
CSP LMC Prototype Documentation

Release 1.0.0

SKA Organization

Nov 22, 2019

Contents:

1	CspMaster Class Documentation	1
2	CSP Subarrays	13
3	CspMaster Class Documentation	29
4	Indices and tables	31
	Python Module Index	33
	Index	35

CspMaster Class Documentation

CspMaster Tango device prototype

CSPMaster TANGO device class for the CSPMaster prototype

```
class CspMaster.CspMaster(*args, **kwargs)
    Bases: sphinx.ext.autodoc.importer._MockObject
```

CSPMaster TANGO device class for the CSPMaster prototype

CspMidCbf

Device property

The CspMidCbf FQDN.

Type: DevString

CspMidPss

Device property

The CspMidPss FQDN.

Type: DevString

CspMidPst

Device property

The CspMidPst FQDN.

Type: DevString

CspSubarrays

Device property

The CspSubarrays FQDN.

Type: array of DevString

SearchBeams

Device property

The CSP Search Beam Capabilities FQDNs.

Type: array of DevString

TimingBeams

Device property

The CSP Timing Beam Capabilities FQDNs.

Type: array DevString

VlbiBeams

Device property

The CSP Vlbi Beam Capabilities FQDNs.

Type: array of DevString

CspTelState

Device property

The CSP TelState FQDN.

Type: DevString

adminMode

Class attribute

The device administrative mode.

Note: This attribute is defined in SKABaseDevice Class from which CspMaster class inherits. To override the attribute *write* method, the *adminMode* attribute is added again (“overload” button enabled in POGO).

commandProgress

Class attribute

Percentage progress implemented for commands that result in state/mode transitions for a large number of components and/or are executed in stages (e.g power up, power down)

Type: DevUShort

cspCbfState

Class attribute

The CbfMaster *State* attribute value.

Type: DevState

cspPssState

Class attribute

The PssMaster *State* attribute value.

Type: DevState

cspPstState

Class attribute

The PstMaster *State* attribute value.

Type: DevState

cspCbfHealthState

Class attribute

The CbfMaster *healthState* attribute value.

Type: DevUShort

cspPssHealthState

Class attribute

The PssMaster *healthState* attribute value.

Type: DevUShort

cspPstHealthState

Class attribute

The PstMaster *healthState* attribute value.

Type: DevUShort

cbfMasterAddress

Class attribute

The CbfMaster FQDN.

Type: DevString

pssMasterAddress

Class attribute

The PssMaster FQDN.

Type: DevString

pstMasterAddress

Class attribute

The PstMaster FQDN.

Type: DevString

cbfAdminMode

Class attribute

The CbfMaster *adminMode* attribute value.

Type: DevUShort

pssAdminMode

Class attribute

The PssMaster *adminMode* attribute value.

Type: DevUShort

pstAdminMode

Class attribute

The PstMaster *adminMode* attribute value.

Type: DevUShort

availableCapabilities

Class attribute

The list of available instances of each capability type.

Note: This attribute is defined in SKAMaster Class from which CspMaster class inherits. To override the attribute *read* method, the *availableCapabilities* attribute is added again (“overload” button enabled in

POGO).

reportSearchBeamState

Class attribute

The *State* attribute value of the CSP SearchBeam Capabilities.

Type: array of DevState.

reportSearchBeamHealthState

Class attribute

The *healthState* attribute value of the CSP SearchBeam Capabilities.

Type: array of DevUShort.

reportSearchBeamAdminMode

Class attribute

The *adminMode* attribute value of the CSP SearchBeam Capabilities.

Type: array of DevUShort.

reportTimingBeamState

Class attribute

The *State* attribute value of the CSP TimingBeam Capabilities.

Type: array of DevState.

reportTimingBeamHealthState

Class attribute

The *healthState* attribute value of the CSP TimingBeam Capabilities.

Type: array of DevUShort.

reportTimingBeamAdminMode

Class attribute

The *adminMode* attribute value of the CSP TimingBeam Capabilities.

Type: array of DevUShort.

reportVlbiBeamState

Class attribute

The *State* attribute value of the CSP VlbiBeam Capabilities.

Type: array of DevState.

reportVlbiBeamHealthState

Class attribute

The *healthState* attribute value of the CSP VlbiBeam Capabilities.

Type: array of DevUShort.

reportVlbiBeamAdminMode

Class attribute

The *adminMode* attribute value of the CSP VlbiBeam Capabilities.

Type: array of DevUShort.

cspSubarrayAddress*Class attribute*

The CSPSubarray FQDNs.

Type: Array of DevString**searchBeamCapAddress***Class attribute*

The CSP SearchBeam Capabilities FQDNs.

Type: Array of DevString**timingBeamCapAddress***Class attribute*

The CSP TimingBeam Capabilities FQDNs.

Type: Array of DevString**vlbiCapAddress***Class attribute*

The CSP VlbiBeam Capabilities FQDNs.

Type: Array of DevString**receptorMembership***Class attribute*

The receptors subarray affiliation.

Type: array of DevUShort.**searchBeamMembership***Class attribute*

The SearchBeam Capabilities subarray affiliation.

Type: array of DevUShort.**timingBeamMembership***Class attribute*

The TimingBeam Capabilities subarray affiliation.

Type: array of DevUShort.**vlbiBeamMembership***Class attribute*

The VlbiBeam Capabilities subarray affiliation.

Type: array of DevUShort.**availableReceptorIDs***Class attribute*

The available receptor IDs.

Type: array of DevUShort.**reportVCCState***TANGO Forwarded attribute.*The *State* attribute value of the Mid CBF Very Coarse Channel TANGO Devices.

Type: array of DevState.

__root_att: /mid_csp_cbf/sub_elt/master/reportVCCState

Note: If the __root_att attribute property is not specified in the TANGO DB or the value doesn't correspond to a valid attribute FQDN, the CspMaster State goes in ALARM.

reportVCCHealthState

TANGO Forwarded attribute.

The healthState attribute value of the Mid CBF Very Coarse Channel TANGO Devices.

Type: an array of DevUShort.

__root_att: /mid_csp_cbf/sub_elt/master/reportVCCHealthState

reportVCCAdminMode

TANGO Forwarded attribute.

The adminMode attribute value of the Mid CBF Very Coarse Channel TANGO devices.

Type: array of DevUShort.

__root_att: /mid_csp_cbf/sub_elt/master/reportVccAdminMode

reportFSPState

TANGO Forwarded attribute.

The State attribute value of the Mid CBF Frequency Slice Processor TANGO devices.

