Windows Server AppFabric Beta 2 Refresh  
Release Notes

Microsoft Corporation

Published: April 2010

### Copyright

This document is provided “as-is”. Information and views expressed in this document, including URL and other Internet Web site references, may change without notice. You bear the risk of using it.

Some examples depicted herein are provided for illustration only and are fictitious.  No real association or connection is intended or should be inferred.

This document does not provide you with any legal rights to any intellectual property in any Microsoft product. You may copy and use this document for your internal, reference purposes.

© 2010 Microsoft Corporation. All rights reserved.

Microsoft, Visual Studio, Windows, Windows PowerShell, and Windows Server are trademarks of the Microsoft group of companies.

All other trademarks are property of their respective owners.

Contents

[Release Notes For Windows Server AppFabric Beta 2 Refresh 4](#_Toc259729696)

[Known Issues with Setup and Configuration 4](#_Toc259729697)

[Automated installation returns control before setup is complete 4](#_Toc259729698)

[Prerequisites for executing setup from a network share 4](#_Toc259729699)

[Upgrade Program cannot be used to upgrade from Beta 1 or Beta 2 to Beta 2 Refresh 5](#_Toc259729700)

[Uninstalling previous versions of AppFabric 5](#_Toc259729701)

[Uninstall/install order for AppFabric and .NET Framework 4 5](#_Toc259729702)

[Multiple .NET Framework 4 folders may cause AppFabric cmdlets to fail 6](#_Toc259729703)

[Uninstalling AppFabric can cause applications configured to use AppFabric features to be non-functional 6](#_Toc259729704)

[Uninstalling an upgraded version of AppFabric requires also uninstalling the previous version 6](#_Toc259729705)

[Uninstall AppFabric before upgrading Vista or Windows Server 2008 to Windows 7 or Windows Server 2008 R2 6](#_Toc259729706)

[Caching Service configuration is not completely removed during uninstallation 7](#_Toc259729707)

[Caching service firewall considerations 7](#_Toc259729708)

[Encrypted configuration data can cause AppFabric configuration to fail 7](#_Toc259729709)

[Use domain accounts to configure AppFabric on a domain controller 8](#_Toc259729710)

[The\inetpub\wwwroot folder must be manually created when IIS is disabled 8](#_Toc259729711)

[Support for server-level configuration of 32-bit applications on a 64-bit Operating System is limited 8](#_Toc259729712)

[Windows PowerShell cmdlets for AppFabric are not supported on WOW64 9](#_Toc259729713)

[AppFabric configuration and enumeration functionality is limited in a shared-content scenario 9](#_Toc259729714)

[Known Issues with Tooling Features 10](#_Toc259729715)

[PowerShell and Dashboard do not show instances that have not persisted yet 10](#_Toc259729716)

[Issues when using custom WCF bindings and behaviors 10](#_Toc259729717)

[Applications from both .NET 2 Framework and .NET 4 Framework are not recommended in the same website 10](#_Toc259729718)

[Starting and stopping of the default application must be done at the application level 10](#_Toc259729719)

[Workflow services require the NET.PIPE Protocol 10](#_Toc259729720)

[Net.pipe warning missing for site level in IIS Manager 11](#_Toc259729721)

[Known Issues with Caching Features 11](#_Toc259729722)

[Unable to add references to AppFabric caching assemblies from Visual Studio 11](#_Toc259729723)

[“Total Write Operations” performance counter differences between cache and host levels 12](#_Toc259729724)

[Network share for cache cluster configuration requires Full Control or Co-owner permissions 12](#_Toc259729725)

[Case sensitivity of region names, key names, and cache names 12](#_Toc259729726)

[Starting the AppFabric Caching Service prior to service configuration 12](#_Toc259729727)

[Creating new caches on a throttled server can cause the server to crash 12](#_Toc259729728)

# Release Notes For Windows Server AppFabric Beta 2 Refresh

This software is for evaluation and deployment planning purposes only. If you plan to install the software on your primary computer, back up your existing data prior to installation.

AppFabric Beta 2 Refresh is an update of AppFabric Beta 2 on top of the .NET Framework 4 RTM release. No big changes have been made since the Beta 2 release. Known issues in AppFabric Beta 2 Refresh have been documented below.

## Known Issues with Setup and Configuration

Configuration has been separated from the Setup wizard. You can run a standalone Configuration wizard after Setup or whenever AppFabric configuration changes are needed. The configuration wizard replaces the System Services, Persistence Database and Monitoring Database IIS modules. For installation information, see Installing Windows Server AppFabric.

