

Tango Client and Device-Server Communication Issues

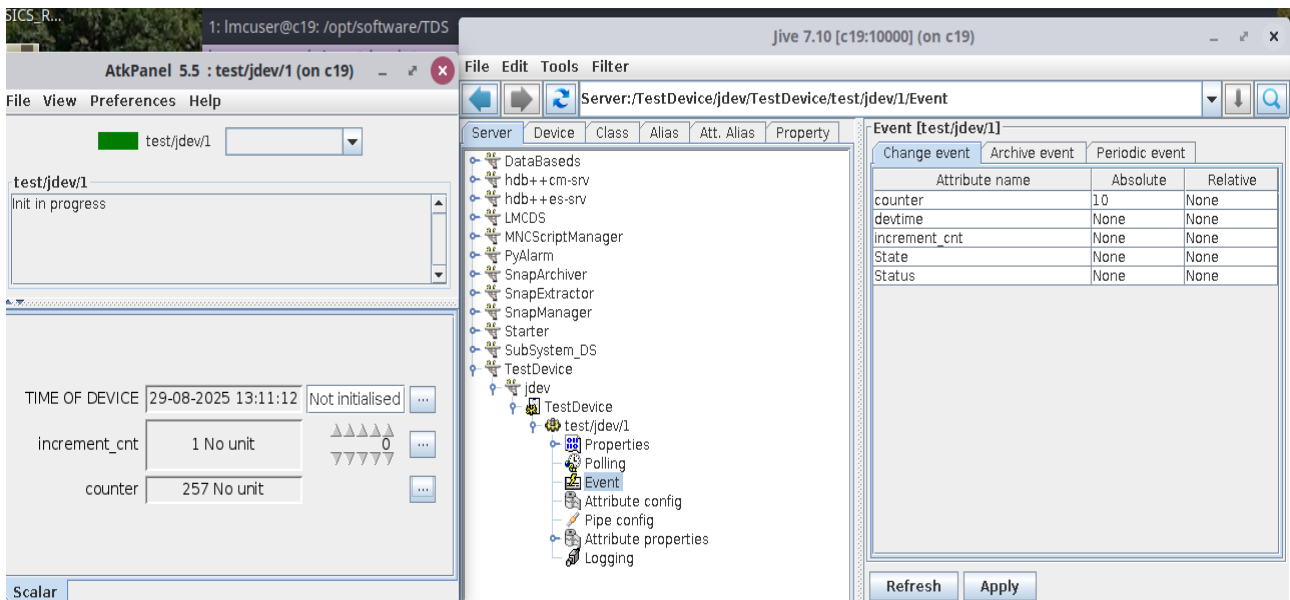
Jitendra Kodilkar , Version-1 Sep 09, 2025
jitendra@ncra.tifr.res.in

1. INTRODUCTION :

This document reports TangoDevice-Server and Client communication related issues using the *CPP/PyTango 9.4.2 , Jtango 9.7.4* versions on Ubuntu 20.04/22.04 LTS platform. In Tango based Control-systems, if the remotely located Device-Server is not accessible over the ethernet (*due to various kind of problems such as network access problem, or Tango-Device server machine hanged, server crashed etc.*), the client application shall not get stuck or hang, as it is suppose to perform time-bound other scheduled activities, such as complete some functionality, or read tango-attributes from other connected device-servers. However, practically client-applications at the user ends appear to be hang, the present report tries to give details of the problem, highlight some Tango communication related issues, and also propose a possible alternative to overcome it.

This report study the Tango-Client behaviour in case of device-server is not accessible by performing the predefined test-cases which vary with the two configuration settings, one is a **transparency** for the client-server reconnection, and other is the **device-timeout** which is either default or set to 5 millisecond for device communication. For this purpose, a '**TestDevice.java**' Tango Device Server is created with the '**devtime**' attribute showing current time of the device, which is polled for every 1 second. And another '**counter**' attribute where the change-event is generated whenever it changes absolute by 10 count value, the counter is increased by 1 per second.

The test-client application '**TangoEventClient.java**' suppose to read the '**devtime**' attribute for every 1 second, also read '**state**' of the device-server for every 3 second. Also, client receives a change-event for every 10 seconds for the '**counter**' attribute. Both the Client and Server is written in Java, **Figure-1** shows a screenshot of Tango-Device server to get an idea.



2. TEST-CASES :

To study a Tango Device Client behaviour, application Device-Server ran on remote machine connected over the Ethernet. In each test case, the device-client application log is saved in the text format where '**devtime**' attribute of the Device-Server is logged per second along with the **State** of

Device-server for every 3 second, also the change-Event along with the **counter** attribute value is printed with a local client-host time i.e. '**Client-Time**'.

