

Contents:
ASDM Table Overview
ASDM Summary of Intent levels/codes
ASDM Master Reference

ASDM Table Overview: Mandatory vs. Optional				
Mandatory Tables	Rows are added at:		Optional Tables	Rows are added at:
Main*	Integration		Beam	ExecBlock
AlmaCorrelatorMode*	Subscan		CalDevice	Scan
Antenna	ExecBlock		CalAtmosphere	Scan
ConfigDescription*	Subscan		Doppler	Scan
DataDescription	Subscan		Ephemeris	ExecBlock
Feed	ExecBlock		ExecBlock	ExecBlock
Field	Subscan		Focus	Subscan
Polarization	Scan		FocusModel	ExecBlock
Processor	Subscan		GainTracking	Subscan
Scan	Scan		History	Subscan
Source	Scan		Observation	ExecBlock
SpectralWindow	Subscan		Pointing	Integration
State	Integration		PointingModel	ExecBlock
SubScan	Subscan		Receiver	ExecBlock
			SBSummary	ExecBlock
			Seeing	Subscan
			SourceParameter	Scan
* Table is not required for non-correlator modes				
			SquareLawDetector	Integration
			Station	ExecBlock
			SwitchCycle	Integration
			TotalPowerData	Integration
			WVMCal	Scan
			Weather	?

ASDM Summary of SB Intent Levels				
SBSummary	SBSummary	SBSummary	Scan	Subscan
SBIntent	SBType	Observation	scanIntent	subscan Intent
UserSpecified	Standard	SingleField	TARGET	
	Expert	Mosaic	CALIBRATION	
	Observatory	SingleDish		
		OTFMosaic	FOCUS	
		OTFSingleDish	HOLOGRAPHY	HOLOGRAPHY
				RASTER
				HOLOGRAPHY
			POINTING	PHASECAL
			ATMOSPHERE	
			POINTINGMODEL	
			PHASECAL	
			AMPLICAL	
			DELAY	
			ANTENNAPOSITIONS	
			PHASECURVE	
			AMPLICURVE	
			SKYDIP	
			POLARIZATION	
			BANDPASS	
			MAP?	PATTERN
			FREGSWITCH	CYCLE
			BEAMSWITCH	CYCLE
			POSITIONSWITCH	CYCLE
			LAST	
			OFFLINE	OFFLINE

Table Key	
Red	Mandatory Table (for correlator observing modes)

Yellow	Indicates question areas/changes
Cyan	Optional Table
Green	Proposed Deletion
Grey	Not applicable; e.g., keys are consumed by all, ASDM content not

ASDM Master Reference: Contains information on the subsystem origination of content, subsystems dependent on content, ASDM->MS mapping										
ASDM Column	CIPT Origin	Consumer	Req.	MS Column	APDM Equiv.	ASDM2MS Processing	ASDM Java type	ASDM C++ type	Shape	Description
Main[<i>integration</i>]				MAIN						Purpose: Data, coordinates, flags; The master table containing links to the data, data context and project structure information. Not required for non-correlator observing modes.
configDescriptionId	DC [subscan]		Y			Not copied; used to obtain DATA_DESC_ID references for MAIN table.	Tag	Tag		Configuration description identifier (>0) providing a direct index into the ConfigDescription sub-table row number. Note that two or more sub-arrays cannot refer to the same configDescriptionId.
fieldId	DC [subscan]		Y	FIELD_ID		NP	Tag	Tag		Field identifier (>0)
time	Control [integration]		Y	TIME		NP	ArrayTime	ArrayTime	-> Mepoch(UTC,1,ns)	Mid-point (not centroid) of data interval.
execBlockId	DC [ExecBlock]		Y	OBSERVATION_ID		not necessary for post processing	Tag	Tag		Execution Block identifier (>0) providing a direct index into the ExecBlock sub-table row number; this corresponds to the MS OBSERVATION table.
stateId	DC [integration]		Y	STATE_ID		NP	Tag[]	vector<Tag>	ConfigDescription.numAntenna	State identifier (>0)
scanNumber	Control [scan]	Filler, QL, TelCal	Y	SCAN_NUMBER		NP	int	int		Scan Number
subscanNumber	Control [subscan]	Filler, QL, TelCal	Y	NS_SUBSCAN_NUMBER		not necessary for post processing; this is the row number	int	int		Subscan Number
integrationNumber	Control [integration]	TelCal	Y	NS_INTEGRATION_NUMBER		not necessary for post processing	int	int		Integration Number
uvw	Control -> DC [integration]	Filler, TelCal	Y	UVW		filler does difference of antenna-based values and re-interpolates to actual observing times.	Length[]	vector<vector<Length>>	ConfigDescription.numAntenna,3	Antenna-based uvw coordinates for the N_a antennas. This is defined at the centroid time of TIME_CENTROID; this is the vector from the center of the array to the Antenna. In the MS, this is the difference, POSITION2-POSITION1 between antennas, ANTENNA2 and ANTENNA1.
exposure	Corr [integration]	Filler, TelCal	Y	EXPOSURE		NP	Interval[]	vector<vector<Interval>>	ConfigDescription.numAntenna, AlmaCorrelatorMode.numBaseband ->Interval(1,ns)	Effective duration of an integration.
timeCentroid	Corr [integration]	Filler, TelCal	Y	TIME_CENTROID		NP	ArrayTime[]	vector<vector<ArrayTime>>	ConfigDescription.numAntenna, AlmaCorrelatorMode.numBaseband ->Mepoch(UTC,1,ns)	Time stamp reflecting the average time the non-blanked data was integrated.
dataOid	Archive -> DC [integration]	Filler, TelCal, Archive	Y	DATA		Unpacked from ALMA data cell.	EntityRef	EntityRef		Reference to the data cell object in the archive (i.e., the archive address).
flagAnt	Control -> Corr [integration]	Operations?	Y	NS_FLAGANT		Not Used.	int[]	vector<int>	ConfigDescription.numAntenna	4-byte vector of size N_antenna. Each bit corresponds to a specified flagging condition: 0: Last WVR calibration failed, 1: Current WVR hardware defect, 2: Last pointing calibration failed, 3: Last temperature scale calibration failed, 4: Temperature scale calibration hardware defect, 5: Last WVR calibration failed, 6: Shadowing, 7: LO1 out of lock, 8: No correlation detected on last calibrator, 9: Too much decorrelation on last correlator.

