
Release Notes

Platform Symphony™
Version 5.1
Release date: April 2011
Last modified: April 29, 2011



Copyright

© 1994-2011 Platform Computing Corporation.

Although the information in this document has been carefully reviewed, Platform Computing Corporation ("Platform") does not warrant it to be free of errors or omissions. Platform reserves the right to make corrections, updates, revisions or changes to the information in this document.

UNLESS OTHERWISE EXPRESSLY STATED BY PLATFORM, THE PROGRAM DESCRIBED IN THIS DOCUMENT IS PROVIDED "AS IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT WILL PLATFORM COMPUTING BE LIABLE TO ANYONE FOR SPECIAL, COLLATERAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, INCLUDING WITHOUT LIMITATION ANY LOST PROFITS, DATA, OR SAVINGS, ARISING OUT OF THE USE OF OR INABILITY TO USE THIS PROGRAM.

We'd like to hear from you

You can help us make this document better by telling us what you think of the content, organization, and usefulness of the information. If you find an error, or just want to make a suggestion for improving this document, please address your comments to doc@platform.com.

Your comments should pertain only to Platform documentation. For product support, contact support@platform.com.

Document redistribution and translation

This document is protected by copyright and you may not redistribute or translate it into another language, in part or in whole.

Internal redistribution

You may only redistribute this document internally within your organization (for example, on an intranet) provided that you continue to check the Platform Web site for updates and update your version of the documentation. You may not make it available to your organization over the Internet.

Trademarks

LSF is a registered trademark of Platform Computing Corporation in the United States and in other jurisdictions.

ACCELERATING INTELLIGENCE, PLATFORM COMPUTING, PLATFORM SYMPHONY, PLATFORM JOB SCHEDULER, PLATFORM ISF, PLATFORM ENTERPRISE GRID ORCHESTRATOR, PLATFORM EGO, and the PLATFORM and PLATFORM LSF logos are trademarks of Platform Computing Corporation in the United States and in other jurisdictions.

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

Linux is the registered trademark of Linus Torvalds in the U.S. and other countries.

Microsoft is either a registered trademark or a trademark of Microsoft Corporation in the United States and/or other countries.

Windows is a registered trademark of Microsoft Corporation in the United States and other countries.

Intel, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Other products or services mentioned in this document are identified by the trademarks or service marks of their respective owners.

Third-party license agreements

<http://www.platform.com/Company/third.part.license.htm>

Contents

What's New?	5
Simplified Application Integration	5
Enhanced Performance and Scalability	5
Resource Management Improvements	6
Usability Enhancements	6
New Platform Support	7
What's Changed?	8
Error and log messages	8
Configuration	9
Commands	9
PERF	10
Scavenging	10
Audit log format	10
PMC	11
Applications	11
Known Issues	12
Message "Could not load file or assembly 'Platform.Symphony.Soam.Net_64.dll'" displays when trying to run .NET application	14

What's New?

Simplified Application Integration

With Platform Symphony 5.1, developers using Microsoft Visual Studio®2008 or 2010 can benefit from compilation-free integrations with existing service code written as .NET assemblies. An easy-to-use wizard guides developers through the process of loading and analyzing existing .NET services. Required code and application profiles are generated automatically.

With this enhancement, developers familiar with Visual Studio can rapidly adapt existing .NET services to use Platform Symphony and perform end-to-end integration testing without leaving the IDE. This makes them more productive and relieves them of the need to learn the Platform Symphony API in detail before integrating an application.

Refer to the *Visual Studio Tutorial* under *Developer Tutorials* in the Symphony Knowledge Center.

Enhanced Performance and Scalability

The maximum number of supported services, cores, and applications have been increased as follows:

- 10000 services per application
- 40000 cores per cluster
- 300 applications per cluster

Simulation:

- Using "Symphony Virtual Cluster" technology to simulate 40000 virtual cores on 5000 virtual hosts.
- Tested performance, failover, functions, daemon footprint, resource allocation, workload scheduling in virtual scale cluster.
- Introduced best practices for large scale cluster tuning.

