Adobe Premiere Pro CS3 Troubleshooting

Legal notices

For legal notices, see http://help.adobe.com/en_US/legalnotices/index.html.

A note to our customers

Adobe provides this searchable PDF of archived technical support documents as a service to our customers who own and continue to enjoy older, unsupported versions of our software products. The information in these documents is not updated, and will become increasingly less accurate as hardware, browsers, and operating systems continue to evolve. Please be aware that these archived documents reflect historical issues and solutions for products that are no longer supported. Adobe does not warrant that the information in this document is accurate.

Contents

Error "Premiere Pro Debug Event" when you select Keyboard Customization (Adobe Premiere Pro CS3, Windows)

Issue

When you select Keyboard Customization in Adobe Premiere Pro CS3 for Windows, the application returns the error message, "Premiere Pro Debug Event. Premiere Pro has encountered an error. [..\..\Src\ShortcutListItem.cpp-239]" followed by the error message, "Sorry, a serious error has occurred that requires Adobe Premiere Pro to shut down. We will attempt to save your project."

Reason

Your system contains a large number of fonts.

Solution

Reduce the number of fonts in the Fonts folder

1   Create a temporary folder on the Desktop or other hard disk location.

2   Navigate to the Windows\Fonts folder, typically installed to C:\Windows\Fonts.

3   Move unnecessary fonts to the temporary folder created in step 1.

4 Open Premiere Pro CS3 and select Edit > Keyboard Customization. If the problem reoccurs, repeat steps 2 and 3 to remove additional fonts. Continue this process until the error no longer occurs.**Note:** When removing unnecessary font from the Windows Fonts folder, be certain not to remove required Windows XP system fonts. See Adobe "TrueType fonts installed by Windows XP" (Technote 326402) for additional details.

Error"...the processor does not support the SSE2 instruction set" when you install (Adobe Premiere Pro CS3)

Issue

When you try to install Adobe Premiere Pro CS3, it returns the error message, "Adobe Premiere Pro CS3 cannot run on this computer because the processor does not support the SSE2 instruction set."

Solution

Make sure you install Adobe Premiere Pro CS3 on a computer that meets the minimum system requirements. For a list of system requirements, see the Adobe website at http://www.adobe.com/products/premiere/systemreqs/.

Additional Information

The SSE2 (Streaming SIMD Extensions 2) instruction set is included with Intel Pentium 4 and later processors, and with AMD Athlon 64 and Opteron and later processors.

Export from Adobe Media Encoder or Adobe Clip Notes doesn't start | Premiere Pro CS3 | Windows

Issue

When you use Adobe Media Encoder or Adobe Clip Notes to export a sequence in Adobe Premiere Pro CS3 on Windows, one of the following behaviors occurs:

• When you click OK in the Export Settings window, the Save File dialog box does not appear.

• When you click Save in the Save File dialog box, nothing happens. Or, the program returns the error "The above filename is invalid" or "The path does not exist."

Solution

If you are in the Save File dialog box, delete prohibited characters (see the table below) from the filename. Then click Save.

If you are in the Export Settings window, take these steps:

1 Close the Export Settings window.

2 In the Project panel, right-click the sequence and select Rename.

3 Delete any of the following characters from the sequence name:

<	Angle bracket (left)
>	Angle bracket (right)
*	Asterisk
\	Backslash
:	Colon
/	Forward slash
?	Question mark
"	Quotation mark
|	Vertical bar

4 Choose File > Export > Adobe Media Encoder or File > Export > Adobe Clip Notes.

5 Adjust the export settings and click OK.

6 In the Save File dialog box, do not add any invalid characters to the filename.

Additional Information

Adobe Premiere Pro CS3 tries to use the exact name of the active sequence as the default filename in the Save File dialog box. If the sequence name contains any of the characters that Windows prohibits in filenames, export doesn't start. Likewise, it doesn't start if you add a prohibited character to the filename in the Save File dialog box,

More Help topics

Error Message: Filename is Invalid or Cannot Contain Any of the Following Characters

JRun 4: Manually Configuring External Web Server Connectors

There may be times when it is necessary to manually configure JRun's external web server connector in order to establish the connection between the JRun server and the web site. This article will demonstrate the steps required for manually configuring a connection to an Apache, IIS and iPlanet web site.