flagPol	Control -> Corr [integration]	Operations ?	Y	NS_FLAGPOL	Not Used.	int[1]	vector<vector<int>>	ConfigDescription.numAntenna, Polarization.numCorr	Unsigned long int (4-bytes) vector. When different from zero, all data concerning the polarization and antenna are affected ((i.e., all basebands - e.g., a problem incident before the down converters)). The filler will use the flags from the binary attachment; 0: Mixer not operational, 1: Total power out of range, 2: LO2 out of lock, 3: Integration totally blanked, 4: Interference detected, 7: No correlation detected on last calibrator, 9: Too much decorrelation on last calibrator.
flagBaseband	Control -> Corr [integration]	Operations ?	Y	NS_FLAGBB	Not Used.	int[1][1]	vector<vector<vector<int>>>	configDescription.numAntenna, Polarization.numCorr, AlmaCorrelatorMode.numBaseband	Is this really needed - can it be deprecated or made optional? The filler will use the flags from the binary attachment. 4-byte vector of size NaNcNbb, 0: Last Temperature scale calibration failed, Tsys currently estimated, 1: Total power out of range, 2: Interference detected, 3: No correlation detected on last calibrator, 4: Integration totally blanked, 9: Too much decorrelation on last calibrator.
flagRow	Control -> Corr [integration]	Operations ?	Y	FLAG_ROW	Not Used.	boolean	bool		The filler will use the flags from the binary attachment.
interval	Control [integration]	Filler, TelCal	Y	INTERVAL	NP	Interval	Interval		Duration of the integration.
subintegrationNumber	Control [integration +]	TelCal, Filler	M		Not Used.	int	int		If it isn't present, then it indicates that the row is an integration.
				ANTENNA1	7th axis of the data cell; the baseline products are: Na(Na-1)/2; cross products are A1.A2, A1.A3, A2.A3,A1.A4, ..., An-1.An		int		Antenna number (>=0) and a direct index into the ANTENNA sub-table row.
				ANTENNA2	7th axis of the data cell; the baseline products are: Na(Na-1)/2; cross products are A1.A2, A1.A3, A2.A3,A1.A4, ..., An-1.An		int		Antenna number (>=0) and a direct index into the ANTENNA sub-table row; for n>0, triple-product data are implied.
				FEED1	Obtained from configDescriptionId		int		Feed number (>=0)
				DATA_DESC_ID	Obtained from configDescriptionId		int		Data description identifier (>=0) and a direct index into the DATA_DESCRIPTION sub-table row.
				PROCESSOR_ID	Obtained from configDescriptionId		int		Processor identifier (>=0) and a direct index into the PROCESSOR sub-table row.
				ARRAY_ID	Not in ASDM; set to 1.		int		Subarray identifier (>=0) which identifies data in separate subarrays.
				SIGMA	Computed from Tsys/Exposure		Float(Nc)	Nc	The estimated rms noise for a single channel for each correlator.
				SIGMA_SPECTRUM	Computed from Tsys/Exposure		Float(Nc,Nf*)	Nc, Nf*	The estimate rms noise for each channel
				WEIGHT	Computed from Tsys/Exposure etc		Float(Nc)	Nc	The weight for the whole data matrix for each correlator, as assigned by the correlator.
				WEIGHT_SPECTRUM	Computed from Tsys		Float(Nc,Nf*)	Nc, Nf*	The weight for each channel in the data matrix, as assigned by the correlator.
				FLAG	Initially all False (all data is okay)		Bool(Nc,Nf*)	Nc, Nf*	An array of Boolean values with the same shape as the DATA, representing the cumulative flags to this data matrix.
				FLAG_CATAGORY	Derived from flagAnt,Pol,Baseband		Bool(Nc,Nf*,Ncat)	Nc, Nf*, Ncat	An array of flag matrices with the same shape as DATA, but indexed by category.
AlmaCorrelatorMode [Subscan]					Add columns to PROCESSOR table as needed - Do we need information for noise computation - basebandConfig (n - provides this info). Need quantization bitsize - put into spectral window table? There is an issue with the order of the spectral windows and the matching to the binary data.				Purpose: Processor mode information. This table contains information on the processor (backend) mode applicable to the current data record.
modelId	DC		Y			Tag	Tag		
numBaseband	ObsPrep -> Control	TelCal	Y			int	int		Baseband number (1-4)
basebandName	ObsPrep -> Control	TelCal	Y		baseBandIndex	int[1]	vector<int>	numBaseband	Baseband name
accumMode	ObsPrep -> Control	TelCal	Y		cAM	Interval	Interval		Processor accumulation mode defining the correlator chip level accumulation duration; 1 or 16 ms (auto/cross)

basebandConfig	ObsPrep -> Control	TelCal, Offline	Y		bBC/IFBMode?		int[]	vector<int>	numBaseband	Baseband configuration mode (1-66) which defines the bandwidth, polarizations and oversampling.
binMode	ObsPrep -> Control		N		binMode		int	int		single_nutator_sidebandseparation
quantization	ObsPrep -> Control		N		quantizationCorrection		boolean	bool		Boolean selection to perform quantization correction on the lag data.
windowFunction	ObsPrep -> Control	TelCal	Y		windowFunction		String	string		Windowing function used: Uniform, Hanning, Hamming, Bartlett, Blackman, Blackman-Harris, Welch
axesOrderArray	DC		N				int[]	vector<int>		Note: This is unchanging for ALMA; Axes order is always: 1: Polarization, 2: Frequency Channel Number, 3: Atmospheric pathlength correction, 4: Bin number, 5: Window number, 6: Baseband number, 7: Baseline products.
Antenna [ExecBlock]					ANTENNA					Purpose: Antenna characteristics. This table contains the global antenna properties for each antenna.
antennald	DC		Y			Implicit in the row number of the table.	Tag	Tag		
name	Control	Filler	Y	NAME		NP	String	string		Antenna name
stationId	DC		Y	STATION		NP	Tag	Tag		Station identifier
type	Control	Filler, TelCal	Y	TYPE		NP	String	string		Antenna type, VertexP, AECP, Vertex, AEM, Mitsubishi 7, Mitsubishi-12.
xPosition	Control	Filler, TelCal	Y	POSITION		Convert to MPosition	Length	Length	-> MPosition (ITRF,3,m)	In right-handed frame. X towards the intersection of the equator and the Greenwich meridian.
yPosition	Control	Filler, TelCal	Y	POSITION		Convert to MPosition	Length	Length		
zPosition	Control	Filler, TelCal	Y	POSITION		Convert to MPosition	Length	Length		Z towards the pole.
time	Control	Filler, TelCal, Operations, Archive	Y			NP	ArrayTime	ArrayTime		Time the antenna position was determined
xOffset	Control	Filler, TelCal	N	OFFSET		Convert to Quanta	Length	Length		Axis offset of mount to feed reference point.
yOffset	Control	Filler, TelCal	N	OFFSET		Convert to Quanta	Length	Length		Axis offset of mount to feed reference point.
zOffset	Control	Filler, TelCal	N	OFFSET		Convert to Quanta	Length	Length		Axis offset of mount to feed reference point.
dishDiameter	Control	Filler, TelCal	Y	DISH_DIAMETER		Convert to Quanta	Length	Length	-> Length(1,m)	Nominal diameter of the dish, as opposed to the effective diameter.
flagRow	Control	Filler, TelCal	N	FLAG_ROW		NP	boolean	bool		Row flag.
assocAntennald			N	NS_ASSOC_ANTENNA_ID		NP	Tag	Tag		Proposed deletion.
				ORBIT_ID		Not used. Orbit identifier. Index used in ORBIT sub-table if ANTENNA_TYPE is "SPACE_BASED".				
				MEAN_ORBIT		Not used. 0: Semi-major axis of orbit (a) in m. 1: ellipticity of orbit (e), 2: Inclination of orbit to the celestial equator (i) in deg, 3: Right ascension of the ascending node (omega) in deg, 4: Argument of perigee (w) in deg.				
				PHASED_ARRAY_ID		Not used.				
				MOUNT		Not used. EQUATORIAL, ALT-AZ, X-Y, SPACE-HALCA				
Beam [ExecBlock]					BEAM	This needs to be added to the MS!			Actual beam information needs to be added to the table.	Purpose: Beam characteristics.
beamId	DC	Filler	N	beamId		Implicit in the row number of the table.	Tag	Tag		
CalDevice [Scan]						Not needed, combine with SysCal?				Purpose: Calibration device characteristics. Who knows this and when.
antennald	Control		Y				Tag	Tag		Antenna identifier.
feedId	DC		Y				int	int		Feed identifier