### Automated installation returns control before setup is complete

Running automated installation returns control to the command line while setup is still running. To ensure that setup does not return until it has completed, invoke automated installation using the start /w command of cmd.exe.

### Prerequisites for executing setup from a network share

When Code Access Security (CAS) settings disallow the execution of code from intranet shares, AppFabric setup will simply crash because it will not be able to execute any code, including the code that displays error messages. For that reason, platform validation logic cannot be used to enforce the proper CAS settings. The solution for this issue is to allow the execution of code from intranet shares. The user will have to ensure that the CAS policy includes the local intranet permissions and setup.exe executes in full trust, as shown in the following examples:

 The following command adds the MyOther.exe assembly to the full trust list for the computer policy: caspol -machine -addfulltrust MyOther.exe

 The following command adds a child code group that gives the share \\netserver\netshare local intranet permissions: caspol -machine -addgroup 1. -url \\netserver\netshare\\* LocalIntranet

For more information on configuring the .NET Framework, see [.NET Framework Configuration Tool](http://go.microsoft.com/fwlink/?LinkId=181958) (Mscorcfg.msc) (http://go.microsoft.com/fwlink/?LinkId=181958).

For more information on configuring CAS, see [Code Access Security Policy Tool](http://go.microsoft.com/fwlink/?LinkId=181960) (Caspol.exe) (http://go.microsoft.com/fwlink/?LinkId=181960).

### Upgrade Program cannot be used to upgrade from Beta 1 or Beta 2 to Beta 2 Refresh

The AppFabric upgrade program does not enable you to upgrade from a Beta 1 or Beta 2 version of AppFabric. Using the program, you can only upgrade from AppFabric Beta 2 Refresh to RTW. You must upgrade from AppFabric Beta 1 or Beta 2 by first uninstalling AppFabric, and then installing a later version.

### Uninstalling previous versions of AppFabric

To uninstall AppFabric, click Start, click All Programs, click Windows Server AppFabric, click Add or remove Features, and then run through the AppFabric setup wizard.

AppFabric does not ship as an application, but as an update of Windows. As a result, you can also uninstall it by opening the Control Panel, clicking Programs, and then under Programs and Features, clicking View installed updates. AppFabric is listed under the Microsoft Windows section as Application Server Extensions for .NET 4.

### Uninstall/install order for AppFabric and .NET Framework 4

The order of uninstalling and installing previous beta versions of AppFabric and the .NET Framework 4 is of critical importance. The wrong order can result in an unrecoverable state.

If you uninstall .NET Framework 4 RC, and then install .NET Framework 4 RTW, and then attempt to uninstall AppFabric Beta 2, the AppFabric uninstallation will fail. The cause of this issue is that AppFabric Beta 2 uninstall has a dependency on the presence on these .NET Framework 4 configuration files:

 %SystemRoot%\Microsoft.NET\Framework[64]\v4.0.21006\config\machine.config

 %SystemRoot%\Microsoft.NET\Framework[64]\v4.0.21006\config\web.config

If .NET Framework 4 v4.0.21006 is uninstalled before AppFabric Beta 2 is uninstalled, the configuration files needed for AppFabric Beta 2 uninstall to succeed are removed causing AppFabric Beta 2 uninstall to fail. To resolve this issue, uninstall AppFabric Beta 2 before upgrading .NET Framework 4 v4.0.21006.

To uninstall and install

|  |
| --- |
| 1. Uninstall AppFabric Beta 2.  2. Uninstall .NET Framework 4 RC.  3. Install .NET 4 RTW.  4. Install AppFabric Beta 2 Refresh. |

If .NET Framework 4 v4.0.21006 has already been removed, you can work around this issue by copying the machine.config and web.config files from the new .NET Framework 4 location into the old .NET Framework 4 location.

### Multiple .NET Framework 4 folders may cause AppFabric cmdlets to fail

If multiple .NET Framework 4 folders are present on a computer, Windows PowerShell cmdlets for AppFabric may not work properly after you uninstall AppFabric Beta 1 and then uninstall .NET Framework 4. As a result, when you run the AppFabric configuration wizard after uninstalling AppFabric Beta 1 and then uninstalling .NET Framework 4, AppFabric may not be configured correctly. You can resolve this issue by deleting any older .NET Framework 4 folders before running the configuration wizard.

