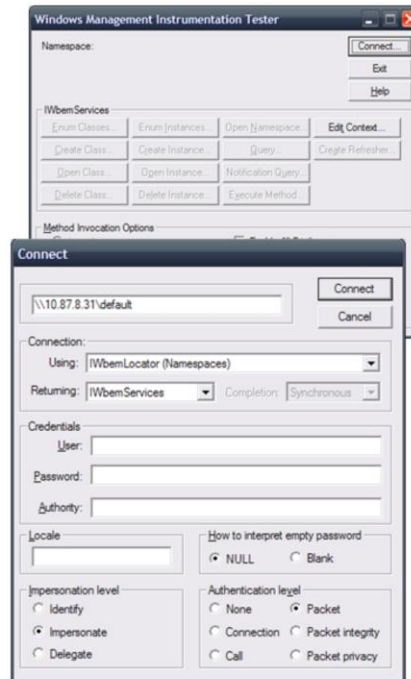
Error code CTJTP1161E

The application is unable to establish the WMI session

1. Run wbemtest
2. Test WMI on the windows gateway by connecting to root\cimv2
3. Test connectivity from the gateway to the target:
 - a. Connect to \\servername\root\cimv2
 - b. Use either the IP or the servername
 In both cases, if the buttons are active, WMI is working



[http://msdn.microsoft.com/en-us/library/aa394559\(VS.85\).aspx](http://msdn.microsoft.com/en-us/library/aa394559(VS.85).aspx)
 Troubleshooting Windows Discovery

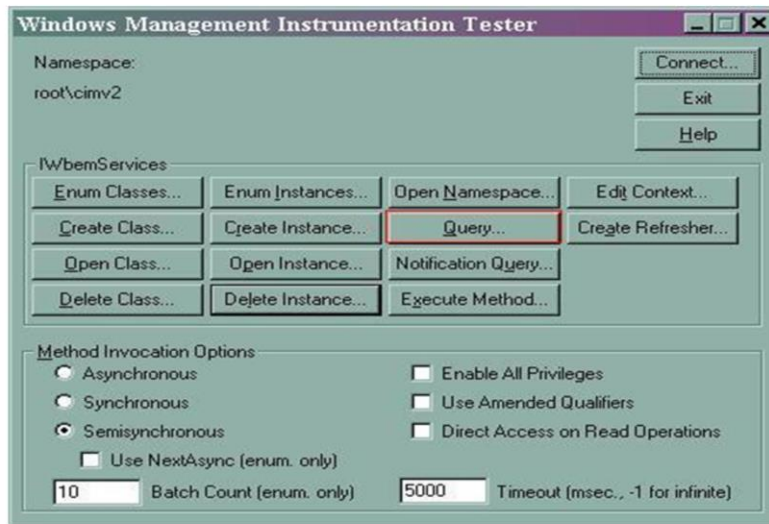


© 2011 IBM Corporation

WMI Connectivity tests using wbemtest.

Steps to test WMI connectivity by using wbemtest are shown. WMI errors from Microsoft are listed at [http://msdn.microsoft.com/en-us/library/aa394559\(VS.85\).aspx](http://msdn.microsoft.com/en-us/library/aa394559(VS.85).aspx).

Running queries from the logs in wbemtest



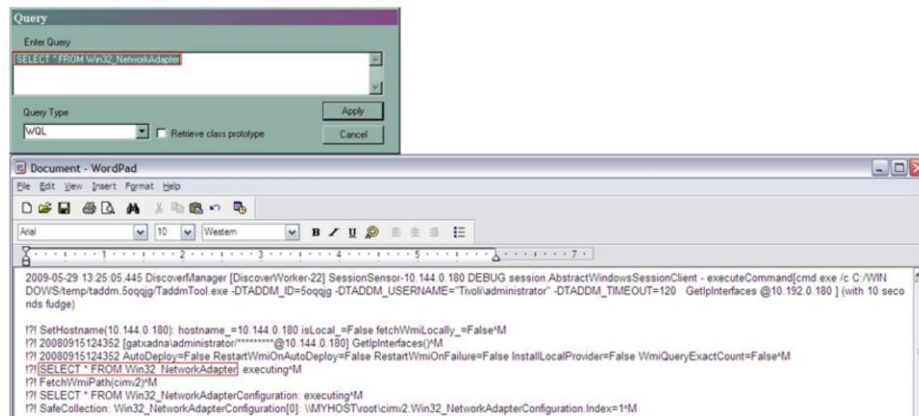
Troubleshooting Windows Discovery

© 2011 IBM Corporation

Running queries from the logs in wbemtest.