spectralWindowId	DC		Y				Tag	Tag		Spectral window identifier
timeInterval	Control		Y				ArrayTime	ArrayTime		Time interval. Should be the same as the Feed table.
numCalload	Control	Filler, TelCal	Y				int	int		Number of the calibration noise or load available for this feed.
noiseCal	Control	Filler, TelCal	N				double[]	vector<double>	numCalload	Noise calibration temperature.
temperatureLoad	Control	Filler, TelCal	N				Temperature[]	vector<Temperature>	numCalload	Physical temperature of the load for device. Either noiseCal or temperatureLoad must be provided but not both.
calEff							float[]	vector<vector<float>>	numCalLoad, Feed.numreceptors	Proposed deletion.
CalAtmosphere (Scan)										Purpose: System calibration information. This table contains the time variable calibration measurements for each antenna.
CalDataID	DC		Y							
CalReductionID	DC		Y							
AntennaName	TelCal		Y	ANTENNA_ID						
NumReceptors	TelCal	Filler, TelCal	Y							
NumFreq	TelCal	Filler, TelCal	Y							
StartTimeValid	TelCal	Filler, TelCal	Y	TIME					(EndTimeValid - StartTimeValid)/2	
EndTimeValid	TelCal	Filler, TelCal	Y	INTERVAL					EndTimeValid - StartTimeValid	
FrequencyRange	TelCal	Filler, QL	Y	SPECTRAL_WINDOW_ID					Row per SPECTRAL_WINDOW_ID for each SPW relevant to the FrequencyRange	
FreqSpectrum	TelCal	Filler, TelCal	Y							
SubType	TelCal	Pipeline, Operations	Y							Single Direction/ SkyDip
Water	TelCal	Filler	Y						Goes to WEATHER table	
TSysSpectrum	TelCal	Filler	Y	TSYS_SPECTRUM						
TRxSpectrum	TelCal	Filler	Y	TRX_SPECTRUM						
TAimSpectrum	TelCal	Filler	Y	TSKY_SPECTRUM						
TauSpectrum	TelCal	TelCal	Y							
SidebandGainSpectrum	TelCal	TelCal	Y							
ForwardEffSpectrum	TelCal	TelCal	Y							
TSys	TelCal	Filler, QL, Telcal	Y	TSYS						
TRx	TelCal	Filler, QL, Telcal	N	TRX						
TAim	TelCal	TelCal	N	TSKY						
Tau	TelCal	Filler, QL, Telcal	N							
SidebandGain	TelCal	TelCal, QL	N							
ForwardEff	TelCal	TelCal	N							
				FEED_ID					Only one value	
				TCAL+, TANT+, FLAGS					From CalDevice	
configDescription (Subscan)										Purpose: Configuration description. The table contains information on the shape of the data cells in the Main table.
configDescriptionId	DC (integration)		Y				Tag	Tag		
antennaId	DC (scan)		Y				Tag[]	vector<Tag>		

dataDescriptionId	DC [subscan]	Filler, TelCal,	Y			This is slightly different from the MS DATA_DESC_ID; the SUBBAND and BASEBAND make the spectral window number. The Correlator axis is the stokes or polarization axis and they get referenced in the POLARIZATION table.	Tag[]	vector<Tag>		
feedId	DC [ExecBlock]		N				int[]	vector<int>	Only a single value for ALMA; never changes.	
processorId	DC [scan]	Filler, TelCal,	Y			This ID goes to the PROCESSOR_ID of the MAIN table	Tag	Tag		
numAntenna	Control [scan]	Filler, TelCal,	Y			Determines number of rows in ANTENNA subtable (>=NUM_ANTENNA).	int	int		
numFeed	Control [ExecBlock]		N				int	int		
numSubBand	ObsPrep -> Control [subscan]	Filler, TelCal,	Y			Determines the number of rows in SPECTRAL_WINDOW subtable (>=NUM_BASEBAND*NUM_SUBBAND).	int[]	vector<int>		
phasedArrayList	Control [scan]		N			All antennas that are phased are considered as a new antenna and they can	int[]	vector<int>		
correlationMode	ObsPrep -> Control [subscan]	Filler, TelCal,	Y		dataProducts	implicit (autocorr has uvw = 0)	int	int		->String - AutoOnly, CrossOnly, AutoAndCross
flagAnt	Control [subscan]		N			Also all rows in the MAIN table with ANTENNA1 and ANTENNA2 equal to the flagged antenna should be flagged.	boolean[]	vector<bool>		4-byte vector of size N_antenna. Each bit corresponds to a specified flagging condition: 0: Last WVR calibration failed, 1: Current WVR hardware defect, 2: Last pointing calibration failed, 3: Last temperature scale calibration failed, 4: Temperature scale calibration hardware defect, 5: Last WVR calibration failed, 6: Shadowing, 7: LO1 out of lock, 8: No correlation detected on last calibrator, 9: Too much decorrelation on last correlator.
atmPhaseCode	ObsPrep -> Control [subscan]	Filler, TelCal,	Y				int	int		->String - Corrected, Uncorrected, CorrectedAndUncorrected
switchCycleId	DC [subscan]	Filler, TelCal,	Y			Corresponds to a different STATE_ID (referenced to a row in STATE subtable). Everything gets repeated in the MAIN table except for DATA for each state.	Tag[]	vector<Tag>		
DataDescription [Subscan]						DATA_DESCRIPTION				Purpose: Spectral and polarization description.
dataDescriptionId	DC		Y			Implicit in the row number of the table.	Tag	Tag		
polarizationId	DC [subscan]	Filler, TelCal,	Y		POLARIZATION_ID	NP	Tag	Tag		
spectralWindowId	DC [subscan]	Filler, TelCal,	Y		SPECTRAL_WINDOW_ID	NP	Tag	Tag		
flagRow			N		FLAG_ROW	NP	boolean	bool		Proposed deletion.
					FLAG_ID					
Doppler [Scan]						DOPPLER				Purpose: Doppler tracking information. This table contains information on the Doppler tracking modes.
dopplerId	DC	Filler	Y		DOPPLER_ID	NP	int	int		
sourceId	DC	Filler	Y		SOURCE_ID	NP	int	int		
transitionIndex	ObsPrep -> Control	Filler	Y		TRANSITION_ID	NP	int	int		
velDef	ObsPrep -> Control	Filler	Y		VELDEF	Convert to MDoppler	Speed	Speed		->MDoppler(1,m.s-1)