Refer to the *Performance tuning for large clusters* section in the *Cluster and Application Management Guide*.

Improved Resource Harvesting

New harvesting options for Platform Symphony 5.1 enable administrators to more efficiently tap idle cycles from resources residing elsewhere in the organization. Harvesting idle capacity improves overall efficiency and can help organizations avoid or defer spending on infrastructure.

Specific options exist for:

- Desktop harvesting
- Server harvesting

Desktop harvesting (also referred to as *desktop scavenging*) has existed for some time, but new features make server harvesting more efficient and practical.

While desktop harvesting is usage-based (detecting the presence of a desktop user), server harvesting is load-based. The server harvesting software accurately monitors non-Symphony load on candidate servers to ensure that Platform Symphony services will not conflict with existing business applications.

Additional enhancements enable administrators to identify key processes related to important applications so that Platform Symphony will immediately stop harvesting cycles when these processes are present on a server that is a candidate for harvesting.

Refer to *Host Scavenging Feature* on page 231 in the *Cluster and Application Management Guide*.

Multiple Repository Servers per cluster

As of Platform Symphony 5.1, multiple repository servers can be configured per cluster. This enables clusters that span geographies to load application services more quickly in parallel, while limiting network traffic over slow wide-area links. This feature also makes it faster to deploy new application services on large individual clusters by enabling compute hosts to load services in parallel.

Refer to *Multiple repository servers* on page 347 in the *Cluster and Application Management Guide*.

SSM physical memory boundary control

You are now able to link boundary events to the physical memory available for each application.

Refer to *Optimizing session manager performance* on page 434 in the *Cluster and Application Management Guide*.

Resource Management Improvements

Platform Symphony 5.1 enables administrators to limit the number of service instances that an application (or session type) can run on a host even if there are more slots available. These more granular controls provide the flexibility to have remaining slots used by other applications residing on compute hosts such as data caches.

Refer to *Limit the number of service instances that can run on a host* on page 421 in the *Cluster and Application Management Guide*.

New sharing policies in Platform Symphony make it easier for organizations with multiple consumers to express loaning and borrowing policies. This enables better utilization and a larger pool of apparent resources for each consumer.

Refer to *Hybrid scheduling policy* on page 161 in the *Cluster and Application Management Guide*.

Usability Enhancements

Graceful host removal

You are now able to remove dynamic hosts from the cluster using the Platform Management Console or CLI. To remove a host, it must be either unavailable or closed without running workload.

Refer to *egosh resource removehost* command on page 42 in the *Symphony Reference* document or *PMC help*.

Comments when closing hosts

It is now possible to add a comment when closing a host from the Platform Management Console or CLI. Use the `EGO_HOST_COMMENT_ENFORCE` parameter in *ego.conf* to enforce mandatory comments during host close.

Refer to the *egosh resource close -c* command and the *EGO_HOST_COMMENT_ENFORCE* parameter on page 38 and 256, respectively, in the *Symphony Reference* document.

Auto-select most recent Sym version to run on

You can now ensure that an application always runs on the latest version of Symphony middleware available in the cluster. A version of Symphony is considered to be available if it is the only version in the cluster or, in the case of multiple Symphony versions in the cluster, it was at some time activated during a direct upgrade.

Refer to *Enable an application to always use the latest Symphony version* on page 359 in the *Cluster and Application Management Guide*.

Package dependencies

Platform Symphony 5.1 provides enhancements for modular applications that share common components. By enabling complex applications to be expressed as collections of smaller packages with defined dependencies, applications become easier to manage and maintain. Disk space is also conserved on repository servers, and deployment speed is enhanced because only altered components need be distributed to compute hosts when updates are made.

Refer to *Managing service package dependencies* on page 348 in the *Cluster and Application Management Guide*.

New Platform Support

- Windows 7
- Windows Server 2008 R2
- Windows HPC Server 2008 R2
- Microsoft Visual Studio 2010
- Microsoft .NET 4.0 runtime
- RHEL 6

What's Changed?

ISF Adaptive Cluster for Symphony

Symphony 5.1 does not support ISF Adaptive Cluster (AC) versions 1.2 and lower. An ISF AC upgrade is necessary.

