

# MTAOS handling of rejected commands

This notebook is used for the level 3 integration tests from test plan LVV-P81 (<https://jira.lsstcorp.org/secure/Tests.jspa#/testPlan/LVV-P81>) as part of test cycle LVV-C176 (<https://jira.lsstcorp.org/secure/Tests.jspa#/testCycle/LVV-C176>). The following tests are currently run as part of this notebook:

- LVV-T2193 (<https://jira.lsstcorp.org/secure/Tests.jspa#/testCase/LVV-T2193>)

Execution steps are separated by horizontal lines. Upon completion, save the notebook and its output as a pdf file to be attached to the test execution in JIRA.

Last updated by E. Dennihy 20211020

---

Load all the needed libraries. Get the remotes ready Code in the notebook including section: "Check the summary state of each CSC".

```
In [1]: %load_ext autoreload
        %autoreload 2
```

```
In [2]: import rubin_jupyter_utils.lab.notebook as nb
        nb.utils.get_node()
```

```
/tmp/ipykernel_23811/1665379685.py:2: DeprecationWarning: Call to deprecate
d function (or staticmethod) get_node. (Please use lsst.rsp.get_node())
    nb.utils.get_node()
```

```
Out[2]: 'yagan06'
```

```
In [3]: import os
        import sys
        import asyncio
        import logging

        import pandas as pd
        import numpy as np

        from matplotlib import pyplot as plt

        from lsst.ts import salobj
        from lsst.ts.observatory.control.maintel import MTCS, ComCam
        from lsst.ts.observatory.control import RotType
```

```
| lsst.ts.utils.tai INFO: Update leap second table
```

```
| lsst.ts.utils.tai INFO: current_tai uses the system TAI clock
```

```
In [4]: logging.basicConfig(format="%(name)s:%(message)s", level=logging.DEBUG)
```

```
In [5]: log = logging.getLogger("setup")
log.level = logging.DEBUG
```

```
In [6]: domain = salobj.Domain()
```

```
In [7]: mtcs = MTCS(domain=domain, log=log)
mtcs.set_rem_loglevel(40)
```

```
| setup.MTCS DEBUG: mtmount: Adding all resources.
| setup.MTCS DEBUG: mtptg: Adding all resources.
| setup.MTCS DEBUG: mtaos: Adding all resources.
| setup.MTCS DEBUG: mtm1m3: Adding all resources.
| setup.MTCS DEBUG: mtm2: Adding all resources.
| setup.MTCS DEBUG: mthexapod_1: Adding all resources.
| setup.MTCS DEBUG: mthexapod_2: Adding all resources.
| setup.MTCS DEBUG: mtrotator: Adding all resources.
| setup.MTCS DEBUG: mtdome: Adding all resources.
| setup.MTCS DEBUG: mtdometrajectory: Adding all resources.
```

```
In [8]: await mtcs.start_task
```

```
| MTHexapod INFO: Read historical data in 0.04 sec
| MTHexapod INFO: Read historical data in 0.05 sec
```

```
Out[8]: [None, None, None, None, None, None, None, None, None, None]
```

```
| MTHexapod.electrical WARNING: tel_electrical DDS read queue is filling:
| 20 of 100 elements
```

---

Ready M1M3: Raise mirror, turn on FB, clear forces

Need to have M1M3 LUT use its inclinometer.

---

Ready M2: Turn on FB, clear forces

Need to have M2 LUT use its inclinometer

---

Get camera hexapod ready: check config; make sure LUT is on, and has valid inputs;  
make sure hex is at LUT position

---

Get M2 hexapod ready: check config; make sure LUT is on, and has valid inputs; make  
sure hex is at LUT position

Slew to the next target. Choose a target such that the rotator stays within a couple of degrees of its initial position. This is because the CCW is not running (MTmount in simulation mode).

```
In [9]: target = await mtcs.find_target(el=60, az=120, mag_limit=8)
        print(target)
```

WARNING: AstropyDeprecationWarning: Transforming a frame instance to a frame class (as opposed to another frame instance) will not be supported in the future. Either explicitly instantiate the target frame, or first convert the source frame instance to a `astropy.coordinates.SkyCoord` and use its `transform\_to()` method. [astropy.coordinates.baseframe]

astroquery WARNING: AstropyDeprecationWarning: Transforming a frame instance to a frame class (as opposed to another frame instance) will not be supported in the future. Either explicitly instantiate the target frame, or first convert the source frame instance to a `astropy.coordinates.SkyCoord` and use its `transform\_to()` method.