You can run queries from the logs in wbemtest.

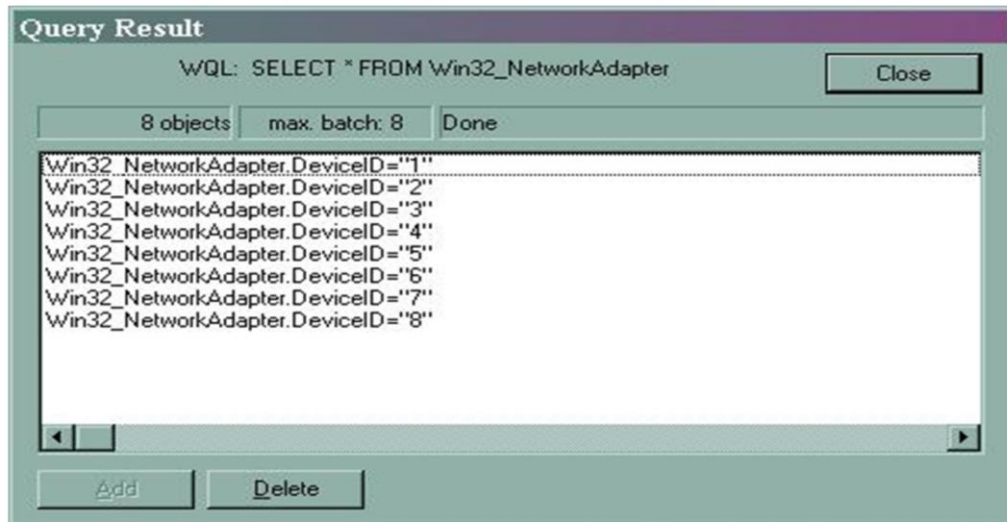
Finding and running the query



Finding and running the query.

Enter query and click **Apply**. The query is shown.

Results



The image shows a 'Query Result' window from an IBM management console. The window title is 'Query Result'. Below the title bar, the WQL query is displayed: 'WQL: SELECT * FROM Win32_NetworkAdapter'. To the right of the query is a 'Close' button. Below the query, there are three status indicators: '8 objects', 'max. batch: 8', and 'Done'. The main area of the window contains a list of eight objects, each represented as a WQL path: 'Win32_NetworkAdapter.DeviceID="1"', 'Win32_NetworkAdapter.DeviceID="2"', 'Win32_NetworkAdapter.DeviceID="3"', 'Win32_NetworkAdapter.DeviceID="4"', 'Win32_NetworkAdapter.DeviceID="5"', 'Win32_NetworkAdapter.DeviceID="6"', 'Win32_NetworkAdapter.DeviceID="7"', and 'Win32_NetworkAdapter.DeviceID="8"'. At the bottom of the window, there are two buttons: 'Add' and 'Delete'.

Troubleshooting Windows Discovery

© 2011 IBM Corporation

Results.

The query result is shown.

testwmi.jy tool

- You can use the testwmi.py tool in dist/support/bin on the Tivoli Application Dependency Discovery Manager tool for verifying end-to-end WMI connection
- The tool uses the ssh credential from the access list for the Windows gateway
- The tool uses the WMI credential from the access list for the Windows target
- You verify Tivoli Application Dependency Discovery Manager server by clicking **Windows Gateway > Target**

testwmi.jy tool.

The testwmi.jy tool is shown.

testwmi.jy tool example

```
[root@tadcm02 bin]# ./testwmi.jy 9.3.4.174
Testing WMI on host 9.3.4.174
...
2006-06-27 16:28:47,785[main] INFO session.SessionFactory - getSession(9.3.4.174) portList=null
2006-06-27 16:28:57,888[main] INFO util.PortScanner - PortScanner: scan for 9.3.4.174 complete; returning: [135]
2006-06-27 16:29:06,875[main] INFO util.PortScanner - PortScanner: scan for 9.3.5.252 complete; returning: [22]
2006-06-27 16:29:07,868[main] INFO session.Ssh2SessionClient - 9.3.5.252: SSH version=[SSH-2.0-1.75 sshlib: WinSSHD 4.13]
reuse=true
2006-06-27 16:29:08,526[main] INFO session.UnscopedGateway - Gateway.prepare(9.3.5.252): first attempt
2006-06-27 16:29:11,043[main] INFO session.UnscopedGateway - Gateway.prepare(9.3.5.252): succeeded!
2006-06-27 16:29:11,531[main] INFO session.AbstractWindowsSessionClient - GetDesiredVersion returns version 20060607 after 0
seconds.
2006-06-27 16:29:12,093[main] INFO session.AbstractWindowsSessionClient - GetVersion returns version 20060607 after 0 seconds.
2006-06-27 16:29:12,719 [main] INFO session.AbstractWindowsSessionClient - GetVersion returns version 20060607 after 0
seconds.
```

