

Kepler - Bug #2315

Unable to run multiple Kepler instances simultaneously

12/09/2005 03:14 PM - Dan Higgins

Status:	Resolved	Start date:	12/09/2005
Priority:	Normal	Due date:	
Assignee:	ben leinfelder	% Done:	0%
Category:	general	Estimated time:	0.00 hour
Target version:	2.0.0	Spent time:	0.00 hour
Bugzilla-Id:	2315		
Description			
On a Windows machine, I have been unable to launch a second instance of Kepler (using a batch file which should run completely separate task) while first instance is still running. Error message is: Caused by: java.lang.NullPointerException at org.kepler.objectmanager.cache.CacheUtil.executeSQLCommand(CacheUtil.java:78)			
Related issues:			
Is duplicate of Kepler - Bug #2320: Cannot run multiple instances of Kepler s...		Resolved	12/16/2005

History

#1 - 12/16/2005 09:13 AM - Dan Higgins

Because of the internal database (hsqldb), one cannot run two instances of kepler simultaneously. The second instance of kepler started results in a very ugly stacktrace but does not terminate the application. This is an example of the stacktrace:

```
java.lang.NullPointerException
at org.kepler.objectmanager.cache.CacheUtil.executeSQLCommand(CacheUtil.java:78)
at org.kepler.objectmanager.cache.CacheManager.<init>(CacheManager.java:101)
at org.kepler.objectmanager.cache.CacheManager.getInstance(CacheManager.java:119)
at org.kepler.moml.KSWLibraryBuilder.buildLibrary(KSWLibraryBuilder.java:82)
at ptolemy.vergil.VergilApplication.openLibrary(VergilApplication.java:230)
at
ptolemy.vergil.VergilApplication._createDefaultConfiguration(VergilApplication.java:418)
at
ptolemy.vergil.VergilApplication._createEmptyConfiguration(VergilApplication.java:439)
at ptolemy.actor.gui.MoMLApplication._parseArgs(MoMLApplication.java:875)
at ptolemy.vergil.VergilApplication._parseArgs(VergilApplication.java:512)
at ptolemy.actor.gui.MoMLApplication.<init>(MoMLApplication.java:208)
at ptolemy.vergil.VergilApplication.<init>(VergilApplication.java:105)
at ptolemy.vergil.VergilApplication$1.run(VergilApplication.java:148)
at java.awt.event.InvocationEvent.dispatch(InvocationEvent.java:178)
at java.awt.EventQueue.dispatchEvent(EventQueue.java:454)
at
java.awt.EventDispatchThread.pumpOneEventForHierarchy(EventDispatchThread.java:201)
at
java.awt.EventDispatchThread.pumpEventsForHierarchy(EventDispatchThread.java:151)
at java.awt.EventDispatchThread.pumpEvents(EventDispatchThread.java:145)
at java.awt.EventDispatchThread.pumpEvents(EventDispatchThread.java:137)
at java.awt.EventDispatchThread.run(EventDispatchThread.java:100)
java.lang.NullPointerException
at org.kepler.objectmanager.cache.CacheUtil.executeSQLCommand(CacheUtil.java:78)
at org.kepler.objectmanager.cache.CacheManager.<init>(CacheManager.java:101)
at org.kepler.objectmanager.cache.CacheManager.getInstance(CacheManager.java:119)
at org.kepler.moml.KSWLibraryBuilder.buildLibrary(KSWLibraryBuilder.java:82)
at ptolemy.vergil.VergilApplication.openLibrary(VergilApplication.java:230)
at
ptolemy.vergil.VergilApplication._createDefaultConfiguration(VergilApplication.java:418)
at
ptolemy.vergil.VergilApplication._createEmptyConfiguration(VergilApplication.java:439)
at ptolemy.actor.gui.MoMLApplication._parseArgs(MoMLApplication.java:875)
at ptolemy.vergil.VergilApplication._parseArgs(VergilApplication.java:512)
at ptolemy.actor.gui.MoMLApplication.<init>(MoMLApplication.java:208)
```

```
at ptolemy.vergil.VergilApplication.<init>(VergilApplication.java:105)
at ptolemy.vergil.VergilApplication$1.run(VergilApplication.java:148)
at java.awt.event.InvocationEvent.dispatch(InvocationEvent.java:178)
at java.awt.EventQueue.dispatchEvent(EventQueue.java:454)
at
java.awt.EventDispatchThread.pumpOneEventForHierarchy(EventDispatchThread.java:201)
at
java.awt.EventDispatchThread.pumpEventsForHierarchy(EventDispatchThread.java:151)
at java.awt.EventDispatchThread.pumpEvents(EventDispatchThread.java:145)
at java.awt.EventDispatchThread.pumpEvents(EventDispatchThread.java:137)
at java.awt.EventDispatchThread.run(EventDispatchThread.java:100)
```

