

GDP - Bug #99

Subscribe seems to be missing data

04/18/2017 05:20 PM - Anonymous

Status:	Closed	Start date:	04/18/2017
Priority:	Normal	Due date:	
Assignee:	Eric Allman	% Done:	0%
Category:	libgdp	Estimated time:	0.00 hour
Target version:			
Description			
When I run <code>\$PTII/bin/capecode \$PTII/ptolemy/actor/lib/jjs/modules/gdp/test/auto/GDPLogSubscribe.xml</code>			
The regression test actor connected to a Subscribe has the following training values: <code>{"nil", "1", "2", "3", "4"}</code>			
However, it never sees "2".			
If I retrain the regression test actor, the new training values are missing a 2: <code>{"nil", "1", "3", "4", "5"}</code>			
Any ideas?			
See the attached screenshot and gdp logging for details.			

History

#1 - 04/18/2017 06:26 PM - Eric Allman

- Status changed from New to In Progress

In digging through the voluminous debug output, I'm seeing the following set of commands issued (search for >>> `_gdp_invoke`):

- OPEN_RA, response NAK_NO_ROUTE
- CREATE, response ACK_DATA_CREATED
- SUBSCRIBE, response ACK_SUCCESS
- APPEND, recno 1, data "1", response ACK_DATA_CREATED
- APPEND, recno 2, data "2", response ACK_DATA_CREATED
- READ, recno 1, response ACK_DATA_CONTENT
- APPEND, recno 2, data "3", response ACK_DATA_CREATED
- READ, recno 2, response ACK_DATA_CONTENT
- SUBSCRIBE, response ACK_SUCCESS
- OPEN_RA, response NAK_NO_ROUTE
- CREATE, response ACK_DATA_CREATED
- SUBSCRIBE, response ACK_SUCCESS
- APPEND, recno 1, data "1", response ACK_DATA_CREATED
- READ, recno 1, response ACK_DATA_CONTENT
- APPEND, recno 2, data "3", response ACK_DATA_CREATED
- READ, recno 2, response ACK_DATA_CONTENT
- APPEND, recno 3, data "4", response ACK_DATA_CREATED
- READ, recno 3, response ACK_DATA_CONTENT
- APPEND, recno 4, data "5", response ACK_DATA_CREATED

- READ, recno 4, response ACK_DATA_CONTENT
- SUBSCRIBE, response ACK_SUCCESS
- SUBSCRIBE, response ACK_SUCCESS
- SUBSCRIBE, response ACK_SUCCESS

The interesting thing here (other than your rather confusing sequence of commands) is that there are two places where the record with content "3" gets written as recno 2. This will overwrite the previous value. (It is supposed to create a "branch", so there are two records with the same number, but that's not implemented yet.) This is all about replication, but it does have this rather odd effect.

The record overwrite is probably because it appears that you have two threads, both writing to the same GCL. The GDP library tries to relieve the application from the burden of keeping track of the current record number. Multiple threads writing to one GCL was not expected (you can have multiple threads, but writing to different GCLs). Unfortunately allowing writes to the same GCL in different threads is a problem that cannot be fully solved — each append includes the record number (that's part of the security model), and you can't be sure that the append in process is going to succeed. However, the library can do a better job of guessing, which I have implemented (but if the guess is wrong things will crash and burn). However, I don't want to push that until I've had a chance to do some testing on it, and I don't have time right now, so rather than giving you untested bits I'm going to defer pushing that.

#2 - 04/19/2017 08:13 AM - Anonymous

That certainly is an interesting set of commands that are generated.

The model is single threaded, it is invoking the various commands in a sequence that is based on time.

