

IBM “External Data Transfer and Web Start” Readme File

About the “External Data Transfer and Web Start” feature

This package contains the “External Data Transfer and Web Start” feature, which allows the Java client to submit workload using JWS without the Platform Symphony environment. This feature supports JWS work with the “Symphony External Task Output Transfer” (RFC4020) and “Common data chunking” (RFC4075) features.

Readme file for: IBM® Platform Symphony

Product/Component Release: 5.2

Update Name: Fix pack

Fix ID: sym-5.2-build233081-frmc

Publication date: 16 April 2014

Last modified date: 16 April 2014

1. Scope.....	3
2. Configuration.....	4
a. Prerequisites.....	4
b. Installation files.....	5
c. Configuration procedure.....	5
d. Verification Procedure.....	9
3. Usage.....	9
a. How this feature works.....	9
b. Use cases.....	9
Case 1: Enable this feature and RFC4020 on Windows 64-bits.....	9
Case 2: Enable this feature and RFC4075 on Windows 32-bits.....	10
Case 3: Client API debug level.....	10
4. Troubleshooting.....	11
5. Copyright and trademark information.....	11

1. Scope

Applicability		
Operating system		Windows 32-bit and Windows 64-bit
Java version		Oracle JRE 1.6
Symphony version		Symphony 5.2
Cluster types		N/A
Apache version	tomcat	6.0.37
Browser		Symphony 5.2 supported Internet Explorer and Firefox
Dependencies		
File system		This feature has no requirements on the file system type.
Signed Applications	Java	All Java binaries, including Symphony Java client applications and Symphony runtime libraries must be signed with jarsigner to run in the JWS framework.
Symphony Environment		The feature is designed to launch a Symphony Java Client in JWS from a browser client, where the Symphony environment is not installed.
Limitations		
Limitation		<ol style="list-style-type: none">1. Does not support other clients except Java.2. Does not support other Symphony security plug-ins except the AD security plug-in.3. Does not support this feature when installed on the Symphony host.4. Does not support DDT work with RFC4020 or RFC4075.5. Does not support MR job.6. Does not support the feature that sets the client environments.7. Do not deploy the client in the Symphony GUI Tomcat server. Note: Do not use Symphony GUI Tomcat as the server.8. Does not support other JREs except Oracle.9. Reference RFC4020 and RFC4075 out-of-scope.

2. Configuration

a. Prerequisites

- Oracle JRE 1.6 installed
- Apache Tomcat 6.0.x
- Symphony 5.2 supported Internet Explorer 9 or Firefox 12
- The JNLP file supports the following properties for Platform Symphony:

Property	Description		Related feature
SOAM_ENABLE_JWS	Enable JWS feature.	Required	RFC4570 RFC4608
EGO_MASTER_LIST	The list of hosts that are master host candidates in the cluster to which you want to connect.	Required	Cluster information
EGO_KD_PORT	Specifies the port number to use to connect to the Symphony cluster.	Required	Cluster information
EGO_SEC_PLUGIN	Specifies the security mechanism to use when connecting to the Symphony cluster.	Required	Cluster information
DDT_WORK_DIR	Stores the DDT input/output data temporarily.	Optional	DDT(patch #232219)
SOAM_DIRECT_DATA_PORT	Defines a port or port range for the client to listen for connections from the service.	Optional	DDT
SOAM_DIRECT_DATA_ADDRESS	Allows a non-default interface on the client host to be specified for communication with the service instance.	Optional	DDT
SOAM_DIRECT_DATA_STORAGE	Defines whether the data resides in client memory or is written to disk.	Optional	DDT
EGO_CLIENT_ADDR	Used with firewalls on the client side and the <code>rfa</code> command. If the SOAM_DIRECT_DATA_PORT is not defined, Symphony will use the value defined in EGO_CLIENT_ADDR. If neither of these variables is defined, Symphony randomly selects a client port to listen on.	Optional	RFC4020 DDT
SOAM_COMMON_DATA_CHUNK_SIZE	The size of each common data chunk.	Optional	RFC4075
SOAM_CLIENT_PARALLEL_FETCH_NUM	The maximum of parallel fetching in the client.	Optional	RFC4020
SOAM_FAM_CONSUMER	The consumer used to start FAM.	Optional	RFC4020