Type: array of DevState.

__root_att: /mid_csp_cbf/sub_elt/master/reportFSPHealthState

reportFSPHealthState

TANGO Forwarded attribute.

The healthState attribute value of the Mid CBF Frequency SLice Processor TANGO Devices.

Type: an array of DevUShort.

__root_att: /mid_csp_cbf/sub_elt/master/reportFSPHealthState

reportFSPAdminMode

TANGO Forwarded attribute.

The adminMode attribute value of the Mid CBF Frequency SLice Processor TANGO Devices.

Type: an array of DevUShort.

__root_att: /mid_csp_cbf/sub_elt/master/reportFSPAdminMode

fspMembership

TANGO Forwarded attribute.

The subarray affiliation of the Mid CBF Frequency SLice Processor TANGO Devices.

Type: an array of DevUShort.

__root_att: /mid_csp_cbf/sub_elt/master/fspMembership

vccMembership

TANGO Forwarded attribute.

The subarray affiliation of the Mid CBF VCC TANGO Devices.

Type: an array of DevUShort.

__root_att: /mid_csp_cbf/sub_elt/master/reportVCCSubarrayMembership

init_device()

always_executed_hook()

delete_device()

Method called on stop/reinit of the device. Release all the allocated resources.

write_adminMode(value)

Write attribute method.

Set the administration mode for the whole CSP element.

Parameters value – one of the administration mode value (ON-LINE, OFF-LINE, MAINTENANCE, NOT-FITTED, RESERVED).

Returns None

read_commandProgress()

Read attribute method.

Returns The commandProgress attribute value.

read_cspCbfState()

Read attribute method.

Returns The CBF Sub-element *State* attribute value.

read_cspPssState()

Read attribute method.

Returns The PSS Sub-element *State* attribute value.

read_cspPstState()

Read attribute method.

Returns The PST Sub-element *State* attribute value.

read_cspCbfHealthState()

Read attribute method.

Returns The CBF Sub-element *healthState* attribute value.

read_cspPssHealthState()

Read attribute method.

Returns The PSS Sub-element *healthState* attribute value.

read_cspPstHealthState()

Read attribute method.

Returns The PST Sub-element *healthState* attribute value.

read_cbfMasterAddress()

Read attribute method.

Returns Return the CBS sub-element Master TANGO Device address.

read_pssMasterAddress()

Read attribute method.

Returns The PSS sub-element Master TANGO Device address.

read_pstMasterAddress ()

Read attribute method.

Returns The PST sub-element Master TANGO Device address.

read_cbfAdminMode ()

Read attribute method.

Returns The CBF sub-element *adminMode* attribute value.

write_cbfAdminMode (value)

Write attribute method.

Set the CBF sub-element *adminMode* attribute value.

Parameters value – one of the administration mode value (ON-LINE, OFF-LINE, MAINTENANCE, NOT-FITTED, RESERVED).

Returns None

Raises `tango.DevFailed` – raised when there is no DeviceProxy providing interface to the CBF sub-element Master, or an exception is caught in command execution.

read_pssAdminMode ()

Read attribute method.

Returns The PSS sub-element *adminMode* attribute value.

write_pssAdminMode (value)

Write attribute method.

Set the PSS sub-element *adminMode* attribute value.

Parameters value – one of the administration mode value (ON-LINE, OFF-LINE, MAINTENANCE, NOT-FITTED, RESERVED).

Returns None

Raises

- `tango.DevFailed` – raised when there is no DeviceProxy providing
- interface to the PSS sub-element Master, or an exception is caught
- in command execution.

read_pstAdminMode ()

Read attribute method.

Returns The PST sub-element *adminMode* attribute value.

write_pstAdminMode (value)

Write attribute method.

Set the PST sub-element *adminMode* attribute value.

Parameters value – one of the administration mode value (ON-LINE, OFF-LINE, MAINTENANCE, NOT-FITTED, RESERVED).

Returns None

Raises

- `tango.DevFailed` – raised when there is no DeviceProxy providing
- interface to the PST sub-element Master, or an exception is caught in command execution.

read_availableCapabilities()

Override read attribute method.

Returns A list of strings with the number of available resources for each capability/resource type.

Example

["Receptors:95", "SearchBeam:1000", "TimingBeam:16", "VlbiBeam:20"]

Raises `tango.DevFailed`

read_reportSearchBeamState()

Class attribute method.

Returns

The *State* value of the CSP SearchBeam Capabilities.

Type: array of DevState.

read_reportSearchBeamHealthState()

Class attribute method.

Returns

The *healthState* attribute value of the CSP SearchBeam Capabilities.

Type: array of DevUShort

read_reportSearchBeamAdminMode()

Class attribute method.

Returns

The *adminMode* of the CSP SearchBeam Capabilities.

Type: array of DevUShort

read_reportTimingBeamState()

Class attribute method.

Returns

The *State* value of the CSP TimingBeam Capabilities.

Type: array of DevState.

read_reportTimingBeamHealthState()

Class attribute method.

Returns

The *healthState* value of the CSP TimingBeam Capabilities.

Type: array of DevUShort.

read_reportTimingBeamAdminMode()

Class attribute method.

Returns

The *adminMode* value of the CSP TimingBeam Capabilities.

Type: array of DevUShort.

read_reportVlbiBeamState ()

Class attribute method.

Returns

The *State* value of the CSP VlbiBeam Capabilities.

Type: array of DevState.

read_reportVlbiBeamHealthState ()

Class attribute method.

Returns

The *healthState* value of the CSP VlbiBeam Capabilities.

Type: array of DevUShort.

read_reportVlbiBeamAdminMode ()

Read attribute method.

Returns

The *adminMode* value of the CSP VlbiBeam Capabilities.

Type: array of DevUShort.

read_cspSubarrayAddress ()

Class attribute method.

Returns

The CSP Subarrays FQDNs if the associated Device Property is defined, otherwise None.

Type: array of DevString

read_searchBeamCapAddress ()

Class attribute method.

Returns

The CSP SearchBeam Capabilities FQDNs if the associated Device Property is defined, otherwise None.

Type: array of DevString

read_timingBeamCapAddress ()

Class attribute method.

Returns

The CSP TimingBeam Capabilities FQDNs if the associated Device Property is defined, otherwise None.

Type: array of DevString

read_vlbiCapAddress ()

Class attribute method.

Returns

The CSP VlbiBeam Capabilities FQDNs if the associated Device Property is defined, otherwise None.