Thus, the device-client expected behaviour as follows :

- (i) In normal client-server network-connected condition, device-client shall print '**devtime**' / '**State**' value for **every second**, and print the received **change-event per ten seconds** along with the **counter** attribute value.
- (ii) In the network disconnected situation, the Client shall wait for maximum **default timeout (3 sec)** OR the **set device-time out (5 milli-sec)** and come out.
- (iii) Disconnected behaviour of the tango-client is noted for more than 10 minutes i.e. up to 15 minutes.
- (iv) The Server shall reconnect once the network is established, and it shall print 'devtime' / 'state' value for every seconds, and shall receive change-event for every 10 seconds associated with the counter attribute of the DS i.e. same behaviour mentioned in enumeration (i).

2.1 Test Setup :

Application	Machine name and OS	Versions
TangoDeviceClient	'e01' , Ubuntu 22.04 LTS	CPP/PyTango 9.4.2 , Jtango 9.7.4
TestDevice	'c19' , Ubuntu 20.04 LTS 'tango://c19:10000/test/jdev/1'	CPP 9.2.5a, PyTango 9.3.3, Jtango 9.5.0

2.2 Test Cases :

◆ **CASE-1: Transparency – True, Timeout - Default (3 sec) :**

- **Device-connected :** In normal condition 'devtime', 'state' are being printed every 1 seconds, and change-event with 'counter' attribute values printed every 10th second.
- **Device-Disconnected :** When the Device-Server disconnected from the client, following behaviour observed.

(i) Read attributes 'devtime' failed with the device-timeout exception.

Client-Time : 26-08-2025 10:09:26 devstate-TANGO exception: Device (test/jdev/1) timed out (>3000 ms)!

Client-Time : 26-08-2025 10:09:33 DevTime-TANGO exception: Device (test/jdev/1) timed out (>3000 ms)!

(ii) KeepAlive Thread create two new connection every time while running the loop of trying to read attributes and receiving event. Out of which one thread try to search the device ' test/jdev/1' in local database of client-machine '**e01**' which couldn't understand.

[KeepAliveThread] INFO org.jacorb.orb.giop - ClientConnectionManager: created new ClientGIOPConnection to 192.168.70.35:10000 (477e063c)

[KeepAliveThread] INFO org.jacorb.orb.giop - ClientConnectionManager: created new ClientGIOPConnection to 127.0.0.1:2809 (5647ca4e)

Error receiving event: device test/jdev/1 not defined in the database e01:10000 !

(iii) Expected Client-Time interval for throwing the exception while waiting given time-out for reading the attributes a (two times '*devtime*' and one time '*state*') is 4 seconds (with 3000 ms timeout + 1 sec extra). But every time after trying to read the '*state*' of the device, **it takes 6 to 7 seconds for reading the '*devtime*' which is expected to take only regular interval of 4 seconds.**

Client-Time : 26-08-2025 10:10:17 DevTime-TANGO exception: Device (test/jdev/1) timed out (>3000 ms)!

[KeepAliveThread] INFO org.jacorb.orb.giop - ClientConnectionManager: created new ClientGIOPConnection to 192.168.70.35:10000 (7ac5ea00)

[KeepAliveThread] INFO org.jacorb.orb.giop - ClientConnectionManager: created new ClientGIOPConnection to 127.0.0.1:2809 (8fd1a74)

Error receiving event: device test/jdev/1 not defined in the database e01:10000 !

[KeepAliveThread] INFO org.jacorb.orb.giop - ClientConnectionManager: found ClientGIOPConnection to 192.168.70.35:36931 (3cbbc1e0)

[TangoEventExample3.LMCDevStat] INFO org.jacorb.orb.giop - ClientConnectionManager: found ClientGIOPConnection to 192.168.70.35:36931 (3cbbc1e0)

Client-Time : 26-08-2025 10:10:21 DevTime-TANGO exception: Device (test/jdev/1) timed out (>3000 ms)!

[TangoEventExample3.LMCDevStat] INFO org.jacorb.orb.giop - ClientConnectionManager: found ClientGIOPConnection to 192.168.70.35:36931 (3cbbc1e0)

[TangoEventExample3.LMCDevStat] INFO org.jacorb.orb.giop - ClientConnectionManager: found ClientGIOPConnection to 192.168.70.35:36931 (3cbbc1e0)

[KeepAliveThread] INFO org.jacorb.orb.giop - ClientConnectionManager: found ClientGIOPConnection to 192.168.70.35:36931 (3cbbc1e0)

Client-Time : 26-08-2025 10:10:25 devstate-TANGO exception: Device (test/jdev/1) timed out (>3000 ms)!

[TangoEventExample3.LMCDevStat] INFO org.jacorb.orb.giop - ClientConnectionManager: found ClientGIOPConnection to 192.168.70.35:36931 (3cbbc1e0)

[KeepAliveThread] INFO org.jacorb.orb.giop - ClientConnectionManager: created new ClientGIOPConnection to 192.168.70.35:10000 (58dee6fa)

[KeepAliveThread] INFO org.jacorb.orb.giop - ClientConnectionManager: created new ClientGIOPConnection to 127.0.0.1:2809 (6392460e)

Error receiving event: device test/jdev/1 not defined in the database e01:10000 !