Ephemeris										Purpose: Ephemeris object information.
ExecBlock [ExecBlock]				EPHEMERIS						
ephemerisId	Control	Filler, TelCal	Y	All columns		Need to add content	Tag	Tag		
ExecBlock [ExecBlock]				OBSERVATION		Site information is derived from Data Repository so not needed in MS. Base* information is not used by CASA. Other information is similar to that in OBSERVATION table.				Purpose: Execute summary. Want addition of content to indicate whether the EB executed to completion.
execBlockId	DC		Y			Implicit in the row number of the table.	Tag	Tag		
antennald	DC		Y				Tag[]	vector<Tag>		Needed for dataset combination.
telescopeName	Control	Filler, Operations	Y	TELESCOPE_NAME		NP	String	string		Should be 'ALMA'
configName	Control	Operations	Y	NS_CONFIG_NAME		NP	String	string		TBD (Configuration names)
numAntenna	Control	Operations	Y	NS_NUM_ANT		NP	int	int		
baseRangeMin	Scheduling	Operations	Y	NS_BASERANGEMIN		NP	Length	Length		
baseRangeMax	Scheduling	Operations	Y	NS_BASERANGEMAX		NP	Length	Length		
baseRmsMinor	Scheduling	Operations	Y	NS_BASERMSMINOR		NP	Length	Length		
baseRmsMajor	Scheduling	Operations	Y	NS_BASERMSMAJOR		NP	Length	Length		
basePa	Scheduling	Operations	Y	NS_BASEPA		NP	Angle	Angle		
timeInterval	Control	Operations	Y	TIME_RANGE		NP	ArrayTimeInterval	ArrayTimeInterval		Time of EB execution.
observerName	ObsPrep -> Control	Filler, Operations	Y	OBSERVER		NP	String	string		Not in SB; must look in Project (Archive) to obtain; change name to PIName; this must go into the SB. Put into SBSummary table.
observingLog	Control		Y	LOG		NP	String[]	vector<string>		This is the shift log. There is no path to this currently.
executeIntent	Scheduling > Control	Operations	Y	SCHEDULE_TYPE		NP	String[]	vector<string>		Options: Queue, Interactive, Dynamic, Manual
projectId	ObsPrep -> Control	Operations, Pipeline	Y	PROJECT		NP	EntityRef	EntityRef		Needed for dataset combination.
siteLongitude	Control	Operations	Y			Data Repository	Angle	Angle		
siteAltitude	Control	Operations	Y			Data Repository	Angle	Angle		
releaseDate			N	RELEASE_DATE		Not filled.	Length	Length		Unknown how to set this.
flagRow			N			Fill all False.	ArrayTime	ArrayTime		Remove.
							boolean	bool		Remove.
Feed [ExecBlock]				FEED						Purpose: Feed characteristics. A feed is a collecting element on an antenna, that shares physical properties and makes sense to calibrate as a single entity. It is an abstraction of an antenna feed and is considered to have one or more receptors that respond to different polarization states. feedNum should be non-zero only for feed arrays.
feedId	DC		Y	FEED_ID		NP	int	int		This is always 0. One feed within a band can only point at a target.
antennald	DC		Y	ANTENNA_ID		NP	Tag	Tag		
spectralWindowId	DC		Y	SPECTRAL_WINDOW_ID		NP	Tag	Tag		
timeInterval	Control		Y	INTERVAL, TIME		NP	ArrayTimeInterval	ArrayTimeInterval		
beamId	DC		N	BEAM_ID		NP	Tag[]	vector<Tag>		
receiverId	DC		Y	NS_RECEIVER_ID		NP	int[]	vector<int>		
numReceptors	Control		Y	NUM_RECEPTORS		NP	int	int		
feedNum	Control		N	NS_FEED_NUM		NP	int	int		This is always 1. Delete
beamOffset	TCDB -> Control	Filler, TelCal	Y	BEAM_OFFSET		NP	double[]	vector<vector<double>>		
polarizationType	Control	Filler, TelCal	Y	POLARIZATION_TYPE		NP	String[]	vector<string>		
polResponse	TCDB -> Control	Filler, TelCal	Y	POL_RESPONSE		NP	Complex[]	vector<vector<Complex>>		

xPosition			N	POSITION		Convert to MPosition	Length	Length		Delete (relevant only for feed arrays)
yPosition			N	POSITION		Convert to MPosition	Length	Length		Delete (relevant only for feed arrays)
zPosition			N	POSITION		Convert to MPosition	Length	Length		Delete (relevant only for feed arrays)
receptorAngle	TCDB -> Control	Filler, TelCal	Y	RECEPTOR_ANGLE			Angle[]	vector<Angle>		
focusReference	TCDB -> Control	Control, Operations	Y	NS_FOCUS_REFERENCE		Likely not filled.	double[][]	vector<vector<double>>		
illumOffset	TCDB -> Control	Filler	N				float	float		Can be used to correct UVW
illumOffsetPa	TCDB -> Control	Filler	N				float	float		Can be used to correct UVW
				FOCUS_LENGTH		Not filled. Focus length. As defined along optical axis.				
				PHASED_FEED_ID		Not filled. Phased feed identifier. Used to point to multiple FEED entries and specifies manner for combination.				
Field (Subscan)				FIELD						Purpose: Field positions. This table defines a field position on the sky. For interferometers, this is the correlated field position. For single dishes, this is the nominal pointing direction.
fieldId	DC		Y	NS_FIELD_ID		Implicit in the row number of the table.	Tag	Tag		
fieldName	ObsPrep -> Control	Filler	Y	NAME		NP	String	string		For each APDM FieldPattern - need a rule to construct the fieldName.
code	ObsPrep -> Control	Filler	N	CODE		NP	String	string		User specified comment. Should change name to FieldComment.
time	Control	Filler	Y	TIME		NP	ArrayTime	ArrayTime		Goes away with proposed polyn class
numPoly	Control	Filler	Y	NUM_POLY		NP	int	int		Goes away with proposed polyn class
delayDir	Control	Filler	Y	DELAY_DIR		NP	Angle[][]	vector<vector<Angle>>	numPoly+1,2	
phaseDir	Control	Filler	Y	PHASE_DIR	phaseCenterCoordinates	NP	Angle[][]	vector<vector<Angle>>	numPoly+1,2	Change this to a polyn class with shape (numPoly+1,2,sampling_step,1) (last is time)?
referenceDir	Control	Filler	Y	REFERENCE_DIR	referenceCoordinates	NP	Angle[][]	vector<vector<Angle>>	numPoly+1,2	
flagRow	Control	Filler	N	FLAG_ROW		NP	boolean	bool		
assocFieldId	ObsPrep -> Control -> DC	Filler	N	NS_ASSOCFIELD_ID		NP	Tag[]	vector<Tag>	?	
ephemerisId	DC	Filler	N	EPHEMERIS_ID		NP	Tag	Tag		
sourceId	DC	Filler	N	SOURCE_ID		NP	int	int		
assocNature	ObsPrep -> Control	Operations, Pipeline	N	NS_ASSOC_NATURE		NP	String	string		phasereferencesource, pattern, mosaic
FlagCmd (never)				FLAG_CMD						This can't be populated on-line.
timeInterval			Y	INTERVAL, TIME		Convert to Quanta	ArrayTimeInterval	ArrayTimeInterval		
type			Y	TYPE		NP	String	string		
reason			Y	REASON		NP	String	string		
level			Y	LEVEL		NP	int	int		
severity			Y	SEVERITY		NP	int	int		
applied			Y	APPLIED		NP	boolean	bool		
command			Y	COMMAND		NP	String	string		
Focus (Subscan)						Not currently used by CASA				Purpose: Focus Information. This table contains information on the antenna based focus as a function of time. Updated only when there is a change (i.e., a CalFocus result is obtained).
antennaId	DC		Y				Tag	Tag		
feedId	DC		Y				int	int		
timeInterval	Control		Y				ArrayTimeInterval	ArrayTimeInterval		
focusModelId	DC		Y				Tag	Tag		
xFocusPosition	TCDB -> Control	Control, TelCal	Y				Length	Length		Used by TelCal for determining CalFocus result; holds (for subsequent scans) the CalFocus result.