MTHexapod.application WARNING: tel\_application DDS read queue is filling: 20 of 100 elements

MTHexapod.actuators WARNING: tel\_actuators DDS read queue is filling: 17 of 100 elements

MTHexapod.actuators WARNING: tel\_actuators DDS read queue is filling: 34 of 100 elements  
HD 72647

```
In [10]: await mtcs.slew_object(target, rot_type=RotType.PhysicalSky, rot=1.9)
```

setup.MTCS INFO: Slewing to HD 72647: 08 32 30.9533 -40 12 35.258

setup.MTCS DEBUG: Setting rotator physical position to 1.9 deg. Rotator will track sky.

setup.MTCS DEBUG: Wait 5.0s for rotator to settle down.

setup.MTCS DEBUG: Workaround for rotator trajectory problem. Moving rotator to its current position: 1.76

setup.MTCS DEBUG: Wait for MTRotator in position event.

setup.MTCS DEBUG: MTRotator in position: False.

setup.MTCS INFO: MTRotator in position: True.

setup.MTCS DEBUG: MTRotator in position True. Waiting settle time 5.0s

setup.MTCS DEBUG: Sending slew command.

setup.MTCS DEBUG: Scheduling check coroutines

setup.MTCS DEBUG: process as completed...

setup.MTCS DEBUG: Monitor position started.

setup.MTCS DEBUG: Waiting for Target event from mtmount.

setup.MTCS DEBUG: mtmount: <State.ENABLED: 2>

setup.MTCS DEBUG: mtptg: <State.ENABLED: 2>

setup.MTCS DEBUG: mtaos: <State.ENABLED: 2>

setup.MTCS DEBUG: mtm1m3: <State.ENABLED: 2>

setup.MTCS DEBUG: mtm2: <State.ENABLED: 2>

setup.MTCS DEBUG: mthexapod\_1: <State.ENABLED: 2>

setup.MTCS DEBUG: mthexapod\_2: <State.ENABLED: 2>

setup.MTCS DEBUG: mtrotator: <State.ENABLED: 2>

```

| setup.MTCS DEBUG: mtdome: <State.ENABLED: 2>
| setup.MTCS DEBUG: mtdometrajectory: <State.ENABLED: 2>
| setup.MTCS DEBUG: Wait for mtmount in position events.
| setup.MTCS DEBUG: Wait for dome in position event.
| setup.MTCS DEBUG: Wait for MTRotator in position event.
| setup.MTCS DEBUG: MTRotator in position: True.
| setup.MTCS DEBUG: MTRotator already in position. Handling potential race
| condition.
| setup.MTCS DEBUG: Wait for MTMount elevation in position event.
| setup.MTCS DEBUG: MTMount elevation in position: True.
| setup.MTCS DEBUG: MTMount elevation already in position. Handling potent
| ial race condition.
| setup.MTCS DEBUG: Wait for MTMount azimuth in position event.
| setup.MTCS DEBUG: MTMount azimuth in position: True.
| setup.MTCS DEBUG: MTMount azimuth already in position. Handling potentia
| l race condition.
| setup.MTCS DEBUG: Mount target: private_revCode: bdc00ba, private_sndSt
| amp: 1652470333.6882071, private_rcvStamp: 1652470333.688368, private_se
| qNum: 47985, private_identity: MTMount, private_origin: 35669, elevatio
| n: 60.218476669063456, elevationVelocity: 0.0031492027756239442, azimut
| h: 119.21955452542977, azimuthVelocity: 0.0009735030670890412, taiTime:
| 1652470333.7475684, trackId: 4, tracksys: SIDEREAL, radesys: ICRS, prior
| ity: 0
| setup.MTCS DEBUG: [Tel]: Az = +119.220[ +0.0]; El = +060.218[ +0.0] [R
| ot]: +001.762[ -0.0] [Dome] Az = +000.000; El = +000.000
| setup.MTCS DEBUG: Dome azimuth in position.
| setup.MTCS DEBUG: Dome elevation in position.
| setup.MTCS DEBUG: No new in position event in the last 3.0s. Assuming MT
| Rotator in position.
| setup.MTCS DEBUG: MTRotator in position True. Waiting settle time 3.0s
| setup.MTCS DEBUG: No new in position event in the last 3.0s. Assuming MT
| Mount elevation in position.
| setup.MTCS DEBUG: MTMount elevation in position True. Waiting settle tim
| e 3.0s
| setup.MTCS DEBUG: No new in position event in the last 3.0s. Assuming MT
| Mount azimuth in position.
| setup.MTCS DEBUG: MTMount azimuth in position True. Waiting settle time
| 3.0s

```

---

clear all corrections using cmd\_resetCorrection

```
In [11]: await mtcs.rem.mtaos.cmd_resetCorrection.start()
```

```
Out[11]: <ddsutil.MTAOS_ackcmd_fd03e870 at 0x7f6d42e0ca90>
```

```
In [12]: await mtcs.rem.mtaos.cmd_issueCorrection.start(timeout=60.)
```

```
Out[12]: <ddsutil.MTAOS_ackcmd_fd03e870 at 0x7f6d42afa310>
```

---

Add 1um of z7 to the system via OFC, issue the corrections.