### Uninstalling AppFabric can cause applications configured to use AppFabric features to be non-functional

When you use Windows Server AppFabric, you may configure a server, site, or application to use AppFabric-specific features. This can cause your applications to be non-functional after you uninstall Windows Server AppFabric. As a result, after you uninstall AppFabric, you may need to clean your application configuration. This can involve locating and cleaning the applicationHost.config file and the Web.config files for each scope.

You can also use the AppFabric Uninstall Cleanup Utility that is available on the AppFabric download center for an automated solution.

For more information, see “Clean Application Configuration After Unstalling AppFabric” in the [AppFabric Installation Guide](http://go.microsoft.com/fwlink/?LinkId=185913) (http://go.microsoft.com/fwlink/?LinkId=185913).

### Uninstalling an upgraded version of AppFabric requires also uninstalling the previous version

If you have upgraded AppFabric to a later version, uninstalling that version of AppFabric will leave the previous version of AppFabric still installed. To uninstall the previous version, open the Control Panel, click Programs and Features, click View Installed Updates, right-click Windows Server AppFabric (KB970622), and then click Uninstall. This issue only occurs on Windows 7 or Windows Server 2008 R2, because AppFabric upgrade is only supported on those operating systems, not on Windows Vista or Windows Server 2008.

### Uninstall AppFabric before upgrading Vista or Windows Server 2008 to Windows 7 or Windows Server 2008 R2

If AppFabric Beta 2 Refresh is installed on Vista or Windows 2008, and the operating system is to be upgraded to Windows 7 or Windows Server 2008 R2, AppFabric needs to be uninstalled before the upgrade and reinstalled after completing the upgrade.

Before upgrading the OS

|  |
| --- |
| 1. Record which AppFabric components are installed.  2. Record any changes to settings made to the Windows Server AppFabric Beta 2 Refresh installation. For example, note changes to the following:  a. configuration file  b. membership of AS\_Administrators and AS\_Observers Windows groups  c. AppFabric Windows services like Event Collection Service, Workflow Management Service, and Caching Service  3. After recording the changes, uninstall AppFabric Beta 2 Refresh, then upgrade the OS, and then reinstall AppFabric Beta 2 Refresh.  4. Reapply settings to the new AppFabric Beta 2 Refresh install on the upgraded OS. |

If you perform an operating system upgrade without uninstalling AppFabric beforehand, then after the upgrade you need to uninstall AppFabric, and then remove the following registry entry:

<keyPath>HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\AppFabric\v1.0\</keyPath>

<valueName>ProductVersion</valueName>

### Caching Service configuration is not completely removed during uninstallation

When using Add-Remove Programs, Caching Service is uninstalled, but node configuration details are left behind. This occurs because unconfigure steps are not performed when using Add-Remove Programs. For more information, see [Clean Application and Clustering Configuration After Uninstalling AppFabric](http://go.microsoft.com/fwlink/?LinkId=190412) (http://go.microsoft.com/fwlink/?LinkId=190412).

### Caching service firewall considerations

After you uninstall the caching features from a distributed cache server, the AppFabric setup program will disable the Windows Server AppFabric: AppFabric Caching Service firewall rule group, removing it as an exception. Upon installation, the setup program creates this group and adds rules to it. However, if you enable the Remote Service Management firewall rule group upon setup, the setup program will not disable it upon uninstallation. You will have to disable Remote Service Management manually on each node of the cluster, if you choose to do so. After any of the caching features of AppFabric has been uninstalled, review your firewall configuration.

### Encrypted configuration data can cause AppFabric configuration to fail

AppFabric Hosting Services cmdlets can read encrypted configuration sections, but cannot write to them. This can result in AppFabric configuration failing if during the configuration process (either through the use of the AppFabric configuration wizard or directly through the use of a PowerShell cmdlet), a cmdlet attempts to write to a section in the Web.config file whose data is entered in an <EncryptedData> section. You will see the following error: “Configuration section encryption is not supported.” (error code 555). This indicates that the configuration section that AppFabric setup was attempting to write to has been encrypted. To resolve this issue, unencrypt the section and run setup again.

### Use domain accounts to configure AppFabric on a domain controller

The AppFabric Configuration Wizard can fail on a domain controller if you use non-domain accounts. Per-service accounts for the Event Collection service and Workflow Management service cannot be added to the AS\_Administrators group or the AS\_Observers group on a domain controller. The reason is that NT Service SIDs are longer than those supported by the Active Directory. When you attempt to initialize a store, this can result in an error that a Windows security principal is invalid.