[KeepAliveThread] INFO org.jacorb.orb.giop - ClientConnectionManager: found ClientGIOPConnection to 192.168.70.35:36931 (3cbbc1e0)

Client-Time : 26-08-2025 10:10:32 DevTime-TANGO exception: Device (test/jdev/1) timed out (>3000 ms)!

(iv) After ~10 minutes, **KeepAliveThred** creation goes on continuous for ~ 10 seconds. And then **ClientMessageRecepter** - "Abnormal connection termination" message appeared. After Abnormal connection termination, **the Client-Application is hang for almost 5 to 6 minutes.**

After , 5 to 6 minutes, KeepAliveThread thread creation for connection tries res-tarted, and 'devtime' attribute and state reading try exception throwing interval changed from 4 to 1-2 seconds.

Client-Time : 26-08-2025 10:25:36 DevTime-TANGO exception: Connection to database failed !

org.omg.CORBA.TRANSIENT: Retries exceeded, couldn't reconnect to 127.0.0.1:2809 vmcid: 0x0 minor code: 0 completed: No

[TangoEventExample3.LMCDevStat] INFO org.jacorb.orb.giop - ClientConnectionManager: created new ClientGIOPConnection to 192.168.70.35:10000 (51937b3)

[TangoEventExample3.LMCDevStat] INFO org.jacorb.orb.giop - ClientConnectionManager: created new ClientGIOPConnection to 127.0.0.1:2809 (7b24c035)

Client-Time : 26-08-2025 10:25:37 devstate-TANGO exception: Connection to database failed !

org.omg.CORBA.TRANSIENT: Retries exceeded, couldn't reconnect to 127.0.0.1:2809 vmcid: 0x0 minor code: 0 completed: No

[TangoEventExample3.LMCDevStat] INFO org.jacorb.orb.giop - ClientConnectionManager: created new ClientGIOPConnection to 192.168.70.35:10000 (582267e4)

[TangoEventExample3.LMCDevStat] INFO org.jacorb.orb.giop - ClientConnectionManager: created new ClientGIOPConnection to 127.0.0.1:2809 (40df177d)

Client-Time : 26-08-2025 10:25:38 DevTime-TANGO exception: Connection to database failed !

org.omg.CORBA.TRANSIENT: Retries exceeded, couldn't reconnect to 127.0.0.1:2809 vmcid: 0x0 minor code: 0 completed: No
 [TangoEventExample3.LMCDevStat] INFO org.jacorb.orb.giop - ClientConnectionManager: created new ClientGIOPConnection to 192.168.70.35:10000 (4c779aa6)
 [TangoEventExample3.LMCDevStat] INFO org.jacorb.orb.giop - ClientConnectionManager: created new ClientGIOPConnection to 127.0.0.1:2809 (53671730)
Client-Time : 26-08-2025 10:25:39 DevTime-TANGO exception: Connection to database failed !

- **Device-Reconnected** : Connection established, attribute reading for the ‘devtime’ and state reading successfully has interval of 1 seconds. **But two events are received separately for ‘counter’ value change for every 10 seconds.**

Error receiving event: No heartbeat from dserver/testdevice/jdev since 40001ms on channel named tcp://192.168.70.35:32959

Event-Time : 26-08-2025 10:27:51 EVENT : change counter value 1573

Event-Time : 26-08-2025 10:27:52 EVENT : tango://c19:10000/test/jdev/1/counter.idl5_change counter value 1574

Client-Time : 26-08-2025 10:27:52 STATE RUNNING 4000

[TangoEventExample3.LMCDevStat] INFO org.jacorb.orb.giop - ClientConnectionManager: found ClientGIOPConnection to 192.168.70.35:36931 (312fefc4)