The model was working until I updated the gdp libraries on April 11:

<http://terra.eecs.berkeley.edu:8080/job/ptII/984/changes> says:

```
Updated gdp libraries to dad48a4..c2f7e02 and 0.7.35.
Updated RHEL gdp to 00cf2f7..c2f7e02.
Updated to gdp 00cf2f7..c2f7e02
```

I looked in to what version of the GDP we were using in build 983, and it is a bit confusing because stderr gets redirected. If I search for "libgdp 0.7.2" in the output from 983 (which worked), I get:

```
bash-4.1$ grep 'libgdp 0.7.2' ~/jobs/ptII/builds/983/log
[junit] @(#)libgdp 0.7.2 (2017-02-08 16:48) b3e16f5d32ca76691f288d0ec9cc18c9830f7a23
[junit] @(#)libgdp 0.7.2 (2017-04-11 04:54) d4712db48058060d35c2c2dad78976148115d24e
[junit] @(#)libgdp 0.7.2 (2017-04-11 04:54) d4712db48058060d35c2c2dad78976148115d24e
```

So, there is a chance I was running the tests with a libgdp from 2017-02-08.

The same command from the 984 log, which has the failure:

```
bash-4.1$ grep 'libgdp 0.7.2' ~/jobs/ptII/builds/984/log
[junit] @(#)libgdp 0.7.2 (2017-04-11 13:58) c2f7e0254821e46b7690a07575df27e282c3a26f++
[junit] @(#)libgdp 0.7.2 (2017-04-12 04:58) f3dbae1e6d83e88db049f37b62434fc6b8978b92
[junit] @(#)libgdp 0.7.2 (2017-04-12 04:58) f3dbae1e6d83e88db049f37b62434fc6b8978b92
```

So, something in the gdp changed between 2017-02-08 and 2017-04-11 that either fixed a bug or introduced a bug

I was curious as to why Create is being invoked twice and it is because when Subscribe gets an updated log name, it creates a new GCL and

resubscribes.

At this point, I'm not sure how to proceed. It seems like I'm missing data in a way that most users would find confusing. The model used to work, and something has changed. I'm not sure what.

#3 - 04/19/2017 09:15 AM - Eric Allman

- *Category set to libgdp*

- *Assignee set to Eric Allman*

Lots of things have changed since 2017-02-08, many of them having to do with locking. I suspect this is because in the process of fixing a deadlock I opened up a window that asynchronous code could sneak into. From your trace, it looks like something in your code must be asynchronous (event-driven perhaps?), since it really looks like there are multiple commands in flight at the same time.

In the meantime, I've tweaked the "next record guess" code so it guesses earlier. I'm not sure if that will fix the problem, but please give it a try (commit:0de5c9d3). You'll have to link with the new library, since the problem is on the client side rather than the server side.

#4 - 04/19/2017 10:43 AM - Anonymous

- *File subscribe2.txt added*

Yes, this is an event driven model.

After updating on the Darwin, I'm still getting similar results.

See the log.

#5 - 04/26/2017 10:15 AM - Eric Allman

Christopher, I found something that might explain your problem. Please pull the latest code, re-link, and try again.

#6 - 04/26/2017 10:29 AM - Anonymous

- *Status changed from In Progress to Closed*

Thanks, that fixed it!

For the record, under Darwin, I ran

```
org/terraswarm/accessor/accessors/web/gdp/adm/gdpUpdatePtII.sh
```

which builds the gdp shared libraries under Darwin, RHEL and Ubuntu. For this script to work, one must have accounts on terra.eecs and swarmnuc001.eecs.

At the end, it is necessary to follow the commands to update the npm library.

Then I ran

```
$PTII/bin/capecode $PTII/ptolemy/actor/lib/jjs/modules/gdp/test/auto/GDPLogSubscribe.xml
```

under Darwin and RHEL.

I've marked this issue as "Closed".

Thanks again for the fix!

Files

Screen Shot 2017-04-18 at 5.14.50 PM.png	304 KB	04/19/2017	Anonymous
subscribeLog.txt	84.7 KB	04/19/2017	Anonymous
subscribe2.txt	39.5 KB	04/19/2017	Anonymous