ASTRI Mini-Array
Data Model

Prepared by:	Name:	*Andrea Bulgarelli*	Signature:		Date:	Jun 18, 2021
Verified by:	Name:	*Gino Tosti*	Signature:		Date:	Jun 18, 2021
Approved by:	Name:	*Gino Tosti*	Signature:		Date:	Jun 18, 2021
Released by:	Name:	*Salvatore Scuderi*	Signature:		Date:	Jun 18, 2021

Main Authors: *Andrea Bulgarelli, Joseph Schwarz, Gino Tosti, Fabrizio Lucarelli, Maria Concetta Maccarone, Alessandro Costa, Vito Conforti, Luca Zampieri, Nicolò Parmiggiani*
Contributor Authors: *Federico Russo, Ciro Bigongiari, Fulvio Gianotti, Pierluca Sangiorgi, Stefano Germani, Saverio Lombardi*

Table of Contents

INDEX OF FIGURES & TABLES

Document History		
Version	***Date***	***Modification***
0.1	Oct 15th, 2019	First version
1.0	Nov 6th, 2019	First release. Version name: V1.0
1.3	Dec 10, 2019	Version aligned with Architecture V 1.3
2.1	Feb 3, 2020	Version aligned with Architecture V 2.1
2.2	Apr 16, 2020	Version aligned with Architecture V 2.2
2.3	Nov 9, 2020	Changes after the Concept Design Review. Version aligned with Architecture V 2.3
2.4	Apr 9, 2021	Changes after internal reviews and discussions
2.5	Jun 18, 2021	Annex B - Avro schema Definition of reading point

1. Introduction

The **ASTRI Mini-Array (MA)** is an INAF ground-based project to construct, deploy and operate a set of nine identical dual-mirror Cherenkov gamma-ray telescopes, and several other auxiliary equipment and infrastructures. The ASTRI Mini-Array scientific objective is to exploit the imaging atmospheric Cherenkov technique to measure the energy, direction and arrival time of gamma-ray photons arriving at the Earth from astrophysical sources. In the almost unexplored energy range 1-300 TeV this technique requires an array of optical telescopes (~ 4 m in diameter) at a site located at an altitude of > 2000m. The telescopes will have reflecting mirrors focusing the Cherenkov UV-optical light produced by atmospheric particle cascades (air-showers), initiated by the primary gamma-ray photons entering in the atmosphere, onto ultrafast (nanosecond timescale) cameras. Most of the collected data will come from the large number of charged primary cosmic-ray initiated air-showers, which will also be recorded, then appropriate data analysis methods will be employed to reduce the level of this background and allow an efficient detection of gamma-rays coming from astrophysical sources.
Besides the gamma-ray scientific program, the ASTRI Mini-Array will also perform:

- Stellar Hambury-Brown intensity interferometry: each of the telescopes of the ASTRI Mini-Array will be equipped with an intensity interferometry module. The Mini-Array layout with its very long baselines (hundreds of meters), will allow, in principle, to obtain angular resolutions down to 50 micro-arcsec. With this level of resolution, it will be possible to reveal details on the surface of bright stars and of their surrounding environment and to open new frontiers in some of the major topics in stellar astrophysics.
- Direct measurements of cosmic rays: 99% of the observable component of the Cherenkov light is hadronic in nature. Even if the main challenge in detecting gamma-rays is to distinguish them from the much higher background of hadronic Cosmic Rays, this background, recorded during normal gamma-ray observations, will be used to perform direct measurements and detailed studies of the Cosmic Rays themselves.

The ASTRI MA telescopes (including the Cherenkov Camera) are an updated version of the ASTRI-Horn Cherenkov Telescope operating at Serra La Nave (Catania, Italy) on Mount Etna.

The nine telescopes will be installed at the Teide Astronomical MA System, operated by the Instituto de Astrofisica de Canarias (IAC), on Mount Teide (~2400 m a.s.l.) in Tenerife (Canary Islands, Spain).

The ASTRI MA System will be operated by INAF on the basis of a host agreement with IAC.

1.1. Purpose

This document provides an overview of the data products of the ASTRI MA System, and their conceptual (and logical) data models.
The goals are:

- list the data products produced by the MA system and the related data product categories;
- show the relationship between the data products;
- define a short identifier for the data product;
- refer to data streams in architectural diagrams without ambiguity;
- define interfaces and data models.

In this document, some colours are used:

- **blue bold**, for definitions;
- **black bold**, for main concepts, or name of systems or functional units;

- **green bold**, for roles covered by human actors.

This document uses the Unified Modeling Language (UML) [RD5] to present conceptual schemas for describing Data Models.

1.2. Scope

This document applies to all application- and domain-level data models and, in some cases, data products, for the software of the **ASTRI Mini-Array System** (a.k.a ASTRI MA System or MA System). This document does not specify the physical view of the data (memory, file data format), or the transport mechanism.

This document provides a conceptual view of the ASTRI MA Data Model.

This document describes data models and their relationship to define interfaces, refer to data streams in architectural diagrams without ambiguity, define a short identifier for the data product.

2. Related Documents

2.1. Applicable documents

[AD1] N. Parmiggiani et al., ASTRI MA Glossary, ASTRI-INAF-LIS-2100-001 , issue 2.5
[AD2] A. Bulgarelli, G. Tosti, et al., ASTRI MA Top Level Use Cases, ASTRI-INAF-SPE-2100-001, issue 2.5
[AD3] A. Bulgarelli et al., ASTRI MA Top Level Software Architecture, ASTRI-INAF-DES-2100-001, issue 2.5
[AD4] ASTRI-MA Software Engineering Management Plan: ASTRI-INAF-PLA-2100-001, issue 1.0
[AD5] ASTRI Mini-Array Data & Documentation Management Plan, ASTRI-INAF-PLA-1000-003, issue 1.2

2.2. Reference documents

[RD1] J. Schwarz, G. Chiozzi, P. Grosbol, H. Sommer, A. Farris, D. Muders, ALMA Project Software Architecture, ALMA-70.15.00.00.001-H-GEN, Version J, 2007-08-13
[RD2] S. Lombardi, L.A.Antonelli, C. Bigongiari, M. Cardillo, F. Lucarelli, M. Perri, A. Stamerra, F. Visconti, “ASTRI data reduction SW in the framework of the Cherenkov Telescope Array”. Proc. of SPIE 2018, paper number 10707-29.
[RD3] E. F. Codd, The Relational Model for Database Management, ADDISON-WESLEY PUBLISHING COMPANY, 1990
[RD4] OPC-UA Information Model https://opcfoundation.org/developer-tools/specifications-opc-ua-information-models/opc-unified-architecture-plcopen-information-model
[RD5] https://www.omg.org/spec/UML/
[RD6] F. Lucarelli, S. Lombardi, “ASTRI SST-2M reconstruction software: SCI-TECH0 parameters”, ASTRI-IR-INAF-3700-081
[RD7] A. Caproni, “Integrated Alarm System Architecture”, ESO-293482

3. Definition and Methodology

A **data product** is **data** produced by devices ([AD3]) or software algorithms. The two main classes of data products are:

1. Raw data: data generated by devices
2. Derived data: a transformation of input data to another data type

Each **data product** is described by a data model. A **data model** describes the purpose, structural elements (the **data type**) and how data products relate to one another and the properties of real-world entities. The description of a data model is provided using a **data modelling language**. Each **data product** is of one **data type**. A **data model category** is a grouping of data models with a similar purpose.

> A **data product** is **data** produced by devices or software algorithms. Each **data product** is described by a **data model**, with a purpose, the structural elements (the data type) and the relationships. Data models are grouped in **data model categories**.

A special data type is an **alarm**, a notification if some conditions that require immediate intervention are verified e.g we could have an alarm conditions if a component or a system is in an abnormal status, i.e. that one or more of its reported and monitored values are out of the normal range.
An alarm means that a component has entered into an abnormal status that requires timely **Operator** or automated software intervention and that the **Operator** has not yet acknowledged this situation.

There are mainly three different levels of data models:

1. **Conceptual**: Conceptual data models, also called domain models, establish the basic concepts and semantics of a given domain and help to communicate these to a wide audience of stakeholders. This data model defines WHAT the system contains. The purpose is to organize, scope and define system concepts and rules.
2. **Logical**: Logical data models add further detail to conceptual model elements and refine the structure of the domain. Defines HOW the data is structured regardless of the implementation (the description of the content of a data product). The purpose is to develop a technical map of rules and data structures. The logical model is a translation of a conceptual model in a data modelling language ready for implementation (e.g. the relational model [RD3], or OPC-UA information model [RD4])
3. **Physical**: This data model describes HOW the system is implemented, i.e. the **data format**, using a specific DBMS system, file format, memory structure, etc..

Data models could be part of **data model categories**, a grouping of data models with a similar purpose.

This document uses the Unified Modeling Language (UML) [RD5] to present conceptual schemas for describing Data Models. These schemas define conceptual classes that (i) may be considered to define a cross-domain application schema, or (ii) may be used in application or functional schemas. The same name may be used in implementations as in the model, so types defined in the UML model may be used directly in application schemas.

3.1. Data Model Domains

The data models of the **ASTRI MA System** can be divided into two **data model domains**: **telescope domain** and **science domain**. Each domain is a data model category.

The **telescope domain** is instrument-centric. The data that flows from this domain (the output of this domain) tends to be ordered in time and related to specific instruments or groups of instruments. In particular, such data are not necessarily associated with Observing Projects (see Sect. 5.). A first example is the monitoring data, which are collected based on a sampling rate and are obtained even when no observation is running. Another example is the pointing data. The inputs to this domain also tend to be instrument-centric. They consist of commands to configure and control instruments.

The **science domain** is scientific observation-centric. This domain is oriented towards carrying out the scientific intent of Observing Projects and is less instrument-oriented. Part of this domain is the scientific data acquired for scientific exploitation purposes.

It is useful to maintain a separation between these two domains because they have radically different concerns and tend to change for independent reasons. The science domain should be separated from changes in the telescope domain. Likewise, the telescope domain should be isolated from changes in the science domain. In particular, the data models in the telescope domain should know nothing about the data models in the science domain; likewise, the science domain should not know anything about what is occurring upstream from it, i.e., in the telescope domain.

Each data product is part of one of these domains. The **Science Data Model** (see Sect. 7.) is the link between these two domains.

The data models of the ASTRI MA System can be divided into telescope domain and science domain. The **telescope domain** is instrument-centric (instruments produce outputs, inputs are commands). The **science domain** is scientific observation-centric. This domain is oriented towards carrying out the scientific intent of Observing Projects. Each **data product** is part of one of these domains. The Science Data Model is the link between these two domains.

Within the telescope domain, the telescope is the whole world. Within the science domain, the telescope is an abstraction. The dataflow is from the telescope domain ("upstream") to the science domain ("downstream"). The software in the telescope domain does not know anything about what is downstream. Anything may be downstream, and downstream software may change or come and go, without the knowledge of anything upstream.

3.2. Data Processing Categories

The same data product can be produced in more than one analysis pipeline. To this respect, each data product is labelled with a predefined **Data Processing Category.** This does not change the *data model* of the data product. The following data categories are defined:

- **A: Prompt Data Products/Processing (Level-A category)**: Prompt data products produced and distributed during the run (see Sect. 13).
- **B: Short-term Data Products/Processing (Level-B category)**: data products (up to preliminary scientific products) produced at the end of a run (after some off-line processing) or at maximum by the next observation day.
- **C: Data Release Products/Processing (Level-C category)**: Data products produced by the full high-quality data processing chain, off-site in MA data centre. These use the best calibrations and algorithms, providing precision and systematics meeting or exceeding MA requirements, and thus are the products intended for final analysis and publication of results.
- **SIM/MC: Simulation Processing**: data products produced by simulations (e.g. Monte Carlo).

3.3. Data Levels

Data Levels represent a naming for the number of data transformations that has been applied to the data.

The two main data level classes of data products are:

- **Raw data**: data produced by devices
- **Derived data**: a transformation of input data to another data type

In order to simplify the development of the MA Software Architecture (see [AD3]), **raw data** can be divided into two data levels:

- **raw data stream**: (**R0**) on-site streamed raw data. R0 content and format is internal to each device / controllable system, such as raw data transmitted from the physical device/system to its respective server in the on-site Data Centre;
- **raw data archived**: (**DL0**) raw archived data from the hardware/software data acquisition. This is the lowest level of data that is intended for long-term storage in the bulk archive and could contain some metadata.

R0 and DL0 could have the same data model but a different data format.

The **derived data** level indicates processed and archived data. Derived data levels depend on the data model. A data model could have one or more derived data levels, e.g. EVT has many data levels (from DL0 to DL5). Other data models could have only one data level.