Type: array of DevString

read_receptorMembership ()

Class attribute method.

Returns The subarray affiliation of the receptors.

read_searchBeamMembership()

Class attribute method.

Returns The subarray affiliation of the Search Beams.

read_timingBeamMembership()

Class attribute method.

Returns The subarray affiliation of the Timing Beams.

read_vlbiBeamMembership()

Class attribute method.

Returns The subarray affiliation of the VLbi Beams.

read_availableReceptorIDs()

Class attribute method.

Returns

The list of the available receptors IDs. The list includes all the receptors that are not assigned to any subarray and, from the side of CSP, are considered “full working”. This means:

- a valid link connection receptor-VCC
- the connected VCC healthState OK

Type: array of DevUShort

Raises `tango.DevFailed` – if there is no DeviceProxy providing interface to the CBF sub-element Master Device or an error is caught during command execution.

is_On_allowed()

TANGO is_allowed method

Command *On* is allowed when:

- state is `STANDBY` and `adminMode` = `MAINTENACE` or `ONLINE` (end state = `ON`)
- state is `DISABLE` and `adminMode` = `MAINTENACE` or `ONLINE` (end state = `ON`)

Returns True if the method is allowed, otherwise False.

On(*argin*)

Class method

Switch-on the CSP sub-elements specified by the input argument. If no argument is specified, the command is issued on all the CSP sub-elements.

The command is executed if the *AdminMode* is `ONLINE` or `MAINTENANCE`.

If the *AdminMode* is `OFFLINE`, `NOT-FITTED` or `RESERVED`, the method throws an exception.

Parameters

- **argin** – the list of sub-element FQDNs to switch-on or an empty list to switch-on the whole CSP Element.
- **Type** – `DevVarStringArray`

Returns None

Raises `tango.DevFailed` – an exception is caught processing the On command for the CBF sub-element or there are no DeviceProxy providing interface to the CSP sub-elements or the AdminMode is not correct.

is_Off_allowed()

TANGO is_allowed method

Command *Off* is allowed when the device *State* is STANDBY.

Returns True if the method is allowed, otherwise False.

Off (*argin*)

Switch-off the CSP sub-elements specified by the input argument. If no argument is specified, the command is issued to all the CSP sub-elements.

Parameters

- **argin** – The list of sub-elements to switch-off. If the array length is 0, the command applies to the whole CSP Element. If the array length is > 1, each array element specifies the FQDN of the CSP SubElement to switch OFF
- **Type** – DevVarStringArray

Returns None

is_Standby_allowed()

TANGO is_allowed method

Command *Standby* is allowed when the device *State* is ON, DISABLE or ALARM.

Returns True if the method is allowed, otherwise False.

Standby (*argin*)

Transit to STANDBY the CSP sub-elements specified by the input argument. If no argument is specified, the command is issued to all the CSP sub-elements.

Parameters

- **argin** – The list of the Sub-element devices FQDNs
- **Type** – DevVarStringArray

Returns None

Raises `tango.DevFailed` – if command fails or if no DeviceProxy associated to the FQDNs.

`CspMaster.main` (*args=None, **kwargs*)

CSP Subarrays

The core CSP functionality, configuration and execution of signal processing, is configured, controlled and monitored via subarrays.

CSP Subarray makes provision to TM to configure a subarray, select Processing Mode and related parameters, specify when to start/stop signal processing and/or generation of output products. TM accesses directly a CSP Subarray to:

- Assign resources
- Configure a scan
- Control and monitor states/operations

2.1 Resources assignment

The assignment of Capabilities to a subarray (*subarray composition*) is performed in advance of a scan configuration. Assignable Capabilities for CSP Mid subarrays are:

- receptors and the associated CBF Very Coarse Channelizers: each VCC processes the input from one receptor.
- CBF Frequency Slice Processors performing one of the available Processing Mode Functions: Correlation, Pulsar Timing Beamforming, Pulsar Search Beamforming, VLBI Beamforming.
- tied-array beams: Search Beams, Timing Beams and Vlbi Beams.

In general resource assignment to a subarray is exclusive, but in some cases (FSPs) the same Capability instance may be used in shared manner by more than one subarray.

Note: of all the listed Capabilities, only FSPs are assigned to subarrays via a scan configuration.

2.1.1 Inherent Capabilities

Each CSP subarray has also four permanently assigned *inherent Capabilities*:

- Correlation

- PSS
- PST
- VLBI

An inherent Capability can be enabled or disabled, but cannot assigned or removed to/from a subarray. They correspond to the CSP Mid Processing Modes and are configured via a scan configuration.

2.2 Scan configuration

TM provides a complete scan configuration to a subarray via an ASCII JSON encoded string. Parameters specified via a JSON string are implemented as TANGO Device attributes and can be accessed and modified directly using the built-in TANGO method *write_attribute*. When a complete and coherent scan configuration is received and the subarray configuration (or re-configuration) completed, the subarray it's ready to observe.

2.3 Control and Monitoring

Each CSP Subarray maintains and report the status and state transitions for the CSP subarray as a whole and for the individual assigned resources.

In addition to pre-configured status reporting, a CSP Subarray makes provision for the TM and any authorized client, to obtain the value of any subarray attribute.

2.4 Class Documentation

CspSubarray TANGO Device Class

CSP subarray functionality is modeled via a TANGO Device Class, named *CspSubarray*. This class exports a set of attributes and methods required for configuration, control and monitoring of the subarray.

class *CspSubarray.CspSubarray* (*args, **kwargs)

CSP subarray functionality is modeled via a TANGO Device Class, named *CspSubarray*. This class exports a set of attributes and methods required for configuration, control and monitoring of the subarray.

CbfSubarrayPrefix

Class property

The CBF sub-element subarray FQDN prefix.

Type: DevString

Example

mid_csp_cbf/sub_elt/subarray_

PssSubarrayPrefix

Class property

The PSS sub-element subarray FQDN prefix.

Type: DevString

Example

mid_csp_pss/sub_elt/subarray_

CspMaster

Device property

The CspMaster FQDN.

Type: DevString

scanID

Class attribute

The identification number of the scan.

Type: DevULong64

corrInherentCap

Class attribute

The CspSubarray Correlation inherent Capability FQDN.

Type: DevString

pssInherentCap

Class attribute

The CspSubarray Pss inherent Capability FQDN.

Type: DevString

pstInherentCap

Class attribute

The CspSubarray Pst inherent Capability FQDN.

Type: DevString

vlbiInherentCap

Class attribute

The CspSubarray Vlbi inherent Capability FQDN.

