

Problem

What does it mean and how to interpret `cbReceive` took too long and `cbReceive` in AVG took too long messages?

Solution

The bulk data NT internally has a mechanism to measure the processing time of `cbReceive` - how long does it take to execute `cbReceive` method. The data that is given for process is a frame of size 64000 bytes (~64kBytes). So to satisfy the request of 64MBytes/sec the frame should be processed in less than 1ms (be aware that the 64MBytes/sec is requirements for the whole bulk data chain (BDC) not just processing the data). If the mechanism finds out that processing time is greater than the value set via API or CDB (for details see [cbReceiveProcessTimeoutSec](#)) it logs the message like:

```
cbReceive took too long (type=57, code=21) ActualProcessTime="0.01004" TotalFrameCount="138" FrameCount="132"
ProcessTimeoutSec="0.01" StreamFlowName="InterferometricStream3#FullSpectralFlow"
```

where we can see that frame # 132 out of 138, took 0.0104s to be processed, and that the process limit (timeout) was set to 0.01s (default value) for `Stream#Flow: InterferometricStream3#FullSpectralFlow`.

This is just a processing time for one frame, so it might be that there is just a glitch or similar if it happens not just here and there, but if it happens often the receiver has to be investigated to see why processing is so slow.

Because it is important that the processing time in average is in the spec bulk data NT provides yet another check that measures the average processing time of all the frames between `cbStar=t` and `=cbStop`. If the average (AVG) processing time exceeds the value set for `cbReceiveAvgProcessTimeoutSec` via API or in CDB (for details see [cbReceiveAvgProcessTimeoutSec](#)) it logs error message like:

```
cbReceive in AVG took too long (type=57, code=22) AvgProcessTimeoutSec="0.005" Throughput="10.2391" CallsCount="
2" StreamFlowName="InterferometricStream3#WVRFlow" ActualAvgProcessTime="0.009"
```

what means that average processing time for `Stream#Flow: InterferometricStream3#WVRFlow` took 0.009s what is more than 0.005 what corresponds to throughput of 10.2391 MBytes/sec, and there was 2 call to `cbReceive` (so the average was done on 2 calls).

-- [BogdanJeram](#) - 2013-09-26

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`?](#)