SOAM_RFA_RETRY_LIMIT	The fetching task output retry limit. .	Optional	RFC4020
SOAM_RFA_TIMEOUT	The timeout in seconds for RFA to retrieve the task output.	Optional	RFC4020

b. Installation files

This RFC includes the following files:

File name	Description
JavaSoamApi.jar_w2k3-x64	Windows 64-bit Symphony Java API library
JavaSoamApi.jar_w2k-x86	Windows 32-bit Symphony Java API library
jnativesoamapi_5.2.0_64.dll	Windows 64-bit Java native lib
jnativesoamapi_5.2.0.dll	Windows 32-bit Java native lib
libvem.dll_win2003-x64	Windows 64-bit Symphony VEM client library
libvem.dll_w2k-x86	Windows 32-bit Symphony VEM client library
soambase.dll_w2k3-x64	Windows 64-bit Symphony client library
soambase.dll_w2k-x86	Windows 32-bit Symphony client library

c. Configuration procedure

1. Prepare the 64-bit Symphony Java Client Library.

A 64-bit Java client launched from a browser in JWS requires a 64-bit Symphony Java client library in the JWS framework to load all dependencies.

The following table lists the runtime dependencies required by a 64-bit Symphony Java client. All the files listed in the table must be included in the 64-bit Symphony Java client library.

File List	File location Symphony Cluster Installation Directory
-----------	---

com_platform_ACE_544_02.dll com_platform_ACE_544_02_SSL.dll com_platform_log4cxx_097_3.dll icudt32.dll icuin32.dll icuuc32.dll jnativesoamapi_5.2.0_64.dll libeay32.dll libevent.dll libsec.dll soam_resources.dll ssleay32.dll	<dir>\soam\5.2\w2k3_x64-vc7-psdk\lib64
rfa.exe	<dir>\1.2.6\bin
rfa.dll	Provided by RFC4020
libxml2.dll sec_ego_default.dll sec_ego_ext_co.dll xml4c_5_8.dll zlib.dll	<dir>\1.2.6\lib
soambase.dll libvem.dll jnativesoamapi_5.2.0_64.dll	provided by this RFC
api.log4j.properties	<dir>\soam\conf

Copy the files in the list above into a directory, for example, c:\soamlib64, and pack all the files into a soamlib64.jar. Ensure soambase.dll, jnativesoamapi_5.2.0_64.dll, and libvem.dll are from this patch delivery:

```
C:\soamlib64>jar cvf soamlib64.jar com_platform_ACE_544_02.dll
com_platform_ACE_544_02_SSL.dll com_platform_log4cxx_097_3.dll
icudt32.dll icuin32.dll icuuc32.dll jnativesoamapi_5.2.0_64.dll
libeay32.dll libevent.dll libsec.dll soam_resources.dll ssleay32.dll
xml4c_5_8.dll zlib.dll libvem.dll libxml2.dll sec_ego_default.dll
soambase.dll sec_ego_ext_co.dll rfa.dll rfa.exe api.log4j.properties
```

2. Prepare the 32-bit Symphony Java Client Library.

A 32-bit Java client launched from a browser in JWS requires a 32-bit Symphony Java client library in the JWS framework to load all dependencies.

The following table lists the runtime dependencies required by a 32-bit Symphony Java client. All the files listed in the table must be included in the 32-bit Symphony Java client library.