JRun's Web Server Configuration tool (wsconfig.jar) uses JNDI to discover running JRun servers. If the wsconfig tool is unable to automatically locate a JRun server (perhaps while setting up a distributed system) you can still manually configure the connection by following the instructions in this article.

• Setup steps common to each web server

• Apache Configuration

- Microsoft IIS Configuration
- iPlanet 4 Configuration
- iPlanet 6 Configuration

Setup steps common to each web server

For these examples, we will assume the use of three JRun servers:

1 Server1 (on proxy port 51001 - connecting to Apache 2.x)

2 Server2 (on proxy port 51002 - connecting to IIS 5)

3 Server3 (on proxy port 51003 - connecting to iPlanet 4 or 6)

We will also assume that the **{jrun-root}** directory is C:\jrun4.

Step 1: Create the following directories under {jrun-root}\lib (one for each connected web site):

C:\jrun4\lib\wsconfig\1 (this folder will contain configuration information for Apache)

C:\jrun4\lib\wsconfig\2 (this folder will contain configuration information for IIS)

C:\jrun4\lib\wsconfig\3 (this folder will contain configuration information for iPlanet)

Step 2: Create a file named *wsconfig.properties* in the C:\jrun4\lib\wsconfig directory. The *wsconfig.properties* file contains the data that the Web Server Configuration tool uses to maintain web server configuration information. It should contain the following information (actual locations should reflect your specific installation directories):

#JRun Web Server Configuration File #Fri May 23 10:19:20 EDT 2003 1=Apache,C:/Apache Group/Apache2/conf,"","","" 1 .srv=localhost,"server1" 2 =IIS,1,false,"" 2 .srv=localhost,"server2" 3 =NES,C:/Netscape/Server4/https-servername/config 3 .srv=localhost,"server3"

Step 3 - Configure the Web Server(s).

 Apache Configuration:

1 Populate the C:\jrun4\lib\wsconfig\1 directory with the following files:

- jrunserver.store - create a file called jrunserver.store and enter the following information:

    proxyservers=127.0.0.1:51001

    (Notice that 51001 corresponds to the port assigned to Server1).

- mod_jrun20.so (extract the appropriate file for your operating system from C:\jrun4\lib\wsconfig.jar)

- **Note:**mod_jrun20.so supports Apache 2.x web servers. To connect to Apache 1.3.x, usemod_jrun.so.

2 Edit the Apache configuration file (located in**{apache-root}/conf/httpd.conf**) and add the following section at the end of the file:

3 # JRun Settings LoadModule jrun_module c:/jrun4/lib/wsconfig/1/mod_jrun20.so <IfModule mod_jrun.c> JRunConfig Verbose false JRunConfig Apialloc false JRunConfig Ssl false JRunConfig Ignoresuffixmap false JRunConfig Serverstore c:/jrun4/lib/wsconfig/1/jrunserver.store JRunConfig Bootstrap 127.0.0.1:51001 #JRunConfig Errorurl <optionally redirect to this URL on errors> AddHandler jrun-handler .jsp .jws </IfModule>

Note the following entries from the Apache configuration file, above:

- LoadModule jrun_module c:/jrun4/lib/wsconfig/1/mod_jrun20.so - points to the folder and file name of the Apache connector

- JRunConfig Serverstore c:/jrun4/lib/wsconfig/1/jrunserver.store - points to the location of the jrunserver.store file
- JRunConfig Bootstrap 127.0.0.1:51001 - corresponds to defined port for Server1

Microsoft IIS Configuration:

1 Extract the jrunwin32.dll from the C:\jrun4\lib\wsconfig.jar and copy it to C:\jrun4\lib\wsconfig\

2 Extract the jrun.dll from C:\jrun4\lib\wsconfig.jar and copy it to C:\jrun4\lib\wsconfig\2

3 Create a jrun.ini file in C:\jrun4\lib\wsconfig\2 . It should contain the following information:

verbose=false scriptpath=/JRunScripts/jrun.dll serverstore=C:/JRun4/lib/wsconfig/2/jrunserver.store bootstrap=127.0.0.1:51002 apialloc=false ssl=false ignoresuffixmap=false #errorurl=<optionally redirect to this URL on errors>