Error and log messages

Error messages

The following error messages are either new or have been updated to improve debuggability.

Message ID	Message
20084	{"Domain <SOAM>: Unable to contact the Symphony system at \"{0}\". It is likely that there is either a network problem or the SOA Middleware is not running. Ensure that the SOA Middleware services have been started. It is also possible that the session manager servicing this application has exceeded its number of client and service connections and has rejected the client connection. Check the session manager log to confirm this issue and try connecting again at a later time when the session manager is not as busy. Additional Details: system error code <{1}>, system error description <{2}>."}
20109	{"Domain <Application> : The value \"{0}\" specified for the configuration parameter \"{1}\" is not valid. The parameter value must be {2}. The default value \"{3}\" will be used instead."}
25274	{"Check your session ID. If you do not see any session information, check the parameter persistSessionHistory in your application profile. Other possible reasons are: the time specified for saving historical information with lastingPeriod has expired; the app was disabled; the session information may be temporarily unavailable while writing history; or you have no permission to view this session information."}
25310	{ "The version of SOAM CLI or WSDL client is not supported by Session Director (SD) in the cluster. The unsupported namespace is <{0}> in SOAP element <{1}>. Use CLI that is supported by SD in the cluster, or specify a different namespace in the WSDL client." }
32138	{"Domain <SOAM>: Service instance manager {0} on host {1} for application {2} could not be started. A new resource will be automatically requested. Check your application profile: SOAM_HOME for SIM may not be correctly configured."}
32140	{ "Domain <Application>: The value <{0}> configured for boundary event <{1}> of boundary element <{2}> is not valid. The value of a boundary event must be a non-negative number. Correct the value of the boundary event and re-register the application profile."}
32141	{ "Domain <Application>: The value <{0}> configured for boundary param <{1}> of boundary element <{2}> is not valid. The value of a boundary param must be a non-negative number. Correct the value of the boundary param and re-register the application profile."}
33017	{ "Domain <VEM>: Failed to verify path for consumer {0} using resource group \"{1}\" , reason: {2}." }
50142	{"Permission denied. Only users with roles <{0}> or permissions <{1}> can perform operation <{2}>. Check your role on consumer <{3}>."}

Message ID	Message
75065	{ "Application <{0}>: Host <{1}> is blocked due to the service instance manager timed out on process startup and attribute blockHostOnTimeout is set to true in application profile." }
75080	{ "Application <{0}>: Host <{1}> is blocked due to the service instance manager cannot startup successfully and attribute blockHostOnTimeout is set to true in application profile." }
85095	{"Domain <EGO>: Cannot read the file C:\EGO50\eservice\rs\cache\SampleApplications\SOASamples\SampleService.pkg\SampleServiceCPP.zip into allocated memory for the requested package on the file system."}
85113	{ "Domain <EGO>: The package <{0}> is not deployed under consumer / or <{1}>. With the short consumer name feature, only packages deployed under root node and leaf nodes can be used. Use soamdeploy utility or PMC to add the package to the service repository." }
95519	{ "Domain <EGO>: Failed to start the FAM agent process on <{0}>. Allocation failed: <{1}>. Client name: <{2}>. Make sure the user is cluster admin, consumer user, or consumer admin on consumer <{3}>, then retry the request." }

Log messages

1. The out-of-box log level configuration for EGO daemons has changed from LOG_WARNING to LOG_NOTICE in ego.conf. As a result, the NOTICE level messages will appear in EGO log files, using the out-of-box log level configuration.
2. The level of the Version String message in EGO logs has changed from LOG_WARNING to LOG_NOTICE.

