MeasurementSet definition version 3.0β

MS v3.0 working group:
Dijkema T.J., Golap K., Hiriart R., Nakazato T., Pokorny M., Pouzols F., Raba,R., van Diepen G.

November, 2020

Contents

1 Introduction

The MeasurementSet (MS) defines a format in which interferometer visibilities and single-dish data are stored. It is implemented in software packages using casa/casacore code. Version 2.0 (Note 229; Kemball and Wieringa 2000) of the MeasurementSet has been in use since 2000 in different packages and at different telescopes. This revision is to correct or help in common usages by adding missing data that was felt needed for data reduction. In V2.0 we feel that sometimes schema structure was defined to help with efficiency issues that was forseen (e.g definition of a FLOAT_DATA column for single dish). In this revision we try to not do that. We realize that a given data structure can be implemented in different ways in software to achieve maximum efficiency or speed. This is the reason why we have tried to keep the information redundancy to a minimum and focused on completeness(although we may attach an addendum in the future to this document, to describe implementation choices especially with regards to IO rates and large data issues) .

The changes are not backward compatible; so converters will be needed between v2.0 and v3.0 and vice versa. Some of the changes help common needs and complaints like column versioning (removing the need of CORRECTED_DATA or FLAG_VERSION) others are expansion for VLBI and non-dish arrays (e.g LOFAR).

Here are some of the major changes from MS V2.0 to MS V3.0:

- **Explicit keys** One of the major complaints of ms v2.0 is the use of row ids in subtables as implicit keys. In v3.0 we are moving to explicit keys; e.g ANTENNA subtable will have a column ANTENNA_ID so that when making a subset MS re-indexing is not needed in the main table. This makes it easier to make a subset of an MS without the need to reindex the data in the main table or any subtables.

- **Single-dish processing:** FLOAT_DATA is replaced by DATA unifying it with interferometer. It is left to the software column implementation to store floats efficiently rather than have multiple columns defined for different data types. We do not need to define in the schematic how the data is to be stored and accessed for efficiency purposes. The implementation can decide to compress data or keep as float or integers but on request a given DATA column cell, for e.g, single precision complex numbers are served.

- **Synthesis calibration:** The need for calibrations table as part the MeasurementSet has been felt by several telescopes and those using v2.0 (e.g ALMA and EVLA) have been using non standard tables to carry calibration type information.

- **Data,weight, flag versioning** Non standard columns (e.g CORRECTED_DATA) or kludges have been used by casa (for e.g) to deal with having version of data, flag or weight. These three columns have the same shapes in every row of an MS V3.0. Whenever changes are made, e.g. in flagging or calibration, and a new version is needed a new triplet of columns will be made. It is left to software implementation to do the smart thing (not making unnecessary copies). E.g if flagging only is done and need to be saved in a new version of FLAG column, the new version of DATA an WEIGHT are just virtual columns pointing back to the previous version.

- **Data Description**The concept of data description is being deprecated. It saved a column of integers but users found it confusing. SPECTRAL_WINDOW_ID and POLARIZATION_ID are explicit columns in the main table.

- **VLBI data reduction:** From v2.0 added an optional subtable INTERFEROMETER_MODEL
- **Phased Array interferometers:** A new optional subtable, PHASED_ARRAY, which will be defining which antennas form a phased array station.
- **BEAM, EPHEMERIDES and SCAN subtables:** In MS V2.0 the BEAM (BEAM_ID), EPHEMERIDES (EPHEMERIS_ID) and SCAN (SCAN_NUMBER) subtables are referred to but were not defined. Here we explicitly define these optional subtables.
- **Baseline dependent averaging (BDA)**. MS V3.0 extends the SPECTRAL_WINDOW table and introduces new subtables for describing BDA data. See https://confluence.skatelescope.org/display/SE/BDA+keywords+draft for more information regarding BDA support.

Some features and reasons for some changes made and not made:

- **Changes:** The changes proposed here are designed to be as incremental as possible and taking into account what usage at different telescopes and software packages has found lacking.
- **Compatibility:** Some of the changes proposed here are not backwards compatible but converters can be written to move data between version 2.0 and 3.0. As there is more information in version 3.0 going to version 2.0 may be lossy.
- **Calibration information:** Many of the direct and peripheral information stored at data collection can be stored as calibration terms (Jones or Mueller matrices). Therefore MS v3.0 will carry in its definition optional calibration tables which may carry monitoring information that can be used to modify the data, weights and flags. The calibration tables definition will be added to this document once it is finalized
- **Storage:** A future document will provide results and suggestion for data storage structures for some commonly used access patterns and storage system in use.
- **Multi measurement sets:** The reason for Multi Measurement Sets (MMS) was to bypass an issue in the impementation in parallel access subsection of an ms could not be locked separately. We therefore postpone the need to define the MMS format if no implementation of multi-lockable Measurementsets is do-able.

2 Changes to existing tables.

This section contains a description of the changes proposed for each table in the MS definition. Section 5 gives a full definition of the v3.0 MS format.

All the subtables not mentioned here will have an explicit ID column; in v3.0 implicit row number being an ID is no longer valid. For example ANTENNA subtable will have a ANTENNA_ID column added as a key.

2.1 MAIN table

- **Removal of FLAG_CATEGORY** This column has not been used effectively. With the versioning of FLAG below it is redundant.
- **Removal of SIGMA** WEIGHT being always considered to be $\frac{1}{\sigma^2}$ this a redundant column.
- **Removal of FLAG_ROW** The purpose of this column was to avoid having to renumber rows when e.g. removing an antenna. This is not necessary with explicit row numbers anymore.
- **Redefinition of WEIGHT to be WEIGHT_SPECTRUM** With most of contemporary intefer-ometers being spectral machines with possibility of distinctive weights per channel, usage of MSv2.0 WEIGHT is low w.r.t WEIGHT_SPECTRUM. In this version WEIGHT column will have the same shape as the data.
- **Versioning of (DATA, WEIGHT, FLAG) triplet** These three columns will have the same shapes. There may be many versions them in an MS. But the active version will be refered to as DATA, WEIGHT and FLAG. The versions need not be copies. For e.g if only DATA is corrected and a new version is made. The active DATA will refer to DATA1 column while the active WEIGHT, FLAG (thus WEIGHT1 and FLAG1) will refer to the to WEIGHT0 and FLAG0 respectively. We leave it to the software to implement these in the most optimal fashion without making duplicate copies. This removes the need for non-standard column like CORRECTED_DATA or FLAG_VERSION etc.
- **Merging of DATA and FLOAT_DATA** It is unnecessary to have explicit DATA and FLOAT_DATA. The software implementation can do the optimal storage in the presence of float data only and serve it as complex.
- **Replacement of DATA_DESC_ID** This unique ID has caused more confusion than the extra column of integers is worth. In this version we are reverting to having explicit POLARIZATION_ID and SPECTRAL_WINDOW_ID which refers to keys in the POLARIZATION and SPECTRAL_WINDOW subtables directly. DATA_DESC_ID column is removed.

2.2 ARRAY table

- **ARRAY_ID** An explicit ID column is newly defined.
- **ARRAY_CENTER** A new column which will give the full Measures position of where the array center is assumed for each ARRAY_ID

2.3 FIELD table

Changes applicable to the FIELD table are discussed in this section.

An explicit FIELD_ID

2.3.1 Direction information

The PHASE_DIR information is to be derived from the ephemeris subtable if the EPHEMERIS_ID is not -1.

A new column optional DURATION column is added; when present in combination with TIME (time origin) it defines the time range for when the information for direction for a given FIELD_ID. In this fashion step function phasecentering that happens in some correlators

2.4 FLAG_CMD

The FLAG_CMD table is made an optional subtable.

2.5 OBSERVATION table

Apart from the explicit OBSERVATION_ID column version 3.0 adds the **ARRAY_CENTER**: A new column which will give the full Measures position of where the array center is assumed for each ARRAY_ID

2.6 SOURCE table

The SOURCE table already had a SOURCE_ID column. In this version it it proposed to generalize the SOURCE_MODEL. Apart from a TableRecord it can take a URL which will point to catalogue style source list or even images representing the model associated with a SOURCE_ID and SPECTRAL_WINDOW_ID.