Compare the corrections sent vs forces and position changes applied. This is currently done in a separate notebook or on Chronograf.

```
In [13]: wavefront_errors = np.zeros(19)
```

```
In [14]: wavefront_errors[3]=1.0
```

```
In [15]: await mtcs.rem.mtaos.cmd_addAberration.set_start(wf=wavefront_errors, timeout=60.)
```

```
Out[15]: <ddsutil.MTAOS_ackcmd_fd03e870 at 0x7f6d42cdfc70>
```

```
In [16]: await mtcs.rem.mtaos.cmd_issueCorrection.start(timeout=60.)
```

```
Out[16]: <ddsutil.MTAOS_ackcmd_fd03e870 at 0x7f6dacc79790>
```

---

Make plots using telemetry from each component to verify the changes in the DOFs. This step does not currently involve running any commands in this notebook. This step must be verified using a separate notebook.

---

Put M2 hexapod in DISABLED state (so that we can test command rejection).

```
In [17]: await mtcs.set_state(salobj.State.DISABLED, components=["mthexapod_2"])
| setup.MTCS DEBUG: [mthexapod_2]::[<State.ENABLED: 2>, <State.DISABLED: 1>]
| setup.MTCS INFO: All components in <State.DISABLED: 1>.
```

---

Add 1um of z7 to the system via OFC. Expect m2 hexapod corrections are rejected, and all other corrections applied, then undone.

```
In [18]: await mtcs.rem.mtaos.cmd_addAberration.set_start(wf=wavefront_errors, timeout=60.)
```

```
Out[18]: <ddsutil.MTAOS_ackcmd_fd03e870 at 0x7f6d3a6b2280>
```

```
In [19]: await mtcs.rem.mtaos.cmd_issueCorrection.start(timeout=60.)
```

---

```

AckError                                     Traceback (most recent call last)
Input In [19], in <cell line: 1>()
----> 1 await mtcs.rem.mtaos.cmd_issueCorrection.start(timeout=60.)

File /opt/lsst/software/stack/conda/miniconda3-py38_4.9.2/envs/lsst-scipipe
-3.0.0/lib/python3.8/site-packages/lsst/ts/salobj/topics/remote_command.py:
485, in RemoteCommand.start(self, data, timeout, wait_done)
      481 cmd_info = CommandInfo(
      482     remote_command=self, seq_num=seq_num, wait_done=wait_done
      483 )
      484 self.salinfo._running_cmds[seq_num] = cmd_info
--> 485 return await cmd_info.next_ackcmd(timeout=timeout)

File /opt/lsst/software/stack/conda/miniconda3-py38_4.9.2/envs/lsst-scipipe
-3.0.0/lib/python3.8/site-packages/lsst/ts/salobj/topics/remote_command.py:
195, in CommandInfo.next_ackcmd(self, timeout)
      193 ackcmd = await self._wait_task
      194 if ackcmd.ack in self.failed_ack_codes:
--> 195     raise base.AckError(msg="Command failed", ackcmd=ackcmd)
      196 return ackcmd
      197 except asyncio.TimeoutError:

AckError: msg='Command failed', ackcmd=(ackcmd private_seqNum=743583437, ac
k=<SalRetCode.CMD_FAILED: -302>, error=1, result="Failed: Failed to apply c
orrection to: ['m2hex']. ")

```

---

Re-enable M2 hexapod Make it ready for AOS

```

In [20]: await mtcs.set_state(salobj.State.ENABLED, components=["mthexapod_2"])

| setup.MTCS DEBUG: [mthexapod_2]::<State.DISABLED: 1>, <State.ENABLED: 2
| >]
| setup.MTCS INFO: All components in <State.ENABLED: 2>.