yFocusPosition	TCDB -> Control	Control, TelCal	Y				Length	Length		Used by TelCal for determining CalFocus result; holds (for subsequent scans) the CalFocus result.
zFocusPosition	TCDB -> Control	Control, TelCal	Y				Length	Length		Used by TelCal for determining CalFocus result; holds (for subsequent scans) the CalFocus result.
focusTracking	Control	TelCal	Y				boolean	bool		True if tracking the target position.
xFocusOffset	Control	TelCal	Y				Length	Length		
yFocusOffset	Control	TelCal	Y				Length	Length		
zFocusOffset	Control	TelCal	Y				Length	Length		
FocusModel [ExecBlock]										Purpose: Describe focus model properties. Add content please (based on Control capabilities); function depending on band, antenna type, meteorological info.
focusModelId	DC	Operations	Y			Not currently used by CASA	Tag	Tag		
FreqOffset (never)										Purpose: Frequency offset information. This table contains frequency offset information to be added directly to the defined frequency labeling in the SpectralWindow table. This allows bands with small time-variable frequency offsets to be labeled as the same spectralWindowId. This is not needed for ALMA as there are no such offsets possible within an EB.
			N	FREQ_OFFSET		Table is never filled for ALMA.				
antennaId	DC		Y				Tag	Tag		
feedId	DC		Y				int	int		
spectralWindowId	DC		Y				Tag	Tag		
timeInterval	Control		Y				ArrayTimeInterval	ArrayTimeInterval		
offset	Control		Y				Frequency	Frequency		
GainTracking [Integration]						samplingLevel should be retained for diagnostic information; BEAM or POINTING table?				Purpose: Antenna gain tracking information.
			N							
antennaId	DC		Y				Tag	Tag		
feedId	DC		Y				int	int		
spectralWindowId	DC		Y				Tag	Tag		
timeInterval	Control		Y				ArrayTimeInterval	ArrayTimeInterval		
attenuator	Control [subscan]	Operations, Pipeline?	Y				float	float		Combination of the IF and baseband attenuator.
samplingLevel			N				float	float		Always fixed. Delete. Currently Optional.
delayoff1	Control		N				Interval	Interval		Need evaluation whether these can be deleted.
delayoff2	Control		N				Interval	Interval		Need evaluation whether these can be deleted.
phaseoff1	Control		N				Angle	Angle		Need evaluation whether these can be deleted.
phaseoff2	Control		N				Angle	Angle		Need evaluation whether these can be deleted.
rateoff1	Control		N				AngularRate	AngularRate		Need evaluation whether these can be deleted.
rateoff2	Control		N				AngularRate	AngularRate		Need evaluation whether these can be deleted.
phaseRefOffset	Control		N				Angle	Angle		Need evaluation whether these can be deleted.
History [Subscan]										Purpose: History information. Can this be used as the script history - messages intended to be served.
				HISTORY						
execBlockId	DC		Y	OBSERVATION_ID		NP	Tag	Tag		
time	Control		Y	TIME		Convert to Quanta	ArrayTime	ArrayTime		
message	Control	Operations	Y	MESSAGE		NP	String	string		
priority			N				String	string		
origin	Control	Operations	Y	ORIGIN		NP	String	string		
objectId			N				String	string		
application			N				String	string		
cliCommand			N				String	string		
appParams			N				String	string		
Observation [ExecBlock]				Content is contained in ExecBlock						Purpose: Observation summary.

observationId	DC		Y				Tag	Tag			
Pointing (Integration)											
				POINTING	Purpose: Antenna pointing information. This table contains information on the primary pointing direction of each antenna as a function of time.						
antennaId	DC		Y	ANTENNAID	NP		Tag	Tag		Antenna identifier.	
timeInterval	Control		Y	INTERVAL	NP		ArrayTimeInterval	ArrayTimeInterval		Time interval.	
name			N	NAME	NP		String	string		Pointing direction name; user specified. Delete.	
numPoly	Control		Y	NUM_POLY	NP		int	int		Series order for the polynomial expressions in Direction and PointingOffset; this will be fixed as 0.	
timeOrigin	Control		Y	TIME_ORIGIN	NP		ArrayTime	ArrayTime		Time origin for the polynomial expressions.	
pointingDirection	Control	Filler, TelCal	Y	DIRECTION	NP		Angle[]	vector<vector<Angle>>	numPoly+1,2	Antenna-based pointing direction (does not include any errors); note: this will always be numPoly=0 for ALMA. Use polyn class as above to encapsulate the polynomial, any step and the time. The difference between the pointingDirection and the target is the offset position for mapping (the offset below are the pointing corrections in addition to the pointing model).	
target	Control	Filler, TelCal	Y	TARGET	NP		Angle[]	vector<vector<Angle>>	numPoly+1,2	Intended direction toward object of interest.	
encoder	Control	Filler, TelCal	Y	ENCODER	NP		Angle[]	vector<Angle>	2	Actual pointing directions (includes all corrections (refraction, etc) plus errors due to wind, etc) in Az/EI frame. The difference between the encoder and the pointingDirection is the accumulated pointing errors for an antenna.	
pointingModelId	Control	Filler, TelCal	Y	POINTING_MODEL_ID	NP		int	int		Pointing Model Identifier.	
pointingTracking	Control	Filler, TelCal	Y	TRACKING	NP		boolean	bool		True if tracking the nominal pointing position.	
phaseTracking			Y	NS_PHASE_TRACKING	NP		boolean	bool		True if tracking the phase center.	
offset	Control	TelCal, Filler	Y	POINTING_OFFSET	NP		Angle[]	vector<vector<Angle>>		The a priori pointing corrections applied by the telescope in pointing to the Direction position (result from TelCal); these values are included in the pointingDirection.	
sourceOffset			N	SOURCE_OFFSET	NP		Angle[]	vector<vector<Angle>>		Offset added on top of the target direction. Setting an offset will change the pointing but not the target direction.	
overTheTop			N	OVER_THE_TOP	NP		boolean	bool		True if the antenna is over the top. Not possible for production antennas (should check if Mitsubishi antennas can do this).	
PointingModel (ExecBlock)											
					Not used by CASA						
Purpose: Pointing Model information.											
antennaId	DC		Y				Tag	Tag			
pointingModelId	DC		Y				int	int			
numCoeff	TelCal -> Control	TelCal, Control	Y				int	int			
coeffName	TelCal -> Control	TelCal, Control	Y				String[]	vector<string>	numCoeff		
coeffVal	TelCal -> Control	TelCal, Control	Y				float[]	vector<float>	numCoeff		
numFormula			Y				int	int		Adopted TPOINT for defining the formula.	
coeffFormula			Y				String[]	vector<string>	numCoeff	Adopted TPOINT for defining the formula.	
Polarization (Subscan)											
				POLARIZATION	Purpose: Polarization information. This table defines the polarization labeling of the data.						
polarizationId	DC		Y			Not required.	Tag	Tag			
numCorr	ObsPrep -> Control, DC(lookup of TFB)	Filler, TelCal	Y	NUM_CORR	NP		int	int		Issue: Should Control do lookup of TCDB and send to DC?	