Type: DevString

cbfSubarrayState

Class attribute

The CBF sub-element subarray State attribute value.

Type: DevState

pssSubarrayState

Class attribute

The PSS sub-element subarray State attribute value.

Type: DevState

cbfSubarrayHealthState

Class attribute

The CBF sub-element subarray healthState attribute value.

Type: DevEnum

enum_labels: ["OK", "DEGRADED", "FAILED", "UNKNOWN",]

pssSubarrayHealthState

Class attribute

The PSS sub-element subarray healthState attribute value.

Type: DevEnum

enum_labels: ["OK", "DEGRADED", "FAILED", "UNKNOWN",]

cbfSubarrayObsState

Class attribute

The CBF sub-element subarray obsState attribute value.

Type: DevEnum

enum_labels: ["IDLE", "CONFIGURING", "READY", "SCANNING", "PAUSED", "ABORTED", "FAULT",]

pssSubarrayObsState

Class attribute

The PSS sub-element subarray obsState attribute value.

Type: DevEnum

enum_labels: ["IDLE", "CONFIGURING", "READY", "SCANNING", "PAUSED", "ABORTED", "FAULT",]

pssSubarrayAddr

Class attribute

The PSS sub-element subarray FQDN.

Type: DevString

cbfSubarrayAddr

Class attribute

The CBF sub-element subarray FQDN.

Type: DevString

validScanConfiguration

Class attribute

The last valid scan configuration JSON-encoded string.

Type: DevString

fsp

Class attribute

The list of receptor IDs assigned to the subarray.

Type: array of DevUShort

vcc

Class attribute

The list of VCC IDs assigned to the subarray.

Type: array of DevUShort

searchBeams

Class attribute

The list of Search Beam Capability IDs assigned to the subarray.

Type: array of DevUShort

timingBeams

Class attribute

The list of Timing Beam Capability IDs assigned to the subarray.

Type: array of DevUShort

vlbiBeams

Class attribute

The list of Vlbi Beam Capability IDs assigned to the subarray.

Type: array of DevUShort

searchBeamsState

Class attribute

The *State* attribute value of the Search Beam Capabilities assigned to the subarray.

Type: array of DevState

timingBeamsState

Class attribute

The *State* attribute value of the Timing Beam Capabilities assigned to the subarray.

Type: array of DevState

vlbiBeamsState

Class attribute

The *State* attribute value of the Vlbi Beam Capabilities assigned to the subarray.

Type: array of DevState

searchBeamsHealthState

Class attribute

The *healthState* attribute value of the Search Beams Capabilities assigned to the subarray.

Type: array of DevUShort.

References

See *Common definition* paragraph for correspondences among Ushort values and label

timingBeamsHealthState

Class attribute

The *healthState* attribute value of the Timing Beams Capabilities assigned to the subarray.

Type: array of DevUShort.

References

See *Common definition* paragraph for correspondences among Ushort values and label

vlbiBeamsHealthState

Class attribute

The *healthState* attribute value of the Vlbi Beams Capabilities assigned to the subarray.

Type: array of DevUShort.

References

See *Common definition* paragraph for correspondences among Ushort values and healthState labels.

timingBeamsObsState

Class attribute

The *obsState* attribute value of the Timing Beams Capabilities assigned to the subarray.

Type: array of DevUShort.

References

See *Common definition* paragraph for correspondences among Ushort values and obsState labels.

receptors

The list of receptors assigned to the subarray.

Forwarded attribute

_root_att: mid_csp_cbf/sub_elt/subarray_N/receptors

vccState

The State attribute value of the VCCs assigned to the subarray.

Forwarded attribute

_root_att: mid_csp_cbf/sub_elt/subarray_N/reportVCCState

vccHealthState

The healthState attribute value of the VCCs assigned to the subarray.

Forwarded attribute

_root_att: mid_csp_cbf/sub_elt/subarray_N/reportVCChealthState

cbfOutputLink

The CBF Subarray output links information.

Forwarded attribute

_root_att: mid_csp_cbf/sub_elt/subarray_N/cbfOutputLinksDistribution

init_device()

Class method

Perform device initialization. during initiazlization the CspSubarray device : * connects to CSP Master and sub-element master devices

- sub-element sub-array devices with the same subarray ID
- subscribes to the sub-element subarrays State,healthState, obsState attributes for change event

always_executed_hook()

delete_device()

read_scanID()

Attribute method

Returns The scan configuration ID.

write_scanID(value)

Note: Not yet implemented.

Attribute method

Set the scan configuration ID to the defined value.

Parameters

- **value** – the scan configuration ID
- **Type** – DevUshort

Returns The scan configuration ID.

read_corrInherentCap ()

Attribute method

Returns

The CspSubarray Correlation Inherent Capability FQDN.

Type: DevString

read_pssInherentCap ()

Attribute method

Returns

The CspSubarray PSS Inherent Capability FQDN.

Type: DevString

read_pstInherentCap ()

Attribute method

Returns

The CspSubarray PST Inherent Capability FQDN.

Type: DevString

read_vlbiInherentCap ()

Attribute method

Returns

The CspSubarray VLBI Inherent Capability FQDN.

Type: DevString

read_cbfSubarrayState ()

Attribute method

Returns

The CBF sub-element subarray *State* attribute value.

Type: DevState

read_pssSubarrayState ()

Attribute method

Returns

The PSS sub-element subarray *State* attribute value.

Type: DevState

read_cbfSubarrayHealthState()

Attribute method

Returns

The CBF sub-element subarray *healthState* attribute value.

Type: DevUShort

read_pssSubarrayHealthState()

Attribute method

Returns

The PSS sub-element subarray *healthState* attribute value.

Type: DevUShort

read_cbfSubarrayObsState()

Attribute method

Returns

The CBF sub-element subarray *obsState* attribute value.

Type: DevUShort

read_pssSubarrayObsState()

Attribute method

Returns

The PSS sub-element subarray *obsState* attribute value.

Type: DevUShort

read_pssSubarrayAddr()

Attribute method

Returns

The PSS sub-element subarray FQDN.

Type: DevString

read_cbfSubarrayAddr()

Attribute method

Returns

The CSP sub-element subarray FQDN.

Type: DevString

read_validScanConfiguration()

Attribute method

Returns

The last programmed scan configuration.

Type: DevString (JSON-encoded)

read_fsp()

Attribute method

Returns

The list of FSP IDs assigned to the subarray.