2.7 SPECTRAL_WINDOW table

- Add an explicit SPECTRAL_WINDOW_ID column.
- Add an optional LO_FREQUENCY column which contains a vector of frequencies. These frequencies are the Local Oscillator frequencies in the chain of frequency conversion.
- Add an optional BDA_SPECTRAL_WINDOW_ID column, which links spectral windows that cover the same spectral window but use different BDA strategies.

3 New sub-tables

New subtables added to MS v3.0 are included in this Section.

3.1 BEAM

This subtable was proposed in v2.0 but we explicitly define it here. It is referenced from the FEED table or from the newly proposed PHASED_ARRAY table here.

3.2 EPHEMERIDES

This optional subtable is explicitly defined and is referred to from the FIELD table via the EPHEMERIS_ID column.

3.3 INTERFEROMETER_MODEL

This optional subtable contains information necessary for VLBI arrays

3.4 PHASED_ARRAY

For multi element station based antennas, this subtable contains information about elements that makes a station antenna, combination scheme etc

3.5 SCAN subtable

This subtable carries the information about the scan intent and information necessary to tie back information from the online system

3.6 CAL_TABLES (Associated tables)

A lot of data processing or online correction information are carried as non standard subtables. The formal existance of calibration tables associated with a given MS will satisfy the needs of many of the non-standard subtables.

4 MS v3.0 layout

There is a MAIN table containing a number of data columns and keys into various subtables. There is at most one of each subtable. The subtables are stored as keywords of the MS, and all defined sub-tables are tabulated below. Optional sub-tables are shown in italics.

Subtables		
Table	Contents	Keys
ANTENNA	Antenna characteristics	ANTENNA_ID
BEAM	Beam information	BEAM_ID
(*DOPPLER*)	Doppler tracking	DOPPLER_ID, SOURCE_ID
(*EPHEMERIDES*)	Ephemeris information for near field objects	EPHEMERIS_ID, TIME
FEED	Feed characteristics	FEED_ID, ANTENNA_ID, TIME, SPECTRAL_WINDOW_ID
FIELD	Field position	FIELD_ID
(*FLAG_CMD*)	Flag commands	TIME
(*FREQ_OFFSET*)	Frequency offset information	FEED_ID, ANTENNAn, FEED_ID, TIME, SPECTRAL_WINDOW_ID
HISTORY	History information	OBSERVATION_ID, TIME
(*INTERFEROMETER_MODEL*)	Information for VLBI observations	ANTENNA_ID, FIELD_ID, SPECTRAL_WINDOW_ID, TIME
OBSERVATION	Observer, Schedule, etc	OBSERVATION_ID
(*PHASED_ARRAY*)	phased array stations information	PHASED_ARRAY_ID, ANTENNA_ID
POINTING	Pointing information	ANTENNA_ID, TIME
POLARIZATION	Polarization setup	POLARIZATION_ID
PROCESSOR	Processor information	PROCESSOR_ID
(*QUALITY_FREQUENCY_STATISTIC*)	Frequency related statistics	Frequency
(*QUALITY_BASELINE_STATISTIC*)	Baseline related statistics	ANTENNA1, ANTENNA2, Frequency
(*QUALITY_TIME_STATISTIC*)	Time related statistics	TIME, Frequency
SCAN	scan information	SCAN_NUMBER
(*SOURCE*)	Source information	SOURCE_ID, SPECTRAL_WINDOW_ID, TIME
SPECTRAL_WINDOW	Spectral window setups	SPECTRAL_WINDOW_ID
STATE	State information	STATE_ID
(*SYSCAL*)	System calibration characteristics	FEED_ID, ANTENNA_ID, TIME, SPECTRAL_WINDOW_ID
(*WEATHER*)	Weather info for each antenna	ANTENNA_ID, TIME

Note that all optional columns are indicated in italics and in parentheses.

4.1 MAIN table: Data, coordinates and flags

main				
Name	Format	Units	Measure	Comments
Columns				
Keywords				
MS_VERSION	Float			MS format version
Key				
TIME	Double	s	EPOCH	Integration midpoint
(TIME_EXTRA_PREC)	Double	s		extraTIME precision
ANTENNA1	Int			First antenna
ANTENNA2	Int			Second antenna
(ANTENNA3)	Int			Third antenna
FEED1	Int			Feed on ANTENNA1
FEED2	Int			Feed on ANTENNA2
(FEED3)	Int			Feed on ANTENNA3
SPECTRAL_WINDOW_ID	Int			Spectral window id.
POLARIZATION_ID	Int			polarization id.
PROCESSOR_ID	Int			Processor id.
(PHASE_ID)	Int			Phase id.
FIELD_ID	Int			Field id.
Non-key attributes				
INTERVAL	Double	s		Sampling interval
EXPOSURE	Double	s		The effective integration time
TIME_CENTROID	Double	s	EPOCH	Time centroid
(PULSAR_BIN)	Int			Pulsar bin number
(PULSAR_GATE_ID)	Int			Pulsar gate id.
SCAN_NUMBER	Int			Scan number
ARRAY_ID	Int			Subarray number
OBSERVATION_ID	Int			Observation id.
STATE_ID	Int			State id.
(BASELINE_REF)	Bool			Reference antenna
UVW	Double(3)	m	UVW	UVW coordinates
(UVW2)	Double(3)	m	UVW	UVW (baseline 2)

4.2 MAIN table: continued

Name	Format	Units	Measure	Comments
Data				
(DATA)	Complex(N_c, N_f)			Complex visibility matrix (synthesis arrays)
(VIDEO_POINT)	Complex(N_c)			Video point
(LAG_DATA)	Complex(N_c, N_l)			Correlation function
WEIGHT	Float(N_c, N_f^*)			Weight for each channel
Flag information				
FLAG	Bool(N_c, N_f^*)			Cumulative data flags

Notes:

Note that N_l= number of lags, N_c= number of correlators, N_f= number of frequency channels, and N_{cat}= number of flag categories.

MS_VERSION The MeasurementSet format revision number, expressed as *major_revision.minor_revision*. This version is 2.0.

SORT_ORDER Sort order as either "ASCENDING" or "DESCENDING".

TIME Mid-point (not centroid) of data interval.

TIME_EXTRA_PREC Extra time precision.

ANTENNA*n* Antenna number (≥ 0), and a direct key index into the ANTENNA sub-table. For $n > 2$, triple-product data are implied.

FEED*n* Feed number (≥ 0). For $n > 2$, triple-product data are implied.

SPECTRAL_WINDOW_ID Spectral window identifier (≥ 0), and a direct key index into the SPECTRAL_WINDOW sub-table.

POLARIZATION_ID Polarization identifier (≥ 0), and a direct key index into the POLARIZATION sub-table.

PROCESSOR_ID Processor indentifier (≥ 0), and a direct key index into the PROCESSOR sub-table.

PHASE_ID Switching phase identifier (≥ 0)

FIELD_ID Field identifier (≥ 0) a direct key index into the FIELD sub-table.

INTERVAL Data sampling interval. This is the nominal data interval and does not include the effects of bad data or partial integration.

EXPOSURE Effective data interval, including bad data and partial averaging.

PULSAR_BIN Pulsar bin number for the data record. Pulsar data may be measured for a limited number of pulse phase bins. The pulse phase bins are described in the PULSAR sub-table and indexed by this bin number.

PULSAR_GATE_ID Pulsar gate identifier (≥ 0), and a direct index into the PULSAR_GATE sub-table *rownr*.

SCAN_NUMBER Arbitrary scan number to identify data taken in the same logical scan. Not required to be unique; (≥ 0) a direct key index into the SCAN sub-table.

ARRAY_ID Subarray identifier (≥ 0), which identifies data in separate subarrays, as defined in Section 3.3.

OBSERVATION_ID Observation identifier (≥ 0), which identifies data from separate observations, as defined in Section 3.3.

