

Infrastructure - Bug #3786

python libclient should gracefully handle server closed connection

2013-06-04 19:55 - Dave Vieglais

Status:	Closed	Start date:	
Priority:	Normal	Due date:	
Assignee:	Roger Dahl	% Done:	100%
Category:	d1_libclient_python	Estimated time:	0.00 hour
Target version:		Story Points:	
Milestone:	None		
Product Version:	*		

Description

Reusing a connection with DataONEBaseClient will fail if the subsequent request is not sent within a few seconds since the server may have closed the connection. This is manifest in an exception "httplib.BadStatusLine", which is really just a symptom of the actual problem, which is that the socket is no longer in a state for writing.

The solution is one of:

1. call connection.close() after each request
2. trap the exception, close the connection, and reissue the call
3. check the socket state before issuing the request

Of these #2 seems most reasonable, since 1. is wasteful, and for 3. the state may have changed between the call to socket.select and the request is made.

To reproduce the issue:

```
import select
import time
import httpplib
import logging
import traceback
from d1_client import d1baseclient

logging.basicConfig(level=logging.DEBUG)
burl = "https://cn.dataone.org/cn"
c = d1baseclient.DataONEBaseClient(burl)
t0 = time.time()
ol = c.listObjects(start=0, count=100)
fileno = c.connection.sock.fileno()
sockok = True
while sockok:
    res = select.select([fileno],[fileno],[fileno],0)
    logging.debug("select response: %s" % str(res) )
    sockok = res[0] == []
    time.sleep(0.5)
logging.info( "Delta = %.2f" % (time.time() - t0) )
try:
    ol = c.listObjects(start=0, count=100)
except httpplib.BadStatusLine as e:
    logging.error(traceback.format_exc())
    c.connection.close()
    ol = c.listObjects(start=0, count=100)
```

History

#1 - 2013-06-04 19:56 - Dave Vieglais

- Description updated

#2 - 2013-07-12 15:27 - Roger Dahl

- Status changed from New to Closed