Type: array of DevUShort.

read_vcc()

Attribute method

Returns

The list of VCC IDs assigned to the subarray.

Type: array of DevUShort.

read_searchBeams()

Attribute method

Returns

The list of Search Beam Capability IDs assigned to the subarray.

Type: array of DevUShort.

read_timingBeams()

Attribute method

Returns

The list of Timing Beam Capability IDs assigned to the subarray.

Type: array of DevUShort.

read_vlbiBeams()

Attribute method

Returns

The list of Vlbi Beam Capability IDs assigned to the subarray.

Type: array of DevUShort.

read_searchBeamsState()

Attribute method

Returns

The Search Beam Capabilities *State* attribute value.

Type: array of DevState

read_timingBeamsState()

Attribute method

Returns

The Timing Beam Capabilities *State* attribute value.

Type: array of DevState

read_vlbiBeamsState()

Attribute method

Returns

The Vlbi Beam Capabilities *State* attribute value.

Type: array of DevState

read_searchBeamsHealthState ()

Attribute method

Returns

The Search Beam Capabilities *healthState* attribute value.

Type: array of DevUShort

read_timingBeamsHealthState ()

Attribute method

Returns

The Timing Beam Capabilities *healthState* attribute value.

Type: array of DevUShort

read_vlbiBeamsHealthState ()

Attribute method

Returns

The Vlbi Beam Capabilities *healthState* attribute value.

Type: array of DevUShort

read_timingBeamsObsState ()

Attribute method

Returns

The Timing Beam Capabilities *obsState* attribute value.

Type: array of DevUShort

is_EndScan_allowed ()

TANGO is_allowed method: filter the external request depending on the current device state.

Check if the Scan method can be issued on the subarray.

The Scan() method can be issue on a subarray if its *State* is *ON*.

Returns True if the command can be executed, otherwise False

EndScan ()

Class method End the execution of a running scan. After successful execution, the CspSubarray *ObsState* is IDLE.

Raises

- `tango.DevFailed` – if the subarray *obsState* is not SCANNING or if an exception
- is caught during the command execution.

Note: Still to implement the check on AdminMode values: the command can be processed only when the CspSubarray is *ONLINE* or *MAINTENANCE*

is_Scan_allowed ()

TANGO is_allowed method: filter the external request depending on the current device state.

Check if the Scan method can be issued on the subarray.

A scan configuration can be performed when the subarray *State* is *ON* (that is, at least one receptor is assigned to it)

Returns True if the command can be executed, otherwise False

Scan (*argin*)
Class method

Start the execution of scan.

Raises `tango.DevFailed` – if the subarray *obsState* is not `READY` or if an exception is caught during the command execution.

Note: Still to implement the check on AdminMode values: the command can be processed only when the CspSubarray is *ONLINE* or *MAINTENANCE*

is_AddReceptors_allowed ()

TANGO is_allowed method: filter the external request depending on the current device state.

Check if the AddReceptors method can be issued on the subarray.

Receptors can be added to a Subarray when its *State* is *OFF* or *ON*.

Returns True if the command can be executed, otherwise False

AddReceptors (*argin*)
Class method

Add the specified receptor IDs to the subarray.

The command can be executed only if the CspSubarray *ObsState* is *IDLE*.

Parameters

- **argin** – the list of receptor IDs
- **Type** – array of DevUShort

Returns None

Raises `tango.DevFailed` – if the CbfSubarray is not available or if an exception is caught during command execution.

Note: Still to implement the check on AdminMode values: the command can be processed only when the CspSubarray is *ONLINE* or *MAINTENANCE*

is_RemoveReceptors_allowed ()

TANGO is_allowed method: filter the external request depending on the current device state.

Check if the method can be issued on the subarray.

Re can be removed from a subarray when its *State* is *ON* or *OFF*.

Returns True if the command can be executed, otherwise False

RemoveReceptors (*argin*)

Remove the receptor IDs from the subarray.

Parameters

- **argin** – The list of the receptor IDs to remove from the subarray.
- **Type** – array of DevUShort

Returns None

Raises `tango.DevFailed` – raised if the subarray *obState* attribute is not IDLE, or when an exception is caught during command execution.

is_RemoveAllReceptors_allowed()

TANGO is_allowed method: filter the external request depending on the current device state.

Check if the method can be issued on the subarray.

Resources can be removed from a subarray when its *State* is ON or OFF-

Returns True if the command can be executed, otherwise False

RemoveAllReceptors()

Class method.

Remove all the assigned receptors from the subarray. :returns: None

Raises `tango.DevFailed` – raised if the subarray *obState* attribute is not IDLE or READY, or when an exception is caught during command execution.

is_ConfigureScan_allowed()

TANGO is_allowed method: filter the external request depending on the current device state.

Check if the ConfigureScan method can be issued on the subarray.

A scan configuration can be performed when the subarray *State* is ON (that is, at least one receptor is assigned to it)

Returns True if the command can be executed, otherwise False

ConfigureScan(argin)

Note: Part of this code (the input string parsing) comes from the CBF project developed by J.Jjang (NRC-Canada)

Class method.

Configure a scan for the subarray.

The command can be executed when the CspSubarray State is *ON* and the ObsState is *IDLE* or *READY*.

If the configuration for the scan is not correct (invalid parameters or invalid JSON) the configuration is not applied and the ObsState of the CspSubarray remains IDLE.

Parameters *argin* – a JSON-encoded string with the parameters to configure a scan.

Returns None

Raises `tango.DevFailed` exception if the CspSubarray ObsState is not valid or if an exception is caught during command execution.

Note: Still to implement the check on AdminMode values: the command can be processed only when the CspSubarray is *ONLINE* or *MAINTENANCE*

AddNumOfSearchBeams(argin)

Note: Still to be implemented

Class method

Add the specified number of Search Beams capabilities to the subarray.

Parameters *argin* – The number of SearchBeams Capabilities to assign to the subarray

Returns None

RemoveNumOfSearchBeams (*argin*)

Note: Still to be implemented

Class method

Remove the specified number of Search Beams capabilities from the subarray.

Parameters *argin* – The number of SearchBeams Capabilities to remove from the subarray.
If equal to the max number of search beam capabilities (1500 for MID), all the search beams are removed.

Returns None

AddTimingBeams (*argin*)

Note: Still to be implemented

Class method

Add the specified Timing Beams Capability IDs to the subarray.

Parameters

- **argin** – The list of Timing Beams Capability IDs to assign to the subarray.
- **Type** – array of DevUShort

Returns None

AddVlbiBeams (*argin*)

Note: Still to be implemented

Class method

Add the specified Vlbi Beams Capability IDs to the subarray.