testwmi.jy tool example.

The testwmi.jy tool example is shown.

wmiexec.jy tool

- You can use the wmiexec.py tool in dist/support/bin on the Tivoli Application Dependency Discovery Manager server
- You can run a command on the target by using WMI

```
[root@taddm02 bin]# ./wmiexec.jy 9.3.4.174 "hostname"
```

```
Testing ...9.3.4.174 hostname...
```

```
2006-06-27 16:26:31,528 [main] INFO util.PortScanner - PortScanner: scan for 9.3.5.252 complete; returning: [22]
```

```
2006-06-27 16:26:32,692 [main] INFO session.Ssh2SessionClient - 9.3.5.252: SSH version=[SSH-2.0-1.75 sshlib: WinSSHD 4.13] reuse=true
```

```
2006-06-27 16:26:33,320 [main] INFO session.UnscopedGateway - Gateway.prepare(9.3.5.252): first attempt
```

```
2006-06-27 16:26:35,729 [main] INFO session.UnscopedGateway - Gateway.prepare(9.3.5.252): succeeded!
```

```
2006-06-27 16:26:36,216 [main] INFO session.AbstractWindowsSessionClient - GetDesiredVersion returns version 20060607 after 0 seconds.
```

```
2006-06-27 16:26:36,734 [main] INFO session.AbstractWindowsSessionClient - GetVersion returns version 20060607 after 0 seconds.
```

```
2006-06-27 16:26:37,390 [main] INFO session.AbstractWindowsSessionClient - GetVersion returns version 20060607 after 0 seconds.
```

```
Result is :taddm99
```

wmiexec.jy tool.

The wmiexec.jy tool is shown.

TADDMTool calls

- You can troubleshoot specific failing calls (found using the logs) directly.
- Found in directory C:\WINDOWS\Temp\taddm.##### on the Windows Gateway

```
TaddmTool.exe taddmtool -DTADDM_ID=12345 -DTADDM_USERNAME=taddm -  
DTADDM_PASSWORD=password123 -DTADDM_INTERACTIVE=yes QueryServices  
@9.43.73.81 servicename
```

TADDMTool calls.

The TADDMTool location and example of the command is shown.

TADDMTool queries (1 of 2)

- AdsiDump
- AdsiDumpLocal
- AdsiDumpRemote
- AdsiEnum
- AdsiEnumLocal
- AdsiEnumRemote
- CheckServices
- Db2Find
- Db2FindSchema
- Db2Svce2Inst
- GetActiveDirectoryLdapParameters
- GetActiveDirectoryNamingContexts
- GetCitrixInformation
- GetEnvironment
- GetEtcServices
- GetFileInfo
- GetFreeDiskSpace
- GetHostId
- GetInstalledSoftware
- GetIpInterfaces
- GetLongPath
- GetNonDefaultServices
- GetNonDefaultServicesWithDescriptions
- GetPortMap
- GetProcessEnvironment
- GetRouteInfo
- GetSecurity
- GetShortPath
- GetSMSInformation
- GetSMSParentChilds

TADDMTool queries (1 of 2).

First page of TADDMTool queries are shown.

TADDMTool queries (2 of 2)

- GetSystemInfo
- GetTaddmToolVersion
- GetWmiClassProperties
- GetWmiClassValues
- Help
- InstallProvider
- Kill
- ListDevices
- ListDNSServers
- ListDrives
- ListIpAddresses
- ListKernelModules
- ListProcesses
- ListShares
- Md5Hash
- NetConnect
- Obscure
- ProbePort
- Ps
- QueryRegistry
- QueryServices
- RestartWmi
- RunCommand
- RunCommandUtf8
- SqlDump
- StartAnchor
- StartWmi
- StopWmi
- TestWmi
- Unobscure
- WinError

TADDMTool queries (2 of 2).