STATE_ID State identifier (≥ 0), as defined in Section 3.1.5.

BASELINE_REF Flag to indicate the original correlator reference antenna for baseline-based correlators (True for ANTENNA1; False for ANTENNA2).

UVW *uvw* coordinates for the baseline from ANTENNE2 to ANTENNA1, i.e. the baseline is equal to the difference POSITION2 - POSITION1. The UVW given are for the TIME_CENTROID, and correspond in general to the reference type for the PHASE_DIR of the relevant field. I.e. J2000 if the phase reference direction is given in J2000 coordinates. However, any known reference is valid. Note that the choice of baseline direction and UVW definition (W towards source direction; V in plane through source and system's pole; U in direction of increasing longitude coordinate) also determines the sign of the phase of the recorded data.

UVW2 *uvw* coordinates for the baseline from ANTENNE3 to ANTENNA1 (triple-product data only), i.e. the baseline is equal to the difference POSITION3 - POSITION1. The UVW given are for the TIME_CENTROID, and correspond in general to the reference type for the PHASE_DIR of the relevant field. I.e. J2000 if the phase reference direction is given in J2000 coordinates. However, any known reference is valid. Note that the choice of baseline direction and UVW definition (W towards source direction; V in plane through source and system's pole; U in direction of increasing longitude coordinate) also determines the sign of the phase of the recorded data.

DATA, LAG_DATA At least one of these columns should be present in a given MeasurementSet. In special cases one or more could be present (e.g., single dish data used in synthesis imaging or a mix of auto and crosscorrelations on a multi-feed single dish). If only correlation functions are stored in the MS, then N_f^* is the maximum number of lags (N_l) specified in the LAG table for this LAG_ID. If both correlation functions and frequency spectra are stored in the same MS, then N_f^* is the number of frequency channels, and the weight information refers to the frequency spectra only. The units for these columns (eg. 'Jy') specify whether the data are in flux density units or correlation coefficients.

VIDEO_POINT The video point for the spectrum, to allow the full reverse transform.

WEIGHT The weight for each channel, with the same shape as DATA, as assigned by the correlator or processor.

FLAG An array of Boolean values with the same shape as DATA (see the DATA item above) representing the cumulative flags applying to this data matrix. Data are flagged bad if the FLAG array element is True.

4.3 ANTENNA: Antenna characteristics

Name	Format	Units	Measure	Comments
Columns				
Keys				
ANTENNA_ID	Int			unique antenna id
Data				
NAME	String			Antenna name
STATION	String			Station name
TYPE	String			Antenna type
MOUNT	String			Mount type:alt-az, equatorial, X-Y, nasmyth, coude, orbiting, bizarre
POSITION	Double(3)	m	POSITION	Antenna X,Y,Z phase reference positions
OFFSET	Double(3)	m	POSITION	Axes offset of mount to FEED REFERENCE point
DISH_DIAMETER	Double	m		Diameter of dish
(ORBIT_ID)	Int			Orbit id.
(MEAN_ORBIT)	Double(6)			Mean Keplerian elements
(PHASED_ARRAY_ID)	Int			Phased array id.

Notes: This sub-table contains the global antenna properties for each antenna in the MS. It is indexed directly from MAIN via ANTENNAn.

ANTENNA_ID ID of the antenna.

NAME Antenna name (e.g. "NRAO_140")

STATION Station name (e.g. "GREENBANK")

TYPE Antenna type. Reserved keywords include: ("GROUND-BASED" - conventional antennas; "SPACE-BASED" - orbiting antennas; "TRACKING-STN" - tracking stations).

MOUNT Mount type of the antenna. Reserved keywords include: ("EQUATORIAL" - equatorial mount; "ALT-AZ" - azimuth-elevation mount; "X-Y" - x-y mount; "SPACE-HALCA" - specific orientation model.)

POSITION In a right-handed frame, X towards the intersection of the equator and the Greenwich meridian, Z towards the pole. The exact frame should be specified in the MEASURE_REFERENCE keyword (ITRF or WGS84). The reference point is the point on the az or ha axis closest to the el or dec axis.

OFFSET Axes offset of mount to feed reference point.

DISH_DIAMETER Nominal diameter of dish, as opposed to the effective diameter.

ORBIT_ID Orbit identifier. Index used in ORBIT sub-table if ANTENNA_TYPE is "SPACE_BASED".

MEAN_ORBIT Mean Keplerian orbital elements, using the standard convention (Flatters 1998):

- **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 (Ω) in deg.

- **4:** Argument of perigee (ω) in *deg*.
- **5:** Mean anomaly (M) in *deg*.

PHASED_ARRAY_ID Phased array identifier. Points to a PHASED_ARRAY sub-table which points back to multiple entries in the ANTENNA sub-table and contains information on how they are combined.

4.4 BDA_TIME_AXIS: Metadata for time table

bda-time-axis				
Name	Format	Units	Measure	Comments
Columns				
Keywords				
BDA_TIME_AXIS_VERSION	String			Version tag.
Key				
BDA_TIME_AXIS_ID	Int			Unique row id.
FIELD_ID	Int			Field id.
BDA_SPECTRAL_WINDOW_ID	Int			Spectral window id.
Data				
IS_BDA_APPLIED	Bool			BDA is applied.
(*SINGLE_FACTOR_PER_BASELINE*) Bool			Baselines have a single averaging factor.	
(*MAX_TIME_INTERVAL*)	Double	s		Maximum time interval.
(*MIN_TIME_INTERVAL*)	Double	s		Minimum time interval.
(*UNIT_TIME_INTERVAL*)	Double	s		Unit time interval.
(*INTEGER_INTERVAL_FACTORS*)	Bool			Interval factors are integers.
(*HAS_BDA_ORDERING*)	Bool			Data is ordered.

BDA_TIME_AXIS_ID Unique id for the row.

FIELD_ID Field identier (≥ 0).

BDA_SPECTRAL_WINDOW_ID Spectral window identifier (≥ 0). Together with the field identifier, this key uniquely identifies a row. The BDA_SPECTRAL_WINDOW_ID values in the SPECTRAL_WINDOW table refer to this key.

IS_BDA_APPLIED BDA has been applied to the time axis.

SINGLE_FACTOR_PER_BASELINE For every baseline, the averaging factor is constant in time. If a specific baseline is averaged to 10 seconds for one timestep, it will be averaged to 10 seconds for every timestep over the selected data range. We expect this property to be true in the normal BDA cases.

MAX_TIME_INTERVAL Maximum TIME_INTERVAL over this subset. With BDA applied, it normally is the time interval of the smallest baselines. This value, together with ordering properties (discussed below), helps a software tool by telling it how long a baseline might have a contribution to an interval.

MIN_TIME_INTERVAL Minimum TIME_INTERVAL over this subset.

UNIT_TIME_INTERVAL This value is basically the original TIME_INTERVAL. If BDA is applied and the shortest baseline has not been averaged down, this value will be equal to MIN_TIME_INTERVAL. If a high-time resolution measurement set is averaged down immediately with BDA, it might be that the shortest baseline is averaged and the longest baseline is not an integer factor of the shortest baseline averaging factor (and thus MIN_TIME_INTERVAL), whereas all baselines are still a multiple of some underlying time interval.

INTEGER_INTERVAL_FACTORS The TIME_INTERVAL - and therefore also the distance (difference between two TIMEs) between two consecutive timesteps - is always an integer multiple of the UNIT_TIME_INTERVAL value. This restriction implies that for all baselines, the first intervals starts at the same time, i.e. there is no offset.

HAS_BDA_ORDERING If a row starts at T_0 (where T_0 = TIME - 0.5 * TIME_INTERVAL) then all visibilities that end before T_0 are before this row. In other words, only overlapping intervals are allowed to not obey time ordering: non-overlapping intervals are strictly ordered.

4.5 BDA_FACTORS

bda-factors				
Name	Format	Units	Measure	Comments
Columns				
Key				
BDA_TIME_AXIS_ID	Int			Reference to BDA_TIME_AXIS.
SPECTRAL_WINDOW_ID	Int			Reference to SPECTRAL_WINDOW.
ANTENNA1	Int			Antenna 1.
ANTENNA2	Int			Antenna 2.
(*ANTENNA3*)	Int			Antenna 3.
Data				
TIME_FACTOR	Int			Time averaging factor.