corrType	ObsPrep -> Control, DC(lockup of TFB)	Filler, TelCal	Y	CORR_TYPE		NP	int[]	vector<int>	numCorr	Change this to a string; include list of possible options.
corrProduct	ObsPrep -> Control, DC(lockup of TFB)	Filler, TelCal	Y	CORR_PRODUCT		NP	int[][]	vector<vector<int>>	numCorr.2	
flagRow				FLAG_ROW		NP	boolean	bool		Correlator does not know if a correlation is bad.
Processor (Subscan)				PROCESSOR						Purpose: Processor characteristics. This table holds the summary of the back-end processing device used to generate the data in the Main table.
processorId	DC		Y			Not required.	Tag	Tag		
modelId	DC		Y	MODE_ID		NP	Tag	Tag		
type	Control	Filler, TelCal	Y	TYPE		NP	String	string		Correlator, Total Power, Holography, Optical, ToBeDiscussed
subType	Control	Filler, TelCal	Y	SUB_TYPE		NP	String	string		eBLC, ACA, PrototypeCorr, TotalPowerDetector, WVR, ToBeDiscussed
flagRow				FLAG_ROW		All False	boolean	bool		Only true if every total power detector in the array failed. But if running with multiple processors, it will save one of them. Retain. Nope. Should have the rows flagged via the Correlator previously.
Receiver (Subscan)			N	RECEIVER		Do we fill this - or distribute to FEED, etc.				Purpose: Receiver properties. This table contains the measured properties of the individual receivers. Indexed in the Feed table. Worrisome sampling period;
receiverId	DC		Y			Write to SPECTRAL_WINDOW table?	int	int		
spectralWindowId	DC		Y				Tag	Tag		
timeInterval	Control		Y				ArrayTimeInterval	ArrayTimeInterval		
numLo	Control	TelCal	Y				int	int		
name	Control	Filler, TelCal	Y				String	string		Hardware serial number.
receiverBand -> frequencyBand	Control	Filler, TelCal	Y				String	string		Band_1, Band_2, ... Band_10
freqLo	Control	Filler, TelCal	Y				Frequency[]	vector<Frequency>	numLo	
sidebandLo	Control	Filler, TelCal	Y				int[]	vector<int>	numLo	
TDewar	Control	TelCal	Y				Temperature	Temperature		Should not be recorded faster than per Scan.
stabilityDuration	TCDB -> Control	Operations	N				Interval	Interval		
stability	TCDB -> Control	Operations	N				double	double		
stabilityFlag							boolean	bool		
dewarName	Control	TelCal	N				String	string		Hardware serial number
SBSummary (ExecBlock)				NS SBSummary						Purpose: Scheduling Block summary.
sbid	DC		Y	NS_SBID		NP	EntityRef	EntityRef		
projectId	ObsPrep -> Control	Operations	Y	PROJECT_ID		NP	EntityRef	EntityRef		
obsUnitSetId	ObsPrep -> Control	Operations	Y	OBSUNITSET_ID		NP	EntityRef	EntityRef		
sbIntent	ObsPrep -> Control		Y	SB_INTENT		NP	String	string		
sbType	Script -> Control		Y	SB_TYPE		NP	String	string		
sbDuration	ObsPrep -> Control		Y	SB_DURATION		NP	Interval	Interval		
numScan			N				int	int		
scanIntent			N				String[]	vector<string>	numScan	

numberRepeats	ObsPrep -> Control	Operations, Pipeline	Y	NUMBER_REPEATS		NP	int	int		
weatherConstraint	ObsPrep -> Control	Operations, Pipeline	Y	WEATHER_CONSTRAINT		NP	String[]	vector<string>		
scienceGoal	ObsPrep -> Control	Operations, Pipeline	Y	SCIENCE_GOAL		NP	String[]	vector<string>		
raCenter	ObsPrep -> Control	Operations, Pipeline	Y	RA_CENTER		NP	double	double		
decCenter	ObsPrep -> Control	Operations, Pipeline	Y	DEC_CENTER		NP	double	double		
frequency	ObsPrep -> Control	Operations, Pipeline	Y	FREQUENCY		NP	double	double		
frequencyBand	ObsPrep -> Control	Operations, Pipeline	Y	FREQUENCY_BAND		NP	String	string		
observingMode	Script -> Control	Operations, Pipeline	Y	OBSERVING_MODE		NP	String[]	vector<string>		
PIName	ObsPrep -> Control	Operations, Pipeline	Y	PI_NAME		NP				
Scan [Scan]										
State table? Add scanIntent, numSubScan, fieldName, (startTime?), (endTime?) (subscanIntent)						Purpose: Scan summary information.				
execBlockId	DC		Y				Tag	Tag		
scanNumber	Control	Filler, TelCal, QL	Y				int	int		
startTime	Control	Filler, TelCal, QL	Y				ArrayTime	ArrayTime		
endTime	Control	Filler, TelCal, QL	Y				ArrayTime	ArrayTime		
scanIntent	Control	Filler, TelCal, QL, Pipeline	Y				String	string		Options: Focus, Holography, Pointing, Atmosphere, PointingModel, Last, Target, PhaseCal, AmpliCal, Delay, AntennaPositions, PhaseCurve, AmpliCurve, SkyDip, Polarization, BandPass, Offline (e.g. OpticalPointing). Defined in: ICD/Control/ControlInterfaces.midi; Should be a vector of values.[Note: This may move to ICD/APDM]
numSubScan	Control	Filler, TelCal, QL	Y				int	int		
sourceName			Y				String	string		Move fieldName to Subscan, add sourceName.
flagRow	Control	Filler, TelCal, QL	Y				boolean	bool		
CalSeeing [Subscan]										
Add to SYSCAL						Purpose: Seeing calibration information.				
timeInterval	TelCal -> Control		Y	All columns		All columns copied without modification.	ArrayTimeInterval	ArrayTimeInterval		
numBaseLength	TelCal -> Control		Y				int	int		
baseLength	TelCal -> Control		Y				Length[]	vector<Length>	numBaseLength	
phaseRms	TelCal -> Control		Y				Angle[]	vector<Angle>	numBaseLength	
seeing	TelCal -> Control		Y				float	float		
exponent	TelCal -> Control		Y				float	float		
Source [Scan]										
SOURCE						Purpose: Source information.				
sourceId	DC			SOURCE_ID		NP	int	int		
spectralWindowId	DC			SPECTRAL_WINDOW_ID		NP	Tag	Tag		
timeInterval	Control			INTERVAL, TIME		NP	ArrayTimeInterval	ArrayTimeInterval		
sourceParameterId	DC		Y			not used.	int	int		
numLines	ObsPrep -> Control	Filler, TelCal, QL	N	NUM_LINES		NP	int	int		
sourceName	ObsPrep -> Control	Filler, TelCal, QL	Y	NAME	sourceName	NP	String	string		

calibrationGroup			N	CALIBRATION_GROUP		NP	int	int		Sources that may be calibrated together (must be calibrated together/share a calibrator).
code	Archive -> Control	Filler, TelCal, QL	N	CODE		NP	String	string		Should be a vector. Need a list (e.g., BANDPASS, PHASE, AMPLITUDE, etc).
direction	ObsPrep -> Control	Filler, TelCal, QL	Y	DIRECTION	sourceCoordinates	convert to MDirection	Angle[]	vector<Angle>		2
properMotion	ObsPrep -> Control	Filler, TelCal	N	PROPER_MOTION	pMRA,pMDec	Convert to Quanta	AngularRate[]	vector<AngularRate>		2
catalog	Control	Filler, TelCal	N	NS_CATALOG		NP	String	string		
position	Control	Filler, TelCal, QL	N	POSITION		convert to MPosition	Length[]	vector<Length>		3
transition	ObsPrep -> Control	Filler, TelCal, QL	N	TRANSITION	transitionName	NP	String[]	vector<string>	numLines	
restFrequency	ObsPrep -> Control	Filler, TelCal, QL	N	REST_FREQUENCY	restFrequency	Convert to Quanta	Frequency[]	vector<Frequency>	numLines	
sysVel	ObsPrep -> Control	Filler, TelCal, QL	N	SYSVEL	sourceVelocity	Convert to Quanta	Speed[]	vector<Speed>	numLines	
sourceModel	Archive -> Control	Filler, TelCal	N	SOURCE_MODEL		NP	String	string		Point, Gaussian
DeltaVel	ObsPrep -> Control	Filler, TelCal, QL	N	NS_DELTA_VEL		NP	Speed	Speed		
rangeVel	ObsPrep -> Control	Filler, TelCal, QL	N	NS_RANGE_VEL		NP	Speed[]	vector<Speed>		2
				PULSAR_ID						
SourceParameter (Subscan)			N			Not used by CASA; some information is in Data Repository or could be added to SOURCE.				Purpose: Source parameter information. This table provides information for known calibration sources.
sourceParameterId	DC		Y				int	int		
sourceId	DC		Y				int	int		Source identifier
timeInterval	Control		Y				ArrayTimeInterval	ArrayTimeInterval		Time when the source parameters were measured; interval over which the parameters are believed valid.
numFreq	ObsPrep -> Control	Pipeline	Y				int	int		Number of frequencies for which the fluxes are provided.
numStokes	ObsPrep -> Control	Pipeline	Y				int	int		Number of stokes parameters with known fluxes.
numDep	ObsPrep -> Control	Pipeline	Y				int	int		Number of dependencies for bootstrapped fluxes; 0: primary flux calibrator; change to string enumeration with options: primary, secondary, tertiary.
stokesParameter	ObsPrep -> Control	Pipeline	Y				int[]	vector<int>	numStokes	I, Q, U, V
flux	ObsPrep -> Control	Pipeline	Y				Flux[][]	vector<vector<Flux>>	numFreq,numStokes	Source fluxes
size	ObsPrep -> Control	Pipeline	N				Angle[][]	vector<vector<Angle>>	numFreq,2	Source sizes
positionAngle	ObsPrep -> Control	Pipeline	N				Angle[]	vector<Angle>	numFreq	Position angle of the major axis
frequency	ObsPrep -> Control	Pipeline	Y				Frequency[]	vector<Frequency>	numFreq	
frequencyInterval	ObsPrep -> Control	Pipeline	Y				Frequency[]	vector<Frequency>	numFreq	
fluxErr	ObsPrep -> Control	Pipeline	Y				Flux[][]	vector<vector<Flux>>	numFreq,numStokes	Uncertainty in the flux
sizeErr	ObsPrep -> Control	Pipeline	N				Angle[][]	vector<vector<Angle>>	numFreq, 2	Uncertainty in the size
positionAngleErr	ObsPrep -> Control	Pipeline	N				Angle[]	vector<Angle>	numFreq	Uncertainty in the position angle.
depSourceParameterId	ObsPrep -> Control	Pipeline	N				int[]	vector<int>	numDep	