Here are some possible resolutions:

- 1) deploy hsqldb as a service and have it run in an independent jvm. Questions: How is startup managed? Are there any performance or resource constraints?
- 2) test for another instance of kepler and terminate the second copy with an appropriate error message
- 3) use a different database system which operates independently from kepler. MySQL is one example. Questions: what are the deployment issues involved here.

#2 - 12/16/2005 09:14 AM - Dan Higgins

- Bug 2320 has been marked as a duplicate of this bug. ***

#3 - 01/24/2006 11:17 AM - Chad Berkley

The only way we could fix this is to have multiple .kepler directories on one machine. This would require a profiling system like morpho has. Since I don't think this is on our list of things to do, i'm going to mark this WONTFIX. If at some point we add a profile system, possibly when we add authentication, this problem should be fixed.

#4 - 01/29/2007 08:15 AM - Christopher Brooks

On kepler-dev, Dan wrote:

Norbert,

The problem with multiple instances of Kepler is due to the internal database (hsq) used for the cache and keeping track of actors for the Kepler actor list. Running multiple workflows can be done using 'momlexecute' or 'ptexecute'. (There are some ant targets for these methods in the build.xml file.) These methods avoid even setting up the database.

Incidentally, I disagree with the 'wontfix' decision that Christopher quoted for the bug. I think that I should be able to work with multiple simultaneous versions of Kepler.

Dan Higgins - NCEAS

Norbert Podhorszki wrote:

Hi Christopher,

Uhhoh, or what.

Then I have to ask that guy, how was he able to execute ten workflows from a script last Fall, since this bug had been resolved a year ago.

If you happened to misunderstand me, I clarify now: I do not want to run even one copy of Kepler on a machine. I (may) want to execute two workflows created in Kepler at once on the same machine ;-)

Best regards
Norbert

On Thu, 25 Jan 2007, Christopher Brooks wrote:

Hi Norbert,
See

http://bugzilla.ecoinformatics.org/show_bug.cgi?id=2315
is marked as resolved and wontfix.

#5 - 05/11/2007 08:34 AM - Christopher Brooks

Matt writes:

The issue is that Kepler uses a backend relational db for caching and several data processing activities. That relational db (hsq) stores its files in a subdir of the ~/.kepler directory. If more than one instance of Kepler tries to start up, you get the exception that Christopher listed.

One potential workaround is to have the script that runs the multiple instances run from different accounts, and therefore each instance would use a different .kepler folder and therefore different .hsq db. The limitation is not really on running more than one instance of Kepler on a machine -- its running more than one instance of Kepler in a single user account. Any workaround that allows each process to have its own .kepler dir will work. Of course, the downside is that these processes will not share the cache, and therefore large data downloads, etc, will have to be repeated for each instance.

#6 - 11/29/2007 11:51 AM - Matt Jones

Reopening this bug regarding the ability to start multiple instances of Kepler simultaneously in one account. The limitation is due to a deficiency in hsql not being able to open the same database under multiple processes. This is something that could be fixed, and should be considered under the Kepler/CORE project as an infrastructure issue.