3.4. Data product identifier

Each data model (DM) describes a data type (the structure) and the relationship of a data product. Some other additional attributes can describe a data model:

- data processing category: when the data is produced in the workflow;
- data levels represent a naming for the number of data transformations that has been applied to the data. The convention is R0 or DLx, x=0, …, N;
- data types (and subtypes): which data is produced.

Some concepts apply only to some data models (e.g. EVT, see Sect. 7.).

The following schema could be used for describing a data product, where each field is optional:

[<data-category>]/[<data-level>].[<data-type>]:[<name>]

3.5. Data Models

The following are the Data Models defined in this document:

1. Observing Project Data Model
2. Observation Execution Data model
3. Cherenkov Camera Data Model
4. Stellar Intensity Interferometry Instrument Data Model
5. Calibration Data Model Category:
    a. Array Calibration Data Model
    b. Cherenkov Camera Calibration Data Model
6. Monitoring Data Model Category:
    a. Monitor Assembly Data Model
    b. Environmental Condition Data Model
    c. Atmosphere Characterisation Data Model
    d. ICT Data model
    e. IPS Data Model
    f. Infrastructure Data Model
    g. Security Data Model
7. Logging Data Model
8. Telescope Data Model
9. Science Data Model
10. Science Simulated Data Model
11. System Configuration Data Model

Each data product is described by one of these data models.

Figure 1 shows the data models and their relationships.

The classification of each data model in a data model domain is described in Sect. 4.

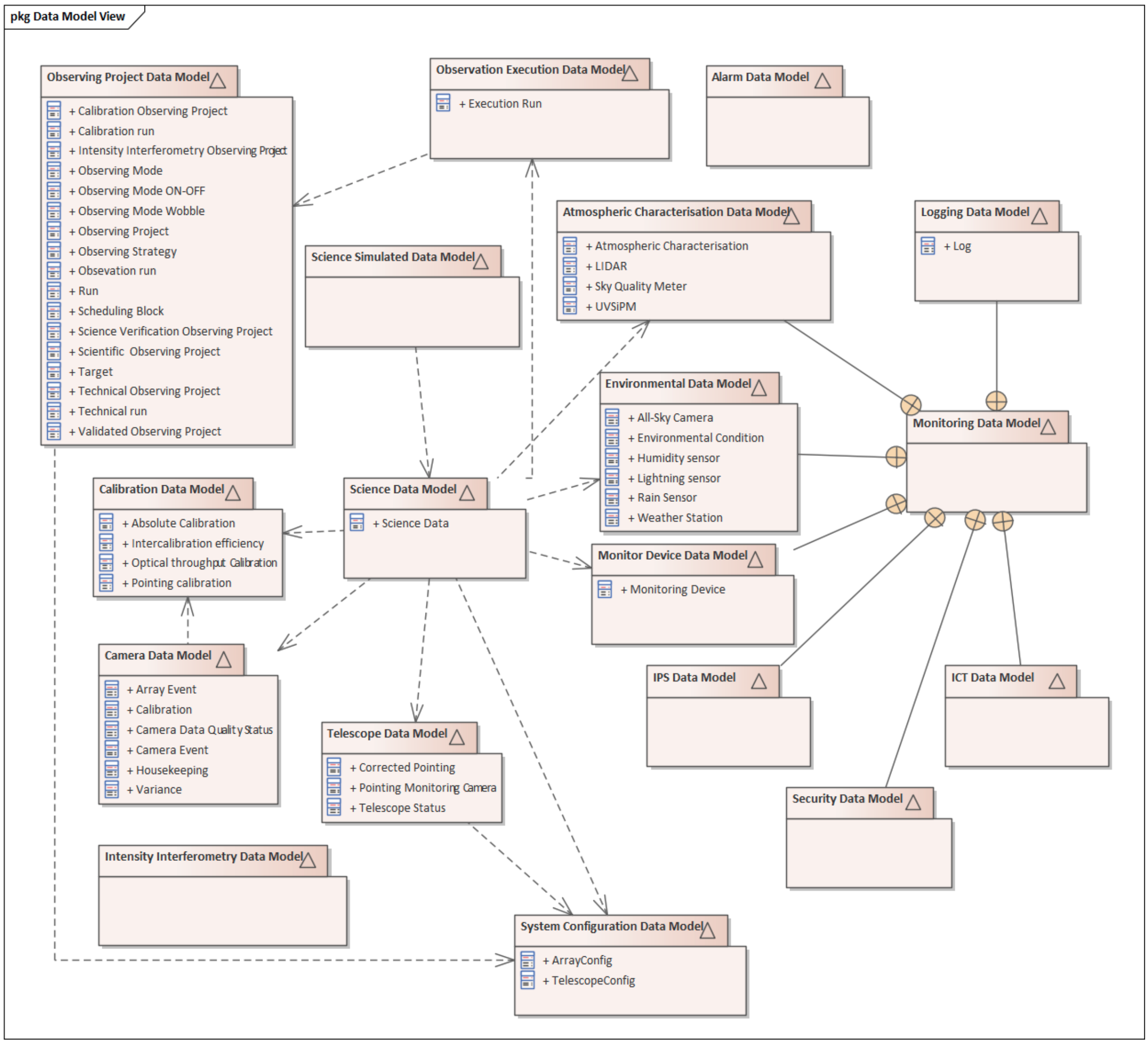

Figure 1: Data Models and their relationships.

4. Science Data Model (SDM)

A science data model is an abstract identification of the types of data, and the internal relationships among such data, that make up a complete astronomical measurement. For these reasons, the **Science Data Model (SDM)** defines the collection of information recorded during an observation that is needed for scientific data analysis**.**

In addition, the Science Data Model (SDM) allows establishing the relationship between acquired data and Observing Projects (see Sect. 4.), to trace all deduced astronomical properties back to raw instrumental data (see [RD1])**.**

The **scientific data** from the MA system have:

1. **observing data**: data produced by a Cherenkov camera or Stellar Intensity Interferometry Instrument; this represents the bulk of the data volume (Cherenkov Camera DM and Stellar Intensity Interferometry Instrument DM);
2. data needed to describe the **observing process** as executed**:** Observing Project (OPDM), Run (Observation Execution DM), pointing information (Telescope DM);
3. **configuration of the system** (System Configuration DM);
4. **monitoring data** acquired during the observation and needed for calibration and data reduction: environmental, atmosphere characterisation, some monitor assemblies (Monitoring DM);
5. logging data (Logging Data Model).

During the observing process, the content of these models is obtained from various hardware/software elements, each sampling at different rates and with limited views of the observing system. Some of this information is static or quasi-static (e.g., configurations) while other information may vary continuously.

The **Data Capture** process, part of the **Central Control** (see [AD3]), takes these pieces and associates all of the information at every time during the observation; the SDM is all of the metadata and auxiliary data along with links.

Science Data Model (SDM) defines the collection of information recorded during an observation that is needed for scientific analysis. It establishes a relationship between the Observing Project and the observation.
Data Capture takes the instrument-centric, time-ordered stream of data, collects and extracts auxiliary information needed in the science domain, and re-organizes them to be useful in data processing in the science domain.
The information collected are: (i) observing data (Cherenkov Camera DM, Stellar Intensity Interferometry Instrument DM), (ii) observing process description (Observing Project DM, Observation Execution DM, Telescope DM), (iii) configuration, (iv) monitoring data (Environmental DM, Atmosphere Characterisation DM), some data products of the Monitoring DM), (iv) some logging data.

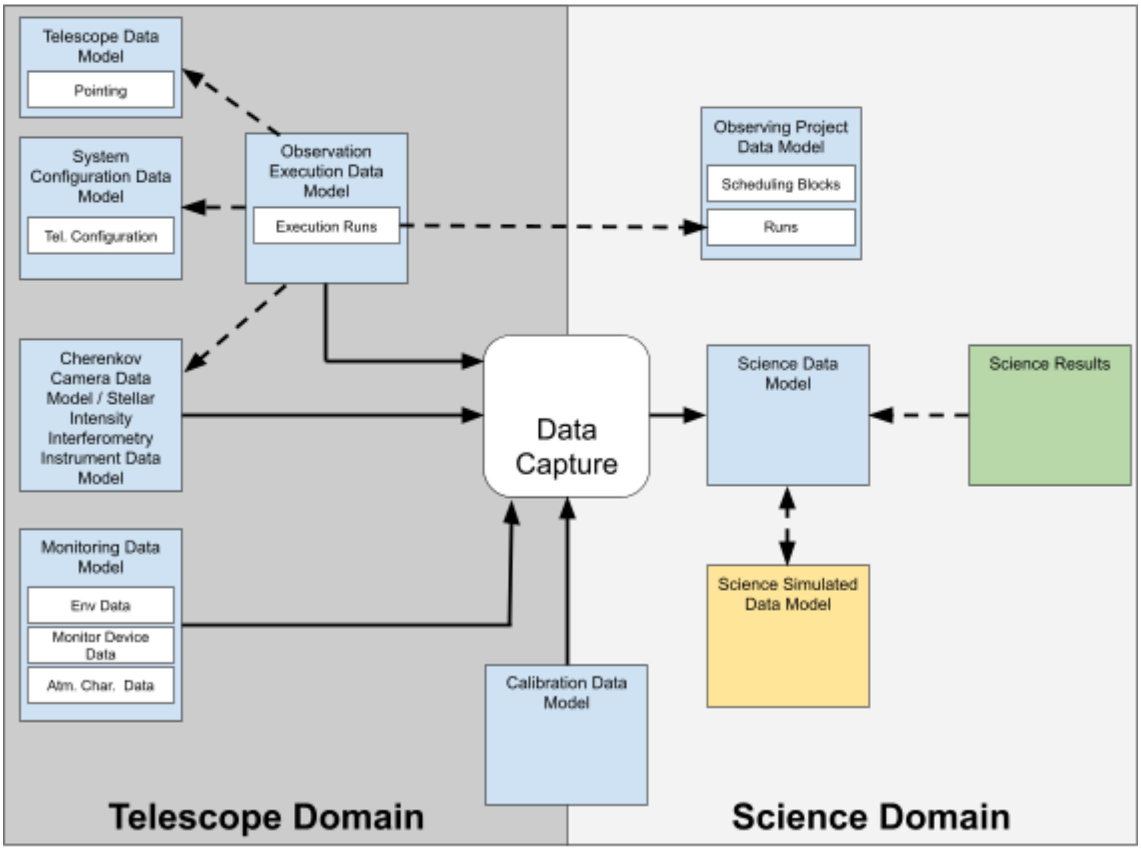

Figure 2: The data models in the telescope and science domains. A solid line indicates data flow, dashed lines indicate referencing. Data flow streams from left ("upstream") to right ("downstream"), although there could be some flow upstream to the Central Control. The double referencing between the SDM and the SSDM means that simulations are linked with the corresponding SDM and vice versa.

Figure 2 shows the data models of telescope and science domains with the connection provided by the Data Capture. Simulations can be treated as real observations if the Science Simulated Data Model has the same structure as the Science Data Model.
Figure 3 shows the Science Data Model.