SPECTRAL_WINDOW										Purpose: Spectral window information. This table describes the properties for each spectral window. A spectral window is both a frequency label for the associated data in the Main table but also represents a generic frequency conversion chain that shares joint physical properties and makes sense to calibrate as a single entity.
spectralWindowId	DC		Y			Implicit in the row number of the table.	Tag	Tag		
numChan	Control	Filler, TelCal	Y	NUM_CHAN		NP	int	int		
name	Control	Filler, TelCal	N	NAME		NP	String	string		
refFreq	ObsPrep ->Control	Filler, TelCal	Y	REF_FREQUENCY		convert to Quanta	Frequency	Frequency		
chanFreq	ObsPrep ->Control	Filler, TelCal	Y	CHAN_FREQUENCY		convert to Quanta	Frequency[]	vector<Frequency>	numChan	Center frequencies for each channel in the matrix. Can be in ascending or descending order. In topographic frame.
chanWidth	ObsPrep ->Control	Filler, TelCal	Y	CHAN_WIDTH		convert to Quanta	Frequency[]	vector<Frequency>	numChan	
measFreqRef			Y	MEAS_FREQ_REF		NP	int	int		
effectiveBw	DC	Filler, TelCl	Y	EFFECTIVE_BW		Convert to Quanta	Frequency[]	vector<Frequency>	numChan	
resolution	ObsPrep ->Control	Filler, TelCal	Y	RESOLUTION		Convert to Quanta	Frequency[]	vector<Frequency>	numChan	
totBandwidth	ObsPrep ->Control	Filler, TelCal	Y	TOTAL_BANDWIDTH		Convert to Quanta	Frequency	Frequency		
netSideband	ObsPrep ->Control	Filler, TelCal	Y	NET_SIDEBAND		NP	int	int		N:SB, DSB, LSB, USB, 2SB LSB, 2SB USB, 2SB LSB SBsep, 2SB USB SBsep +
IFConvChain			Y	IF_CONV_CHAIN		NP	int	int		Identification of the electronic signal path for the case of multiple (simultaneous) IFs.
freqGroup	ObsPrep ->Control	Filler, TelCal	N	FREQ_GROUP		NP	int	int		User Specified.
freqGroupName	ObsPrep ->Control	Filler, TelCal	N	FREQ_GROUP_NAME		NP	String	string		User Specified. For later processing (calibration/imaging).
flagRow	Control	Filler, TelCal	Y	FLAG_ROW		NP	boolean	bool		
assocSpectralWindowId	Control	Filler, TelCal	N	ASSOC_SPW_ID		NP	Tag[]	vector<Tag>	?	Indicates a derived spw (e.g., channel 0).
dopplerId	DC	Filler, TelCal	N	DOPPLER_ID		NP	int	int		
bbcNo	Control	Filler, TelCal	N	BBC_NO		NP	int	int		
bbcSideband	Control	Filler, TelCal	N	BBC_SIDEBAND	sideBand	NP	int	int		Upper, Lower; change to string.
assocNature	Control	Filler, TelCal	N	ASSOC_NATURE RECEIVER_ID		NP	String[]	vector<string>	?	
SquareLawDetector [ExecBlock]			N			Not used by CASA				Purpose: Square Law detector information
modelId	DC		Y				Tag	Tag		
numBaseBand	Control	TelCal	Y				int	int		
basebandName	TCDB ->Control	TelCal	Y				String	string	?	baseband names = 1,2,3,4
State [Integration]				STATE						Purpose: State information. This table defines the state parameters for a particular data record as they refer to external loads, calibration sources or references, and also characterizes the observing mode of the data.
stateId	DC		Y				Tag	Tag		
sig	Control	Filler, TelCal, QL	Y	SIG			boolean	bool		
ref	Control	Filler, TelCal, QL	Y	REF			boolean	bool		

calloadNum	Control	Filler, TelCal, QL	Y	CAL or LOAD?			int	int		
obsMode	Control	Filler, TelCal, QL	Y	OBS_MODE			String	string		
obsIntent	Control	Filler, TelCal, QL	Y	NS_OBS_INTENT			String	string		
flagRow	Control	Filler, TelCal, QL	Y	FLAG_ROW			boolean	bool		
weight			N	NS_WEIGHT			float	float		
Station [ExecBlock]										
stationId	DC		Y				Tag	Tag		
name	Control	Operations	Y				String	string		
position	Control	Operations	Y				Length[]	vector<Length>	->MPosition(ITRF,3,m)	
type	Control	Operations	Y				String	string		AntennaPad, CommissioningPad, WeatherStation
SubScan [Subscan]										
execBlockId	DC		Y				Tag	Tag		
scanNumber	Control		Y				int	int		
subscanNumber	Control		Y				int	int		
startTime	Control	Filler, TelCal, QL	Y				ArrayTime	ArrayTime		
endTime	Control	Filler, TelCal, QL	Y				ArrayTime	ArrayTime		
fieldName	Control	Filler, TelCal, QL	Y				String	string		
subscanIntent	Control	Filler, TelCal, QL	Y				String	string		HOLOGRAPHYRASTER, HOLOGRAPHYPHASECAL, OFFLINE
subscanMode	Control	Filler, TelCal, QL	N				String	string		Nutating, Unswitched, FrequencySwitched
numberIntegration	Control	Filler, TelCal, QL	Y				int	int		
numberSubintegration	Control	Filler, TelCal, QL	Y				int[]	vector<int>	numberIntegration	
flagRow	Control	Filler, TelCal, QL	Y				boolean	bool		
SwitchCycle [Integration]										
switchCycleId	DC		Y	All columns (NS_*)			Tag	Tag		
numStep	Obsprep -> Control	TelCal, Filler, QL	Y		numSwitchCycles		int	int		
weightArray	ObsPrep -> Control	TelCal, Filler, QL	Y				float[]	vector<float>	numStep	
offsetArray	ObsPrep -> Control	Filler, TelCal, QL	Y				Angle[]	vector<vector<Angle>>	numStep,2	
freqOffsetArray	ObsPrep -> Control	Filler, TelCal, QL	Y				Frequency[]	vector<Frequency>	numStep	
TotalPowerDetector [Integration]										
			N	NS_TPM						This table contains raw data originating from the radiometers which monitor the amount of precipitable water vapor in the pointing direction of each antenna; also used to monitor the receiver stability. 4x2GHz channels recorded at up to 2 kHz.
configDescriptionId	DC		Y			Use to obtain DATA_DESC_ID	Tag	Tag		ALMA configuration description
fieldId	DC		Y	FIELD_ID			Tag	Tag		Field identifier
time	Control		Y	TIME			ArrayTime	ArrayTime		Mid-point of time.

