What should I do if scriptForPI breaks at scriptForImagingPrep step due to an error?

Melissa Hoffman - 2022-03-09 - Historical Articles

For projects that were imaged using the pipeline after April 2017, scriptForPI.py may produce an error when trying to restore data.

```
---------------------------------------------------------------------------
RuntimeError                 Traceback (most recent call last)
/home/casa/packages/RHEL6/release/casa-release-4.7.2-el6/lib/python2.7/casapy.py in <module>()
----> 1

/home/casa/packages/RHEL6/release/casa-release-4.7.2-el6/lib/python2.7/casapy.py in <module>()
341 if impreppresent:
    342     print 'Executing scriptForImagingPrep.py ...'
--> 343     execfile('../script/scriptForImagingPrep.py')

/home/casa/packages/RHEL6/release/casa-release-4.7.2-el6/lib/python2.7/casapy.py in <module>()
    341 if impreppresent:
    342     print 'Executing scriptForImagingPrep.py ...'
    --> 343     execfile('../script/scriptForImagingPrep.py')

344

345 if (savingslevel>=3) and os.path.exists('calibrated.ms'):

/home/casa/packages/RHEL6/release/casa-release-4.7.2-el6/lib/python2.7/casapy.py in <module>()
42 os.system('rm -rf ' + concatvis + '.flagversions')
```
concat(vis=vislist2,
        concatvis=concatvis)

/home/casa/packages/RHEL6/release/casa-release-4.7.2-el6/lib/python2.7/concat_cli.pyc in __call__(self, vis, concatvis, freqtol, dirtol, respectname, timesort, copypointing, visweightscale, forcesingleephemfield)

#return False

---> casac.casac.utils().verify(mytmp, trec['concat'], True)

scriptstr=['']

saveinputs = self.__globals__['saveinputs']

/home/casa/packages/RHEL6/release/casa-release-4.7.2-el6/lib/python2.7/__casac__/utils.pyc in verify(self, *args, **kwargs)

RuntimeError: Parameter verification failed

scriptForPI.py will not split out the calibrated science SPWs into uid_*.ms.split.cal if the imaging pipeline was used, which
scriptForImagingPrep.py looks for, causing the scriptForPI.py to break.

If you only want the calibrated data, it is sufficient to stop here and use the uid_*.ms, keeping in mind that the calibrated data is stored in the CORRECTED data column, and all sources are in the ms.

If you want to run the imaging pipeline, you can start from here, without changing the uid_*.ms.

If you want to run scriptForPI.py all the way through, change

```python
if ( nummkim > 1 ):
    print "Imaging pipeline was used."
    usedimpipe = False
```

Tags
calibration
restore
restoredata
restoring
scriptforpi