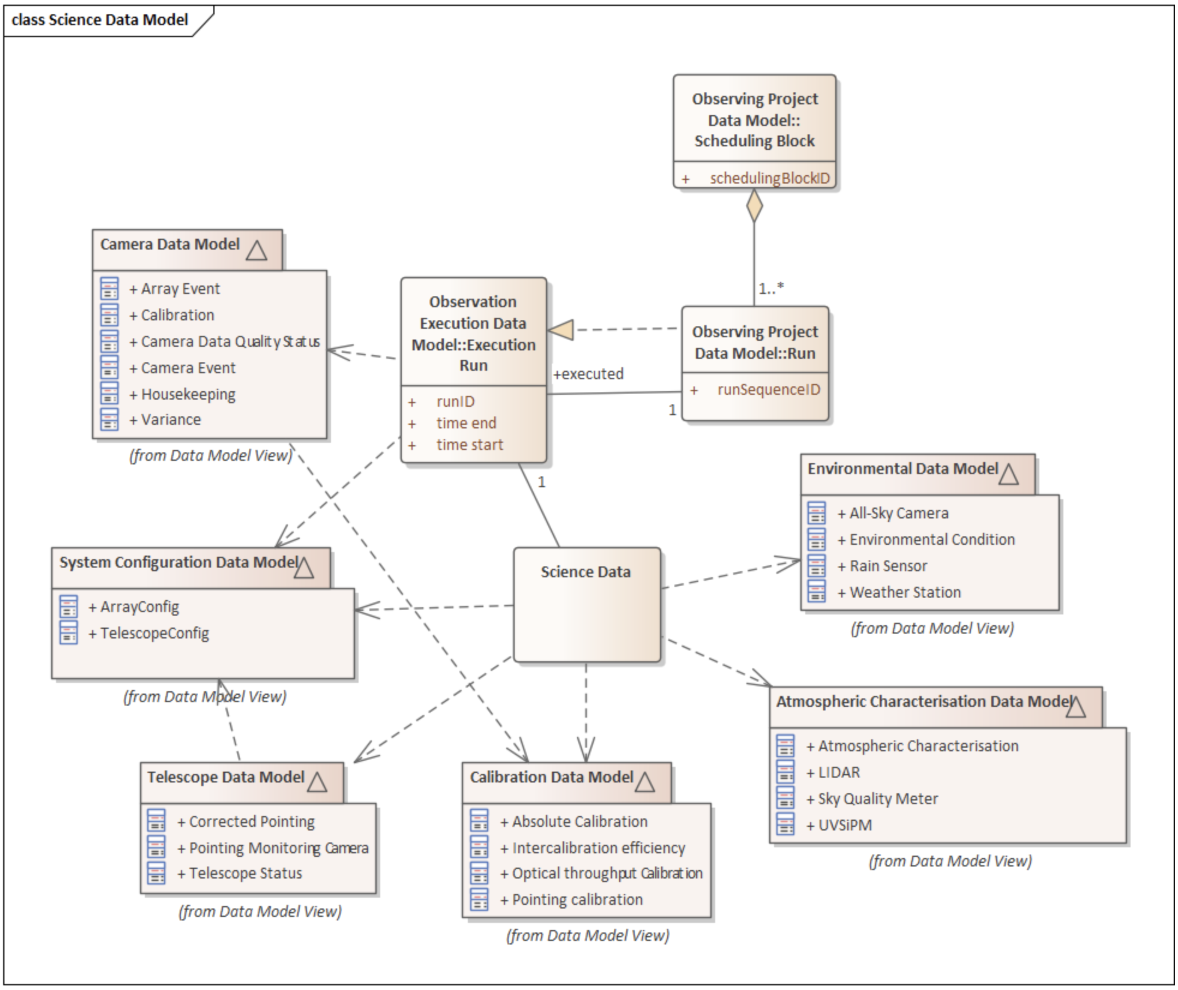

Figure 3: Science Data Model.

5. Observing Project Data Model (OPDM)

The MA software system is envisioned as an end-to-end data flow system, which handles the information and operations required to conduct all tasks from the moment a **Science User** creates an Observing Project until the resulting data are returned.

The Observing Project is the structure that accompanies and directs a scientific or engineering observation from its inception through the Scheduling Block preparation, execution, processing, final archiving and delivery.

5.1. Target

- **Target**: A Target is a location to be observed, either celestial or terrestrial, specified by coordinates in the appropriate reference system (*e.g.*, RA & Dec, galactic, geocentric, ...).

5.2. Observing objects

- **Observing Project (OP)**: a description of a scientific project to observe a target/a set of targets. A **Science User** submits an **Observing Project** to observe a target or a set of targets. An OP contains a description of the Observing Strategies and the associated observing modes, as well as the scientific or technical justification. An OP may span different nights. An OP has a unique identifier.

  Types of OPs are:
  - scientific (for scientific observations). These can be:
    - Cherenkov Scientific Observations;
    - Stellar Intensity Interferometry observations;
  - science verification (used during the SVP);
  - technical (e.g., for maintenance, AIV activities);
  - calibration (e.g. for array calibration activities).

- **Validated Observing Project (VOP):** an OP technically and scientifically evaluated and validated.
- **Scheduling Block (SB)**: The SB is the smallest sequence of observing instructions that can be scheduled. Changes in atmosphere characterisation conditions at the MA site and array operational conditions can change the kinds of observations that can be carried out. OPs are divided into Scheduling Blocks. Each SB has an associated Observing Strategy, with an Observing Mode description and associated constraints: NSB level constraints; environmental conditions and atmosphere characterisation constraints; array status constraints.
  A Scheduling Block is further divided into a set of **Observing Blocks (OBs)**.
  The end of a Scheduling Block may be specified in terms of a maximum amount of time or when certain well-defined science goals have been reached. A SB is atomic, in the sense that it cannot be restarted in the middle; a SB either is complete, fails or is terminated by the

Operator. Execution of several SBs may often be necessary to perform all observations needed to reach the final scientific result.

- **Observing Block**: a unit of a Scheduling Block with a planned start time and stop time (*i.e.*, a continuous observation), characterised by a unique ID, a single target position of a step of the given Observing Mode. Each observing block has an execution time limit (typically 15-20 minutes in Wobble observing mode, higher on ON/OFF observing mode).
  Observing blocks are able to carry out the setups, calibrations, and target observations necessary to ensure that the acquired data are properly calibrated and used in the construction of the final data product.

  Types of observing blocks are:
  1. Calibration OB
  2. Observation OB
  3. Technical OB

  An example of a Scheduling Block composed by a sequence of Observing Blocks can be:
  - calibration
  - observation: first wobble target position (for wobble observing mode)
  - observation: second wobble target position
  - observation: first wobble target position
  - observation: second wobble target position

Figure 4 reports a UML schema of the Observing Project hierarchy.

5.2.1. Observing Strategy

An **Observing Strategy** defines the configuration of the array, the observing mode and the constraints for the observation of a Target.

Parameters used to describe an Observing Strategy:

1. **Observation Purpose:** Calibration, Observation, Technical
2. **Pointing_Coordinate_System**: Equatorial (*RA-DEC) /* Horizontal *(Alt-Az)*
3. **Nominal_Pointing:** pointing coordinates for the Target (*a single coordinate (a.k.a. Sky Position or the centre of a region of interest) or a list of coordinates (e.g., a grid for a survey)*)
4. **Observing mode**: describes how the choice of the pointing positions allows reaching the scientific goals (through *Targets*) relative to scientific aspects and analysis constraints (statistics and systematic errors). Some examples:
   a. WOBBLE science observing mode for Cherenkov observations, the source is not located at the centre of the Cherenkov camera. Typically, the telescope points (at least) at two different positions symmetrically distant from the target gamma-ray source in the sky (the distance depending on the Cherenkov camera FoV and on the expected gamma-ray source extension). The two positions are tracked alternatively for a few tens of minutes, typically. This procedure provides a simultaneous measurement of the signal and the background. Therefore, there is no need a priori for additional observation time to be dedicated to OFF observations.
   b. ON/OFF science observing mode for Cherenkov observations: in ON-OFF science observation mode, the potential gamma-ray source is tracked (ON) in order to have

its nominal position always located in the centre of the camera. In this case, additional dedicated data must be taken to obtain reliable background data. Therefore, a nearby sky region in RA-Dec (OFF), at the same zenith angle of the ON field, without any known gamma-ray source, is selected and tracked.

c. Staring: Stellar Intensity Interferometry Instrument (SI3) observing mode, for intensity interferometry observations.

d. Fixed (Cosmic Rays).

5. **Zenith_Range:** minimum and maximum allowable zenith angle during the SB; the SB will not be executed unless this condition is met. It could be possible to have a limited choice - *e.g.*, bounds could be 30, 45, 60 degrees - optional parameter.
6. **Precision_Pointing:** pointing precision (low wind, low-temperature gradient).
7. **Requested_Time:** the amount of observation time on the Target.
8. **Allowed_Time_Range(s):** Observation time and epoch and time sequences and/or constraints. Table of observing windows when the observations are allowed (by default the full year, but for multi-wavelength campaigns or periodic sources may be more specific). Many concepts are merged here and the development of more detailed science use cases will help to clarify better the scenarios, pre and post-conditions, and how to manage a possible sequence of targets - optional parameter
9. **Tracking mode**: how to follow the target during the observation (e.g. drift)
10. **Minimum number of telescopes**: e.g. if there are some telescopes off-line.
11. **Atmosphere characterisation constraints** (see Sect. 16.2):
    a. **Minimal_Sky_Quality:** Specifying the category of atmospheric quality (transmission); the Scheduling Block will not be executed unless these conditions are met.
    b. **NSB_Range:** Minimum and maximum allowed NSB, mainly due to the moon; the SB will not be executed unless these conditions are met. The idea is to have a list of values.
12. **Instrument Type**: Cherenkov Camera or Optical Camera or SI3
13. **Instrument Configuration:** For each instrument this is the set of parameters needed to optimize the instrument performances for the observation purpose. These parameters will be defined initially on the basis of the simulations and calibration activities taking into account the different environmental, atmospheric, and sky brightness conditions. This configuration represents the dynamic part of the configuration of an instrument. See Sect. 13.1. for more details.

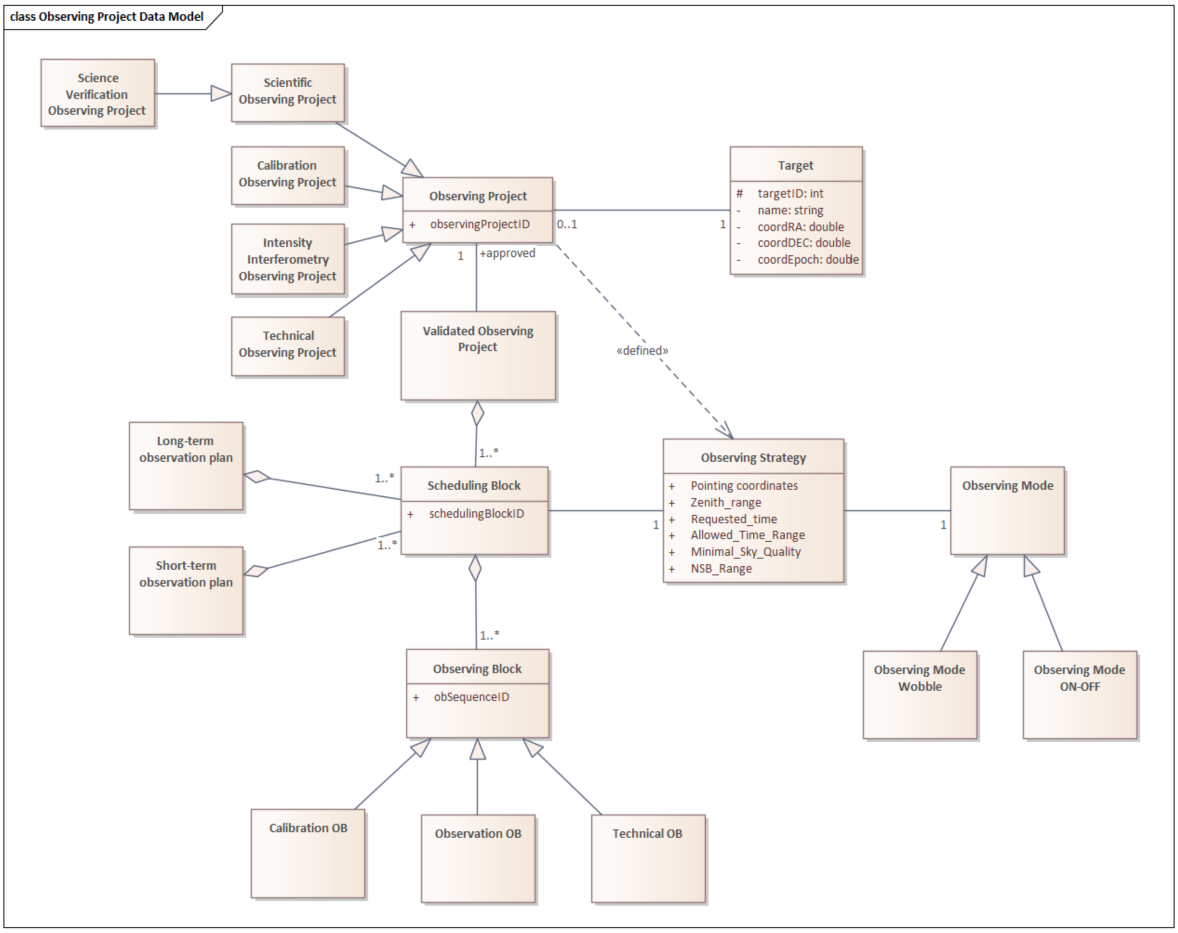

Figure 4: Observing Project Data Model hierarchy.

5.3. Observation plans

- **Long-term observation plan**: the yearly plan of observations of the MA, that consists of a list of Scheduling Blocks (SB) to schedule the observations during each night.
- **Short-term observation plan**: a list of Scheduling Blocks selected from the long-term observation plan to be scheduled during the night.

6. Observation Execution Data Model

The **Observation Execution DM** describes the observing process as executed by the **Central Control** (see [AD3]). The **Run** data type stores the **runID**, the start-time and the end-time. This DM maintains a distinction between the planned observing time, as given in the schedule, and the actual observing time, as measured during the actual observation. The planned times are given by the OPDM; the observed times are stored in the Run tables.