To resolve this issue, use domain accounts to configure AppFabric on a domain controller. To do so, collocate the Active Directory Directory Service role with the Application Server role as follows:

1. Run the Event Collection service and Workflow Management service under domain accounts, rather than per-service accounts.

2. Add the domain accounts used for the services to the group (AS\_\*) that will be used to connect to the associated monitoring or persistence database.

3. Add the Event Collection service account to the perflog user group manually. This enables the Event Collection service to be able to read from or write to the Event Tracing for Windows (ETW) session.

4. Whenever you pass any Windows principal to any AppFabric Hosting Services cmdlet, such as Initialize-ASMonitoringSqlDatabase or Initialize-ASPersistenceSqlDatabase, use an account in either builtin users or domain accounts. Do not use an account in local user/groups.

### The\inetpub\wwwroot folder must be manually created when IIS is disabled

If you install the IIS Manager with IIS disabled, you must create the \inetpub\wwwroot folder after the AppFabric Installation Wizard is run. Otherwise, the default Web site will be mapped to a non-existing folder. This is true for the Standard, Enterprise, Datacenter, and Foundations editions. For AppFabric installations on these editions, create this folder manually.

### Support for server-level configuration of 32-bit applications on a 64-bit Operating System is limited

There are limitations in the ability to edit the configuration of a 32-bit application on a server that uses a 64-bit operating system. On a 64-bit server, if you use the AppFabric IIS Manager Extensions and Windows PowerShell cmdlets for AppFabric to make default configuration settings at the server (root) level for a 32-bit application, the runtime will not use those configuration settings.

The IIS Manager Extensions and AppFabric cmdlets make these configuration settings in the machine.config and web.config files in the <drive>:\Windows\Microsoft.NET\Framework64\<build>\config folder. These settings are not made in the Framework folder that is used for 32-bit applications, and as a result, the runtime will not use the configuration settings when running the application. On a 64-bit server, IIS Manager can only be launched in 64-bit mode and Windows PowerShell cmdlets for AppFabric can only be used from a 64-bit process. The runtime does not have an issue, because it determines whether to use 32-bit or 64-bit configuration settings based upon whether the application pool is 32-bit or 64-bit. The following server-level default configuration settings are affected by this issue:

 Event Collection service

 Workflow Management service

 Monitoring store

 Persistence store

 AppFabric default connection string

 Persistence providers

 endToEndTracing configuration under <Diagnostics> (related to the monitoring level).

You can avoid this issue by making default settings for a 32-bit application on a 64-bit server at the site or application level, not the server level. This issue does not occur with default configuration settings at the site or application level. You can also make default settings at the server level manually in the machine.config and web.config files, although you can encounter issues when you do so because of the presence of configuration settings in both the Framework64 and Framework folders.

### Windows PowerShell cmdlets for AppFabric are not supported on WOW64

On a server that is running 64-bit Windows, Windows PowerShell cmdlets for AppFabric can only be executed in a 64-bit process. If you must run an AppFabric cmdlet in a 32-bit process, you can write your AppFabric cmdlet script against 64-bit Windows PowerShell, and for the portion of the script that must be run from a 32-bit process, launch a 32-bit process from the 64-bit script. You can also launch a 64-bit process from a 32-bit process, and then launch your 64-bit script from that 64-bit process.

### AppFabric configuration and enumeration functionality is limited in a shared-content scenario

This occurs when an application that is hosted on an IIS server has a physical directory on a remote File server, and the IIS logon user does not have access to the physical directory. In this case, the logon user will not be able to see all the services and their configuration data. This limitation applies to both the AppFabric IIS Manager extensions and the AppFabric cmdlets.

This will occur if the following two settings are made. First, in the Advanced Settings dialog box for an application hosted on the IIS server, you set the physical path for the application to a physical directory on a File server, and you set the credentials to a user other than the logged-on user. Second, in the Security tab of the Properties dialog box for the folder on the File server, you only grant access permissions to the user other than the logged-on user, but not to the logged-on user.

## Known Issues with Tooling Features

### PowerShell and Dashboard do not show instances that have not persisted yet

Workflow instances that have not persisted yet are not returned by the Get-ASAppServiceInstance cmdlet or displayed on the AppFabric dashboard.