Client-Time : 26-08-2025 10:27:53 DEVICETIME VAR : 26-08-2025 10:27:52

Client-Time : 26-08-2025 10:27:54 DEVICETIME VAR : 26-08-2025 10:27:53

Client-Time : 26-08-2025 10:27:55 STATE RUNNING 4000

Client-Time : 26-08-2025 10:27:56 DEVICETIME VAR : 26-08-2025 10:27:55

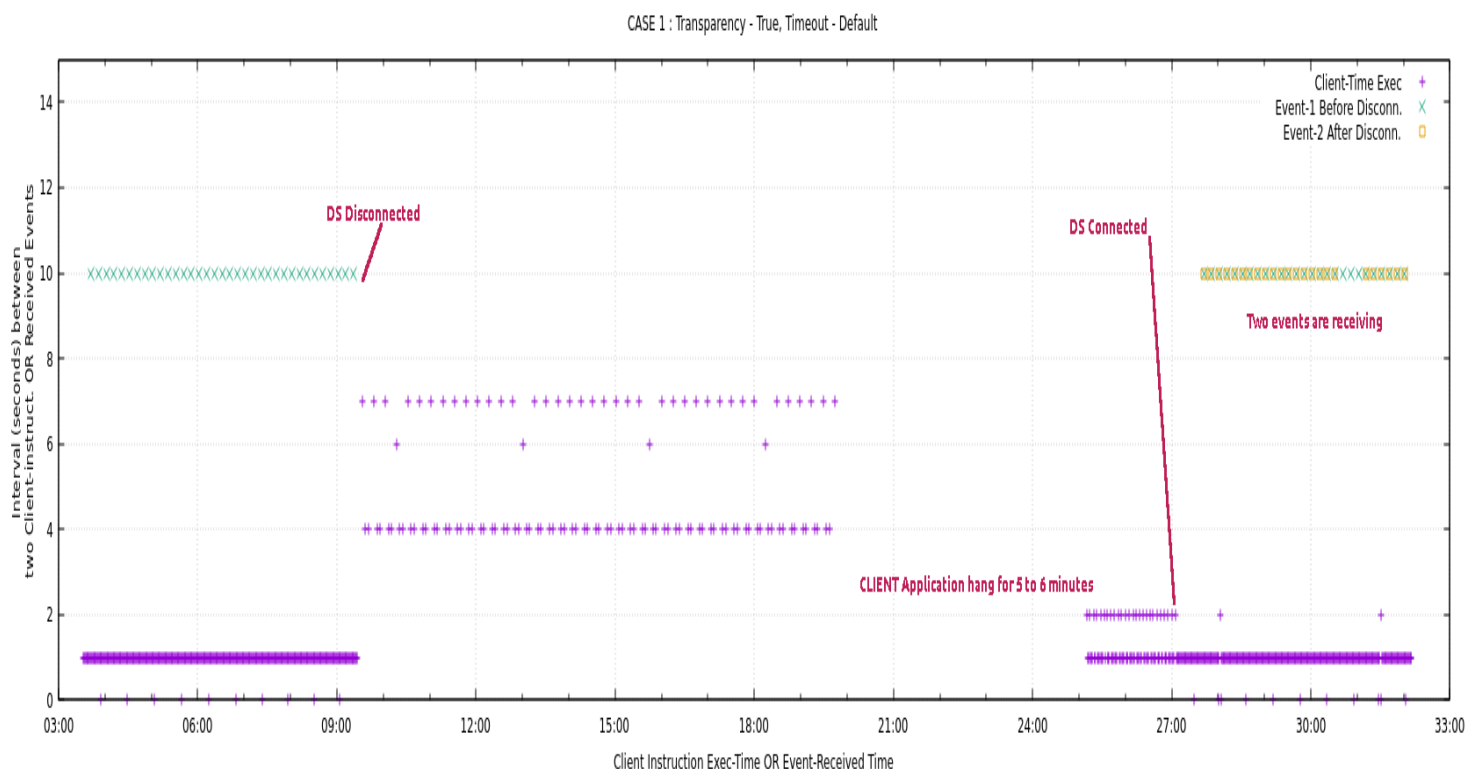


Figure 1 :- Case 1 : Transparency – True, Timeout Default (3000 ms)

Figure 1 shows the client application instruction-execution and event-received time on X-axis Vs time interval in seconds between two consecutive instruction execution, and event-received time.

◆ **CASE-2 : Transparency – False , Timeout – Default :**

- **Device-connected :** In normal condition ‘devtime’, ‘state’ are being printed every one second, and change-event with ‘counter’ attribute values printed every 10th second.

• **Device Disconnected :**

(i) In case of Transparency ‘false’ condition, the read-attributes and read status do show expected time interval of ~ 3 to 4 seconds.

Like **CASE-1**, the KeepAliveThread creates two new *ClientGIOPConnections* every time which tries to connect the Device-Server host machines, and local Tango-Client host. For local-host, it gives error message “*device test/jdev/1 not defined in the database e01:10000*” where ‘e01’ is a local-host.

(ii) In case of Transparency-fail condition, the **time interval of 3 to 4 seconds only** noticed between tries of reading ‘devtime’ and ‘state’ attributes. Also, after 10 minutes of waiting **Client application did not hang** as noticed in ‘Case-1’.

- **Device Reconnected :** Same as **CASE-1** with two events appearing after reconnection !!.

Case-2 Connect, Disconnect and Reconnect Client Exec-time/Event Receive time-stamp vs interval between two consecutive instructions or receiving events shown in **Figure-2**.