Configuration

- The default values for EGO_ADJUST_SHARE_TO_WORKLOAD and EGO_RECLAIM_FROM_SIBLINGS in ego.conf have changed from "N" to "Y".
- EGO_KD_CLIENT_PORT_RANGE has been added to ego.conf to control the range of used local ports.
- EGO_MINIMUM_SLOTS_ALERT has been added to ego.conf.
- EGO_HOST_COMMENT_ENFORCE has been added to ego.conf. if the parameter is configured as Y, EGO will reject "close host" if a comment is not attached.
- The default value for MEM_HIGH_MARK in wsm.conf has been changed to 1536.
- Two new host types have been added to the ego.shared file: SOL1032 and SOLX86.
- A new client env variable SOAM_CONNECT_TIMEOUT has been introduced to allow the client more time to establish a connection with Symphony.

Commands

CLI

- "egoconfig mgghost <share>" now copies all service xml files to share.
- "egosh resource view" output has been changed to support a new "acu" column. This column represents Adjusted CPU Utilization for desktop or server scavenging.
- "egosh alloc unblock" has been improved including the ability to specify short host names.

What's Changed?

- The SI Startup Failures field in the output of the soamview app command now shows the PID of the service instance.
- "egosh consumer view" output format has changed:

Symphony 5.0

```
Policy:  
ResourceGroupName: RG  
Priority: 0  
Hybrid Owned Slots: 15 Hybrid Reserve: 10 Hybrid Limit: 15  
Hybrid Quota: 10
```

Symphony 5.1

```
Policy:  
ResourceGroupName: RG  
Priority: 0  
Hybrid Owned Slots: 15  
Hybrid Reserve: 10  
Hybrid Limit: 15  
Hybrid Quota: 10
```

PERF

- A new parameter "Cipher" has been added to datasource.xml.
- The length of LAST_FAILURE_REASON in TASK_ATTRIBUTES has been increased from 256 to 2048 characters.
- New Java options have been added: -DCLOSE_PLC_REGISTER_THREAD and -DPERF_DB_TIMEOUT
- The following text was added to the "RESOURCE_GROUP" column in the "CONSUMER_RESOURCE_ALLOCATION" schema table: "Note that "all" is a reserved value signifying all resource groups that are associated with the consumer."
- A new database table named 'CONSUMER_PLAN_NON_LEAF' has been created to store allocation info for non-leaf nodes.

Scavenging

- The initial state of a scavenging host will always be closed.
- ELIM_RUNINACTIVESESSION_GROUP=elim.sa is set as default in Windows Vista/7/2008.
- Behavior improvement when user idle time threshold is set to 0. Now, the server scavenging host will be closed as long as the CPU is considered busy.
- CLI command will be rejected if a non-valid value is given for scavenging parameter

Audit log format

The following EGO audit log formats have been added:

DATE/TIME	TYPE	USER	OBJECT	ID	ACTION	DETAIL
<i>time_stamp</i>	CONTROL	<i>user_name</i>	HOST	<i>host_name</i>	removed	-
<i>time_stamp</i>	CONTROL	<i>user_name</i>	HOST	<i>host_name</i>	remove_failed	<i>error_msg</i>

PMC

- The "relative allocation" figure on the "Monitor Resource Allocation" page no longer includes slots from the "InternalResourceGroup" group.
- "Authentication failed because you are already logged in with account:<Admin>. Log out of <Admin> first and then log in with a different account."
- The PMC now supports finding session names by wildcards '*' and '?'.
- In `pmc_ego_conf.xml`, you can now add parameter `ManagementResourceGroup` to define the resource groups belonging to management allocations.
- PMC behavior has been improved. Now, if a browser is open and logged on to the PMC, opening a new browser will bypass the log on and go directly to the dashboard.

Applications

- `VERSION_NUM` can no longer be configured as an environment variable in an application profile
- Messages have been updated for service package dependencies.
- A namespace has been added to the `Deployment` element in the `deployment.xml` file:
`<Deployment xmlns="http://www.platform.com/Symphony/Deployment">`
- Applications and service packages can now be removed if the consumer does not exist.

Known Issues

Antivirus software can block Symphony daemons

If you install Symphony on a host that is running antivirus software such as ESET NOD32, the following Symphony daemons may get quarantined, preventing the cluster from starting up.

- wsm
- wsgserver
- vemkd
- egosh
- pem

Restoring the quarantined daemons allows the cluster to start up.