BDA_TIME_AXIS_ID Refers to a row in the BDA_TIME_AXIS table, since a single baseline may have different fields. The BDA_TIME_AXIS row contains the FIELD_ID and common values for multiple baselines regarding time averaging.

SPECTRAL_WINDOW_ID Refers to a row in the SPECTRAL_WINDOW table, since a single baseline may have multiple spectral windows.

ANTENNAn Antenna identier, as indexed from ANTENNAn in MAIN. Together, the antenna identifiers determine the baseline.

TIME_FACTOR Time averaging factor for the given baseline, field (via the BDA_TIME_AXIS table) and spectral window. The effective time interval for the baseline is this TIME_FACTOR times the UNIT_TIME_INTERVAL value in the BDA_TIME_AXIS table.

4.6 BEAM: Beam information

Name	Format	Units	Measure	Comments
Columns				
Key				
BEAM_ID	Int			Beam id.
Data				
TYPE	String			definition type from a fixed set of strings. E.g POLYNOMIAL, AIRY, NUMERIC, IMAGE
COEFFICIENTS	Double(NUM_COEFFICIENTS)			Depending on TYPE would be polynomial coefficients or numeric values rescaled at 1GHz
BEAM_ROTATION	Int			sign of rotation w.r.t parallactic angle
(*ALTERNATE_URL*)	String url			for types that cannot be expressed as coefficients e.g IMAGE

Notes: This sub-table contains Beam information. Referred to from the FEED and PHASED_ARRAY subtables.

BEAM_ID Beam identifier

TYPE The way the beam information is expressed. Fixed set of strings possible (POLYNOMIAL, AIRY, NUMERIC, IMAGE, GAUSSIAN,INVERSEPOLYNOMIAL, COSPOLYNOMIAL, ZERNIKE)

COEFFICIENTS The coefficients for the expressed type. All the beams are defined at 1GHz; exceptio for IMAGE: it can have the spectral coordinates expressed in the image

BEAM_ROTATION sign of beam rotation angle with respect to parallatic angle.

ALTERNATE_URL Information on how to access beams that are expressed as images (from some telescope archive for e.g).

4.7 CORRELATOR_TYPE: Doppler tracking information

Name	Format	Units	Measure	Comments

Notes: This optional sub-table contains correlator information referred to from the PROCESSOR subtable.

4.8 DOPPLER: Doppler tracking information

Name	Format	Units	Measure	Comments
Columns				
Key				
DOPPLER_ID	Int			Doppler tracking id.
SOURCE_ID	Int			Source id.
Data				
TRANSITION_INDEX	Int			Transition index. Indexes into transitions list in SOURCE
VELDEF	Double	m/s	Doppler	Velocity definition of Doppler shift.

Notes: This sub-table contains frame information for different Doppler tracking modes. It is indexed from the SPECTRAL_WINDOW_ID sub-table (with SOURCE_ID as a secondary index) and thus allows the specification of a source-dependent Doppler tracking reference for each SPECTRAL_WINDOW. This model allows multiple possible transitions per source per spectral window, but only one reference at any given time.

DOPPLER_ID Doppler identifier, as used in the SPECTRAL_WINDOW sub-table.

SOURCE_ID Source identifier (as used in the SOURCE sub-table).

TRANSITION_INDEX This index selects the appropriate line from the list of transitions stored for each SOURCE_ID in the SOURCE table.

VELDEF Velocity definition of the Doppler shift, e.g., RADIO or OPTICAL velocity in m/s.

4.9 EPHEMERIDES: Ephemerides information

Name	Format	Units	Measure	Comments
Keywords				
OBSLOC	String			observer's location for ephemerides
Columns				
Key				
EPHEMERIS_ID	Int			ephemeris id.
Data				
TIME	Double	s	EPOCH	Time of direction measurement
DIRECTION	Double(2)	rad	DIRECTION	Direction of object from obsloc position.
DISTANCE	Double	km		Distance from observer at time
RADIAL_VELOCITY	Double	m/s		Radial velocity of source
(*SHAPE*)	Double(3)	rad		Apparent elliptical shape in angular units (Major, minor, pos. angle)
(*ALTERNATE_URL*)	String			Location of Ephemerides table for this ephemeris_id

Notes: This sub-table contains ephemeris information as referred to from FIELD table.

Either multiple rows of information for a given EPHEMERIS_ID is given in this table with time or a URL is provided to get an ephemeris table of known format (e.g the one used by Measures in casacore)

EPHEMERIS_ID identifier referenced in FIELD table

TIME Epoch of observation of object

DIRECTION Measures direction of object at given time and seen from OBSLOC position.

RADIAL_VELOCITY Velocity of object

SHAPE Apparent Elliptical shape of object

ALTERNATE_URL Location of a ephemeris table in documented format.

4.10 FEED: Feed characteristics

Name	Format	Units	Measure	Comments
Columns				
Key				
ANTENNA_ID	Int			Antenna id
FEED_ID	Int			Feed id
SPECTRAL_WINDOW_ID	Int			Spectral window id.
TIME	Double	s	EPOCH	Interval midpoint
INTERVAL	Double	s		Time interval
Data description				
NUM_RECEPTORS	Int			# receptors on this feed
Data				
BEAM_ID	Int			Beam model
BEAM_OFFSET	Double(2, NUM_RECEPTORS)	rad	DIRECTION	Beam position offset (on sky but in antenna reference frame).
(FOCUS_LENGTH)	Double	m		Focus length
(PHASED_FEED_ID)	Int			Phased feed
POLARIZATION_TYPE	String (NUM_RECEPTORS)			Type of polarization to which a given RECEPTOR responds.
POL_RESPONSE	Complex (NUM_RECEPTORS, NUM_RECEPTORS)			Feed polzn. response
POSITION	Double(3)	m	POSITION	Position of feed relative to feed reference position for this antenna
RECEPTOR_ANGLE	Double (NUM_RECEPTORS)	rad		The reference angle for polarization.

Notes: A feed is a collecting element on an antenna, such as a single horn, that shares joint physical properties and makes sense to calibrate as a single entity. It is an abstraction of a generic antenna feed and is considered to have one or more RECEPTORs that respond to different polarization states. A FEED may have a time-variable beam and polarization response. Feeds are numbered from 0 on each separate antenna for each SPECTRAL_WINDOW_ID. Consequently, FEED_ID should be non-zero only in the case of feed arrays, i.e. multiple, simultaneous beams on the sky at the same frequency and polarization.

ANTENNA_ID Antenna number, as indexed from ANTENNAn in MAIN.

FEED_ID Feed identifier, as indexed from FEEDn in MAIN.

SPECTRAL_WINDOW_ID Spectral window identifier. A value of -1 indicates the row is valid for all spectral windows.

TIME Mid-point of time interval for which the feed parameters in this row are valid. The same Measure reference used for the TIME column in MAIN must be used.

INTERVAL Time interval.

NUM_RECEPTORS Number of receptors on this feed. See POLARIZATION_TYPE for further information.

BEAM_ID Beam identifier. Points to an optional BEAM sub-table defining the primary beam and polarization response for this FEED. A value of -1 indicates that no associated beam response is defined.

BEAM_OFFSET Beam position offset, as defined on the sky but in the antenna reference frame.

FOCUS_LENGTH Focus length. As defined along the optical axis of the antenna.

PHASED_FEED_ID Phased feed identifier. Points to a PHASED_FEED sub-table which in turn points back to multiple entries in the FEED table, and specifies the manner in which they are combined.

POLARIZATION_TYPE Polarization type to which each receptor responds (e.g. "R","L","X" or "Y"). This is the receptor polarization type as recorded in the final correlated data (e.g. "RR"); i.e. as measured after all polarization combiners.