Parameters

- **argin** – The list of Vlbi Beams Capability IDs to assign to the subarray.
- **Type** – array of DevUShort

Returns None

AddSearchBeamsID (*argin*)

Note: Still to be implemented

Class method

Add the specified Search Beams Capability IDs to the subarray. This method requires some knowledge of the internal behavior of the PSS machine, because Search Beam capabilities with PSS pipelines belonging to the same PSS node, can't be assigned to different subarrays.

Parameters

- **argin** – The list of Search Beams Capability IDs to assign to the subarray.
- **Type** – array of DevUShort

Returns None

References

AddNumOfSearchBeams

RemoveSearchBeamsID (*argin*)

Note: Still to be implemented

Class method

Remove the specified Search Beam Capability IDs from the subarray.

Parameters

- **argin** – The list of Timing Beams Capability IDs to remove from the subarray.
- **Type** – Array of unsigned short

Returns None

RemoveTimingBeams ()

Note: Still to be implemented

Class method

Remove the specified Timing Beam Capability IDs from the subarray.

Parameters

- **argin** – The list of Timing Beams Capability IDs to remove from the subarray.
- **Type** – Array of DevUShort

Returns None

RemoveVlbiBeams ()

Note: Still to be implemented

Class method

Remove the specified Vibi Beam Capability IDs from the subarray.

Parameters

- **argin** – The list of Timing Beams Capability IDs to remove from the subarray.
- **Type** – Array of DevUShort

Returns None

is_EndSB_allowed()

TANGO is_allowed method: filter the external request depending on the current device state.

Check if the EndSB method can be issued on the subarray. The EndSB method can be issued on a subarrays when its *State* is *ON*.

Returns True if the command can be executed, otherwise False

EndSB()

Class method

Set the subarray *ObsState* to *IDLE*.

The command is executed only when the CspSubarray *State* is *ON* and *ObsState* is *READY* or *IDLE*.

Raises tango.DevFailed exception if the CspSubarray *ObsState* is not valid or if an exception is caught during command execution.

Note: Still to implement the check on AdminMode values: the command can be processed only when the CspSubarray is *ONLINE* or *MAINTENANCE*

CspSubarray.**main**(args=None, **kwargs)

CspMaster Class Documentation

```
class global_enum.HealthState
```

An enumeration.

```
OK = 0
```

```
DEGRADED = 1
```

```
FAILED = 2
```

```
UNKNOWN = 3
```

```
class global_enum.AdminMode
```

An enumeration.

```
ONLINE = 0
```

```
OFFLINE = 1
```

```
MAINTENANCE = 2
```

```
NOTFITTED = 3
```

```
RESERVED = 4
```

```
class global_enum.ControlMode
```

An enumeration.

```
REMOTE = 0
```

```
LOCAL = 1
```

```
class global_enum.ObsMode
```

An enumeration.

```
IDLE = 0
```

```
IMAGING = 1
```

```
PULSARSEARCH = 2
```

```
PULSARTIMING = 3
```

```
DYNAMICSPECTRUM = 4
TRANSIENTSEARCH = 5
VLBI = 6
CALIBRATION = 7
class global_enum.ObsState
    An enumeration.
    IDLE = 0
    CONFIGURING = 1
    READY = 2
    SCANNING = 3
    PAUSED = 4
    ABORTED = 5
    FAULT = 6
```

CHAPTER 4

Indices and tables

- `genindex`
- `modindex`
- `search`

C

CspMaster, [1](#)
CspSubarray, [14](#)

G

global_enum, [29](#)

A

ABORTED (*global_enum.ObsState* attribute), 30
 AddNumOfSearchBeams() (*CspSubarray.CspSubarray* method), 24
 AddReceptors() (*CspSubarray.CspSubarray* method), 23
 AddSearchBeamsID() (*CspSubarray.CspSubarray* method), 25
 AddTimingBeams() (*CspSubarray.CspSubarray* method), 25
 AddVlbiBeams() (*CspSubarray.CspSubarray* method), 25
 AdminMode (*class in global_enum*), 29
 adminMode (*CspMaster.CspMaster* attribute), 2
 always_executed_hook() (*CspMaster.CspMaster* method), 7
 always_executed_hook() (*CspSubarray.CspSubarray* method), 18
 availableCapabilities (*CspMaster.CspMaster* attribute), 3
 availableReceptorIDs (*CspMaster.CspMaster* attribute), 5

C

CALIBRATION (*global_enum.ObsMode* attribute), 30
 cbfAdminMode (*CspMaster.CspMaster* attribute), 3
 cbfMasterAddress (*CspMaster.CspMaster* attribute), 3
 cbfOutputLink (*CspSubarray.CspSubarray* attribute), 18
 cbfSubarrayAddr (*CspSubarray.CspSubarray* attribute), 16
 cbfSubarrayHealthState (*CspSubarray.CspSubarray* attribute), 15
 cbfSubarrayObsState (*CspSubarray.CspSubarray* attribute), 16
 CbfSubarrayPrefix (*CspSubarray.CspSubarray* attribute), 14
 cbfSubarrayState (*CspSubarray.CspSubarray* at-

tribute), 15
 commandProgress (*CspMaster.CspMaster* attribute), 2
 ConfigureScan() (*CspSubarray.CspSubarray* method), 24
 CONFIGURING (*global_enum.ObsState* attribute), 30
 ControlMode (*class in global_enum*), 29
 corrInherentCap (*CspSubarray.CspSubarray* attribute), 15
 cspCbfHealthState (*CspMaster.CspMaster* attribute), 2
 cspCbfState (*CspMaster.CspMaster* attribute), 2
 CspMaster (*class in CspMaster*), 1
 CspMaster (*CspSubarray.CspSubarray* attribute), 15
 CspMaster (*module*), 1
 CspMidCbf (*CspMaster.CspMaster* attribute), 1
 CspMidPss (*CspMaster.CspMaster* attribute), 1
 CspMidPst (*CspMaster.CspMaster* attribute), 1
 cspPssHealthState (*CspMaster.CspMaster* attribute), 3
 cspPssState (*CspMaster.CspMaster* attribute), 2
 cspPstHealthState (*CspMaster.CspMaster* attribute), 3
 cspPstState (*CspMaster.CspMaster* attribute), 2
 CspSubarray (*class in CspSubarray*), 14
 CspSubarray (*module*), 14
 cspSubarrayAddress (*CspMaster.CspMaster* attribute), 4
 CspSubarrays (*CspMaster.CspMaster* attribute), 1
 CspTelState (*CspMaster.CspMaster* attribute), 2