```

---

Re-issue the correction.

```

In [21]: await mtcs.rem.mtaos.cmd_addAberration.set_start(wf=wavefront_errors, timeou

Out[21]: <ddsutil.MTAOS_ackcmd_fd03e870 at 0x7f6d3a68d0d0>

In [22]: await mtcs.rem.mtaos.cmd_issueCorrection.start(timeout=60.)

Out[22]: <ddsutil.MTAOS_ackcmd_fd03e870 at 0x7f6d4b41a760>

```

---

Reject the latest corrections.

```

In [23]: await mtcs.rem.mtaos.cmd_rejectCorrection.start()

Out[23]: <ddsutil.MTAOS_ackcmd_fd03e870 at 0x7f6d3a6a9640>

```

```
In [24]: await mtcs.rem.mtaos.cmd issueCorrection.start(timeout=60.)
```

```
Out[24]: <ddsutil.MTAOS_ackcmd_fd03e870 at 0x7f6d4b738e80>
```

Add 2um of z7 via OFC

```
In [25]: wavefront_errors[3] = 2.0
```

```
In [26]: wavefront errors
```

[illegible]

```
In [27]: await mtcs.rem.mtaos.cmd_addAberration.set_start(wf=wavefront_errors, timeout=10)
```

```
Out[27]: <ddsutil.MTA0S_ackcmd_fd03e870 at 0x7f6d3a68e5e0>
```

```
In [28]: await mtcs.rem.mtaos.cmd issueCorrection.start(timeout=60.)
```

```
Out[28]: <ddsutil.MTAOS_ackcmd_fd03e870 at 0x7f6d3a7e8d60>
```

Stop Tracking

```
In [29]: await mtcs.stop_tracking()
```

```
| setup.MTCS DEBUG: Stop tracking.
```

Wrap up. Put each component to the following states: mtaos --> standby m1m3 --> standby m2 --> standby camera hex --> standby m2 hex --> standby

```
In [30]: await mtcs.set state(salobj.State.STANDBY, components=["mtaos"])
```

```
setup.MTCS DEBUG: [mtaos]::[<State.ENABLED: 2>, <State.DISABLED: 1>, <State.STANDBY: 5>]
```

```
setup.MTCS INFO: All components in <State.STANDBY: 5>.
```

```
In [31]: await mtcs.lower m1m3()
```

```
setup.MTCS DEBUG: M1M3 current detailed state {<DetailedState.ACTIVEENGI  
NEERING: 11>, <DetailedState.ACTIVE: 7>}, executing command...
```

```
setup.MTCS DEBUG: process as completed...
```

```
setup.MTCS DEBUG: M1M3 detailed state 8
```

```
setup.MTCS DEBUG: mtm1m3: <State.ENABLED: 2>
```

```
setup.MTCS DEBUG: mtm1m3: <State.ENABLED: 2>
```

```
setup.MTCS DEBUG: M1M3 detailed state 5
```

```
In [32]: await mtcs.set_state(salobj.State.STANDBY, components=["mtm1m3"])
```

```
setup.MTCS DEBUG: [mtm1m3]::[<State.ENABLED: 2>, <State.DISABLED: 1>, <State.STANDBY: 5>]
```

```
setup.MTCS INFO: All components in <State.STANDBY: 5>.
```

```
In [33]: await mtcs.set_state(salobj.State.STANDBY, components=["mtm2"])
```

```
setup.MTCS ERROR: Unable to transition mtm2 to <State.STANDBY: 5> NoneType: None
```

```
.
Traceback (most recent call last):
  File "/opt/lsst/software/stack/conda/miniconda3-py38_4.9.2/envs/lsst-scipipe-3.0.0/lib/python3.8/site-packages/lsst/ts/salobj/topics/remote_command.py", line 193, in next_ackcmd
    ackcmd = await self._wait_task
  File "/opt/lsst/software/stack/conda/miniconda3-py38_4.9.2/envs/lsst-scipipe-3.0.0/lib/python3.8/site-packages/lsst/ts/salobj/topics/remote_command.py", line 218, in _basic_next_ackcmd
    ackcmd = await asyncio.wait_for(
  File "/opt/lsst/software/stack/conda/miniconda3-py38_4.9.2/envs/lsst-scipipe-3.0.0/lib/python3.8/asyncio/tasks.py", line 501, in wait_for
    raise exceptions.TimeoutError()
asyncio.exceptions.TimeoutError
```

