

Problem

Why does manager print an error stack trace that begins with "Failed to invoke 'components_unavailable()'"?

Solution

Sometimes manager prints out enormous error stack traces complaining that it cannot invoke some method of a client such as this:

```
[ Connected to 146.88.1.153:57431 ]
2004-08-03T21:57:01.300 Client 'ControlTestR0a' logged in.

#...

[ Retrying to connect to 146.88.1.153:57431 ]
[ Retrying to connect to 146.88.1.153:57431 ]
[ Retrying to connect to 146.88.1.153:57431 ]
[ Retrying to connect to 146.88.1.153:57431 ]
[ Retrying to connect to 146.88.1.153:57431 ]
2004-08-03T21:57:46.882 RemoteException caught while invoking 'Client.components_unavailable' on ClientInfo =3D
{ name =3D 'ControlTestR0a', Handle (0x2d60001) =3D { type =3D CLIENT, key =3D (0xd6), id =3D (0x1) } }.
[ Retrying to connect to 146.88.1.153:57431 ]
[ Retrying to connect to 146.88.1.153:57431 ]
[ Retrying to connect to 146.88.1.153:57431 ]
[ Retrying to connect to 146.88.1.153:57431 ]
2004-08-03T21:57:46.876 Failed to invoke 'components_unavailable()' method.
2004-08-03T21:57:48.887 Exception report:
----->( 0 )<-----
org.omg.CORBA.TRANSPORT: Retries exceeded, couldn't reconnect to 146.88.1.153:57431 vmcid: 0x0 minor
code: 0 completed: No
    at org.jacorb.orb.connection.Client_TCP_IP_Transport.connect(Unknown Source)
    at org.jacorb.orb.connection.TCP_IP_Transport.write(Unknown Source)
    at org.jacorb.orb.connection.GIOPConnection.write(Unknown Source)
    at org.jacorb.orb.CDROutputStream.write(Unknown Source)
    at org.jacorb.orb.connection.ServiceContextTransportingOutputStream.write_to(Unknown Source)
    at org.jacorb.orb.connection.GIOPConnection.sendMessage(Unknown Source)
    at org.jacorb.orb.connection.ClientConnection.sendRequest(Unknown Source)
    at org.jacorb.orb.Delegate.invoke(Unknown Source)
    at org.omg.CORBA.portable.ObjectImpl._invoke(ObjectImpl.java:457)
    at si.ijs.maci._ClientStub.components_unavailable(_ClientStub.java:31)
    at com.cosylab.acs.maci.plugin.ClientProxy.components_unavailable(ClientProxy.java:131)
    at com.cosylab.acs.maci.manager.ManagerImpl$1ComponentUnavailableTask.run(ManagerImpl.java:4050)
    at EDU.oswego.cs.dl.util.concurrent.PooledExecutor$Worker.run(PooledExecutor.java:728)
    at java.lang.Thread.run(Thread.java:534)
-----

#...
```

Answer:

A client that has connected to manager did not disconnect properly following the guidelines ACS has set fourth in documentation and examples. In simple terms, either:

- the *disconnect* method of a Python *PySimpleClient* object was not invoked before exiting the Python interpreter **or**
- the *cleanUp* method of a Java *alma.acs.component.client.ComponentClient* object was not invoked before exiting the JVM

-- [DavidFugate](#) - 14 Aug 2004

Related articles

- [How can more people do development with ACS on the same machine without disturbing each other?](#)
- [Which ports are used by ACS?](#)
- [Problems connecting to ACS servers on a remote machine: bad /etc/hosts](#)
- [Why does the getComponent method of ZLegacy/ACS.ContainerServices return an object of type None?](#)
- [Why are some of my print statements not showing up in the container output section of acscommandcenter?](#)