POL_RESPONSE Polarization response at the center of the beam for this feed. Expressed in a linearly polarized basis $(\vec{\mathbf{e}}_{\mathbf{x}}, \vec{\mathbf{e}}_{\mathbf{y}})$ using the IEEE convention.

POSITION Offset of feed relative to the feed reference position for this antenna (see ANTENNA sub-table).

RECEPTOR_ANGLE Polarization reference angle. Converts into parallactic angle in the sky domain.

4.11 FIELD: Field positions for each source

Name	Format	Units	Measure	Comments
Columns				
Key				
FIELD_ID	Int			Field id.
Data				
NAME	String			Name of field
CODE	String			Special characteristics of field
TIME	Double	s	EPOCH	Time origin for the directions and rates
NUM_POLY	Int			Series order
DELAY_DIR	Double(2, NUM_POLY+1)	rad	DIRECTION	Direction of delay center.
PHASE_DIR	Double(2, NUM_POLY+1)	rad	DIRECTION	Phase center.
REFERENCE_DIR	Double(2, NUM_POLY+1)	rad	DIRECTION	Reference center
SOURCE_ID	Int			Index in Source table
(EPHEMERIS_ID)	Int			Ephemeris id.

Notes: The FIELD 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.

FIELD_ID Field id.

NAME Field name; user specified.

CODE Field code indicating special characteristics of the field; user specified.

TIME Time reference for the directions and rates. Required to use the same TIME Measure reference as in MAIN.

NUM_POLY Series order for the *_DIR columns.

DELAY_DIR Direction of delay center; can be expressed as a polynomial in time. Final result converted to the defined Direction Measure type.

PHASE_DIR Direction of phase center; can be expressed as a polynomial in time. Final result converted to the defined Direction Measure type.

REFERENCE_DIR Reference center; can be expressed as a polynomial in time. Final result converted to the defined Direction Measure type. Used in single-dish to record the associated reference direction if position-switching has already been applied. For interferometric data, this is the original correlated field center, and may equal DELAY_DIR or PHASE_DIR.

SOURCE_ID Points to an entry in the optional SOURCE subtable, a value of -1 indicates there is no corresponding source defined.

EPHEMERIS_ID Points to an entry in the EPHEMERIS sub-table, which defines the ephemeris used to compute the field position. Useful for moving, near-field objects, where the ephemeris may be revised over time.

4.12 FLAG_CMD: Flag commands

Name	Format	Units	Measure	Comments
Columns				
Key				
TIME	Double	s	EPOCH	Mid-point of interval
INTERVAL	Double	s		Time interval
Data				
TYPE	String			FLAG or UNFLAG
REASON	String			Flag reason
LEVEL	Int			Flag level
SEVERITY	Int			Severity code
APPLIED	Bool			True if applied in MAIN
COMMAND	String			Flag command

Notes: The FLAG_CMD sub-table defines global flagging commands which apply to the data in MAIN, as described in Section 3.1.8.

TIME Mid-point of the time interval to which this flagging command applies. Required to use the same TIME Measure reference as used in MAIN.

INTERVAL Time interval.

TYPE Type of flag command, representing either a flagging ("FLAG") or un-flagging ("UNFLAG") operation.

REASON Flag reason; user specified.

LEVEL Flag level (≥ 0); reflects different revisions of flags which have the same REASON.

SEVERITY Severity code for the flag, on a scale of 0-10 in order of increasing severity; user specified.

APPLIED True if this flag has been applied to MAIN, and update in FLAG_CATEGORY and FLAG. False if this flag has not been applied to MAIN.

COMMAND Global flag command, expressed in the standard syntax for data selection, as adopted within the project as a whole.

4.13 FREQ_OFFSET: Frequency offset information

Name	Format	Units	Measure	Comments
Columns				
Key				
ANTENNA1	Int			Antenna 1.
ANTENNA2	Int			Antenna 2.
FEED_ID	Int			Feed id.
SPECTRAL_WINDOW_ID	Int			Spectral window id.
TIME	Double	s	EPOCH	Interval midpoint
INTERVAL	Double	s		Time interval
Data				
OFFSET	Double	Hz		Frequency offset

Notes: The table contains frequency offset information, to be added directly to the defined frequency labeling in the SPECTRAL_WINDOW sub-table as a Measure offset. This allows bands with small, time-variable, ad hoc frequency offsets to be labeled as the same SPECTRAL_WINDOW_ID, and calibrated together if required.

ANTENNA*n* Antenna identifier, as indexed from ANTENNA*n* in MAIN.

FEED_ID Antenna identifier, as indexed from FEED*n* in MAIN.

SPECTRAL_WINDOW_ID Spectral window identifier.

TIME Mid-point of the time interval for which this offset is valid. Required to use the same TIME Measure reference as used in MAIN.

INTERVAL Time interval.

OFFSET Frequency offset to be added to the frequency axis for this spectral window, as defined in the SPECTRAL_WINDOW sub-table. Required to have the same Frequency Measure reference as CHAN_FREQ in that table.

4.14 HISTORY: History information

Name	Format	Units	Measure	Comments
Columns				
Key				
TIME	Double	s	EPOCH	Time-stamp for message
OBSERVATION_ID	Int			Points to OBSERVATION table
Data				
MESSAGE	String			Log message
PRIORITY	String			Message priority
ORIGIN	String			Code origin
OBJECT_ID	String			Originating ObjectID
APPLICATION	String			Application name
CLI_COMMAND	String(*)			CLI command sequence
APP_PARAMS	String(*)			Application paramters

Notes: This sub-table contains associated history information for the MS.

TIME Time-stamp for the history record. Required to have the same TIME Measure reference as used in MAIN.

OBSERVATION_ID Observation identifier (see the OBSERVATION table)

MESSAGE Log message.

PRIORITY Message priority, with allowed types: ("DEBUGGING", "WARN", "NORMAL", or "SEVERE").

ORIGIN Source code origin from which message originated.

OBJECT_ID Originating ObjectID, if available, else blank.

APPLICATION Application name.

CLI_COMMAND CLI command sequence invoking the application.

APP_PARAMS Application parameter values, in the adopted project-wide format.

4.15 INTERFEROMETER_MODEL: VLBI Interferometer information

Name	Format	Units	Measure	Comments
Columns				
Key				
TIME	Double	s	EPOCH	Time-stamp for message
ANTENNA_ID	Int			Antenna id.
FIELD_ID	Int			field id.
SPECTRAL_WINDOW_ID	Int			Spectral window id.
Data				
PHASE_DELAY	Double(N_r, num_poly)	s		Phase delay polynomial for each receptor
GROUP_DELAY	Double(N_r, num_poly)	s		Group delay polynomial for each receptor
PHASE_RATE	Double(N_r, num_poly)			Rate of change of phase delay
GROUP_RATE	Double(N_r, num_poly)			Rate of change of group delay
DISP_DELAY	Double(N_r, nunm_poly)	s		Dispersive delay
DISP_DELAY_RATE	Double(N_r, num_poly)			Dispersive delay rate
CLOCK_ERROR	Double(N_r, num_poly)			Clock error as delay polynomial
CLOCK_ERROR_RATE	Double(N_r, num_poly)			Rate of clock error modelled as a delay rate polynomial

Notes: This sub-table contains information associated for VLBI

TIME Time-stamp as origin for all time based polynomials model.

FIELD_ID Field id, FIELD subtable

ANTENNA_ID antenna id, antenna subtable

SPECTRAL_WINDOW_ID Spectral window id.

PHASE_DELAY phase delay modelled as a time polynomial for each receptor.

PHASE_RATE rate of change for phase delay .

GROUP_DELAY Group delay.

GROUP_RATE Group delay rate.

