

Application: gvSIG desktop - gvSIG bugs #4322

Nasty StackOverflowError in plugin system while SPI lookup

08/16/2016 08:59 PM - Andrea Antonello

Status:	Closed	% Done:	0%
Priority:	Urgent	Spent time:	0.00 hour
Assignee:			
Category:	Application		
Target version:	2.3.0-2442 (rev. org.gvsig.desktop-2.0.149)	Add-on version:	
Severity:	Critical	Add-on build:	
gvSIG version:	2.3.0	Add-on resolve version:	
gvSIG build:	2441	Add-on resolve build:	
Operative System:	Linux	Proyecto:	
Keywords:		Hito:	
Has patch:	Yes		
Add-on name:	Unknown		

Description

Recently some changes have been done to the org.gvsig.andami.plugins.PluginClassLoader class.

These seem to be potentially dangerous for certain SPI lookup sequences. It can happen that the org.gvsig.andami.plugins.PluginClassLoader.getResources(String) method ends up in a recursion loop (in debug I saw pass the main app and the jcrs plugins often) ending in a stackoverflow as:

```
java.lang.StackOverflowError
at java.lang.StringCoding$StringEncoder.encode(StringCoding.java:304)
at java.lang.StringCoding.encode(StringCoding.java:344)
at java.lang.String.getBytes(String.java:918)
at java.io.UnixFileSystem.getBooleanAttributes0(Native Method)
at java.io.UnixFileSystem.getBooleanAttributes(UnixFileSystem.java:242)
at java.io.File.exists(File.java:819)
at sun.misc.URLClassPath$FileLoader.getResource(URLClassPath.java:1245)
at sun.misc.URLClassPath$FileLoader.findResource(URLClassPath.java:1212)
at sun.misc.URLClassPath$1.next(URLClassPath.java:240)
at sun.misc.URLClassPath$1.hasMoreElements(URLClassPath.java:250)
at java.net.URLClassLoader$3$1.run(URLClassLoader.java:601)
at java.net.URLClassLoader$3$1.run(URLClassLoader.java:599)
at java.security.AccessController.doPrivileged(Native Method)
at java.net.URLClassLoader$3.next(URLClassLoader.java:598)
at java.net.URLClassLoader$3.hasMoreElements(URLClassLoader.java:623)
at sun.misc.CompoundEnumeration.next(CompoundEnumeration.java:45)
at sun.misc.CompoundEnumeration.hasMoreElements(CompoundEnumeration.java:54)
at sun.misc.CompoundEnumeration.next(CompoundEnumeration.java:45)
at sun.misc.CompoundEnumeration.hasMoreElements(CompoundEnumeration.java:54)
at org.gvsig.andami.plugins.PluginClassLoader.getResources(PluginClassLoader.java:392)
at org.gvsig.andami.plugins.PluginClassLoader.getResources(PluginClassLoader.java:397)
```

```
at org.gvsig.andami.plugins.PluginClassLoader.getResources(PluginClassLoader.java:397)
at org.gvsig.andami.plugins.PluginClassLoader.getResources(PluginClassLoader.java:397)
at org.gvsig.andami.plugins.PluginClassLoader.getResources(PluginClassLoader.java:397)
at org.gvsig.andami.plugins.PluginClassLoader.getResources(PluginClassLoader.java:397)
....
```

This has been tested with previous versions (ex. build 2426) and it works.

Cesar supplied me with a andami plugin containing the PluginClassLoader without the latest changes and the problem is not there.

I mark this as urgent, since I would not trust this behavior not to happen also at startup in certain conditions.

Related issues:

Related to Application: gvSIG desktop - gvSIG bugs # 4296: Scripting composer...

Closed

07/25/2016

Associated revisions

Revision 42929 - 08/21/2016 10:57 AM - Joaquín del Cerro Murciano

refs #4322, patch from Cesar Martinez. Fix infinite loop in getResources of PluginClassLoader.

History

#1 - 08/16/2016 09:27 PM - Cesar Martinez Izquierdo

The problem seems to be caused by the following combined causes:

- A getResources() method was added to PluginClassLoader, overriding default ClassLoader behaviour (revision <https://redmine.gvsig.net/redmine/projects/gvSIG-desktop/repository/revisions/42774>)
- A dependency loop is created between org.gvsig.geoprocess.app.sexante and org.gvsig.geoprocess.app.mainplugin plugins

Checking the config.xml files:

- org.gvsig.geoprocess.app.sexante declares a dependence on org.gvsig.geoprocess.app.mainplugin on its config.xml
- org.gvsig.geoprocess.app.mainplugin DOES NOT declare a dependence on org.gvsig.geoprocess.app.sexante, however org.gvsig.geoprocess.app.sexante becomes included on the pluginLoaders list at runtime (can be checked by debugging)

I think this may be caused by the system used to register gvSIG & Sextante geoprocesses, which probably modifies the classloader dependences of org.gvsig.geoprocess.app.mainplugin at runtime (but it should be investigated and confirmed)

#2 - 08/17/2016 03:40 PM - Cesar Martinez Izquierdo

- Has patch set to Yes
- File pluginClassLoaderGetResources.diff added

I have created a patch to avoid the infinite loop when a cross-reference is found.

I think we should re-design the way in which Sextante/gvSIG algorithms are loaded, removing the existing cross-reference, as it might cause problems on other parts of gvSIG.

For the moment, this patch solves the most evident problem.

#3 - 08/21/2016 10:58 AM - Joaquín del Cerro Murciano

- Status changed from New to Fixed

Metido el parche.

Gracias.

#4 - 08/21/2016 11:21 AM - Joaquín del Cerro Murciano

- Related to gvSIG bugs #4296: Scripting composer: double-clicking on a text file causes a StackOverflowError added

#5 - 08/23/2016 11:38 AM - Álvaro Anguix

- Status changed from Fixed to Closed

Files

pluginClassLoaderGetResources.diff	1.54 KB	08/17/2016	Cesar Martinez Izquierdo
------------------------------------	---------	------------	--------------------------