File List	File location Symphony Cluster Installation Directory
com_platform_ACE_544_02.dll com_platform_ACE_544_02_SSL.dll com_platform_log4cxx_097_3.dll icudt32.dll icuin32.dll icuuc32.dll libeay32.dll libevent.dll libsec.dll msvc71.dll soam_resources.dll ssleay32.dll xml4c_5_8.dll	<dir>\soam\5.2\win32-vc7\lib
iconv.dll libxml2.dll sec_ego_default.dll zlib1.dll sec_ego_ext_co.dll	<dir>\1.2.6\lib
rfa.exe	<dir>\1.2.6\lib\bin
rfa.dll	Provided by RFC4020
soambase.dll jnativesoamapi_5.2.0.dll libvem.dll	Provided by this RFC
api.log4j.properties	<dir>\soam\conf

to a directory, for example, c:\soamlib32, and pack all the files into a soamlib32.jar. Ensure soambase.dll and libvem.dll are from this patch delivery:

```
C:\soamlib32>jar cvf soamlib32.jar com_platform_ACE_544_02.dll
com_platform_ACE_544_02_SSL.dll com_platform_log4cxx_097_3.dll
icudt32.dll icuin32.dll icuuc32.dll jnativesoamapi_5.2.0.dll
libeay32.dll libevent.dll libsec.dll msvc71.dll soam_resources.dll
ssleay32.dll xml4c_5_8.dll iconv.dll libvem.dll libxml2.dll
sec_ego_default.dll zlib1.dll soambase.dll sec_ego_ext_co.dll
rfa.dll rfa.exe api.log4j.properties
```

3. Use jarsigner to sign all libraries to run in JWS:

For 64-bit Java applications, use jarsigner to sign:

- Your Symphony Java client
- soamlib64.jar created in step 1 above
- JavaSoamApi.jar renamed from JavaSoamApi.jar_w2k3-x64 in this patch
-

For 32-bit Java applications, use jarsigner to sign:

- Your Symphony Java client
- Soamlib32.jar created in Step 2. above
- JavaSoamApi.jar renamed from JavaSoamApi.jar_w2k-x86 in this patch

4. Copy the signed files into the directory to be used by your Tomcat server, for example, a "soamclients" directory.

5. Configure the following attributes in the "resources" section in the .jnlp file used to launch the Symphony client in JWS. Refer to Section 2.1 to get the Symphony supported properties.

```
<?xml version="1.0" encoding="utf-8"?>
<!-- JNLP File -->
<jnlp
  spec="1.0+"
  codebase="http://IP:Port/jwstest/apps"
  href="jwstest.jnlp">
  <information>
    <title>jws test</title>
    <vendor>test</vendor>
    <homepage href="jwstest.html"/>
    <description>ws test</description>
    <icon href="images/jwstest.jpg"/>
    <offline-allowed/>
  </information>
  <security>
    <all-permissions/>
  </security>
  <resources>
    <j2se version="1.6"/>
    <nativelib href="lib/soamlib.jar"/>
    <jar href="lib/asyncclient.jar"/>
    <jar href="lib/JavaSoamApi.jar"/>
    <property name="SOAM_ENABLE_JWS" value="Y"/>
    <property name="EGO_MASTER_LIST" value="mastername"/>
    <property name="EGO_KD_PORT" value="7870"/>
    <property name="EGO_SEC_PLUGIN" value="sec_ego_ext_ad"/>
    <property name="DDT_WORK_DIR" value="c:\DDT"/>
    <property name="SOAM_DIRECT_DATA_PORT" value="28888"/>
  </resources>
  <application-desc
    class="com.platform.symphony.samples.SampleApp.client.AsyncClient"/>
  </jnlp>
```

Note:

- Use the corresponding soamlib32.jar or soamlib64.jar according to your client type.

- Use the correct `JavaSoamApi.jar_w2k3-x64` or `JavaSoamApi.jar_w2k-x86` according to your client type.
- Property `SOAM_ENABLE_JWS` must be configured to `Y` to indicate the JWS environment.
- `EGO_MASTER_LIST` and `EGO_KD_PORT` must be configured to connect to the cluster where the application is running.