The unique **run identifier** (**runID**) is used to identify unequivocally the acquired data and the operations carried out.

The Run has direct links with the systems that execute an observing block, i.e. the Cherenkov Camera and Telescope DM and their System Configurations.

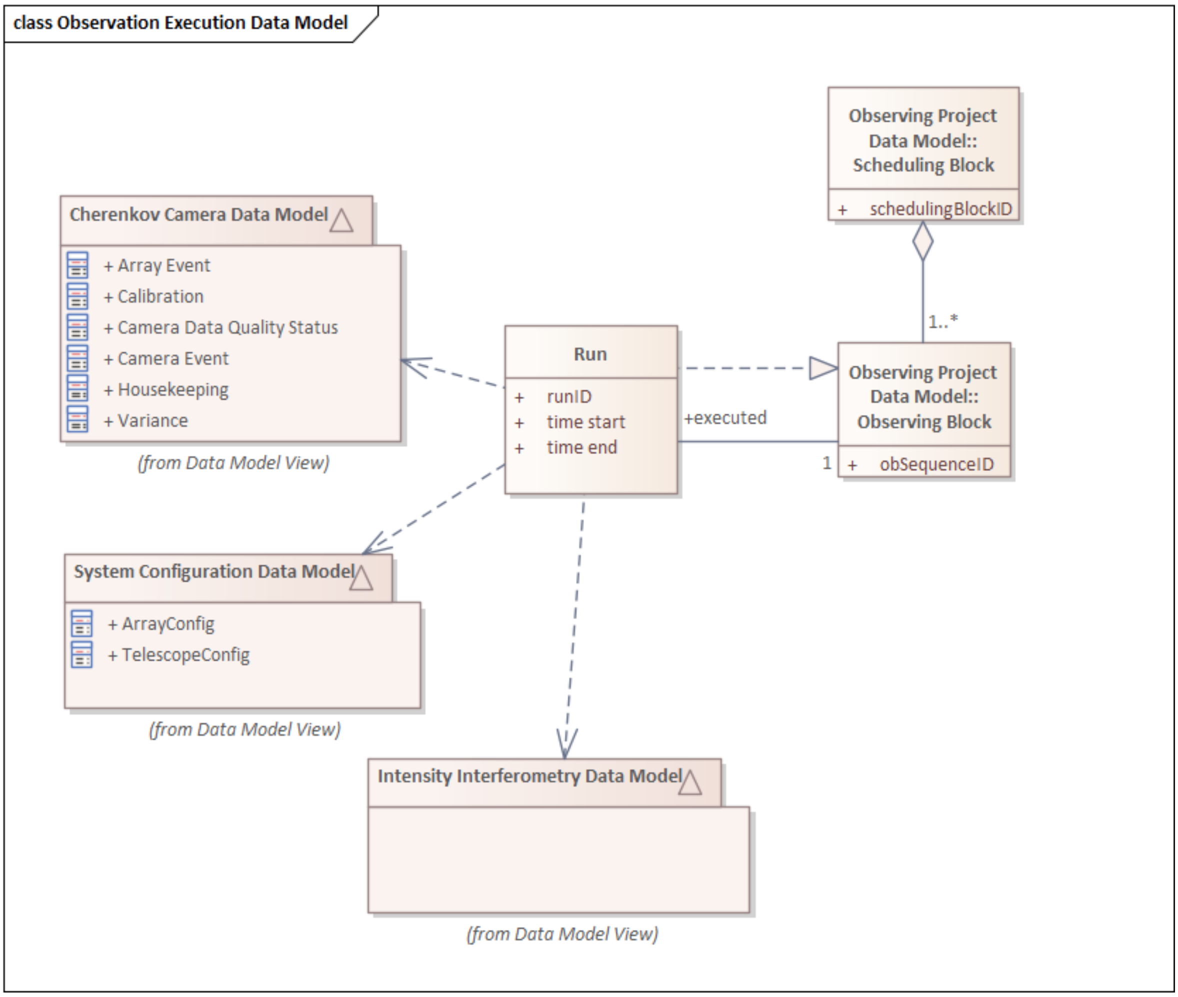

Figure 5: Observation Execution Data Model

7. Cherenkov Camera Data Model (CAMDM)

The Cherenkov Camera Data Model (CAMDM) is the data model of the data produced by the Cherenkov Cameras. Figure 6 shows the Cherenkov Camera Data Model.

7.1. Data association

The **Data Association** allows determining to which part of the MA system a data product is associated:

1. TEL: data at telescope level;
2. ARRAY: data at the array level, after the array trigger;
3. TARG: data per science target.

7.2. Data types

The data products CAM are part of the Cherenkov Camera Data Model (CAMDM) and can be divided into:

- **EVT (camera event)**: produced by the Cherenkov Camera. Contains data that changes for every triggered event (e.g. an Air-Shower Event, Array-Level Event, Local Event, Cherenkov Camera Event), with typically a high rate, which may be more than one kilohertz at the R0 data level. Examples include shower images/cubes, shower parameters, and trigger information. This is typically the highest volume and most complex data stream. Subtype examples:
  - PIXEL data values that can be represented as Cherenkov camera images for each event, e.g. one value per pixel, with optional time-sampled information as an additional axis.
  - PARAM a set of per-image or per-event parameters.
- **VAR (variance)**: the “variance” maps of the focal plane are produced by the Cherenkov Camera periodically, used also to monitor the temporal variations of the NSB flux.
- **CAL (calibration)**: produced by the Cherenkov Camera. Contains data produced for calibration activities.
- **HK (housekeeping)**: Cherenkov cameras housekeeping data, for the monitoring of the camera status.

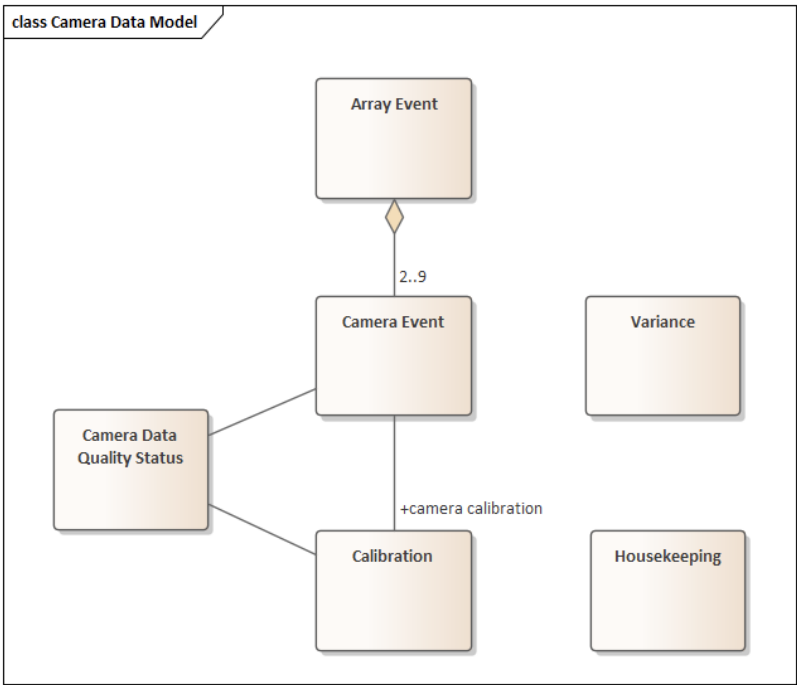

Figure 6: Cherenkov Camera Data Model

7.3. Data Levels

7.3.1. Raw Data Levels

These data levels are applicable to all data types:

- **R0.CAM**: (raw data stream) on-site streamed raw data. R0 content and format is internal to the Cherenkov camera and is the raw data transmitted from the camera to the camera data acquisition. R0 includes also timestamps;
- **DL0.CAM**: (raw data archived) raw data from the hardware/software data acquisition. This is the lowest level of data that is intended for long-term storage in the bulk archive. This includes all CAM data products;
- **DL1.CAM**: data generated to check the data quality;
- **EVT0**: (raw data archived) the DL0.EVT data type;
- **EVT0.TRIG**: EVT0 after software stereo array trigger.

DL0 is saved on disk with two data formats: DL0-RAW, that is the binary (as generated by the Cherenkov Camera back-end), and DL0-FITS, that is the FITS format obtained by the pre-processing of the DL0-RAW.

7.3.2. EVT Data Levels

EVT Calibrated and Reconstructed: Calibrated raw Cherenkov camera images (where the camera readout signals in ADC have been converted into photo-electrons [pe]), and Reconstructed Cherenkov images (where the calibrated images have been cleaned and the Hillas parametrization has been applied).

See [RD2] for more details.

- **EVT1**: (processed) telescope-wise calibrated and reconstructed EVT0 data:
  - **EVT1a**: telescope-wise calibrated data (calibrated image charge)
  - **EVT1b**: telescope-wise cleaned and parameterized data (Hillas parameters, and a usable telescope pattern) (*telescope-wise image parameters*)
  - **EVT1c**: telescope-wise fully reconstructed data (*telescope-wise energy, arrival direction, particle identity discrimination, parameters per telescope*)
- **EVT2**: (reconstructed) array-wise reconstructed air-shower parameters such as energy, direction, particle ID, and related signal discrimination parameters.
  - **EVT2a**: array-wise merged data (*array-wise event parameter*)
  - **EVT2b**: array-wise fully reconstructed data (*array-wise energy, arrival direction, particle identity discrimination per event*)

EVT Science Data: science-ready data product containing the final gamma-like event-list (EVT3) to be delivered to the Science Users for scientific analysis

- **EVT3**: (**gamma-like event-list**). Sets of selected air-shower events obtained with a single final set of reconstruction and discrimination parameters[1].

7.3.3. CAL Data Levels

CAL (calibration data): used for cameras, optics, and array calibrations

- **CAL0**: telescope-wise raw calibration data, triggered by calibration hardware.
- **CAL1**: telescope-wise calibration coefficients/models to be applied to EVT0 data. Produced by the first stage of the data processing pipeline.

7.3.4. MC (Monte Carlo) Simulation Data Levels

We can define the following naming convention:

- SIM/DLx.EVT = MCx where x=0, 1, 2, 3. We have the following data products:
  - **MC0**
  - **MC1: MC1a, MC1b, MC1c**
  - **MC2: MC2a, MC2b**

  analogous to EVT*n,* but including extra MC information.

[1] More than one set of parameters might be foreseen in case different classes of final reduced EVT3 events shall be provided.

- SIM/DLx.CAL = MC-CALx, where x=0, 1, similar to CAL*n*, but from MC simulation.

8. Stellar Intensity Interferometry Instrument Data Model (SI3DM)

This data model is related to data acquired from the SI^3 (Table 1). It consists of a telescope identifier and the time tag of an event in a strongly compressed data format (5 bytes per hit). The reconstructed uncompressed time tag is 17 bytes per hit and contains information on the detector identifier and some flags. This data model has the following scientific data product:

- **SI^3**: data produced by the SI^3

Table 1: SI^3 Data Model

Fields	Example	Frequency Sampling [Hz]	Data-type / Unit	Provided by	Purpose	Description
Telescope ID	1, 2, ...		Int	SI^3	Identify the telescopes for correlating the signals	Telescope identification number
Data output	40 bit integer		Int64	SI^3	Determine the timestamp of the hit/even, the detector ID and some flags	Time tag of the hit/event

8.1. Data Levels

The following data levels are foreseen:

- **DL0.SI3**: raw data acquired and saved on disk;
- **DL1.SI3**: reconstructed event list;
- **DL2.SI3**: event lists calibrated and referred to UTC;
- **DL2a.SI3**: cleaned event lists;
- **DL2b.SI3**: segmented event lists;
- **DL3.SI3:** time coincidences.

9. Calibration Data Model (CALDM)

The **Calibration Data Model (CALDM)** describes Cherenkov calibrations reduced either in real-time or in post-processing by data processing operations. The goal is to determine calibration parameters, with a secondary aim of providing useful feedback on the state and stability of the instrument to the MA staff. Figure 7 shows the Array Calibration Data Model.

9.1. Array Calibration Data Model

The array calibration could require different steps and procedures in different phases; in any case, it is performed analysing the data acquired by the ASTRI MA Cherenkov camera(s) (mainly scientific EVT and variance VAR data) observing a given source, natural or artificial.