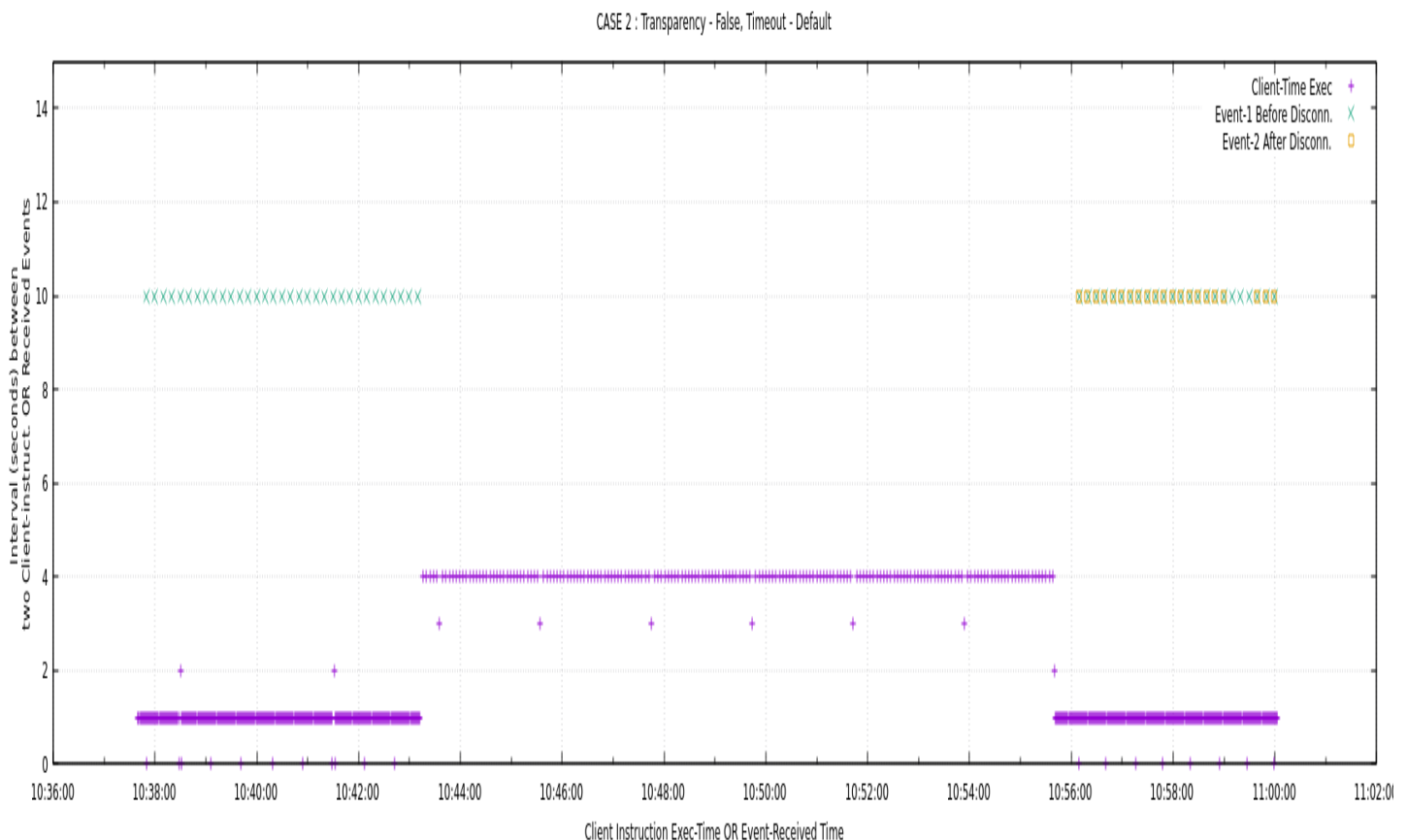


Figure 2 :- Case 2 : Transparency – False, Timeout Default (3000 ms)

◆ **CASE-3 : Transparency – True, Timeout - 5 milli-sec (when the DS disconnected) :**

In Case-3 , the default time-out of 3000 ms is allowed only in case of Client- Device Server connection is established. Otherwise whenever DS is not communicating using the **set_timeout_millis()**, **setDev_timeout()** calls the device time-out is set 5 milli-second for the Device and Admin (dserver) device.

- **Device-connected :** In normal condition ‘devtime’, ‘state’ are being printed every one second, and change-event with ‘counter’ attribute values printed every 10th second.
- **Device Disconnected :**
 - (i) Expected Client-Time interval for throwing the exception while waiting given time-out for reading the attributes a (two times ‘**devtime**’ and one time ‘**state**’) is ~1 seconds (with 5 ms timeout + 1 sec extra). **But every time after trying to read the ‘state’ of the device, it takes 6 to 7 seconds for reading the ‘devtime’ whereas it is expected to take only regular interval of 1 second of interval at the most. Note that for reading two ‘devtime’ attributes interval is coming 1 seconds, but after ‘state’ try reading, interval is of 6 to 7 second.**

Client-Time : 26-08-2025 12:04:11 DevTime-TANGO exception: Device (test/jdev/1) timed out (>5 ms)!

[TangoEventExample3.LMCDevStat] INFO org.jacorb.orb.giop - ClientConnectionManager: found ClientGIOPConnection to 192.168.70.35:42549 (3cbbc1e0)

Client-Time : 26-08-2025 12:04:12 DevTime-TANGO exception: Device (test/jdev/1) timed out (>5 ms)!

[TangoEventExample3.LMCDevStat] INFO org.jacorb.orb.giop - ClientConnectionManager: found ClientGIOPConnection to 192.168.70.35:42549 (3cbbc1e0)

[KeepAliveThread] INFO org.jacorb.orb.giop - ClientConnectionManager: created new ClientGIOPConnection to 192.168.70.35:10000 (10dfe867)

[KeepAliveThread] INFO org.jacorb.orb.giop - ClientConnectionManager: created new ClientGIOPConnection to 127.0.0.1:2809 (10bf801b)

Error receiving event: device test/jdev/1 not defined in the database e01:10000 !

