TIF used by the Test Camera OS

- TIF commands: SETUP, SETINSD
- TIF functions: tifGetFitsStart, tifGetFitsEnd
- instrument workstation: local copy of TCS OLDB

TIF access in the Test Camera OS is centralized with a single interface class (ins0TCS_INTERFACE).

TIF used by the Test Camera Templates

Templates access to TCS is centralized with a single itcl class (tcblibTCS) that logically belongs to OS. This class:

- is implemented using TIF facilities (TCS is **never** accessed directly)
- contains specialized functions needed by Test Camera templates, e.g.:
  IsAgActive: is the telescope guiding ?
  IsAoActive: is Active Optics running ?
- takes care of checking OS state and modes (is TCS available ?).

```
proc tb_img_acq { parArrays obsInfo } {
    # 1. begin
    ...
    catch { delete object obs }
    catch { delete object tcs }
    tcblibOBS obs
    tcblibTCS tcs
    tplLog "Starting acquisition."

    # 2. check that TCS is used
    if { [tcs IsUsed] == 0 } {
         tplLog "Warning: TCS is not available. \
                 No commands are sent to TCS."
    }

    # 3. preset telescope
    obs Setup "-file $TPL(REFSUP) -function ..."
    ...

    # 6. wait for AO and autoguiding
    checkAbortFlag
    if { [tcs IsAgActive -wait] == 0 } {
          error "Autoguiding not started."
    }
    checkAbortFlag
    if { [tcs IsAoActive -wait] == 0 } {
          error "Active optics not started."
    }

    tplLog "Acquisition template finished."
    return {}
}
#____oOo____
```

```
# TCS procedures
class tcblibTCS {
    ...
    # method IsAgActive
    #
    # return 1 if the autoguider is active, 0 otherwise.
    # if option = ‘-wait’: ask to start autoguiding and wait.
    method IsAgActive { {option “”} } {
        # 1. If TCS not used: return
        if { [IsUsed] == 0 && $debug == 0 } {
            return 1
        }

        # 2. check option
        ...

        # 3. get autoguider state
        set agActive 0
        if { $debug } {
            set agState [uif GetValue “Autoguiding State” 122]
        } else {
            set agState [seq_dbReadSymbolic $ptAgState]
        }
        if { $agState == 122 } {
            set agActive 1
            set wait 0
        }

        ...
        return $agActive
    }
    ...
}
```

Test Camera #2 Rotating Pupil Mask

The rotating pupil mask has to follow the parallactic angle of the telescope.