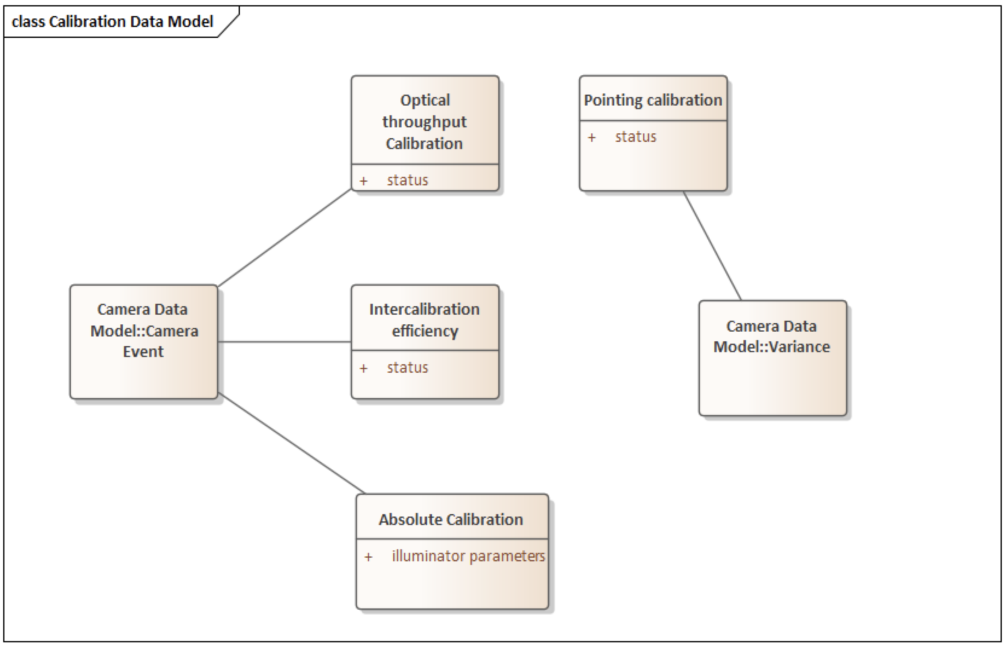

Figure 7: Array Calibration Data Model

9.2. Cherenkov Camera Calibration Data Model

The Cherenkov Camera Calibration Data Model refers to the Cherenkov Camera Data Model, data type CAL.

10. Monitoring Data Model

The Monitoring Data Model is a data model category that includes:

1. Monitor Assembly Data Model (MONDM);
2. Environmental Data Model (ENVDM);
3. Atmosphere Characterisation Data Model (ATMDM);

ICT Data Model (ICTDM), IPS Data Model (IPSDM), Infrastructure Data Model and Security Data Model (SECDM) are also part of the Monitoring Data Model and they will be described in dedicated documents.

10.1. Environmental Data Model (ENVDM)

The **Environmental Data Model (ENVDM)** describes weather and environment (ENV) monitoring data from the **Environmental Monitoring System** (see [AD3]), that includes weather stations, rain sensors, humidity sensors, and all-sky cameras. Weather data includes pressure, humidity, and other weather information, timestamp, and also the station-ID of the originating data.

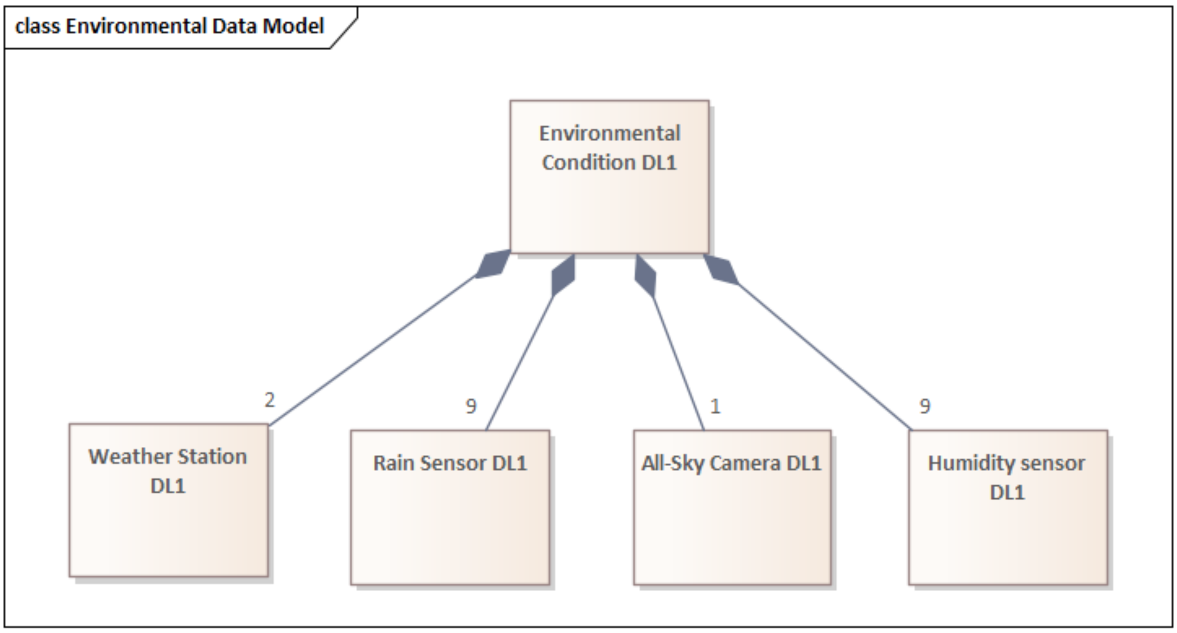

Figure 8: Environmental Data Model

Two data levels are foreseen:

- **DL0.ENV**: the raw data produced by the devices;
- **DL1.ENV**: the high-level data produced after a processing of the DL0 data.

Table 2, Table 3, Table 4 and Table 5 describe some of the main DL1.ENV parameters provided by weather stations, rain sensors, the all-sky camera (ASC), and humidity sensors devices.

Table 2: DL1 data level of the WEATHER STATION

Fields	Example	Data-type / Unit	Provided by	Purpose	Description
Station ID	1, 2	Int	Weather Station	Observation status Data Quality Check Data Selection	Weather station identification number
Timestamp			Weather Station		Date and time of acquisition
External Temperature	17.1	Float / [°C]	Weather Station	Observation status Data Quality Check Data Selection	
Wind Speed	12.8	Float / [km/h]	Weather Station	Observation status Emergency state Data Quality Check Data Selection	
External Relative Humidity	33	Float / [%]	Weather Station	Observation status Data Quality Check Data Selection	
Atmospheric Pressure	748	Float / [hPa]	Weather Station	Observation status Data Quality Check Data Selection	
Dew Point	4.8	Float / [°C]	Weather Station	Observation status Data Quality Check Data Selection	
Rain		Float	Weather Station	Observation status	
WS Status		Int	Weather Station	Observation status Data Quality Check Data Selection	

Table 3: DL1 data level of the RAIN SENSOR

Fields	Example	Data-type / Unit	Provided by	Purpose	Description
Rain sensor ID		Int	Rain sensor	Observation status	Rain sensor identification number
Timestamp			Rain sensor		Date and time of acquisition
Rain status		Float	Rain sensor	Observation status	
Rain sensor Status		Int	Rain sensor	Observation status	

Table 4: DL1 data level of the ALL SKY CAMERA (ASC)

Fields	Example	Data-type / Unit	Provided by	Purpose	Description
ASC ID		Int	All Sky Camera	Data Quality Check / Data Selection Observation status	All-sky camera identification number
Timestamp			All Sky Camera		Date and time of acquisition
Window Sky % Coverage Cloudy	47	Float / [%]	All Sky Camera	Data Quality Check / Data Selection	
All Sky % Coverage Cloudy (cloudiness index)	12	Float / [%]	All Sky Camera	Data Quality Check / Data Selection Observation status	
ASC Status		Int	All Sky Camera	Data Quality Check / Data Selection Observation status	

Table 5: DL1 data level of the HUMIDITY SENSOR

Fields	Example	Data-type / Unit	Provided by	Purpose	Description
Sensor ID		Int	Humidity sensor	Observation status	Humidity sensor identification number
Timestamp			Humidity sensor		Date and time of acquisition
Humidity			Humidity sensor	Observation status	Data
Humidity status		Float	Humidity sensor	Observation status	

10.2. Atmosphere Characterisation Data Model (ATMDM)

The **Atmosphere Characterisation Data Model (ATMDM)** is based on the acquisition and analysis of data from the Atmosphere Characterisation System (see [AD3]) acquired during the night and contemporary to the ASTRI MA observation runs for atmosphere characterisation (ATM). Main data sources are:

- **LIDAR** data: atmospheric extinction (at the beginning of the observation night or start of operations) and atmospheric transmission. The presence of dust is also checked.
- **SQM** data: sky brightness (continuously); a value in mag/arcsec^2 with a predefined frequency (e.g. once 2 sec)
- **UVSiPM** data: counts/second/pixel and counts/second integrated over all pixels (continuously); each UVSiPM acquisition (e.g. once per second) provides an array of 64

elements (8x8 pixel of the sensor) referring the counts/pixel content; a local control software (inner to UVSiPM on-board computer) also provides, over all 64 pixels, the integrated counts/second that is function of the night sky background. The evaluation of the diffuse NSB in physical units is externally performed by another software.

The ATM data will be used in the data quality check procedure, to understand if an observation must be carried out, and as support during the analysis of scientific observations to classify and select good quality data with respect to non-optimal atmosphere characterisation conditions.

Although labelled as ENV Monitoring System, the cloudiness index around the telescopes pointing provided by the analysis of All-Sky Camera images (window sky coverage, see Table 4) has to be considered part of the ATM Data Model for data quality check and data selection purposes.

Two data levels are foreseen:
- **DL0.ATM**: the raw data produced by the devices;
- **DL1.ATM**: the high-level data produced after a processing of the DL0 data.

Table 6, Table 7, Table 8 and Table 9 describe, respectively, the DL1 data provided by the SQM, the UVSiPM, and the LIDAR ATM assemblies.

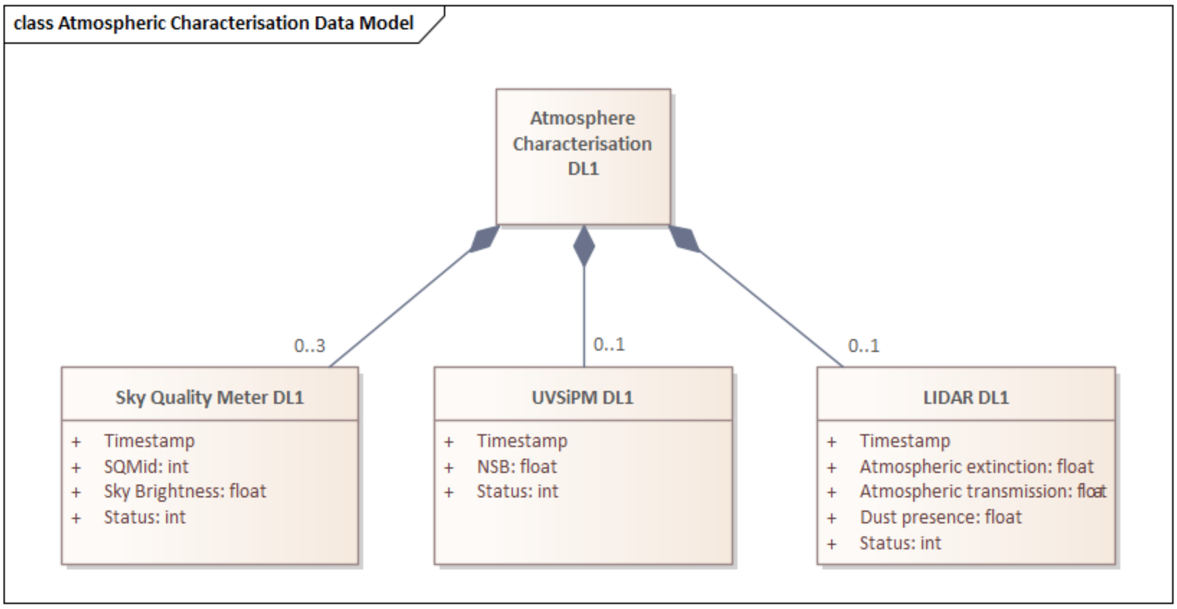

Figure 9: Atmosphere Characterisation DL1 Data Model

Table 6: DL1 data level of the SKY QUALITY METER (SQM)
Frequency: 1 Hz

Fields	Example	Data-type / Unit	Provided by	Purpose	Description
SQM ID	1...3	Int	Sky Quality Meter	Data Quality Check / Data Selection	SQM identification number (one for each telescope)
Timestamp		datetime	Sky Quality Meter		Date and time of acquisition
Sky Brightness	21.25	Float / [Mag/Sq Arcsec.]	Sky Quality Meter	Data Quality Check / Data Selection	Directly measured by the SQM, it is the value of visible sky brightness integrated in the SQM FoV
SQM Status		Int	Sky Quality Meter	Data Quality Check / Data Selection	Report the status of the device. This will depends on the type of SQM installed and could be an integer.