Memory control on Solaris management hosts

The SSM on Solaris management hosts cannot trigger lower level boundary events for process memory.

EGO start-up on Windows Vista and later OS

EGO reports "permission deny" after running "egosh ego start" to start Symphony. This is due to the User Account Control (UAC) on Vista and later operating systems such as Windows 2008. As a workaround, run "egosh ego start" in a command window that has administrator permission.

Connector for Excel upgrade

If you have Symphony 5.1 installed, it is recommended that you deploy and use the 5.1 version of the Excel Connector service.

If you want to run version 4.x of the Excel Connector service in a 5.1 cluster, you must do one of the following:

- Copy the `com_platform_log4cxx_097_1.dll` file to the `5.1lib` directory.
- Repackage the 4.x Excel Connector service with `com_platform_log4cxx_097_1.dll` and redploy it.

If you want to run version 5.0 of the Excel Connector service in a 5.1 cluster, you must do one of the following:

- Copy the `com_platform_log4cxx_097_2.dll` file to the `5.1lib` directory.
- Repackage the 5.0 Excel Connector service with `com_platform_log4cxx_097_2.dll` and redeploy it.

Wildcard is not supported when querying tasks by task tag

The PMC and CLI do not support wildcard character "*" when querying tasks by task tag.

PMC cannot display multi-byte characters

The use of multi-byte characters causes an invalid byte error in the PMC. To avoid this error, limit text to valid characters only.

Migrating your .NET applications from a previous release of Symphony

A .NET application built using Symphony 5.0 or earlier versions will not automatically run in a Symphony 5.1 environment. This is a known issue caused by the .NET embedded "Publisher's Policy" infrastructure that is strictly enforced by the framework. Any published assembly with a version number that is different from your current Symphony release will be treated as incompatible by the framework. This incompatibility can be resolved in one of two ways:

1. Recompile your application using the Symphony 5.1 SDK. By recompiling, your application will automatically reference the new version of the assembly and will run without problems. This is the best option and is the strategy recommended by Microsoft for maximum compatibility.
2. Create a configuration file for your client and service to override the "Publisher's Policy". Since the Symphony 5.1 assembly is designed to be fully compatible with code using a previous version, it is safe to override the publisher's policy and allow the framework to ignore the version number difference between Symphony 5.1 and your application's dependence on a previous release of Symphony.

The override can be achieved by creating a file named <complete executable filename>.config. For example, if your client executable has the name "SyncClient.exe" then you would need to create a file named "SyncClient.exe.config" and place it in the same location as the executable. The .NET framework will automatically parse the file and use it to override the "Publisher's Policy". Here is an example of the .config file contents:

```
<?xml version="1.0"?>
<configuration>
  <runtime>
    <assemblyBinding xmlns="urn:schemas-microsoft-com:asm.v1">
      <publisherPolicy apply="no"/>
      <dependentAssembly>
        <assemblyIdentity name="Platform.Symphony.Soam.Net"
          publicKeyToken="ebe3ab0caa75a162" culture="neutral" />
        <bindingRedirect oldVersion="5.0.0.0"
          newVersion="5.1.0.0"/>
      </dependentAssembly>
    </assemblyBinding>
  </runtime>
</configuration>
```

If your service is to run in the Symphony 5.1 environment, you need to add the .config file to the service package and redeploy the package.

Message "Could not load file or assembly 'Platform.Symphony.Soam.Net_64.dll' displays when trying to run .NET application

This issue is related to VS2010's multi-targeting feature, which was first introduced with VS2008 and .NET Framework 4.0.

What is multi-targeting

With the past few releases of Visual Studio, each Visual Studio release only supported a specific version of the .NET Framework. For example, VS 2002 only worked with .NET 1.0, VS 2003 only worked with .NET 1.1, and VS 2005 only worked with .NET 2.0.

Starting with VS 2008, MS supported the "multi-targeting" feature, which means that Visual Studio now supports targeting multiple versions of the .NET Framework, and developers are able to take advantage of the new features Visual Studio provides without having to always upgrade existing projects and deployed applications in order to use a new version of the .NET Framework library.