DISP_DELAY Dispersive delay

DISP_DELAY_RATE Dispersive delay rate

CLOCK_ERROR

CLOCK_ERROR_RATE

4.16 OBSERVATION: Observation information

Name	Format	Units	Measure	Comments
Columns				
Key				
OBSERVATION_ID	Int			Observation id.
Data				
TELESCOPE_NAME	String			Telescope name
ARRAY_CENTER	Double(3)	m	POSITION	Reference position for array
TIME_RANGE	Double(2)	s	EPOCH	Start, end times
OBSERVER	String			Name of observer(s)
LOG	String(*)			Observing log
SCHEDULE_TYPE	String			Schedule type
SCHEDULE	String(*)			Project schedule
PROJECT	String			Project identification string.
RELEASE_DATE	Double	s	EPOCH	Target release date

Notes: This table contains information specifying the observing instrument or epoch. See the discussion in Section 3.3 for details. It is indexed directly from MAIN via OBSERVATION_ID.

OBSERVATION_ID Observation id. key

TELESCOPE_NAME Telescope name (e.g. "WSRT" or "VLBA").

ARRAY_CENTER Reference position used by the correlator for e.g.

TIME_RANGE The start and end times of the overall observing period spanned by the actual recorded data in MAIN. Required to use the same TIME Measure reference as in MAIN.

OBSERVER The name(s) of the observer(s).

LOG The observing log, as supplied by the telescope or instrument.

SCHEDULE_TYPE The schedule type, with current reserved types ("VLBA-CRD", "VEX", "WSRT", "ATNF").

SCHEDULE Unmodified schedule file, of the type specified, and as used by the instrument.

PROJECT Project code (e.g. "BD46")

RELEASE_DATE Project release date. This is the date on which the data may become public.

4.17 PHASED_ARRAY: phased array station information

Name	Format	Units	Measure	Comments
Columns				
Key				
ANTENNA_ID	Int			Antenna ID.
PHASED_ARRAY_ID	Int			phased array id.
Data				
POSITION	Double(3)	m	POSITION	Position of antenna field
COORDINATE_SYSTEM	Double(3,3)	m	DIRECTION	Local coordinate system
ELEMENT_OFFSET	Double(3, N_{ant})	m	POSITION	Offset per element
(*BEAM_ID*)	Int			Beam id.
ELEMENT_FLAG	Bool(N_{ant})			Flag of elements in array

Notes: The table contains information about phased array information. It refers to antenna ids in ANTENNA table, FEED_ID for feed used in forming this phased array.

ANTENNA_ID Index into the ANTENNA table to show to which ANTENNA this phased array belongs. Note that this is an n-to-1 mapping: one ANTENNA can consist of multiple PHASED_ARRAYs.

POSITION Position of the antenna field in absolute ITRF coordinates

COORDINATE_SYSTEM (cartesian) direction vectors in ITRF (or measure defined) describing the local field coordinate system. This defines the antenna field plane, and the 'up' direction, the normal direction to the antenna field plane. Note that in general this is not the zenith direction. The coordinate system is necessary to evaluate the beam pattern directions. The coordinate system can also be used for describing polarization alignment.

ELEMENT_OFFSET Relative offsets of each element from POSITION

ELEMENT_FLAG flag for invalid antennas in array not used.

BEAM_ID Beam id as defined in BEAM table.

4.18 POINTING: Antenna pointing information

Name	Format	Units	Measure	Comments
Columns				
Key				
ANTENNA_ID	Int			Antenna id.
TIME	Double	s	EPOCH	Interval midpoint
INTERVAL	Double	s		Time interval
Data				
NAME	String			Pointing position desc.
NUM_POLY	Int			Series order
TIME_ORIGIN	Double	s	EPOCH	Origin for the polynomial
DIRECTION	Double(2, NUM_POLY+1)	rad	DIRECTION	Antenna pointing direction
TARGET	Double(2, NUM_POLY+1)	rad	DIRECTION	Target direction
(POINTING_OFFSET)	Double(2, NUM_POLY+1)	rad	DIRECTION	A priori pointing correction
(SOURCE_OFFSET)	Double(2, NUM_POLY+1)	rad	DIRECTION	Offset from source
(ENCODER)	Double(2)	rad	DIRECTION	Encoder values
(POINTING_MODEL_ID)	Int			Pointing model id.
TRACKING	Bool			True if on-position
(ON_SOURCE)	Bool			True if on-source
(OVER_THE_TOP)	Bool			True if over the top

Notes: This table contains information concerning the primary pointing direction of each antenna as a function of time. Note that the pointing offsets for inidividual feeds on a given antenna are specified in the FEED sub-table with respect to this pointing direction.

ANTENNA_ID Antenna identifier, as specified by ANTENNAn in MAIN.

TIME Mid-point of the time interval for which the information in this row is valid. Required to use the same TIME Measure reference as in MAIN.

INTERVAL Time interval.

NAME Pointing direction name; user specified.

NUM_POLY Series order for the polynomial expressions in DIRECTION and POINTING_OFFSET.

TIME_ORIGIN Time origin for the polynomial expansions.

DIRECTION Antenna pointing direction, optionally expressed as polynomial coefficients. The final result is interpreted as a Direction Measure using the specified Measure reference.

TARGET Target pointing direction, optionally expressed as polynomial coefficients. The final result is interpreted as a Direction Measure using the specified Measure reference. This is the true expected position of the source, including all coordinate corrections such as precession, nutation etc.

POINTING_OFFSET The a priori pointing corrections applied by the telescope in pointing to the DIRECTION position, optionally expressed as polynomial coefficients. The final result is interpreted as a Direction Measure using the specified Measure reference.

SOURCE_OFFSET The commanded offset from the source position, if offset pointing is being used.

ENCODER The current encoder values on the primary axes of the mount type for the antenna, expressed as a Direction Measure.

TRACKING True if tracking the nominal pointing position.

ON-SOURCE True if the nominal pointing direction coincides with the source, i.e. offset-pointing is not being used.

OVER-THE-TOP True if the antenna was driven to this position "over the top" (az-el mount).

4.19 POLARIZATION: Polarization setup information

Name	Format	Units	Measure	Comments
Columns				
Data description columns				
NUM_CORR	Int			# correlations
Data				
CORR_TYPE	Int(NUM_CORR)			Polarization of correlation
CORR_PRODUCT	Int(2, NUM_CORR)			Receptor cross-products

Notes: This table defines the polarization labeling of the DATA array in MAIN, and is directly indexed by POLARIZATION_ID.

NUM_CORR The number of correlation polarization products. For example, for (RR) this value would be 1, for (RR, LL) it would be 2, and for (XX,YY,XY,YX) it would be 4, etc.

CORR_TYPE An integer for each correlation product indicating the Stokes type as defined in the Stokes class enumeration.

CORR_PRODUCT Pair of integers for each correlation product, specifying the receptors from which the signal originated. The receptor polarization is defined in the POLARIZATION_TYPE column in the FEED table. An example would be (0,0), (0,1), (1,0), (1,1) to specify all correlations between two receptors.

4.20 PROCESSOR: Processor information

Name	Format	Units	Measure	Comments
Columns				
Data				
TYPE	String			Processor type
SUB_TYPE	String			Processor sub-type
TYPE_ID	Int			Processor type id.
MODE_ID	Int			Processor mode id.
(PASS_ID)	Int			Processor pass number

Notes: This table holds summary information for the back-end processing device used to generate the basic data in the MAIN table. Such devices include correlators, radiometers, spectrometers, pulsar-timers, amongst others. See Section 4.0.4 for further details.

TYPE Processor type; reserved keywords include (”CORRELATOR” - interferometric correlator; ”SPECTROMETER” - single-dish correlator; ”RADIOMETER” - generic detector/integrator; ”PULSAR-TIMER” - pulsar timing device).

SUB_TYPE Processor sub-type, e.g. ”GBT” or ”JIVE”.

TYPE_ID Index used in a specialized sub-table named as *subtype_type*, which contains time-independent processor information applicable to the current data record (e.g. a JIVE_CORRELATOR sub-table). Time-dependent information for each device family is contained in other tables, dependent on the device type.

MODE_ID Index used in a specialized sub-table named as *subtype_type_mode*, containing information on the processor mode applicable to the current data record. (e.g. a GBT_SPECTROMETER_MODE sub-table).

PASS_ID Pass identifier; this is used to distinguish data records produced by multiple passes through the same device, where this is possible (e.g. VLBI correlators). Used as an index into the associated table containing pass information.