Table 7: DL1 data level of the UVSiPM
Frequency: 1 Hz

Fields	Example	Data-type / Unit	Provided by	Purpose	Description
Frequency: 1 Hz					
Timestamp		datetime	UVSiPM		Date and time of acquisition
NSB level		Float counts/sec	UVSiPM (on-line)	Data Quality Check / Data Selection	Counts/sec integrated in all 64 pixels forming UVSiPM
UVSiPM Status		Int	UVSiPM	Data Quality Check / Data Selection	Report the status of the device (on/off/nominal/degraded /safe/fault)

Table 8: DL1 data level of the LIDAR

Fields	Example	Data-type / Unit	Provided by	Purpose	Description
Timestamp		datetime	LIDAR		Date and time of acquisition
Atmospheric extinction		Float	LIDAR	Data Quality Check / Data Selection	
Atmospheric transmission		Float	LIDAR	Data Quality Check / Data Selection	

Dust presence		Float	LIDAR	Data Quality Check / Data Selection	
LIDAR Status		Int	LIDAR	Data Quality Check / Data Selection	Report the status of the device (on/off/nominal/degraded/safe/fault)

Table 9: DL0 data level of the UVSiPM
Frequency: off-line

Fields	**Example**	**Data-type / Unit**	**Provided by**	**Purpose**	**Description**
Timestamp		datetime	UVSiPM		Date and time of acquisition
data stream		counts/pixel	UVSiPM (on-line)	Data Quality Check / Data Selection	Array of 64 elements at the given acquisition frequency containing counts/pixel
Diffuse NSB flux		physical units [ph/(m2 ns sr) or ph/m2 ns deg2]	off-line UVSiPM + Software	Data Quality Check / Data Selection	Flux of the diffuse NSB. Data can be download after the observations

10.3. Monitor Assembly Data Model (MONDM)

The **Monitor Assembly Data Model (MONDM)** describes time-series data that are used to monitor the status or quality of hardware devices, software components, or other data products (the **monitoring points**). These typically update periodically during the operation of the array, or during data processing, at a rate typically much slower than scientific data and faster than the rate at which successive Runs are started. Data are acquired from Assemblies or Devices (see [AD3]).

Monitoring points are represented by a MON port that shall be exposed by each device or assembly of the **MA system** (see [AD3]). Figure 10 shows the logical model where are represented the Assembly nodes of the MA system, with the MON port. A monitoring point could also be an **alarm**.

Table 10 and Table 11 show the list of parameters of the monitoring points (MONDM).

Monitoring points configuration are stored in the **System Configuration DB (SCDB)**.

Monitoring points subtype (based on generator) could be:

- MEASUREMENTS: devices information like tracking positions;
- STATUS: the status of a particular hardware or software component;
- DQ: the quality-control data.

Monitoring points subtype (based on contained information) could be:

- INSTANT: tables of instantaneously measured monitoring quantities (time vs quantity);
- STATS: averaged statistics for monitoring quantities measured over a time-range;

- HISTO: distribution of monitoring quantities from which it is possible to extract statistics.

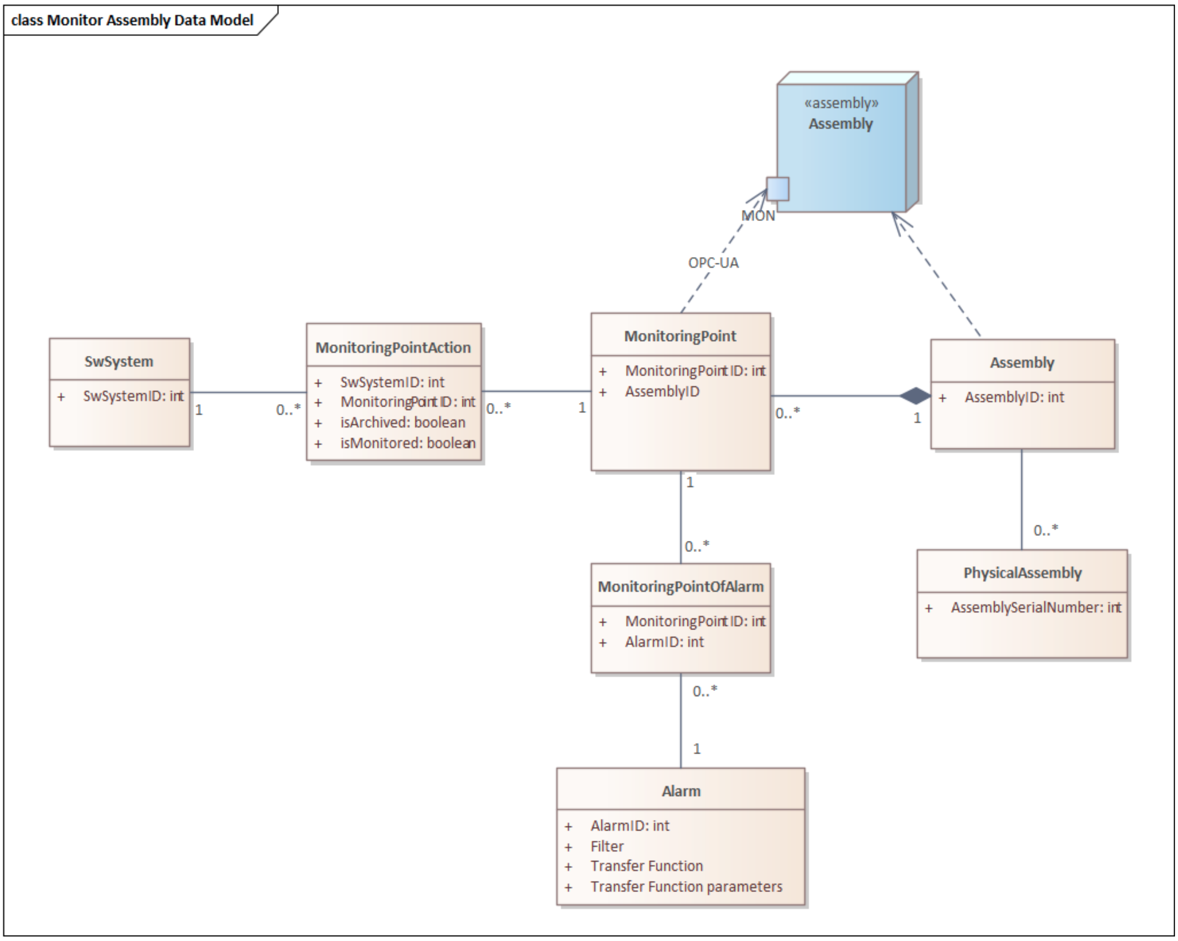

Figure 10: Monitoring Data Model (MONDM)

Table 10: Monitoring point

Fields	Example	Data-type / Unit	Provided by	Description	IDL GET
MonitoringPointID		int	interface definition	A unique ID that identify the monitoring point in the ASTRI MA system	
Short Name					X
AssemblyID		int	interface definition	Device Name	Actionee ?

AssemblySerialNumber		string	interface definition	Serial Number of the assembly. This field is empty if no serial number is specified or if the MonitoringPoint model is the same for all physical assemblies.	
component	opc.tcp://192.168.100.128:52520/OPCUA/ASTRI_WS2_Server	String	interface definition	(OPC-UA) Name of the OPC-UA Server or (ACS) Name of the Software Component where this property is exposed, should correspond to the monitoring data source it is serving	
Node	ns=2;s=ws_exttmp	string	interface definition	(OPC-UA node) **fully qualified NodeId** of the node or (ACS) The display name of the property, e.g. 'Current'	X
ServerTimeStamp	2019-11-26T12:42:12.0	ACS (OMG Time).	software acquisition	Source time (when the property is received by the server) OMG Time, or read by the ACS client 100-ns since Oct 15, 1582. See note (*)	
SourceTimeStamp	2019-11-26T12:42:12.0	ACS (OMG Time).	monitoring device	Source Time (when the property is produced by the device) Stamp OMG Time, 100-ns since Oct 15, 1582. See note (*) below.	
StatusCode	good,bad non initialized	enum symbols: [“good”,”bad”, ”not initialized”]	monitoring device		
Data Type		symbols: ["STRINGSEQ", "DOUBLESEQ", "LONGSEQ", "ULONGSEQ","BOOLEANSEQ", "PATTERN", "ENUM", "INTSEQ", "UINTSEQ", "FLOAT", "FLOATSEQ"]	monitoring device	Type of the Value	X
data		array<double | int | long | string | float | boolean>	monitoring device	Contains time-series data that are used to monitor the status or quality of hardware, algorithms, or other data products. These typically update periodically during the operation of the array, or during data processing, at a rate typically much slower than event data and faster than the length of a typical Observation Block. Examples would be slow-control information like	

				tracking positions, weather monitoring data, or the status or quality-control data of a particular hardware or software component	
Sampling interval	5	sec	monitoring device	Sampling Interval in seconds	X
Default value			monitoring device		X
Units	Deg	string	monitoring device	A string representing the units (normally base SI units or combinations of SI units) of the quantity represented by the property. E.g. 'A'	X
Alarm low			interface definition	parameters needed to apply the Transfer Function. Below this value alarm is set. (Optional)	
Alarm high			interface definition	parameters needed to apply the Transfer Function. Above this value alarm is set. (Optional)	
Withdraw alarm low			interface definition	parameters needed to apply the Transfer Function. Above this value alarm is cleared. (Optional)	
Withdraw alarm high			interface definition	parameters needed to apply the Transfer Function. Below this value alarm is cleared. (Optional)	
Operation modes					X
Expected execution time (s)					X
Maximum execution time (s)					X
Description	The Monitor shows the current position of the AZ axis in Sky Coordinate System (zero point at North). The offset is included. Tpoint and Refraction contributions are	string	interface definition	Description of the monitoring point.	X

	not taken into account.				

TBD: ADS variable, CMD/MODE value

Table 11: Software system that use the Monitoring point

Fields	Example	Data-type / Unit	Provided by	Description
SwSystem	All, TCS, Monitoring System, Alarm System			name of the sw subsystem that acquire/use the monitoring point
MonitoringPointID		int	See Table "Monitoring Point"	A unique ID that identify the monitoring point in the ASTRI MA system
IsArchived	Y/N	boolean	interface definition	Define if monitoring point is archived by a particular SwSystem
IsMonitored	Y/N	boolean	interface definition	Define if monitoring point is monitored

*ACS is using OMG timestamps out-of-the-box, but ASTRI-ma decided to use TAI seconds since 1970-01-01T00:00:00.0 for timestamps. For the time being we use ACS OMG time out of the box, and later expect a way to use TAI in the monitoring stream.

An **alarm** is an audible and/or visible notification indicating to the **Operator** an equipment malfunction, process deviation, or abnormal condition requiring a timely response (ISA/IEC definition). The Alarm Signal is appropriate to the urgency required based on the criticality of the condition.

The system for alarm generation shall be based on the Integrated Alarm System (IAS) [RD7] and shall receive monitoring points or alarms from hardware devices through OPCU-UA subscription and from software processes like ACS components. IAS works on the basis of Input Output values (IASOs) which, in addition to the input values from the monitored systems, could be synthetic parameters or alarms generated from the input. The outputs are generated by Computing Elements (ASCEs) on the basis of a Transfer Function (TF). An IASIO could act both as input and output for different ACSEs. Input and outputs for the ASCEs are managed by Distributed Alarm System Units (DASUs) which also expose the generated Alarms. The IASIOs, ASCEs and DASUs, are defined in the IAS Configuration Database (IAS-CDB). The data coming from devices and external software processes is propagated to the IAS through dedicated Plugins (one per system) which need their own configuration information.

The alarm system data model describes the information which is needed to raise alarms on the monitored systems and to understand their meaning. In the cases where the monitored point is an alarm, the Alarm System shall define an Alarm object which shall be either CLEARED (not set) or set with one out of four possible levels listed in the table. IASIOs shall have an associated Validity (RELIABLE / UNRELIABLE) based on the production time stamp compared to the present time.
IASIOs, Computational Elements and DASUs all have an identifier that allows to uniquely identify them at run time. In order to identify a specific instance of an IASIO the Time Stamp is needed.

The high level information which shall be provided by the subsystems is listed in Table 12.