During handling of the above exception, another exception occurred:

```
Traceback (most recent call last):
  File "/opt/lsst/software/stack/conda/miniconda3-py38_4.9.2/envs/lsst-scipipe-3.0.0/lib/python3.8/site-packages/lsst/ts/salobj/csc_utils.py", line 157, in set_summary_state
    await cmd.start(timeout=timeout)
  File "/opt/lsst/software/stack/conda/miniconda3-py38_4.9.2/envs/lsst-scipipe-3.0.0/lib/python3.8/site-packages/lsst/ts/salobj/topics/remote_command.py", line 485, in start
    return await cmd_info.next_ackcmd(timeout=timeout)
  File "/opt/lsst/software/stack/conda/miniconda3-py38_4.9.2/envs/lsst-scipipe-3.0.0/lib/python3.8/site-packages/lsst/ts/salobj/topics/remote_command.py", line 209, in next_ackcmd
    raise base.AckTimeoutError(
lsst.ts.salobj.base.AckTimeoutError: msg='Timed out waiting for command acknowledgement', ackcmd=(ackcmd private_seqNum=1803175370, ack=<SalRetCode.CMD_NOACK: -301>, error=0, result='No command acknowledgement seen')
```

The above exception was the direct cause of the following exception:

```
Traceback (most recent call last):
  File "/opt/lsst/software/stack/conda/miniconda3-py38_4.9.2/envs/lsst-scipipe-3.0.0/lib/python3.8/site-packages/lsst/ts/salobj/csc_utils.py", line 159, in set_summary_state
    raise RuntimeError(
RuntimeError: Error on cmd=cmd_standby, initial_state=2: msg='Timed out waiting for command acknowledgement', ackcmd=(ackcmd private_seqNum=1803175370, ack=<SalRetCode.CMD_NOACK: -301>, error=0, result='No command acknowledgement seen')
```



```

-----
RuntimeError                                Traceback (most recent call last)
Input In [33], in <cell line: 1>()
----> 1 await mtcs.set_state(salobj.State.STANDBY, components=["mtm2"])

```

File ~/auto-op-env-packages/ts\_observatory\_control/python/lsst/ts/observatory/control/remote\_group.py:732, in RemoteGroup.set\_state(self, state, overrides, components)

```

    729         self.log.debug(f"[{comp}]:{ret_val[i]!r}")
    731 if error_flag:
--> 732     raise RuntimeError(
    733         f"Failed to transition {failed_components} to "
    734         f"{salobj.State(state)!r}."
    735     )
    736 else:
    737     self.log.info(f"All components in {salobj.State(state)!r}.")

```

RuntimeError: Failed to transition ['mtm2'] to <State.STANDBY: 5>.

```
In [34]: await mtcs.set_state(salobj.State.STANDBY, components=["mthexapod_1"])
```

```

| setup.MTCS DEBUG: [mthexapod_1]: [<State.ENABLED: 2>, <State.DISABLED: 1>, <State.STANDBY: 5>]
| setup.MTCS INFO: All components in <State.STANDBY: 5>.

```

```
In [35]: await mtcs.set_state(salobj.State.STANDBY, components=["mthexapod_2"])
```

```

| setup.MTCS DEBUG: [mthexapod_2]: [<State.ENABLED: 2>, <State.DISABLED: 1>, <State.STANDBY: 5>]
| setup.MTCS INFO: All components in <State.STANDBY: 5>.

```

```
In [36]: await mtcs.standby()
```

```

| setup.MTCS DEBUG: [mtmount]: [<State.ENABLED: 2>, <State.DISABLED: 1>, <State.STANDBY: 5>]
| setup.MTCS DEBUG: [mtptg]: [<State.ENABLED: 2>, <State.DISABLED: 1>, <State.STANDBY: 5>]
| setup.MTCS DEBUG: [mtaos]: [<State.STANDBY: 5>]
| setup.MTCS DEBUG: [mtm1m3]: [<State.STANDBY: 5>]
| setup.MTCS DEBUG: [mtm2]: [<State.STANDBY: 5>]
| setup.MTCS DEBUG: [mthexapod_1]: [<State.STANDBY: 5>]
| setup.MTCS DEBUG: [mthexapod_2]: [<State.STANDBY: 5>]
| setup.MTCS DEBUG: [mtrotator]: [<State.ENABLED: 2>, <State.DISABLED: 1>, <State.STANDBY: 5>]
| setup.MTCS DEBUG: [mtdome]: [<State.ENABLED: 2>, <State.DISABLED: 1>, <State.STANDBY: 5>]
| setup.MTCS DEBUG: [mtdometrajectory]: [<State.ENABLED: 2>, <State.DISABLED: 1>, <State.STANDBY: 5>]
| setup.MTCS INFO: All components in <State.STANDBY: 5>.

```

```
In [ ]:
```