4.21 SCAN: Scan information

Name	Format	Units	Measure	Comments
Columns				
Key				
SCAN_NUMBER	Int			Scan number id; referred from the Main table
Data				
SCAN_INTENT	String			Fixed set of string defining the intent of the scan
(*EXECBLOCK_ID*)	Int			Execution block id
(*TIME*)	Double	s	EPOCH	start time of scan
(*INTERVAL*)	Double	s		time from begining from which scan is valid

Notes: This sub-table contains associated scan information for the MS.

SCAN_NUMBER Scan number as referred from main table

SCAN_INTENT Intent for the scan (an scan number may have multiple intents). Fixed set of string for a telescope e.g CAL_POINTING or TARGET

EXECBLOCK_ID A number that is unique to the observation execution block. Used to get more info from the online system of some telescopes.

TIME Time of start for that scan

TIME_INTERVAL interval for which this scan is observed from TIME

4.22 SOURCE: Source information

Name	Format	Units	Measure	Comments
Columns				
Key				
SOURCE_ID	Int			Source id
TIME	Double	s	EPOCH	Midpoint of time for which this set of parameters is accurate
INTERVAL	Double	s		Interval
SPECTRAL_WINDOW_ID	Int			Spectral Window id
Data description				
NUM_LINES	Int			Number of spectral lines
Data				
NAME	String			Name of source as given during observations
CALIBRATION_GROUP	Int			# grouping for calibration purpose
CODE	String			Special characteristics of source, e.g. Bandpass calibrator
DIRECTION	Double(2)	rad	DIRECTION	Direction (e.g. RA, DEC)
(POSITION)	Double(3)	m	POSITION	Position (e.g. for solar system objects)
PROPER_MOTION	Double(2)	rad/s		Proper motion
(TRANSITION)	String(NUM_LINES)			Transition name
(REST_FREQUENCY)	Double(NUM_LINES)	Hz	FREQUENCY	Line rest frequency
(SYSVEL)	Double(NUM_LINES)	m/s	RADIAL VELOCITY	Systemic velocity at reference
(SOURCE_MODEL)	TableRecord or String			Default csm
(PULSAR_ID)	Int			Pulsar id.

Notes: This table contains time-variable source information, optionally associated with a given FIELD_ID.

SOURCE_ID Source identifier (≥ 0), as specified in the FIELD sub-table.

TIME Mid-point of the time interval for which the data in this row is valid. Required to use the same TIME Measure reference as in MAIN.

INTERVAL Time interval.

SPECTRAL_WINDOW_ID Spectral window identifier. A -1 indicates that the row is valid for all spectral windows.

NUM_LINES Number of spectral line transitions associated with this source and spectral window id. combination.

NAME Source name; user specified.

CALIBRATION_GROUP Calibration group number to which this source belongs; user specified.

CODE Source code, used to describe any special characteristics f the source, such as the nature of a calibrator. Reserved keyword, including ("BANDPASS CAL").

DIRECTION Source direction at this TIME.

POSITION Source position (x, y, z) at this TIME (for near-field objects).

PROPER_MOTION Source proper motion at this TIME.

TRANSITION Transition names applicable for this spectral window (e.g. "v=1, J=1-0, SiO").

REST_FREQUENCY Rest frequencies for the transitions.

SYSVEL Systemic velocity for each transition.

SOURCE_MODEL Reference to an assigned component source model table or a URL to access source model

PULSAR_ID An index used in the PULSAR sub-table to define further pulsar-specific properties if the source is a pulsar.

4.23 SPECTRAL_WINDOW: Spectral window description

spectral-window				
Name	Format	Units	Measure	Comments
Columns				
Key				
SPECTRAL_WINDOW_ID	Int			Spectral window id
Data description columns				
NUM_CHAN	Int			# spectral channels.
Data				
NAME	String			Spectral window name.
REF_FREQUENCY	Double	Hz	FREQUENCY	The reference frequency.
CHAN_FREQ	Double(NUM_CHAN)	Hz	FREQUENCY	Center frequencies for each channel in the data matrix.
CHAN_WIDTH	Double(NUM_CHAN)	Hz		Channel width for each channel in the data matrix.
MEAS_FREQ_REF	Int			FREQUENCY Measure ref.
EFFECTIVE_BW	Double(NUM_CHAN)	Hz		The effective noise bandwidth of each spectral channel.
RESOLUTION	Double(NUM_CHAN)	Hz		The effective spectral resolution of each channel.
TOTAL_BANDWIDTH	Double	Hz		The total bandwidth for this window.
NET_SIDEBAND	Int			Net sideband.
(BBC_NO)	Int			Baseband converter no.
(BBC_SIDEBAND)	Int			BBC sideband.
IF_CONV_CHAIN	Int			The IF conversion chain.
(RECEIVER_ID)	Int			Receiver id.
(LO_FREQUENCY)	Double			LO frequency.
FREQ_GROUP	Int			Frequency group.
FREQ_GROUP_NAME	String			Frequency group name.
(DOPPLER_ID)	Int			Doppler id.
(ASSOC_SPW_ID)	Int(*)			Associated spw_id.
(ASSOC_NATURE)	String(*)			Nature of association.
(BDA_SPECTRAL_WINDOW_ID)	Int			BDA set id.

Notes: This table describes properties for each defined spectral window. A spectral window is both a frequency label for the associated DATA array in MAIN, but also represents a generic frequency conversion chain that shares joint physical properties and makes sense to calibrate as a single entity.

SPECTRAL_WINDOW_ID spectral window id key.

NUM_CHAN Number of spectral channels.

NAME Spectral window name; user specified.

REF_FREQUENCY The reference frequency. A frequency representative of this spectral window, usually the sky frequency corresponding to the DC edge of the baseband. Used by the calibration system if a fixed scaling frequency is required or in algorithms to identify the observing band.

CHAN_FREQ Center frequencies for each channel in the data matrix. These can be frequency-dependent, to accommodate instruments such as acousto-optical spectrometers. Note that the channel frequencies may be in ascending or descending frequency order.

CHAN_WIDTH Nomical channel width of each spectral channel. Although these can be derived from CHAN_FREQ by differencing, it is more efficient to keep a separate reference to this information.

MEAS_FREQ_REF Frequency Measure reference for CHAN_FREQ. This allows a row-based reference for this column in order to optimize the choice of Measure reference when Doppler tracking is used. Modified only by the MS access code.

EFFECTIVE_BW The effective noise bandwidth of each spectral channel.

RESOLUTION The effective spectral resolution of each channel.

TOTAL_BANDWIDTH The total bandwidth for this spectral window.

NET_SIDEBAND The net sideband for this spectral window.

BBC_NO The baseband converter number, if applicable.

BBC_SIDEBAND The baseband converter sideband, is applicable.

IF_CONV_CHAIN Identification of the electronic signal path for the case of multiple (simultaneous) IFs. (e.g. VLA: AC=0, BD=1, ATCA: Freq1=0, Freq2=1)

RECEIVER_ID Index used to identify the receiver associated with the spectral window. Further state information is planned to be stored in a RECEIVER sub-table.

LO_FREQUENCY LO frequency used for this spectral window

FREQ_GROUP The frequency group to which the spectral window belongs. This is used to associate spectral windows for joint calibration purposes.

FREQ_GROUP_NAME The frequency group name; user specified.

DOPPLER_ID The Doppler identifier defining frame information for this spectral window.

ASSOC_SPW_ID Associated spectral windows, which are related in some fashion (e.g. "channel-zero").

ASSOC_NATURE Nature of the association for ASSOC_SPW_ID; reserved keywords are ("CHANNEL-ZERO" - channel zero; "EQUAL-FREQUENCY" - same frequency labels; "SUBSET" - narrow-band subset).

BDA_SPECTRAL_WINDOW_ID An id that links a set of spectral windows that cover the same (true/original) spectral window. It is the equal for all spectral windows where the only difference is the amount of frequency averaging. This id refers to the BDA_SPECTRAL_WINDOW_ID in the BDA_TIME_AXIS table, which contains information regarding the time averaging for the spectral window.