Table 12: Alarms High Level Information

Field	Example / Available Options	Data Type / Unit	Description
AlarmID		int	Unique ID of the alarm
MonitoringPointId list		int	List of monitoring point id used in the transfer function to generate the alarm
Short Name		string	Name of the Alarm
Refresh Time	1000	ms	Optional. Minimum refresh time needed
Filter	Time Average	string	Optional. Possible data values manipulations to apply before checking for Alarms
Value Type	ALARM , LONG, INT, SHORT, BYTE, DOUBLE, FLOAT, BOOLEAN, CHAR, STRING, ARRAYOFDOUBLES, ARRAYOFLONGS, TIMESTAMP	string	Type of Data
Transfer Function	Min Max Threshold	string	Algorithm or Conditions to raise an alarm. Not valid for ALARM Data Type
Transfer Function parameters	alarmHighOn=40, alarmHighOff=38, alarmLowOn=-10, alarmLowOff=-8	null | double | int | long | float | bool	parameters needed to apply the Transfer Function. alarmHighOn = Above this value alarm is set. (Optional) alarmHighOff = Below this value alarm is cleared. (Optional) alarmLowOn = Below this value alarm is set. (Optional) alarmLowOff = Above this value alarm is cleared. (Optional)
contacts	Carlo Rossi carlo.rossi@inaf.it +393485556677	string	contact information (name, email, mobile phone)
emails	[carlo.rossi@inaf.it]	Array of strings	Optional. Address to automatically send e-mails

			when the Alarm changes in state SET/CLEAR
Related Alarm Priority	CRITICAL HIGH MEDIUM LOW		Priority for related alarm when set (not cleared)
Web page			Link to a web page that describes possible actions for the Operator.
properties		array/null	Optional. List of properties
Short Description	Weather Station External Temperature	string	Short description of the monitored value. This shall be used also for dysplaying purposes

11. Logging Data Model (LOGDM)

The **Logging Data Model (LOGDM)** describes log records, i.e. either events that occur in the system or actions between human operators and the system itself. All the following logs are composed by a timestamp, produced at the moment of their creation, plus their specific information.

Table 13 shows the main parameters describing the model of the LOG data.

In the case of devices, logging points are represented by a LOG port that could be exposed by a device or assembly or by a generic software component of the **MA system**. Figure 11 shows the logical model that represents the Device and Assembly nodes of the MA system, with the LOG port.

Table 13: Logs

Fields	**Example**	**Data-type / Unit**	**Provided by**	**Description**
LogEntryOriginID		string	Software Component	**Uniquely** Identifies the Log Origin
LogEntryID		string	Software Component	**Unique ID** assigned to the log event.
LogEntryLevel	Trace,Debug, Info, Notice, Warning, Error, Critical, Alert, Emergency	enum symbols: ["TRACE", "DEBUG", "INFO", "NOTICE", "WARNING", "ERROR", "CRITICAL", "ALERT", "EMERGENCY"]	Software Component	Log Entry Level for the log event
LogEntryOriginType	ComputingSystemLog,ACSLog,SoftwareLog,AssemblyLog,GUILog,OperatorLog	enum symbols: ["ComputingSystemLog","ACSLog","SoftwareLog","AssemblyLog,"GUILog","OperatorLog"]	Software Component	Source of the Log Entry
Timestamp	2019-11-26T12:42:12.0	double	Software Component	It specifies the exact time when the log entry

				was submitted. The time is encoded in ISO 8601 format with a precision to one millisecond. The time is specified in TAI.
Message		string	Software Component	text of the log message

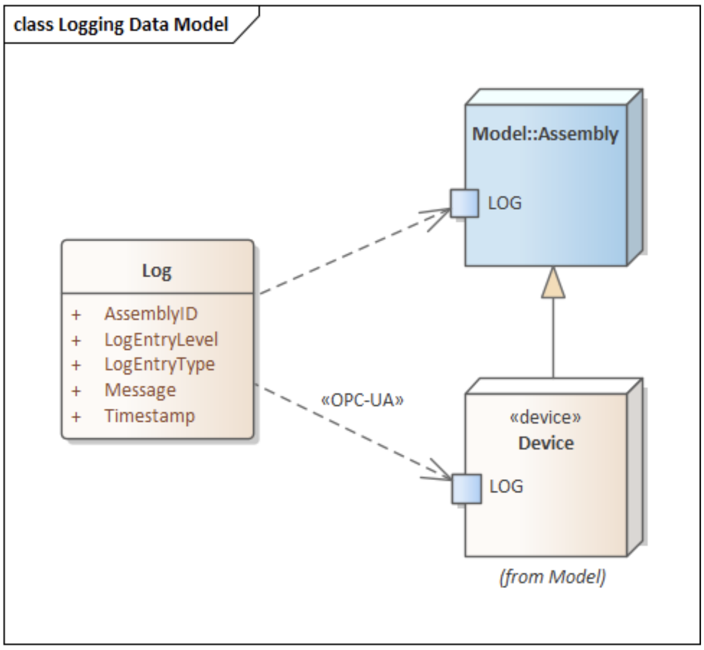

Figure 11: Logging Data Model (LOGDM)

LogEntryOriginType is:

1. **ComputingSystemLog**: logging info with timestamp produced by computing systems.
2. **OperatorLog**: Operator comments and other logging information should be entered in the Observing Log Table (or logbook), which has not yet been adopted.
3. **GUILog**: description of all the actions performed by the operator via the telescope GUI. Furthermore, it includes events that occur at a high level for those subsystems controlled and/or monitored by the GUI itself.
4. **AssemblyLog**: log generated by a hardware assembly. e.g. THCUERRORLOG: error information, for the telescope's subsystem, raised by the Telescope Health Control Unit (THCU). These logs contain error code, origin system, the severity of the error and a brief description. Please note that some log may contain an additional timestamp if the error event described occurred before the creation of the log itself.
5. **SoftwareLog**: a collection of logging info produced by other frameworks that are not ACS.
6. **ACSLog**: a collection of the logging info produced by the ACS framework.

Different LogEntry types could have different fields. E.g. Table 14 reports the *additional* fields of an **ACSLog**.

Table 14: ACS Software Log (additional fields)

Fields	Example	Data-type / Unit	Provided by	Description
host		string	Software Component	The name of the computer on which the log entry is generated.
process		string	Software Component	The name of the process from which the log entry is generated.
File				The identification of the source file
Line				The line number in the source code where the log entry was submitted
Routine	Activator::Init or just init()			The fully-qualified name of the subroutine (function) where the log entry was submitted from
Thread				The identification of the thread
Context				Any additional context information supplied by the issuer of the log entry
StackId				Identification of a bundle of related log entries. All log entries in a bundle are caused by the same “root” log entry
StackLevel	The root log entry has a StackLevel of 0, the immediate log entries caused by the root log entry have a StackLevel of 1, etc			
LogId	“file not found”, “out of memory”, “container starting”, etc.			Optional Attribute: which uniquely identifies the log entry’s class
Priority		int		Priority is measured as an integer number ranging from 1 to 15, where 1 is lowest and 15 highest priority. The value of 0 indicates the default priority.

Audience		enum symbols: ["Operator",Developer,"Sysadmin","DBA,"Scilog"]		A specified Audience for the log event defined audience in acscommon.idl.

12. Telescope Data Model (TELDM)

The **Telescope Data Model (TELDM)** describes the primary pointing position of each telescope that is stored in the pointing table. The **pointing table** stores both the commanded pointing direction and the pointing direction reported by the telescope.

The data model can be divided in sub-categories:

1. Telescope status
2. Pointing information
3. Pointing Monitoring Camera information

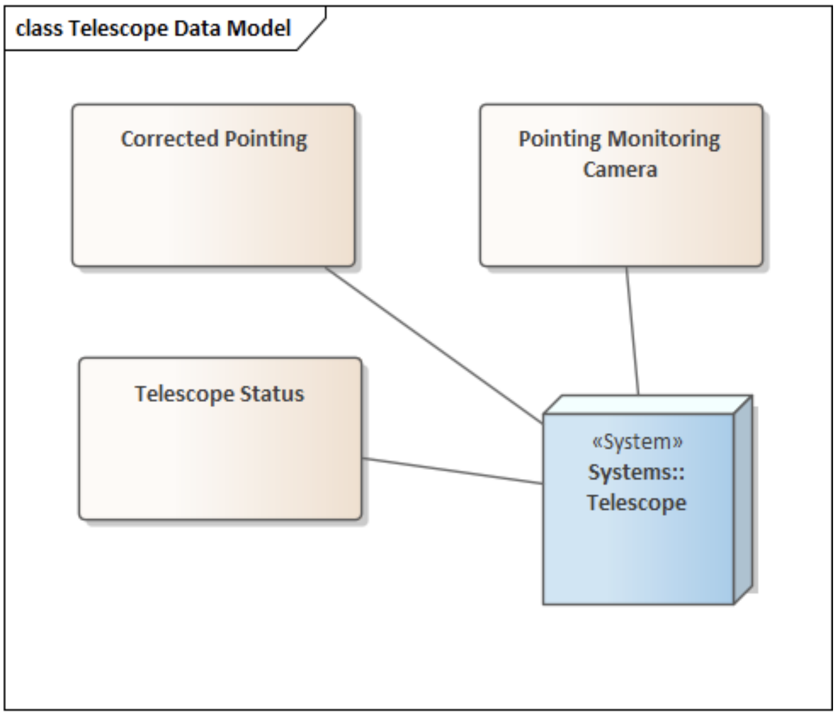

Figure 12: Telescope Data Model

Table 15, Table 16, and Table 17 show, respectively, the main parameters of the TEL data model (which includes telescope status, telescope pointing information and pointing monitoring camera information).

Table 15: Telescope status information

Fields	Example	Data-type / Unit	Provided by	Purpose	Description
Timestamp					Time of acquisition
Telescope ID	1..9	Int / [deg]		Scientific Analysis	Telescope identification number
Telescope Configuration ID					Check SCDB to understand elements of the telescope and position
Telescope Status		Float / [deg]		Scientific Analysis	

Table 16: MOUNT/TCU Pointing information

Fields	Example	Data-type / Unit	Provided by	Purpose	Description
Timestamp					Time of acquisition
Mount ID					
Mount configuration ID					Check SCDB.
Pointing model ID					Check SCDB.
IRS					TAI-UTC, DUT1, X_PM, Y_PM ftp://maia.usno.navy.mil/ser7/ser7.dat
Actual RA		Float / [deg]	TCU	Scientific Analysis	
Actual Dec		Float / [deg]	TCU	Scientific Analysis	
Elevation angle of tracking position after offset correction		Float / [deg]	TCU	Scientific Analysis	
The azimuth angle of tracking position after offset correction		Float / [deg]	TCU	Scientific Analysis	
Target pointed (yes/no)		Bool	TCU	Data Quality Check / Data Selection	
Target tracked (yes/no)		Bool	TCU	Data Quality Check / Data Selection	
Correction for refraction (yes/no)		Bool	TCU	Data Quality Check / Data Selection	

Pointing model status (yes/no)		Bool	TCU	Data Quality Check / Data Selection	
Mount status		Int	TCU	Data Quality Check / Data Selection	

Table 17: Pointing Monitoring Camera information

Fields	Example	Data-type / Unit	Provided by	Purpose	Description
Timestamp					Time of acquisition
PMC id		Int	PMC	Scientific Analysis	
PMC configuration ID					Check TCB
Pointing model ID					to move pointing from PMC to camera
RA pointing from PMC astrometry		Float / [deg]	PMC	Scientific Analysis	
RA pointing error		Float / [deg]	PMC	Scientific Analysis	
DEC pointing from PMC astrometry		Float / [deg]	PMC	Scientific Analysis	
DEC pointing error		Float / [deg]	PMC	Scientific Analysis	
Zenith Angle of tracking position		Float / [deg]	PMC	Scientific Analysis	
Zenith Angle of tracking position error		Float / [deg]	PMC	Scientific Analysis	
Azimuth Angle of tracking position		Float / [deg]	PMC	Scientific Analysis	
Azimuth Angle of tracking position error		Float / [deg]	PMC	Scientific Analysis	
PMC Status		Int	PMC		

13. System Configuration Data Model

The **System Configuration Data Model** describes configuration of MA and all data required to:

1. start the system (hardware and software), reacting, in the process, to any differences between what is expected and what is registered as subsystems come online;
2. make possible the determination from **monitoring points** of the physical and logical location of the component that produced it, and, conversely, from a component (e.g., a telescope) in the system, retrieve any subset (e.g., sampled in time) of its monitoring points for trend or fault analysis;
3. define the alarms of the assembly;
4. define the control points of the assembly;
5. define the sequence of operation for the startup, configuration and shutdown of subsystems and software systems;
6. predict, for purposes of planning (e.g., for preparation of Observing Proposals and Scheduling Blocks to be executed), future system configurations;
7. provide a unique source for all quasi-static data that characterize the system, e.g the supporting data listed below.
8. define the **Critical Item**, i.e. an item whose failure or malfunction can cause degradation of the performance of any system or subsystem the repair of which would be costly or require undue time.
9. maintain a history of hardware and software configurations so that, in the event of anomalies or failures, the involved hardware or software components can be unambiguously identified;

13.1. System Configuration DB (SCDB)

The **SCDB** or **System Configuration DB** stores the data of the **System Configuration Data Model**, including history. The configuration is characterised by the complexity of the system, that generates a large number of possible combinations of assemblies. Each assembly of the system should be connected with monitoring points that generate data in a time-ordered stream which is not easily mapped to the hierarchical structure of the elements that produces them. A solution is needed that manages and maintains these relationships, ensuring correctness and consistency, in an integrated way.