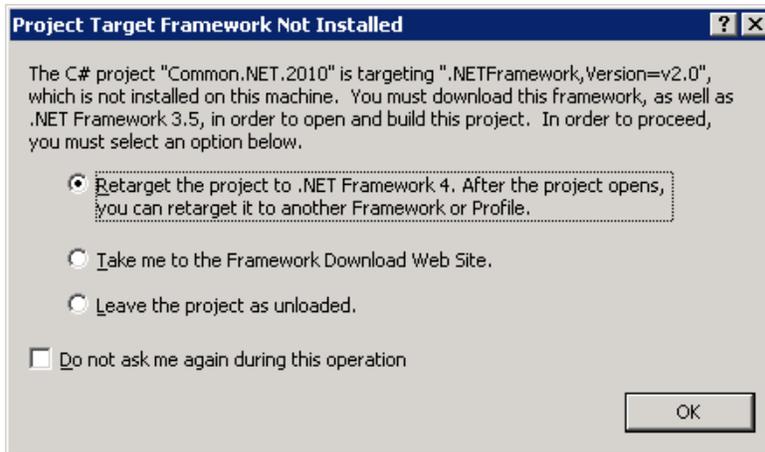
Now when you open an existing project or create a new one with VS 2008, you can choose which version of the .NET Framework to work with, and the IDE will update its compilers and feature-set to match this. Among other things, this means that features, controls, projects, item-templates, and assembly references that do not work with that version of the framework will be hidden, and when you build your application, you'll be able to take the compiled output and copy it onto a machine that only has an older version of the .NET Framework installed knowing that the application will work.

When Symphony MS compiler support extended to VS2010, this feature became a little tricky because there are some default behavior changes between VS2008 & VS 2010. VS2008, by default, can pick up the correct dependency libraries even though a C# binary uses a higher version of .NET Framework than its dependent libraries; VS2010, by default, does not do it for .NET 4.0. The details follow.

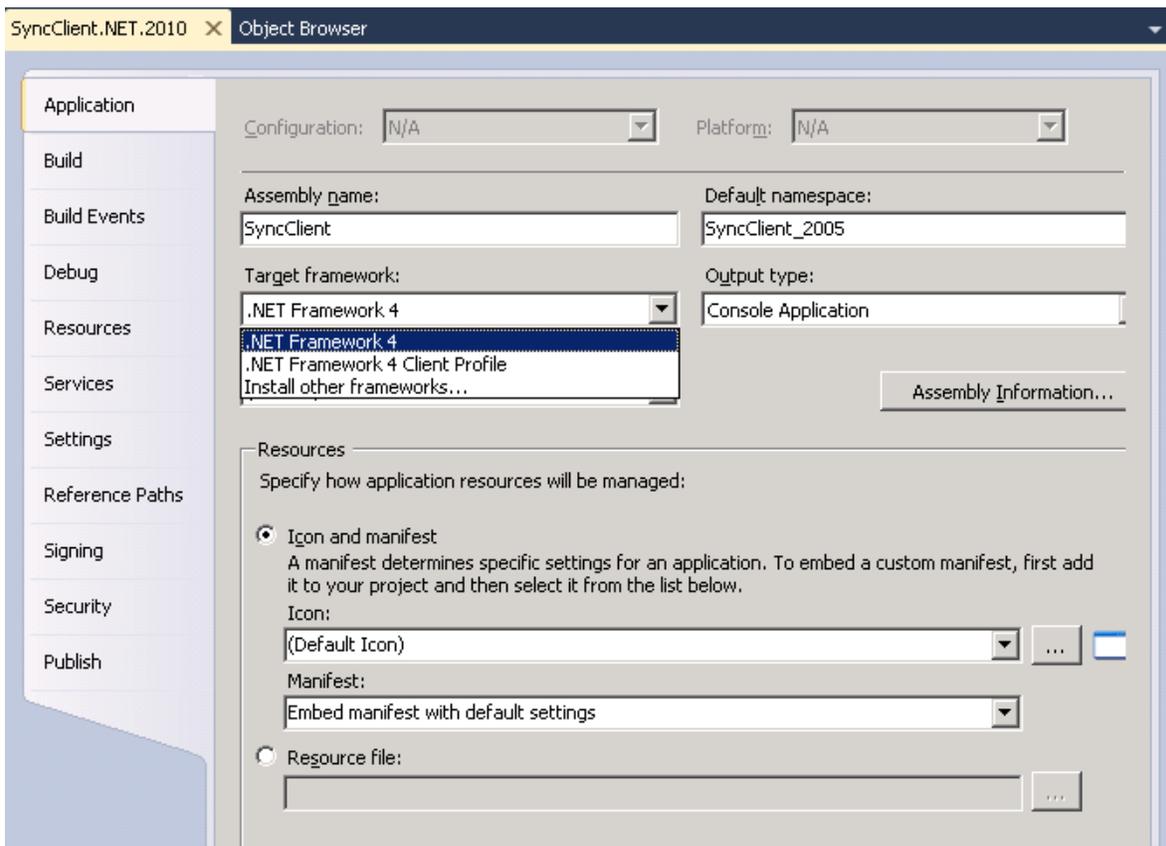
Which case does Symphony support for VS2010?