Second page of TADDMTool queries are shown.

Common issues

- Common permissions missing in these areas:
 - Process level token. (See technote for checking.)
 - Network service
 - whoami.exe, a command that is part of the resource kit that can show permissions, roles, and other information
- Changing permissions at the domain, but replication does not occur down to the server
 - A file called login.log or winlogin.log in system32 shows the permissions that are replicated down
 - The following website can inform about interpreting security setting log files:
<http://technet.microsoft.com/en-us/library/cc787154.aspx>
 - After you get the policy changed, you must clean up the Tivoli Application Dependency Discovery Manager files and permit them to be redeployed

Common issues.

Common Tivoli Application Dependency Discovery Manager issues are shown.

Replacing a process level token

By default, this process is granted to the discovery account:
LOCAL SERVICE, NETWORK SERVICE

1. Click **Start > Run**
2. In the **Open** field, type **gpedit.msc**, and click **OK**
3. In the new window, navigate to **Local Computer Policy > Computer Configuration > Windows Settings > Security Settings > Local Policies > User Rights Assignment**
4. In the right pane, look for **Replace a process level token**
The security setting includes LOCAL SERVICE and NETWORK SERVICE
5. If you cannot add a group or account to this list, try adding Local Administrators to a group that is already in the list

Replacing a process level token.

Replacing a process level token steps are shown.

Cleaning up a Tivoli Application Dependency Discovery Manager WMI deployment

See technote at this website:

<http://www-304.ibm.com/support/docview.wss?uid=swg21426185>

Cleaning up a Tivoli Application Dependency Discovery Manager WMI deployment.

Refer to the technote.

Summary

Now that you have completed this module, you can perform these tasks:

- Describe Windows Management Implementation (WMI)
- Describe Windows Discovery challenges
- Describe Windows Discovery troubleshooting steps
- Describe and use wmicmgmt, and use wmicdiag tools
- Describe and use wbemtest to query WMI classes, test connectivity, and run queries
- Describe and use testwmi, and use wmicexec tools
- Describe TADDMTTool calls and queries

Summary.

Now that you have completed this module, you can perform the following tasks:

- Describe WMI, Windows Discovery challenges, and troubleshooting steps.
- Describe and use wmicmgmt, and use wmicdiag tools.
- Describe and use wbemtest to query WMI classes, test connectivity, and run queries.
- Describe and use testwmi, and use wmicexec tools.
- Describe TADDMTTool calls and queries.



Trademarks, disclaimer, and copyright information

IBM, the IBM logo, ibm.com, and Tivoli 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 other IBM trademarks is available on the web at "[Copyright and trademark information](http://www.ibm.com/legal/copytrade.shtml)" at <http://www.ibm.com/legal/copytrade.shtml>

THE INFORMATION CONTAINED IN THIS PRESENTATION IS PROVIDED FOR INFORMATIONAL PURPOSES ONLY. Other company, product, or service names may be trademarks or service marks of others.

THE INFORMATION CONTAINED IN THIS PRESENTATION IS PROVIDED FOR INFORMATIONAL PURPOSES ONLY. WHILE EFFORTS WERE MADE TO VERIFY THE COMPLETENESS AND ACCURACY OF THE INFORMATION CONTAINED IN THIS PRESENTATION, IT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. IN ADDITION, THIS INFORMATION IS BASED ON IBM'S CURRENT PRODUCT PLANS AND STRATEGY, WHICH ARE SUBJECT TO CHANGE BY IBM WITHOUT NOTICE. IBM SHALL NOT BE RESPONSIBLE FOR ANY DAMAGES ARISING OUT OF THE USE OF, OR OTHERWISE RELATED TO, THIS PRESENTATION OR ANY OTHER DOCUMENTATION. NOTHING CONTAINED IN THIS PRESENTATION IS INTENDED TO, NOR SHALL HAVE THE EFFECT OF, CREATING ANY WARRANTIES OR REPRESENTATIONS FROM IBM (OR ITS SUPPLIERS OR LICENSORS), OR ALTERING THE TERMS AND CONDITIONS OF ANY AGREEMENT OR LICENSE GOVERNING THE USE OF IBM PRODUCTS OR SOFTWARE.

© Copyright International Business Machines Corporation 2011. All rights reserved.

© 2011 IBM Corporation