execBlockId	DC		Y	OBSERVATION_ID		NP	Tag	Tag		Observation/ExecBlock identifier
stateId	DC		Y	STATE_ID		NP	Tag[]	vector<Tag>	ConfigDescription.numAntenna	State identifier
scanNumber	Control	Filler	Y	SCAN_NUMBER		NP	int	int		Scan number
subscanNumber	Control	Filler	Y	SUB_SCAN		NP	int	int		Subscan number
integrationNumber	Control	Filler	Y	NA		Not filled.	int	int		Integration number
uvw	DC?	Filler	Y	UVW		Interpolation to scan based times.	Length[]	vector<vector<Length>>	ConfigDescription.numAntenna,3	UVW values
exposure	Control	Filler	Y	EXPOSURE		NP	Interval[]	vector<vector<Interval>>	ConfigDescription.numAntenna, AlmaCorrelatorMode.numBaseband	Exposure time.
timeCentroid	Control	Filler	Y	TIME_CENTROID		NP	ArrayTime[]	vector<vector<ArrayTime>>	ConfigDescription.numAntenna, AlmaCorrelatorMode.numBaseband	Time centroid
floatData	Control	Filler	Y	FLOAT_DATA		NP	float[][]	vector<vector<vector<float>>>>		Total Power data.
flagAnt	Control	Filler	Y	FLAG_ANT		NP	int[]	vector<int>	ConfigDescription.numAntenna	4-byte vector of size N_antenna. Each bit corresponds to a specified flagging condition: 0: Last WVR calibration failed, 1: Current WVR hardware defect, 2: Last pointing calibration failed, 3: Last temperature scale calibration failed, 4: Temperature scale calibration hardware defect, 5: Last WVR calibration failed, 6: Shadowing, 7: LO1 out of lock, 8: No correlation detected on last calibrator, 9: Too much decorrelation on last correlator.
flagPol	Control	Filler	Y	FLAG_POL		NP	int[]	vector<vector<int>>		Unsigned long int (4-bytes) vector. When different from zero, all data concerning the polarization and antenna are affected ((i.e., all basebands - e.g., a problem incident before the down converters)). The filler will use the flags from the binary attachment; 0: Mixer not operational, 1: Total power out of range, 2: LO2 out of lock, 3: Integration totally blanked, 4: Interference detected, 7: No correlation detected on last calibrator, 9: Too much decorrelation on last calibrator.
flagRow	Control	Filler	Y	FLAG_ROW		NP	boolean	bool		Flag row
interval	Control	Filler	Y	INTERVAL		NP	Interval	Interval		Time interval
subintegrationNumber	Control		N	NA		Not filled.	int	int		Subintegration number (channel averaged data).
WVMCal [Scan]				NS WVMCAL		All columns copied with no modification				This table contains time-variable calibration methods for radiometric corrections.
antennaId	DC		Y	ANTENNA_ID		NP	Tag	Tag		Antenna identifier
spectralWindowId	DC		Y	SPECTRAL_WINDOW_ID		NP	Tag	Tag		Spectral window identifier
timeInterval	Control		Y	TIME, INTERVAL		NP	ArrayTimeInterval	ArrayTimeInterval		Mid-point of the time interval over which the data are valid.
operationMode			Y	OPERATION_MODE		NP	String	string		corrected, uncorrected, correctedAndUncorrected
numPol	TelCal -> Control	Operations	Y	NUM_POL		NP	int	int		Series order of the polynomial expansion.
freqOrigin	TelCal -> Control	Operations	Y	FREQ_ORIGIN		NP	Frequency	Frequency		Frequency origin for the polynomial expansion
pathCoeff	TelCal -> Control	Operations	Y	PATH_COEFF		NP	double[]	vector<double>	numPoly	Conversion formula coefficients of WVR to path length.
calibrationMode	TelCal -> Control	Operations	Y	CALIBRATION_MODE		NP	String	string		apriori, empirical
wvrefModel	TelCal -> Control	Operations	N	WVREF_MODEL		NP	float	float		Calibration parameter which describes the reference model for the calibration of a radiometer.
Weather [Unknown]				WEATHER						This table contains mean external atmospheric and weather information.
stationId	DC		Y	ANTENNA_ID		NP	Tag	Tag		Station identifier.
timeInterval	Control		Y	INTERVAL, TIME		NP	ArrayTimeInterval	ArrayTimeInterval		Note timeInterval will provide information to supply both the TIME and INTERVAL columns.
h2o	Control	Filler	Y	H2O		From CalAtmosphere table, must be linked to the same time interval.	float	float	numReceptors	This is not in the ASDM table and so must be obtained from CalAtmosphere.
pressure	Control	Filler	Y	PRESSURE		NP	Pressure	Pressure	-> Pressure(1,hPa)	Ambient atmospheric pressure
relHumidity	Control	Filler	Y	REL_HUMIDITY		NP	Humidity	Humidity	-> Humidity(1,m-2)	Ambient relative humidity

temperature	Control	Filler	Y	TEMPERATURE		NP	Temperature	Temperature	-> Temperature(1,K)	Ambient temperature
windDirection	Control	Filler	Y	WIND_DIRECTION		NP	Angle	Angle	-> Angle(1,rad)	Average wind direction
windSpeed	Control	Filler	Y	WIND_SPEED		NP	Speed	Speed	-> Velocity(1,m.s-1)	Average wind speed
windMax	Control	Filler	Y	WIND_MAX		NP	Speed	Speed	-> Velocity(1,m.s-1)	Maximum wind speed
dewPoint	Control	Filler	N	DEW_POINT		NP	Temperature	Temperature	-> Temperature(1,K0)	Dew point temperature
h2oFlag	Control	Filler	Y	H2O_FLAG		NP	boolean	bool		Water flag
pressureFlag	Control	Filler	Y	PRESSURE_FLAG		NP	boolean	bool		Pressure flag
relHumidityFlag	Control	Filler	Y	REL_HUMIDITY_FLAG		NP	boolean	bool		RH flag
temperatureFlag	Control	Filler	Y	TEMPERATURE_FLAG		NP	boolean	bool		Temperature flag
dewPointFlag	Control	Filler	N	DEW_POINT_FLAG		NP	boolean	bool		Dew point flag
windDirectionFlag	Control	Filler	Y	WIND_DIRECTION_FLAG		NP	boolean	bool		Wind direction flag
windSpeedFlag	Control	Filler	Y	WIND_SPEED_FLAG		NP	boolean	bool		Wind speed flag
windMaxFlag	Control	Filler	Y	WIND_MAX_FLAG		NP	boolean	bool		Wind max flag