#7 - 12/13/2007 01:51 PM - Dan Higgins

TO be handled as part of KeplerCore

#8 - 06/07/2008 12:40 PM - Daniel Crawl

Another solution is to run the HSQL server in the first Kepler's JVM; subsequent Kepler instances could then contact the server. This is possible with HSQL 1.8.0:

<http://hsqldb.org/doc/guide/ch01.html#N101A8>

(Kepler currently appears to be using 1.7.2).

#9 - 03/05/2009 05:22 PM - ben leinfelder

I've played with the current (1.7.2) HSQL and got it to run in "Server" mode within the first Kepler JVM. Subsequent Kepler launches skip the server startup and then just connect to it. I believe we get 10 simultaneous connections.

I put the DB server launch call in the KeplerApplication class, but maybe it should be somewhere else?

I'd like some feedback before checking in such a large (albeit small) change.

#10 - 03/06/2009 08:45 AM - Daniel Crawl

What happens when you close the first Kepler instance; are the connections to the subsequent Keplers closed or does the first Kepler process stay alive until all the subsequent Keplers quit?

What about running the HSQL server in its own process?

We could have a config file that specifies how to access the cache: embedded, embedded server, or standalone server.

#11 - 03/06/2009 03:09 PM - ben leinfelder

good point about closing the first Kepler - the db server would die with it.

Now I've got it so that the server starts up if need be when a connecting is requested - so if the first Kepler is terminated (and the hsql server with it) the second Kepler can then start up the server when a connection is needed.

BUT: I had to refactor some of the CacheManager class (and a few others) so that there is no DBConnection member variable (otherwise the conn that is originally made when the object is instantiated can become "broken" when the server's JVM switches. It means more calls to getDBConnection() (and also conn.close()) - but I think that should be alright.

#12 - 03/06/2009 03:46 PM - ben leinfelder

committed the changes to DBConnectionFactory and also the CacheManager and LSIDTree so that connections are not saved and sitting around broken when HSQL switches JVMs.

#13 - 03/17/2009 11:03 AM - ben leinfelder

Realized this bug can be closed.

If issues arise with the the way this is working (in Server mode) then we'll open new bugs and go from there.

#14 - 04/25/2009 02:13 PM - Christopher Brooks

When I run two instances of Kepler using "ant run", I see the message:

```
[run] Building Kars...
[run] [Server@2fc7db]: [Thread[HSQldb Server @2fc7db,6,main]]: run()/openServerSocket():
[run] java.net.BindException: Address already in use
[run]   at java.net.PlainSocketImpl.socketBind(Native Method)
[run]   at java.net.PlainSocketImpl.bind(PlainSocketImpl.java:359)
[run]   at java.net.ServerSocket.bind(ServerSocket.java:319)
[run]   at java.net.ServerSocket.<init>(ServerSocket.java:185)
[run]   at java.net.ServerSocket.<init>(ServerSocket.java:97)
[run]   at org.hsqldb.HsqlSocketFactory.createServerSocket(Unknown Source)
[run]   at org.hsqldb.Server.openServerSocket(Unknown Source)
[run]   at org.hsqldb.Server.run(Unknown Source)
[run]   at org.hsqldb.Server.access$000(Unknown Source)
[run]   at org.hsqldb.Server$ServerThread.run(Unknown Source)
[run] Opening user preferences PtolemyPreferences.xml...
[run] * Attempting to get ResourceBundle for SVG defaults
[run]   svgRenderingMethod = SVG_BATIK_RENDERING *
[run] 5179 ms. Memory: 26736K Free: 2941K (11%)
```

However, it seems like both versions work ok. I've seen this error under both Windows and Mac OS X.

Is there any chance we can get rid of the exception that is displayed?
It makes it seem like the second instance of Kepler is not working.

#15 - 04/27/2009 10:19 AM - ben leinfelder

set the error writer for the DB server to null unless debug log level is enabled.
this prevents the stacktrace from showing up when it is not actually a critical error

#16 - 03/27/2013 02:19 PM - Redmine Admin

Original Bugzilla ID was 2315