[KeepAliveThread] INFO org.jacorb.orb.giop - ClientConnectionManager: found ClientGIOPConnection to 192.168.70.35:42549 (3cbbc1e0)

[TangoEventExample3.LMCDevStat] INFO org.jacorb.orb.giop - ClientConnectionManager: found ClientGIOPConnection to 192.168.70.35:42549 (3cbbc1e0)

Client-Time : 26-08-2025 12:04:13 devstate-TANGO exception: Device (test/jdev/1) timed out (>5 ms)!

[TangoEventExample3.LMCDevStat] INFO org.jacorb.orb.giop - ClientConnectionManager: found ClientGIOPConnection to 192.168.70.35:42549 (3cbbc1e0)

Client-Time : 26-08-2025 12:04:20 DevTime-TANGO exception: Device (test/jdev/1) timed out (>5 ms)!

[TangoEventExample3.LMCDevStat] INFO org.jacorb.orb.giop - ClientConnectionManager: found ClientGIOPConnection to 192.168.70.35:42549 (3cbbc1e0)

Client-Time : 26-08-2025 12:04:21 DevTime-TANGO exception: Device (test/jdev/1) timed out (>5 ms)!

(ii) Like **Case-1** of Transparency set True, After ~10 minutes, **KeepAliveThred** creation goes on continuos for ~ 10 seconds. And then **ClientMessageRecepter** - “Abnormal connection termination” message appeared. After Abnormal connection termination, **the Client-Application is hang for almost 5 to 6 minutes.**

After , 5 to 6 minutes, KeepAliveThread thread creation for connection tries restarted, and ‘devtime’ attribute and state reading try exception throwing interval changed from 4 to 1-2 seconds.

- **Device Reconnected :** Same as **CASE-1** with two events appearing after reconnection !!.

Case-3 *Connect, Disconnect and Reconnect* Client Exec-time/Event Receive time-stamp vs interval between two consecutive instructions or receiving events shown in **Figure-3**.

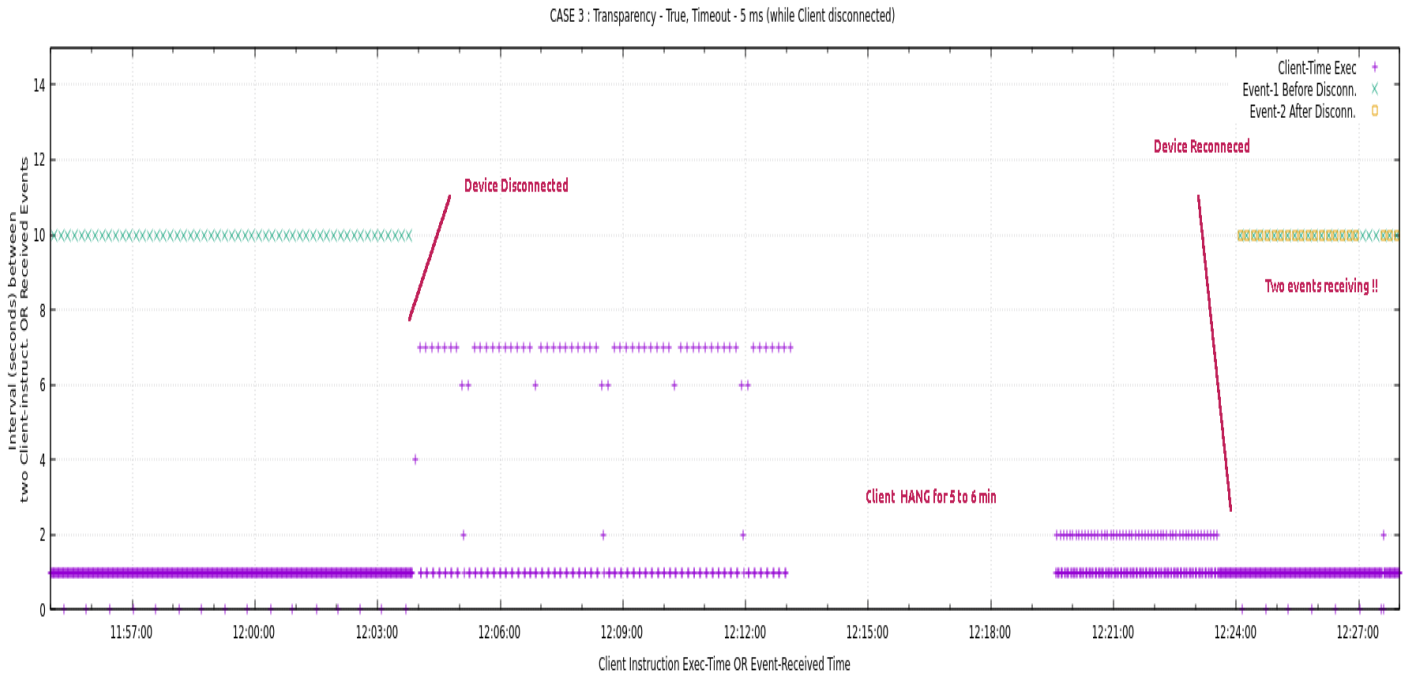


Figure 3 :- Case 3 : Transparency – True, Timeout Tuned to 5 ms in case of Device disconnected (otherwise 3000 ms).