### Issues when using custom WCF bindings and behaviors

If your services use customer behaviors or bindings, you will not be able to configure these settings in AppFabric directly. This is because AppFabric uses MWA configuration API (native to IIS) to read/write configuration. If you want to be able to configure them, you will need to convert these custom element sections to MWA-compatible schemas and place that schema in %SystemRoot%\System32\inetsrv\config\schema. MWA will automatically pick up this schema and parse the custom section in configuration correctly. You can then edit these configuration sections using IIS’s native configuration editor (IIS Configuration Editor) that ships inside of IIS Manager, not the AppFabric Configuration Editor.

### Applications from both .NET 2 Framework and .NET 4 Framework are not recommended in the same website

AppFabric tools provide first-class support for configuring .NET 4 framework applications, and do not explicitly provide tool support for .NET 2 framework applications. Therefore, mixing .NET 2 and .NET 4 applications within the same website may create issues such as mismatched configurations which could break your applications. We recommend that you not mix .NET 2 and .NET 4 applications and instead put them in different websites so you can manage them more successfully in AppFabric.

### Starting and stopping of the default application must be done at the application level

Starting and stopping of applications is not enabled at the site level.

### Workflow services require the NET.PIPE Protocol

The Workflow Management Service (WMS) requires the net.pipe protocol to control workflow instances. To use net.pipe, you must add the net.pipe binding on the web site hosting workflow services, and enable the net.pipe protocol on the application hosting workflow services. Do not change these settings if any instances of the service have persisted, as the wrong binding may be stored in the database.

To add the net.pipe binding to a website

|  |
| --- |
| 1. Open IIS Manager and select the website.  2. In the Actions pane, click Edit Bindings.  3. Click Add.  4. Select net.pipe, and set the Binding Information to an asterisk (\*). |

To enable the net.pipe protocol on an application

|  |
| --- |
| 1. Open IIS Manager and select the application.  2. In the Actions pane, click Advanced Settings.  3. In the Enabled Protocols box, update the value to include net.pipe. Separate protocols with a comma. For example, the value should be http,net.pipe. |

### Net.pipe warning missing for site level in IIS Manager

When you configure applications or services at the site level in IIS Manager, a warning should be displayed when the Workflow Persistence tab or the Workflow Host Management tab is selected in order to remind the user to enable net.pipe on the individual applications. The warning would indicate that persistence or the instance control may not function correctly if net.pipe is not enabled for the individual applications at the site level. This warning is displayed correctly at the server level; however, the warning is missing at the site level.

## Known Issues with Caching Features

### Unable to add references to AppFabric caching assemblies from Visual Studio

The AppFabric caching assemblies are stored in the .\Windows\System32\AppFabric directory. On a 64-bit operating system, the Add Reference feature of Visual Studio is unable to browse to this location. There are two current workarounds for this problem:

 Open the Visual Studio project file in an XML editor and manually add the necessary references. This is described in online documentation that describes how to prepare the cache client development environment.

 Another solution is to browse to the pseudo-directory .\Windows\SysNative from the Add Reference dialog.

For more information about the cause of this issue, see <http://msdn.microsoft.com/en-us/library/aa384187(VS.85).aspx>.

### “Total Write Operations” performance counter differences between cache and host levels

The performance counter “Total Write Operations” can be added for both the “AppFabric Caching:Cache” and the “AppFabric Caching:Host” counter categories. When the default cache is used by client applications, the total number of writes on the host may differ from the total number of writes across all caches on the host. This happens due to internal creation of regions in the default cache.

### Network share for cache cluster configuration requires Full Control or Co-owner permissions

When using a network share to manage the cache cluster configuration settings, you must have Full Control or Co-owner permissions on the share. If you are using the net share command, specify "full" as in the following example:

net share sharename /grant:user,FULL

### Case sensitivity of region names, key names, and cache names

Region and key names are always case-sensitive. The case-sensitivity of cache names is dependent on how the configuration information is stored.

 XML - The cache name is case-sensitive.

 SQL Server - The cache name is case-sensitive if the SQL Server database uses a case-sensitive collation. The cache name is case-insensitive if the SQL Server database uses a case-insensitive collation.

### Starting the AppFabric Caching Service prior to service configuration

When the AppFabric Caching Service is started manually prior to configuration, it creates a single-host cache cluster with “localhost” as the only cache host. Instead of relying on this default behavior, you should always use the AppFabric Configuration Wizard to configure Caching Services prior to starting them from Windows PowerShell with the Start-CacheCluster command.

### Creating new caches on a throttled server can cause the server to crash

When a server is in a throttled state, available memory is less than 4% or available memory on the system goes below the paged out memory for the process. In this state, creating new AppFabric caches can cause the server to fail. In most cases, proper capacity planning should avoid this scenario for a server.