D

DEGRADED (*global_enum.HealthState* attribute), 29
 delete_device() (*CspMaster.CspMaster* method), 7
 delete_device() (*CspSubarray.CspSubarray* method), 18
 DYNAMIC SPECTRUM (*global_enum.ObsMode* attribute), 29

E

EndSB () (*CspSubarray.CspSubarray method*), 27
EndScan () (*CspSubarray.CspSubarray method*), 22

F

FAILED (*global_enum.HealthState attribute*), 29
FAULT (*global_enum.ObsState attribute*), 30
fsp (*CspSubarray.CspSubarray attribute*), 16
fspMembership (*CspMaster.CspMaster attribute*), 6

G

global_enum (*module*), 29

H

HealthState (*class in global_enum*), 29

I

IDLE (*global_enum.ObsMode attribute*), 29
IDLE (*global_enum.ObsState attribute*), 30
IMAGING (*global_enum.ObsMode attribute*), 29
init_device () (*CspMaster.CspMaster method*), 7
init_device () (*CspSubarray.CspSubarray method*), 18
is_AddReceptors_allowed () (*CspSubarray.CspSubarray method*), 23
is_ConfigureScan_allowed () (*CspSubarray.CspSubarray method*), 24
is_EndSB_allowed () (*CspSubarray.CspSubarray method*), 27
is_EndScan_allowed () (*CspSubarray.CspSubarray method*), 22
is_Off_allowed () (*CspMaster.CspMaster method*), 12
is_On_allowed () (*CspMaster.CspMaster method*), 11
is_RemoveAllReceptors_allowed () (*CspSubarray.CspSubarray method*), 24
is_RemoveReceptors_allowed () (*CspSubarray.CspSubarray method*), 23
is_Scan_allowed () (*CspSubarray.CspSubarray method*), 22
is_Standby_allowed () (*CspMaster.CspMaster method*), 12

L

LOCAL (*global_enum.ControlMode attribute*), 29

M

main () (*in module CspMaster*), 12
main () (*in module CspSubarray*), 27
MAINTENANCE (*global_enum.AdminMode attribute*), 29

N

NOTFITTED (*global_enum.AdminMode attribute*), 29

O

ObsMode (*class in global_enum*), 29
ObsState (*class in global_enum*), 30
Off () (*CspMaster.CspMaster method*), 12
OFFLINE (*global_enum.AdminMode attribute*), 29
OK (*global_enum.HealthState attribute*), 29
On () (*CspMaster.CspMaster method*), 11
ONLINE (*global_enum.AdminMode attribute*), 29

P

PAUSED (*global_enum.ObsState attribute*), 30
pssAdminMode (*CspMaster.CspMaster attribute*), 3
pssInherentCap (*CspSubarray.CspSubarray attribute*), 15
pssMasterAddress (*CspMaster.CspMaster attribute*), 3
pssSubarrayAddr (*CspSubarray.CspSubarray attribute*), 16
pssSubarrayHealthState (*CspSubarray.CspSubarray attribute*), 15
pssSubarrayObsState (*CspSubarray.CspSubarray attribute*), 16
PssSubarrayPrefix (*CspSubarray.CspSubarray attribute*), 14
pssSubarrayState (*CspSubarray.CspSubarray attribute*), 15
pstAdminMode (*CspMaster.CspMaster attribute*), 3
pstInherentCap (*CspSubarray.CspSubarray attribute*), 15
pstMasterAddress (*CspMaster.CspMaster attribute*), 3
PULSARSEARCH (*global_enum.ObsMode attribute*), 29
PULSARTIMING (*global_enum.ObsMode attribute*), 29

R

read_availableCapabilities () (*CspMaster.CspMaster method*), 8
read_availableReceptorIDs () (*CspMaster.CspMaster method*), 11
read_cbfAdminMode () (*CspMaster.CspMaster method*), 8
read_cbfMasterAddress () (*CspMaster.CspMaster method*), 7
read_cbfSubarrayAddr () (*CspSubarray.CspSubarray method*), 20
read_cbfSubarrayHealthState () (*CspSubarray.CspSubarray method*), 19
read_cbfSubarrayObsState () (*CspSubarray.CspSubarray method*), 20
read_cbfSubarrayState () (*CspSubarray.CspSubarray method*), 19