◆ **CASE-4 : Transparency – False, Timeout - 5 milli-second (when the DS disconnected) :**

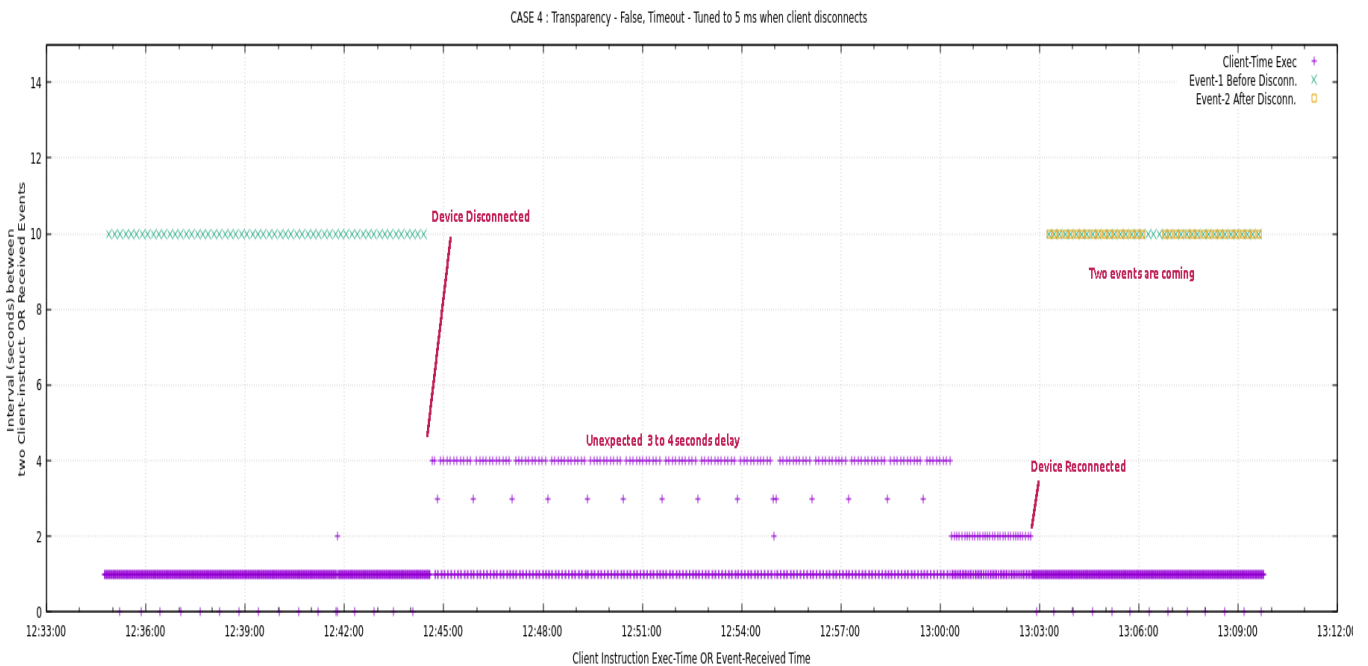


Figure 4 :- Case 4 : Transparency – False, Timeout Tuned to 5 ms in case of Device disconnected (otherwise 3000 ms).

- **Device-connected :** In normal condition ‘devtime’, ‘state’ are being printed every one second, and change-event with ‘counter’ attribute values printed every 10th second.

- **Device Disconnected :**

(i) Expected Client-Time interval for throwing the exception while waiting given time-out for reading the attributes a (two times '**devtime**' and one time '**state**') is ~1 seconds (with 5 ms timeout + 1 sec extra). **But every time after trying to read the '**state**' of the device, it takes 3 to 4 seconds for reading the '**devtime**' whereas it is expected to take only regular interval of 1 second of interval at the most.** Note that for reading two '**devtime**' attributes interval is coming 1 seconds, but after '**state**' try reading, interval is of 3 to 4 second.

Client-Time : 26-08-2025 12:48:29 devstate-TANGO exception: Device (test/jdev/1) timed out (>5 ms)!

[TangoEventExample3.LMCDevStat] INFO org.jacorb.orb.giop - ClientConnectionManager: found ClientGIOPConnection to 192.168.70.35:41287 (3cbbc1e0)

Client-Time : 26-08-2025 12:48:33 DevTime-TANGO exception: Device (test/jdev/1) timed out (>5 ms)!

[TangoEventExample3.LMCDevStat] INFO org.jacorb.orb.giop - ClientConnectionManager: found ClientGIOPConnection to 192.168.70.35:41287 (3cbbc1e0)

Client-Time : 26-08-2025 12:48:34 DevTime-TANGO exception: Device (test/jdev/1) timed out (>5 ms)!