6. Restart Tomcat.

d. Verification Procedure

Run the Symphony Java Client in JWS from a browser client.

3. Usage

a. How this feature works

All client environments must be defined as property in the JNLP file. Refer to Section 2.1 for the Symphony supported properties in the JNLP file.

“Symphony External Task Output Transfer” and “Common data chunking” are not supported in DDT. If you enable “Symphony External Task Output Transfer” or “Common data chunking”, this RFC will not support DDT. This RFC can work with DDT without “Symphony External Task Output Transfer” and “Common data chunking”.

b. Use cases

Case 1: Enable this feature and RFC4020 on Windows 64-bits

Summary:

With this feature enabled, the client can submit Java workload using JWS without the Platform Symphony environment. Note: Enable the “Symphony External Task Output Transfer” feature.

Pre-conditions:

1. Symphony 5.2 Java client application.
2. Web server that runs Apache Tomcat 6.0.37 and Java 1.6.
3. AD security plug-in enabled for the Symphony cluster.

Scenario:

1. Enable RFC4020. Note: Refer to the RFC4020 readme file.
2. Prepare the 64-bit Symphony Java Client Library.
3. Use jarsigner to sign:
 - Your Symphony Java client
 - `soamlib64.jar`
 - `JavaSoamApi.jar` renamed from the `JavaSoamApi.jar_w2k3-x64` in this patch
4. Copy the signed files into the directory to be used by your Tomcat server, for example, a

“soamclients” directory.

5. Configure the RFC4020 client environment in the JNLP file.
6. Restart the Tomcat server.

Post-condition:

1. Run the Symphony Java Client in JWS from a browser client.
2. The client task output directory will be created in the JWS cache directory.
3. The workload is finished successfully.

Case 2: Enable this feature and RFC4075 on Windows 32-bits

Summary:

With this feature enabled, the client can submit Java workload using JWS without the Platform Symphony environment. Note: Enable the “Common data chunking” feature.

Pre-condition:

1. Symphony 5.2 Java client application.
2. Web server that runs Apache Tomcat 6.0.37 and Java 1.6.
3. AD security plug-in enabled for the Symphony cluster.

Scenario:

1. Enable RFC4075. Note: Refer to the RFC4075 readme file.
2. Use jarsigner to sign:
 - Your Symphony Java client
 - Soamlib32.jar
 - JavaSoamApi.jar renamed from JavaSoamApi.jar_w2k-x86 in this patch
3. Copy the signed files into the directory to be used by your Tomcat server, for example, a soamclients directory.
4. Configure the RFC4075 client environment in the JNLP file.
5. Restart the Tomcat server.

Post-condition:

1. Run the Symphony Java Client in JWS from a browser client.
2. The workload is finished successfully.

Case 3: Client API debug level

Summary:

With this feature enabled, the client can open the debug API log.

Pre-condition:

1. Symphony 5.2 Java client application.
2. Web server that runs Apache Tomcat 6.0.37 and Java 1.6.
3. AD security plug-in enabled for the Symphony cluster.

Scenario:

1. Modify the client API log level in api.log4j.properties.

2. Prepare the Symphony Java Client Library.
3. Use jarsigner to sign:
 - Your Symphony Java client
 - soamlib64.jar
 - JavaSoamApi.jar in this patch
4. Copy the signed files into the directory to be used by your Tomcat server, for example, a `soamclients` directory.
5. Restart the Tomcat server.

Post-condition:

1. Run the Symphony Java Client in JWS from a browser client.
2. The client API log is found in the Java work directory (desktop by default).

4. Troubleshooting

When the value of property about RFC4020 and RFC4075 is invalid, `SoamException` will be thrown. To see the error message in the JWS client, you must catch and handle it, for example, write it to a file.

5. Copyright and trademark information

© Copyright IBM Corporation 2014

U.S. Government Users Restricted Rights - Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

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