Case 1: Compile the Symphony C# sample in a host where only .NET Framework 4.0 is installed

Not only 64bit, but when you open the 32bit C# sample with VS2010 in a host where only .NET Framework 4.0 is installed, there will be a dialog to show the multi-target issue. The reason is on this host, only .NET Framework 4.0 is installed, regardless of whether the Framework comes from the separate Framework install package or from the VS2010's built-in package.



Visual Studio 2010 includes only .NET Framework 4.0. To target earlier versions of the .NET Framework, you must have the .NET Framework 3.5 Service Pack 1 (SP1) installed. The .NET Framework 3.5 SP1 includes .NET Framework 2.0, .NET Framework 3.0, and .NET Framework 3.5. On this host, we will only see the dialog shown below.



For .NET Framework 4.0, when the target execute binary has an earlier version dependency, like Symphony SDK Platform.Symphony.Soam.Net.dll that is built from .NET Framework 2.0, the following exception will happen during runtime:

Could not load file or assembly 'Platform.Symphony.Soam.Net_64.dll' or one of its dependencies. A dynamic link library (DLL) initialization routine failed. (Exception from HRESULT: 0x8007045A)

Note:

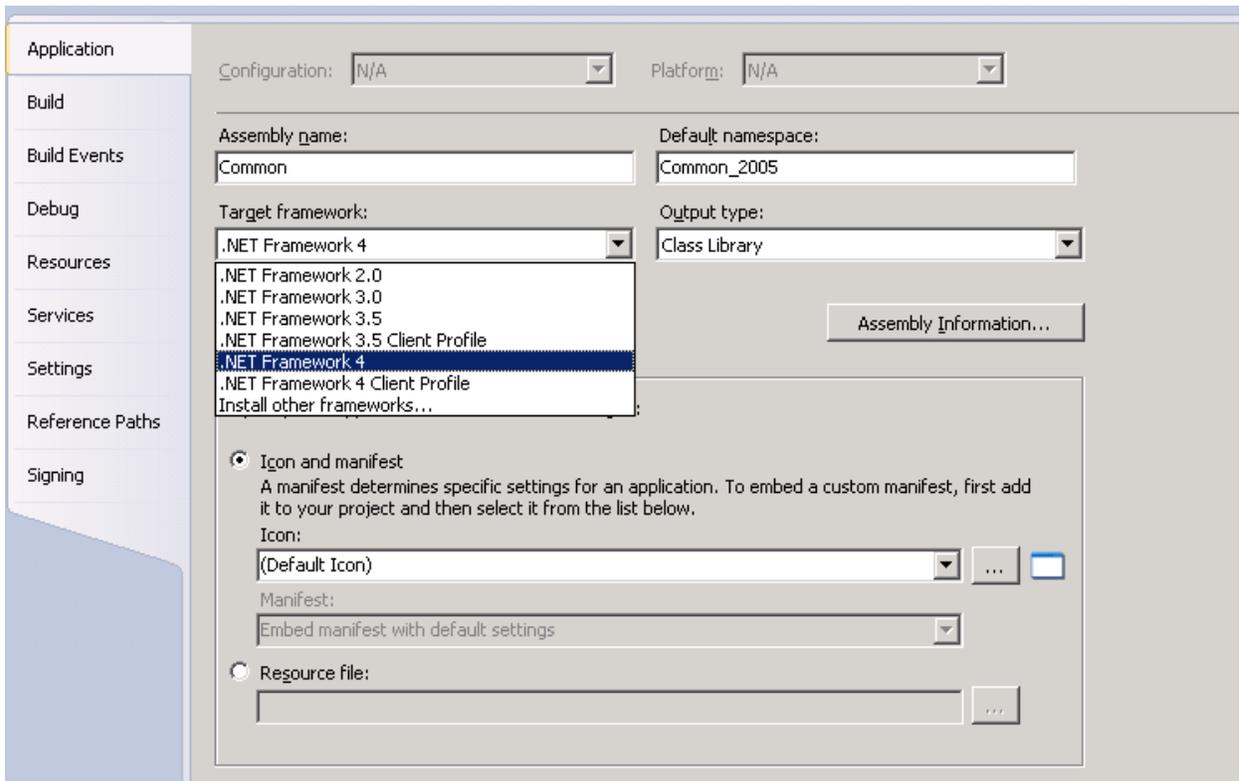
The message content may specify a different dll name, etc for a different Framework version & host type.

