Runtime Infrastructure - Bug #12234

Deadlock in web clients if the websocket connectivty gets cut at the wrong timing (even if restored after)

11/20/2026 10:03 AM - Şerban Bursuc

Status:	Review	**Start date:**	
Priority:	High	**Due date:**	
Assignee:	Şerban Bursuc	**% Done:**	100%
Category:		**Estimated time:**	0.00 hour
Target version:			
billable:	No	**reviewer:**	Hynek Cihlar
vendor_id:	GCD	**production:**	No
case_num:		**env_name:**	
version_reported:		**topics:**	
version_resolved:			

Description

History

#1 - 11/20/2026 10:08 AM - Şerban Bursuc

An accidental reproduction was done in #12082-82. Not a regression and it affects all FWD GUI projects that use a web client.

In that video a background task was at random killing the websocket connection on the client port. Typing while in this environment worked for the most part, until it didn't. The client stops responding completely at some point, even if the websocket connection comes back.

Doing a jstack showed:

```
"main-fwd" #1 prio=5 os_prio=0 cpu=5918.03ms elapsed=803.69s tid=0x00007c9ac8018aa0 nid=0x85d in Object.wait()
  [0x00007c9acdb26000]
   java.lang.Thread.State: WAITING (on object monitor)
    at java.lang.Object.wait(java.base@17.0.20.1/Native Method)
    - waiting on <no object reference available>
    at java.lang.Object.wait(java.base@17.0.20.1/Object.java:338)
    at com.goldencode.p2j.ui.client.driver.web.WebClientProtocol.waitForResult(WebClientProtocol.java:1560)
    - locked <0x00000000f9c69f98> (a java.lang.Object)
    at com.goldencode.p2j.ui.client.driver.web.WebClientProtocol.waitForResult(WebClientProtocol.java:1508)
    at com.goldencode.p2j.ui.client.gui.driver.web.GuiWebSocket.getTextWidths(GuiWebSocket.java:4388)
    at com.goldencode.p2j.ui.client.gui.driver.web.GuiWebDriver.cacheTextMetrics(GuiWebDriver.java:4329)
    at com.goldencode.p2j.ui.client.gui.EditorGuiImpl.computeLineWidths(EditorGuiImpl.java:4310)
    at com.goldencode.p2j.ui.client.gui.EditorGuiImpl.splitLine(EditorGuiImpl.java:2490)
    at com.goldencode.p2j.ui.client.Editor.placeChar(Editor.java:2468)
    at com.goldencode.p2j.ui.client.Editor.placeChar(Editor.java:2396)
    at com.goldencode.p2j.ui.client.Editor.processKeyEvent(Editor.java:1123)
    at com.goldencode.p2j.ui.client.gui.EditorGuiImpl.processKeyEvent(EditorGuiImpl.java:978)
    at com.goldencode.p2j.ui.client.widget.AbstractWidget$1.run(AbstractWidget.java:3364)
    at com.goldencode.p2j.ui.chui.ThinClient.eventBracket(ThinClient.java:20646)
    at com.goldencode.p2j.ui.chui.ThinClient.eventDrawingBracket(ThinClient.java:20588)
    at com.goldencode.p2j.ui.chui.ThinClient.independentEventDrawingBracket(ThinClient.java:20446)
    at com.goldencode.p2j.ui.client.widget.AbstractWidget.processSystemKey(AbstractWidget.java:3345)
    at com.goldencode.p2j.ui.chui.ThinClient.checkForSystemEvent(ThinClient.java:19665)
    at com.goldencode.p2j.ui.chui.ThinClient.waitForEvent(ThinClient.java:19450)
    at com.goldencode.p2j.ui.chui.ThinClient.waitForWorker(ThinClient.java:15991)
    at com.goldencode.p2j.ui.chui.ThinClient.lambda$waitForWorker$80(ThinClient.java:15515)
    at com.goldencode.p2j.ui.chui.ThinClient$$Lambda$553/0x00007c9a543a69c8.get(Unknown Source)
    at com.goldencode.p2j.ui.chui.ThinClient.lambda$doInteractive$103(ThinClient.java:21748)
    at com.goldencode.p2j.ui.chui.ThinClient$$Lambda$554/0x00007c9a543a6bf0.run(Unknown Source)
    at com.goldencode.p2j.ui.chui.ThinClient.doInteractive(ThinClient.java:21720)
    at com.goldencode.p2j.ui.chui.ThinClient.doInteractive(ThinClient.java:21748)
    at com.goldencode.p2j.ui.chui.ThinClient.waitForWorker(ThinClient.java:15514)
    at com.goldencode.p2j.ui.chui.ThinClient.waitFor(ThinClient.java:15457)
    at com.goldencode.p2j.ui.chui.ThinClient.waitFor(ThinClient.java:15406)
```

A server-side method is waiting for the browser to respond, but this particular request fell off because the browser couldn't respond to it, the websocket connection was destroyed. Even if the websocket connection came back, this particular request kept on waiting for browser.

The issue is fixed by activating the watchdog: close the browser tab.

#4 - 11/23/2026 10:17 AM - Șerban Bursuc

- Priority changed from Normal to High

- Status changed from New to WIP

- Assignee set to Șerban Bursuc

- % Done changed from 0 to 100

- reviewer Hynek Cihlar added

Committed to 12234a rev. 16866.

Summary of the changes:

- Every request the client waits for goes through sendRequest(). Binary ones are wrapped in a MSG_REQUEST envelope with the request ID: they already carried an ID after their type byte, but the page could not tell a request from other messages. JSON requests already carry a msgId key.
- The requesting thread waits on WebClientProtocol.lock for the answer with that ID.
- A new websocket makes the waiting thread send the request again: onConnect increments connections and calls notifyAll() (Jetty websocket thread), so the thread (main UI thread) returns from lock.wait(), sees the changed count, sends the same request on the new socket and goes back to lock.wait(). sendRequest() itself still returns only with the answer, on a timeout or on an interrupt (FWD server interrupt, CTRL-C).
- The wait ends without an answer only on a timeout, which few callers give (the broadcast channel pings, pingOnWebSocket, getBrowserDetails - all GUI requests wait indefinitely), or on an interrupt:; sendRequest then returns null.
- The browser handles each request ID once. A repeated ID gets the answer kept from the first time, or nothing while the first one is still being handled (an open file dialog), whose answer then goes out on the current socket. The last 64 requests are kept.
- An answer nobody waits for any more is dropped instead of kept forever: a late answer after a timeout or interrupt, or a second answer to a request sent twice. At most 64 such IDs are remembered.

I have attached a slideshow presentation for those at GCD to read: link

The changes need some diagrams and explanation.

The FWD code review flagged 4 major issues which are from trunk existing already, and as this is quite a major change by itself already I think I will leave the branch as is.

Priority raise to match a customer network issues.

#5 - 11/23/2026 10:35 AM - Șerban Bursuc

- Status changed from WIP to Review