`read_commandProgress()` (*CspMaster.CspMaster method*), 7
`read_corrInherentCap()` (*CspSubarray.CspSubarray method*), 19
`read_cspCbfHealthState()` (*CspMaster.CspMaster method*), 7
`read_cspCbfState()` (*CspMaster.CspMaster method*), 7
`read_cspPssHealthState()` (*CspMaster.CspMaster method*), 7
`read_cspPssState()` (*CspMaster.CspMaster method*), 7
`read_cspPstHealthState()` (*CspMaster.CspMaster method*), 7
`read_cspPstState()` (*CspMaster.CspMaster method*), 7
`read_cspSubarrayAddress()` (*CspMaster.CspMaster method*), 10
`read_fsp()` (*CspSubarray.CspSubarray method*), 20
`read_pssAdminMode()` (*CspMaster.CspMaster method*), 8
`read_pssInherentCap()` (*CspSubarray.CspSubarray method*), 19
`read_pssMasterAddress()` (*CspMaster.CspMaster method*), 7
`read_pssSubarrayAddr()` (*CspSubarray.CspSubarray method*), 20
`read_pssSubarrayHealthState()` (*CspSubarray.CspSubarray method*), 20
`read_pssSubarrayObsState()` (*CspSubarray.CspSubarray method*), 20
`read_pssSubarrayState()` (*CspSubarray.CspSubarray method*), 19
`read_pstAdminMode()` (*CspMaster.CspMaster method*), 8
`read_pstInherentCap()` (*CspSubarray.CspSubarray method*), 19
`read_pstMasterAddress()` (*CspMaster.CspMaster method*), 7
`read_receptorMembership()` (*CspMaster.CspMaster method*), 10
`read_reportSearchBeamAdminMode()` (*CspMaster.CspMaster method*), 9
`read_reportSearchBeamHealthState()` (*CspMaster.CspMaster method*), 9
`read_reportSearchBeamState()` (*CspMaster.CspMaster method*), 9
`read_reportTimingBeamAdminMode()` (*CspMaster.CspMaster method*), 9
`read_reportTimingBeamHealthState()` (*CspMaster.CspMaster method*), 9
`read_reportTimingBeamState()` (*CspMaster.CspMaster method*), 9
`read_reportVlbiBeamAdminMode()` (*CspMaster.CspMaster method*), 10
`read_reportVlbiBeamHealthState()` (*CspMaster.CspMaster method*), 10
`read_reportVlbiBeamState()` (*CspMaster.CspMaster method*), 9
`read_scanID()` (*CspSubarray.CspSubarray method*), 18
`read_searchBeamCapAddress()` (*CspMaster.CspMaster method*), 10
`read_searchBeamMembership()` (*CspMaster.CspMaster method*), 11
`read_searchBeams()` (*CspSubarray.CspSubarray method*), 21
`read_searchBeamsHealthState()` (*CspSubarray.CspSubarray method*), 21
`read_searchBeamsState()` (*CspSubarray.CspSubarray method*), 21
`read_timingBeamCapAddress()` (*CspMaster.CspMaster method*), 10
`read_timingBeamMembership()` (*CspMaster.CspMaster method*), 11
`read_timingBeams()` (*CspSubarray.CspSubarray method*), 21
`read_timingBeamsHealthState()` (*CspSubarray.CspSubarray method*), 22
`read_timingBeamsObsState()` (*CspSubarray.CspSubarray method*), 22
`read_timingBeamsState()` (*CspSubarray.CspSubarray method*), 21
`read_validScanConfiguration()` (*CspSubarray.CspSubarray method*), 20
`read_vcc()` (*CspSubarray.CspSubarray method*), 21
`read_vlbiBeamMembership()` (*CspMaster.CspMaster method*), 11
`read_vlbiBeams()` (*CspSubarray.CspSubarray method*), 21
`read_vlbiBeamsHealthState()` (*CspSubarray.CspSubarray method*), 22
`read_vlbiBeamsState()` (*CspSubarray.CspSubarray method*), 21
`read_vlbiCapAddress()` (*CspMaster.CspMaster method*), 10
`read_vlbiInherentCap()` (*CspSubarray.CspSubarray method*), 19
`READY` (*global_enum.ObsState attribute*), 30
`receptorMembership` (*CspMaster.CspMaster attribute*), 5
`receptors` (*CspSubarray.CspSubarray attribute*), 18
`REMOTE` (*global_enum.ControlMode attribute*), 29
`RemoveAllReceptors()` (*CspSubarray.CspSubarray method*), 24
`RemoveNumOfSearchBeams()` (*CspSubarray.CspSubarray method*), 25
`RemoveReceptors()` (*CspSubarray.CspSubarray method*), 25

method), 23

RemoveSearchBeamsID () (CspSubarray.CspSubarray method), 26

RemoveTimingBeams () (CspSubarray.CspSubarray method), 26

RemoveVlbiBeams () (CspSubarray.CspSubarray method), 26

reportFSPAdminMode (CspMaster.CspMaster attribute), 6

reportFSPHealthState (CspMaster.CspMaster attribute), 6

reportFSPState (CspMaster.CspMaster attribute), 6

reportSearchBeamAdminMode (CspMaster.CspMaster attribute), 4

reportSearchBeamHealthState (CspMaster.CspMaster attribute), 4

reportSearchBeamState (CspMaster.CspMaster attribute), 4

reportTimingBeamAdminMode (CspMaster.CspMaster attribute), 4

reportTimingBeamHealthState (CspMaster.CspMaster attribute), 4

reportTimingBeamState (CspMaster.CspMaster attribute), 4

reportVCCAdminMode (CspMaster.CspMaster attribute), 6

reportVCCHealthState (CspMaster.CspMaster attribute), 6

reportVCCState (CspMaster.CspMaster attribute), 5

reportVlbiBeamAdminMode (CspMaster.CspMaster attribute), 4

reportVlbiBeamHealthState (CspMaster.CspMaster attribute), 4

reportVlbiBeamState (CspMaster.CspMaster attribute), 4

RESERVED (global_enum.AdminMode attribute), 29

S

Scan () (CspSubarray.CspSubarray method), 23

scanID (CspSubarray.CspSubarray attribute), 15

SCANNING (global_enum.ObsState attribute), 30

searchBeamCapAddress (CspMaster.CspMaster attribute), 5

searchBeamMembership (CspMaster.CspMaster attribute), 5

SearchBeams (CspMaster.CspMaster attribute), 1

searchBeams (CspSubarray.CspSubarray attribute), 16

searchBeamsHealthState (CspSubarray.CspSubarray attribute), 17

searchBeamsState (CspSubarray.CspSubarray attribute), 17

Standby () (CspMaster.CspMaster method), 12

T

timingBeamCapAddress (CspMaster.CspMaster attribute), 5

timingBeamMembership (CspMaster.CspMaster attribute), 5

TimingBeams (CspMaster.CspMaster attribute), 2

timingBeams (CspSubarray.CspSubarray attribute), 17

timingBeamsHealthState (CspSubarray.CspSubarray attribute), 17

timingBeamsObsState (CspSubarray.CspSubarray attribute), 18

timingBeamsState (CspSubarray.CspSubarray attribute), 17

TRANSIENTSEARCH (global_enum.ObsMode attribute), 30

U

UNKNOWN (global_enum.HealthState attribute), 29

V

validScanConfiguration (CspSubarray.CspSubarray attribute), 16

vcc (CspSubarray.CspSubarray attribute), 16

vccHealthState (CspSubarray.CspSubarray attribute), 18

vccMembership (CspMaster.CspMaster attribute), 6

vccState (CspSubarray.CspSubarray attribute), 18

VLBI (global_enum.ObsMode attribute), 30

vlbiBeamMembership (CspMaster.CspMaster attribute), 5

VlbiBeams (CspMaster.CspMaster attribute), 2

vlbiBeams (CspSubarray.CspSubarray attribute), 17

vlbiBeamsHealthState (CspSubarray.CspSubarray attribute), 17

vlbiBeamsState (CspSubarray.CspSubarray attribute), 17

vlbiCapAddress (CspMaster.CspMaster attribute), 5

vlbiInherentCap (CspSubarray.CspSubarray attribute), 15

W

write_adminMode () (CspMaster.CspMaster method), 7

write_cbfAdminMode () (CspMaster.CspMaster method), 8

write_pssAdminMode () (CspMaster.CspMaster method), 8

write_pstAdminMode () (CspMaster.CspMaster method), 8

write_scanID () (CspSubarray.CspSubarray method), 18