The workaround is to add an optional attribute named "useLegacyV2RuntimeActivationPolicy" into a config file. If we choose .NET Framework 4.0 as the target framework to build the binaries, the {exename}.config file will be generated automatically by VS2010, e.g., SyncClient.exe.config; we can then add the attribute to it. For example:

```
<?xml version="1.0"?>
<configuration>
  <startup useLegacyV2RuntimeActivationPolicy="true">
    <supportedRuntime version="v4.0"
      sku=".NETFramework,Version=v4.0"/>
  </startup>
</configuration>
```

Case 2 : Compile Symphony C# sample in a host where multiple .NET frameworks are installed

If a host has .NET Framework 3.5 or other earlier version (like 2.0, 3.0) installed, the multi-target setting will look like the following:



If we choose to build with .NET Framework 4.0, since Symphony SDK Platform.Symphony.Soam.Net.dll is built from .NET Framework 2.0. the following exception will happen during runtime:

Unhandled Exception: System.IO.FileLoadException: Mixed mode assembly is built against version 'v1.1.4322' of the runtime and cannot be loaded in the 4.0 runtime without additional configuration information at Platform.Symphony.Samples.SyncClient.Main(String[] args)

Note:

The message content may specify a different dll name, etc for a different Framework version & host type.

The workaround is to add an optional attribute named "useLegacyV2RuntimeActivationPolicy" into a config file. If we choose .NET Framework 4.0 as the target framework to build the binaries, the {exename}.config file will be generated automatically by VS2010, e.g., SyncClient.exe.config; we can then add the attribute to it. For example:

```
<?xml version="1.0"?>
<configuration>
  <startup useLegacyV2RuntimeActivationPolicy="true">
    <supportedRuntime version="v4.0"
      sku=".NETFramework,Version=v4.0"/>
  </startup>
</configuration>
```

If you choose .NET Framework 2.0, 3.0, or 3.5 to build our sample with VS2010, no extra config file needs to be updated.

For VS2008, since it only supports to 3.5 SP1, you do not encounter this issue even though we select multiple versions of the .NET Framework.

Reference

Multiple targeting overview: <http://msdn.microsoft.com/en-us/library/bb398197.aspx>Troubleshooting .NET

Framework Targeting Errors: <http://msdn.microsoft.com/en-us/library/cc668079.aspx>How to: Target a Specific

.NET Framework Version or Profile: <http://msdn.microsoft.com/en-us/library/bb398202.aspx>

<startup> Element: <http://msdn.microsoft.com/en-us/library/bbx34a2h.aspx>