[KeepAliveThread] INFO org.jacorb.orb.giop - ClientConnectionManager: found ClientGIOPConnection to 192.168.70.35:41287 (3cbbc1e0)

[TangoEventExample3.LMCDevStat] INFO org.jacorb.orb.giop - ClientConnectionManager: found ClientGIOPConnection to 192.168.70.35:41287 (3cbbc1e0)

[KeepAliveThread] INFO org.jacorb.orb.giop - ClientConnectionManager: created new ClientGIOPConnection to 192.168.70.35:10000 (7145f692)

[KeepAliveThread] INFO org.jacorb.orb.giop - ClientConnectionManager: created new ClientGIOPConnection to 127.0.0.1:2809 (49c1b67)

Error receiving event: device test/jdev/1 not defined in the database e01:10000 !

[KeepAliveThread] INFO org.jacorb.orb.giop - ClientConnectionManager: found ClientGIOPConnection to 192.168.70.35:41287 (3cbbc1e0)

Client-Time : 26-08-2025 12:48:35 devstate-TANGO exception: Device (test/jdev/1) timed out (>5 ms)!

[TangoEventExample3.LMCDevStat] INFO org.jacorb.orb.giop - ClientConnectionManager: found ClientGIOPConnection to 192.168.70.35:41287 (3cbbc1e0)

Client-Time : 26-08-2025 12:48:39 DevTime-TANGO exception: Device (test/jdev/1) timed out (>5 ms)!

[TangoEventExample3.LMCDevStat] INFO org.jacorb.orb.giop - ClientConnectionManager: found ClientGIOPConnection to 192.168.70.35:41287 (3cbbc1e0)

Client-Time : 26-08-2025 12:48:40 DevTime-TANGO exception: Device (test/jdev/1) timed out (>5 ms)!

(ii) In case of **Transparencey-fail** condition, after 10 minutes of waiting **Client application did not hang** as noticed in '**Case-2**'.

- **Device Reconnected :** Same as **CASE-1** with two events appearing after reconnection !!.

3. SUMMARY

(i) Trasparency set to 'True' value hang the Client application for 5 to 6 minutes after 10 minutes of device disconnection.

(ii) In both the Transparency set to Fail or True, the Client Exception after trying to read the state has interval of 6 to 7 secons (with default timeout) or 4 to 3 seconds (Timeout set 5 ms) which is more than the expected by '3 to 4' seconds. *Is it possible to avoid this extra 3 to 4 seconds ?*

(iii) After reconnection, two events are received from the device-server.

(iv) In case of both Transparency True or False, Device Connects automatically and restored.

(vi) How is behaviour in the latest Tango Release 10, for above (i) to (iv) cases.

(vii) To avoid client application, which is better way ?

- (a) Set transparency to 'False' value, and set device timeout to 5 milli-second when the device is not communicating, and restore default timeout when it is communicating.
OR
(b) Remove the device-proxy and try fresh proxy to reconnect.

Reference :

[1] Reconnection and Exception :

<https://tango-controls.github.io/cppTango-docs/recon.html>

Facts :

(i) Each Attribute/Command has its own polling buffer but the polling buffer depth is the same for all Attributes/Commands .

(ii) A Client is able to read data from (a) Real Equipment (DEVICE) (b) Last record from the Polling Buffer of DS (CACHE) . (c) The polling buffer with fall-back to real-quipment (CACHE_DEVICE).

(iii) Polling threads can be defined per device for a single Device-server, but not attribute-wise.

(iv) Tango 9 , if all attributes have same period , then `always_executed_hook()`, `read_attr_hardware()` called only once, one can revert to `read_attribute()` behaviour instead of `read_attributeS()` of tango-9

(v) Just one client reading one non-pollled attribute (...yes, *unfortunately exactly that slow one...*) can break your device.

- Each client reading a device triggers independent execution of device methods, possibly down to the hardware access.

(vi) If the polling period is the same for all polled attrs/cmds, then:

- Equipment access frequency $\cong 1 / ((\text{polling period}) \times (\text{nr. of polled cmds/attrs}))$
- Polling period = 1s
- Nr. of polled cmds/attrs = 10
- Equipment access every 100ms roughly.

Keep some spare CPU time for the device to to other jobs (...clients?) • At least 50%

It's useless (and inconvenient) setting up 1s polling period for 10 attributes on a TANGO device talking to a serial line equipment that takes 200ms round-trip which takes 2 seconds. (And even for 100ms round trip there won't be time for other activity or jobs.)

(vii) Every 10 seconds server send Heartbeat Event to all clients. Clients has keepAlive Thread which wakes up every 10 seconds to verify server status, and send Event-heartbeat subscription command every 200 seconds. Server stop sending Heartbeat event to the client if the last subscription is older than 600 seconds.

(viii) If event period is 1000 ms, and polling of attribute is 400 ms , then ***change event*** may occur after only 1200 ms (i.e. 400×3).