4 Create the jrunserver.store file in C:\jrun4\lib\wsconfig\2 Edit as follows:

proxyservers=127.0.0.1:51002

5 Using the IIS Management Console add a virtual directory named JRunScripts to the web site The Local Path property of the folder should point to**C:\jrun4\lib\wsconfig\2** and Execute Permissions must be set to "Scripts and Executables"

6 Add the JRunConnector Filter to the ISAPI Filter properties of the web site by:

- Right clicking on the web site
- Selecting the ISAPI Filters tab
- Click the Add button
- In the "Filter Name" text box, enter "JRun Connector Filter"
- In the "Executable" text box, browse to or enter"C:\jrun4\lib\wsconfig\2\jrun.dll"
- Restart the World Wide Web Publishing service (in Control Panel) and check the ISAPI Filter property for the web site. The JRun Connector Filter should show a high priority with a green arrow pointing upwards.

iPlanet 4 Configuration:

1 Extract the **jrun_nsapi35.so** file from the C:\jrun4\lib\wsconfig.jar file and copy it to theC:\jrun4\lib\wsconfig\3 folder.

2 Create a jrunserver.store file in the C:\jrun4\lib\wsconfig\3 folder. Edit it as follows:

proxyservers=127.0.0.1:51003

3 Add the following to the Init section of the obj.conf file for the iPlanet server (note each Init is a single line):

Init fn="load-modules" shlib="C:/JRun4/lib/wsconfig/2/jrun_nsapi35.dll" funcs="jruninit,jrunfilter,jrunservice" Init fn="jruninit" serverstore="C:/JRun4/lib/wsconfig/3/jrunserver.store" bootstrap="127.0.0.1:51003" verbose="false" apialloc="false" ssl="false" ignoresuffixmap="false"

4 Add the following to the Object name="default" section of the obj.conf file:

NameTrans fn="jrunfilter" serverstore="C:/JRun4/lib/wsconfig/3/jrunserver.store" ObjectType fn=type-by-exp exp=*.jsp type="jrun-internal/" ObjectType fn=type-by-exp exp=*.jws type="jrun-internal/" Service method=(GET|POST) type="jrun-internal/" fn="jrunservice"

5 Add the following to the end of the obj.conf file:

<Object name="jrun"> PathCheck fn="jrunfilter" Service fn="jrunservice" </Object>

6 Restart the web server.

iPlanet 6 Configuration:

1 Use steps 1 and 2 of iPlanet 4 configuration

2 Add the following to the object-default section of the obj.conf file:

   NameTrans fn="jrunfilter" serverstore="C:/JRun/lib/wsconfig/3/jrunserver.store" bootstrap="127.0.0.1:51003" ObjectType fn=type-by-exp exp=*.jsp type="jrun-internal/" ObjectType fn=type-by-exp exp=*.jws type="jrun-internal/" Service method=(GET|POST) type="jrun-internal/" fn="jrunservice"

3 Add the following to the end of the obj.conf file:

   <Object name="jrun"> PathCheck fn="jrunfilter" Service fn="jrunservice" </Object>

4 Add the following to the end of the magnus.conf file (note each Init is a single line):

   Init fn="load-modules" shlib="C:/JRun/lib/wsconfig/3/jrun_nsapi35.dll" funcs="jruninit,jrunfilter,jrunservice" Init fn="jruninit" serverstore="C:/JRun/lib/wsconfig/3/jrunserver.store" bootstrap="127.0.0.1:51003" verbose="false" apialloc="false" ssl="false" ignoresuffixmap="false"

5 Restart the web server.

Multiplexed MPEG2-DVD files exported with PCM audio from Adobe Media Encoder import with no audio.

Issue

When you export any source file to a multiplexed MPEG2-DVD file with the default PCM audio in Adobe Media Encoder, the resulting file imports into Premiere without audio.

Solution

Export the source file using either MPEG audio or Dolby audio.

Additional information

The audio in the exported file is silent and no waveforms are drawn on import.