4.24 STATE: State information

Name	Format	Units	Measure	Comments
Columns				
Key				
STATE_ID	Int			State id key
Data				
SIG	Bool			Signal
REF	Bool			Reference
CAL	Double	K		Noise calibration
LOAD	Double	K		Load temperature
SUB_SCAN	Int			Sub-scan number
OBS_MODE	String			Observing mode

Notes: 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 record, as an aid to defining the scheduling heuristics. It is indexed directly via STATE_ID in MAIN.

SIG True if the source signal is being observed.

REF True for a reference phase.

CAL Noise calibration temperature (zero if not added).

LOAD Load temperature (zero if no load).

SUB_SCAN Sub-scan number (≥ 0), relative to the SCAN_NUMBER in MAIN. Used to identify observing sequences.

OBS_MODE Observing mode; defined by a set of reserved keywords characterizing the current observing mode (e.g. "OFF-SPECTRUM"). Used to define the schedule strategy.

4.25 SYSCAL: System calibration

Name	Format	Units	Measure	Comments
Columns				
Key				
ANTENNA_ID	Int			Antenna id
FEED_ID	Int			Feed id
SPECTRAL_WINDOW_ID	Int			Spectral window id
TIME	Double	s	EPOCH	Midpoint of time for which this set of parameters is accurate
INTERVAL	Double	s		Interval
Data				
(PHASE_DIFF)	Float	rad		Phase difference between receptor 0 and receptor 1
(TCAL)	Float (N_r)	K		Calibration temp
(TRX)	Float (N_r)	K		Receiver temperature
(TSKY)	Float (N_r)	K		Sky temperature
(TSYS)	Float (N_r)	K		System temp
(TANT)	Float (N_r)	K		Antenna temperature
(TANT_TSYS)	Float(N_r)			$\frac{T_{ant}}{T_{sys}}$
(TCAL_SPECTRUM)	Float (N_r, N_f)	K		Calibration temp
(TRX_SPECTRUM)	Float (N_r, N_f)	K		Receiver temperature
(TSKY_SPECTRUM)	Float (N_r, N_f)	K		Sky temperature spectrum
(TSYS_SPECTRUM)	Float (N_r, N_f)	K		System temp
(TANT_SPECTRUM)	Float (N_r, N_f)	K		Antenna temperature spectrum
(TANT_TSYS_SPECTRUM)	Float (N_r,N_f)			$\frac{T_{ant}}{T_{sys}}$ spectrum
Flags				
(PHASE_DIFF_FLAG)	Bool			Flag for PHASE_DIFF
(TCAL_FLAG)	Bool			Flag for TCAL
(TRX_FLAG)	Bool			Flag for TRX
(TSKY_FLAG)	Bool			Flag for TSKY
(TSYS_FLAG)	Bool			Flag for TSYS
(TANT_FLAG)	Bool			Flag for TANT
(TANT_TSYS_FLAG)	Bool			Flag for $\frac{T_{ant}}{T_{sys}}$

Notes: This table contains time-variable calibration measurements for each antenna, as indexed on feed and spectral window. Note that N_r= number of receptors, and N_f= number of frequency channels.

ANTENNA_ID Antenna identifier, as indexed by ANTENNAn in MAIN.

FEED_ID Feed identifier, as indexed by FEEDn in MAIN.

SPECTRAL_WINDOW_ID Spectral window identifier.

TIME Mid-point of the time interval for which the data in this row are valid. Required to use the same TIME Measure reference as that in MAIN.

INTERVAL Time interval.

PHASE_DIFF Phase difference between receptor 0 and receptor 1.

TCAL Calibration temperature.

TRX Receiver temperature.

TSKY Sky temperature.

TSYS System temperature.

TANT Antenna temperature.

TANT_TSYS Antenna temperature over system temperature.

TCAL_SPECTRUM Calibration temperature spectrum.

TRX_SPECTRUM Receiver temperature spectrum.

TSKY_SPECTRUM Sky temperature spectrum.

TSYS_SPECTRUM System temperature spectrum.

TANT_SPECTRUM Antenna temperature spectrum.

TANT_TSYS_SPECTRUM Antenna temperature over system temperature spectrum.

PHASE_DIFF_FLAG True if PHASE_DIFF flagged.

TCAL_FLAG True if TCAL flagged.

TRX_FLAG True if TRX flagged.

TSKY_FLAG True if TSKY flagged.

TSYS_FLAG True if TSYS flagged.

TANT_FLAG True if TANT flagged.

TANT_TSYS_FLAG True if TANT_TSYS flagged.

4.26 WEATHER: weather station information

Name	Format	Units	Measure	Comments
Columns				
Key				
ANTENNA_ID	Int			Antenna number
TIME	Double	s	EPOCH	Mid-point of interval
INTERVAL	Double	s		Interval over which data is relevant
Data				
(H2O)	Float	m^{-2}		Average column density of water
(IONOS_ELECTRON)	Float	m^{-2}		Average column density of electrons
(PRESSURE)	Float	hPa		Ambient atmospheric pressure
(REL_HUMIDITY)	Float			Ambient relative humidity
(TEMPERATURE)	Float	K		Ambient air temperature for an antenna
(DEW_POINT)	Float	K		Dew point
(WIND_DIRECTION)	Float	rad		Average wind direction
(WIND_SPEED)	Float	m/s		Average wind speed

Notes: This table contains mean external atmosphere and weather information.

ANTENNA_ID Antenna identifier, as indexed by ANTENNAn from MAIN.

TIME Mid-point of the time interval over which the data in the row are valid. Required to use the same TIME Measure reference as in MAIN.

INTERVAL Time interval.

H2O Average column density of water.

IONOS_ELECTRON Average column density of electrons.

PRESSURE Ambient atmospheric pressure.

REL_HUMIDITY Ambient relative humidity.

TEMPERATURE Ambient air temperature.

DEW_POINT Dew point temperature.

WIND_DIRECTION Average wind direction.

WIND_SPEED Average wind speed.

5 Units and Measures

5.1 Units and Measures

Name	Format	Units	Measure	Comments
Column keywords				
QuantumUnits	String Vector			Units of the data in the column
MEASINFO	Record			subfields defining the unit
Subfields of MEASINFO				
type	String			lowercase name of the Measure type (e.g., frequency)
Ref	String			Uppercase name of the frame (e.g., LSRK)
VarRefCol	String			Name of the column containing the frames
VarRefTypes	String Vector			All frame names for this type
VarRefCodes	Int Vector			All frame codes for this type

QuantumUnits Its value is a vector of strings, usually containing a single value (e.g., rad) applying to all values in the column. If the column cells contain array values, QuantumUnits can define multiple units. For example, the telescope position could be stored as longitude, latitude, height with units rad,rad,m.

type Its value is one of the Measures types which are epoch, position, direction, frequency, doppler, radialvelocity and earthmagnetic.

Ref This field is used if the entire column has the same frame. Its value is one the the frame names for the given Measures type. For instance: J2000, B1950, ECLIPTIC, etc. for the direction measure.

VarRefCol This field is used if each row in the column can have its own reference frame. Its value is the name of the column containing the frame. The column can contain strings giving the frame name or can contain integers giving the frame codes. If integers are used, the following two fields (VarRefTypes and VarRefCodes) can be defined to make the stored frame codes resilient to possible future changes.

VarRefTypes and VarRefCodes These fields can be used if VarRefCol is defined. They contain a vector of strings and integers giving the the frame names and codes for this Measures type. They map a frame code to its name. It makes it possible that the Measures classes can renumber the frame codes without affecting the stored frame codes (the frame names are not changed).
These fields are optional. If VarRefTypes is not defined, it is assumed that the frame codes in the table are the same as in the Measures classes.

Notes: The possible units can be found in the Casacore Unit classes. They can be listed using the TaQL command `'show units'`.
The possible measure types and frames can be found in the Casacore Measures classes (such as MDirection). They can be listed using the TaQL command `'show meastypes'`.

6 References

Kemball, A.J., Wieringa, M.H., 2000, casacore Note 229.