Figure 13 shows a logical model of the SCDB with the hierarchical structure of System, Assembly, and Device described in [AD3] and their connection with the logical model. Details will be described in a detailed design document.

The configuration of the scientific instruments (Cherenkov Camera, SI3 and Optica Camera) can be divided in

- baseline configuration, saved in the SCDB
- dynamic configuration, that depends by the scientific objective of the observation, and is part of the Observing Mode (see Sect. 5.2.1.).

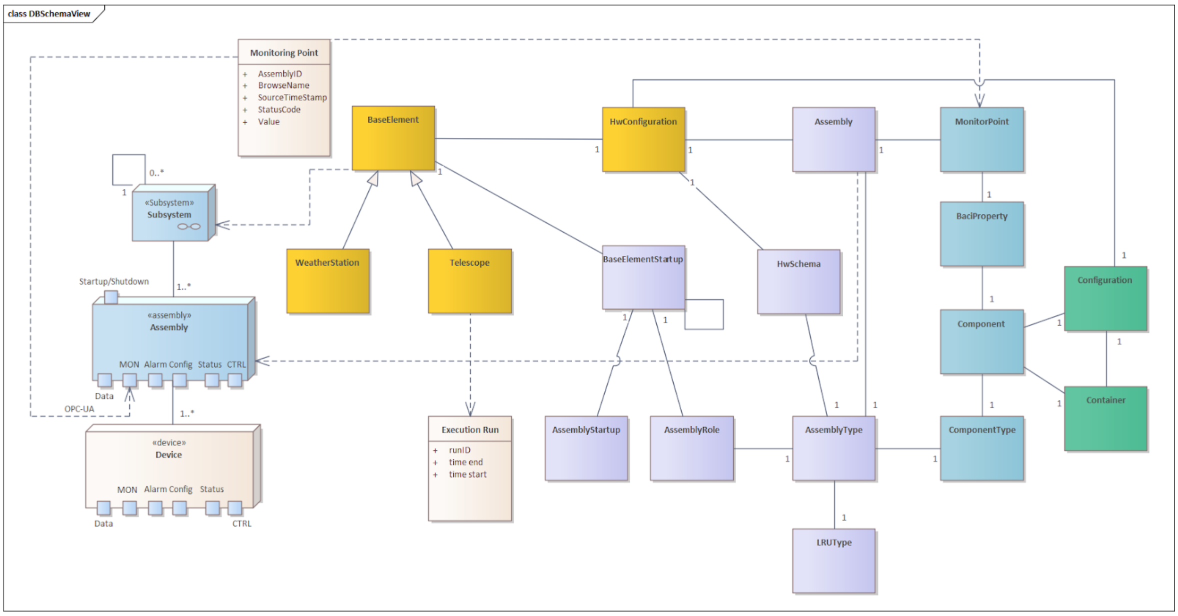

Figure 13: A logical model of the SCDB with the hierarchical structure of System, Element, Assembly, and Device described in [AD3] and their connection with the logical model.

13.2. Supporting data

- **SVC** (service) contains data that acts as a service to an observation, hardware or software component. Generally, these are quantities that change rarely. This data must include a validity range or a list of dependent/parent data products. Subtype examples include:
  - **LUT**: look-up-tables which are generally low-level data needed by various algorithms or hardware components. Examples might include scaled-parameter tables for Cherenkov data reduction and analysis, tables of leap-second corrections for time conversion. Produced by the MC simulation processing system. Data levels:
    - **LUT1**: look-up-tables/models used by telescope-wise discrimination and reconstruction algorithms to estimate energy, arrival direction, and particle identity discrimination parameters of the air-shower events;
    - **LUT2**: look-up-tables/models used by array-wise discrimination and reconstruction algorithms to estimate energy, arrival direction, and particle identity discrimination parameters of the stereo event.
  - **IRF**: instrumental-response functions: high-level instrumental response function data in common representation (effective area, energy response, point-spread function, background distribution model, etc.), expressed in instrumental coordinates or in functional form. Produced by the MC simulation processing and data processing systems. Data levels:
    - **IRF2**: global IRFs covering all of the instrumental phase-space.

        - **IRF3**: reduced observation-related IRFs generated by filtering the IRF2 over several parameters, weighted by the observation configuration parameters of a particular event dataset.
    - **ML-MODEL**: pre-trained machine learning model, in a common format for long-term archiving.
    - **CALDB** (calibration database): set of instrumental and pre-computed quantities stored in a dedicated (calibration) database available for being used throughout the entire Cherenkov data processing pipeline for being used throughout the entire scientific data reduction chain.
- **CONFIG**: configuration information for an instrument or software tool/algorithm. Generally, this would be represented as a series of optionally-hierarchical key-value pairs or linked database tables.

14. Science Simulated Data Model (SSDM)

The **Science Simulated Data Model (SSDM)** is an abstract identification of the types of data, data products and metadata used and generated by the **Simulation System** software (see [AD3]). The Science Simulated Data Model defines the collection of information generated through MC simulations, used for the characterization of the Cherenkov events and the definition of the expected array performances through the instrument response functions (IRFs) needed for scientific analysis.

The SSDM allows to establish the connection between the acquired scientific raw data to the astronomical properties of the observed target. LUTs and IRFs are produced through the Simulation System and are used by the **Data Processing System** to derive physical properties of the observed targets.

Science Simulated Data Model (SSDM) The science simulated data model defines the collection of information generated through MC simulations, used for the characterization of the Cherenkov events and the definition of the expected array performances through the instrument response functions needed for scientific analysis.

Table 18 and Table 19 show the parameters used in the **System Configuration DM** for the MC simulations: Table 18 shows only fixed parameters, while Table 19 contains parameters used in the simulations which can vary with time. These parameters (like mirror reflectivity, instrument PSF, optical throughput, etc) need to be measured on a periodic basis in order to properly assess the period-wise IRFs calculated from the MC production. The frequency of measurement for the parameters in Table 19 can be linked to each new ASTRI MA MC production (e.g., two or three times per year).

Table 18: Fixed input parameters for Monte Carlo simulations

Parameter	Value	Data-type / Unit	Provided by	Purpose	Description
Site location	28.303017 -16.506095	Float/deg	Array design		Latitudine e Longitudine
Altitude	2330	Float/deg	Array design		Altitude above sea-level
Telescope positions (X, Y, Z)		Float/m	Array design		Telescope positions
Magnet	30.968 22.965	Float/microT	Dedicated software		Geomagnetic field H and Z components
Magnetic declination	4.67	Float/deg	Dedicated software		Geomagnetic field declination

Atmospheric model	-		Dedicated software		Atmospheric transmission as a function of wavelength and emission height
NSB sky map	-		Direct measurement		NSB as a function of zenith and azimuth
Night-sky background			Dedicated software		NSB rate
NSB off-axis			Dedicated software		NSB dependence on off-axis angle
Primary diameter	424.1	Float/cm	Design/Direct measurement		Diameter of primary mirror
Primary mirror parameters			Design/Direct measurement		Parameters describing primary mirror shape
Primary segmentation			Design/Direct measurement		Primary mirror tiles
Secondary diameter	180.0	Float/cm	Design/Direct measurement		Diameter of secondary mirror
Secondary shadow diameter			Design/Direct measurement		Diameter of secondary mirror support
Secondary shadow offset			Design/Direct measurement		-
Secondary mirror parameters			Design/Direct measurement		Parameters describing secondary mirror shape
Secondary baffle			Design/Direct measurement		-
Focal length	215.0	Float/cm	Design/Direct measurement		
Effective focal length	215.191	Float/cm	Direct measurement		
Focus offset			Design/Direct measurement		
Mirror offset			Design		
Axes offsets			Design		
Focal surface parameters			Design		-
Telescope transmission			Dedicated software		Shadowing by masts, stays, etc.
Camera body diameter			Design		

Camera pixels			Design		
Camera depth			Design		
Trigger telescopes			Array configuration		Minimum number of triggered telescopes
Trigger pixels			Camera settings		Minimum number of triggered pixels
Tel-trig min time			Camera settings		Minimum overlapping time
Discriminator pulse shape			Camera settings		
Discriminator amplitude			Camera settings		
Discriminator threshold			Camera settings		
Discriminator var threshold			Camera settings		
Discriminator gate length			Camera settings		
FADC pulse shape			Camera settings		
FADC var sensitivity			Camera settings		
FADC max signal			Camera settings		Saturation level

Table 19: Variable input parameters for Monte Carlo simulations

Parameter	Example	Data-type / Unit	Provided by	Purpose	Description
Atmospheric transmission			LIDAR		
Telescope random angle			TCU	Tracking	
Telescope random error			TCU	Tracking	
Mirror align random horizontal			Direct measurement	PSF evaluation	
Mirror align random vertical			Direct measurement	PSF evaluation	
Mirror align random distance			Direct measurement	PSF evaluation	
Mirror reflection random angle			Direct measurement	PSF evaluation	
Mirror reflectivity			Direct measurement		Reflectivity as a function of wavelength and incident angle
Mirror secondary reflectivity			Direct measurement		Reflectivity as a function of wavelength and incident angle

Camera filter			Direct measurement		Camera window transmission as a function of wavelength and incident angle
Quantum efficiency			Direct measurement		SiPM Photon Detection Efficiency
QE variation			Direct measurement		PDE pixel by pixel variation
SiPM photoelectron spectrum			Direct measurement		single photo-electron spectrum
FADC noise			Direct measurement	Electronic noise	
FADC LG noise			Direct measurement	Electronic noise	
FADC amplitude			Direct measurement		
FADC LG amplitude			Direct measurement		
FADC pedestal			Direct measurement	Pedestal	
FADC VAR pedestal			Direct measurement	Pedestal variation	
FADC LG pedestal			Direct measurement	Pedestal	
FADC LG VAR pedestal			Direct measurement	Pedestal variation	

15. Science Results Data Model (SRDM)

The **Science Results Data Model (SRDM)** is an abstract identification of the types of data, data products and metadata produced by a data pipeline.

For Cherenkov observations they are generated by the **Cherenkov Data Pipeline** and released as science-ready data and science products to the **Science Users**.

In particular, the data products released and archived in to the **Science Archive** will be:

- **Gamma-like event-list** (EVT3): sets of selected air-shower events with a single final set of reconstruction and discrimination parameters. Produced by the **Cherenkov Data Selection** subsystem of the Cherenkov Data Pipeline. These are the primary science-ready data products delivered to the Science Users for scientific analysis.
- **Reduced instrument response functions** (IRF3): IRFs are required for the transformation of DL3 data products (e.g. array-wise event lists, and instrument, environmental and atmosphere characterisation information) into physical quantities and final science products. Generated by the **MC/LUT IRF Generation** and reduced by the **Cherenkov Data Selection** subsystems of the **Cherenkov Data Pipeline**.
- **Good-time intervals** (GTI): continuous time intervals where science-ready data can be considered acceptable for the final scientific analysis. Generated by the **Cherenkov Data Pipeline** based on instrumental, environmental and atmosphere characterisation information.
- **Science products**: preliminary and final scientific data products (DL4) in physical units, on a target-basis, such as detection plots, spectra, sky-maps, and light-curves, generated in an automated way by the **Cherenkov Data Scientific Analysis** subsystem. These products, if needed, can be further processed and merged to get high-level observatory (legacy) data.
- **Legacy products**: high-level (DL5) “legacy” data products, compliant with the Virtual Observatory (VO) standards and tools, such as ASTRI MA survey sky-maps, diffuse gamma-ray background models, and/or ASTRI MA source catalogues.

For Intensity Interferometry observations they are generated by the **Intensity Interferometry Data Pipeline**:

- **DL3.SI3:** time coincidences;
- **DL4.SI3**: diagram of the temporal correlation.

16. Annex A

16.1. Atmosphere Characterisation parameters needed for data quality check and data selection

The following table lists the **atmosphere characterisation parameters to be taken into account for data quality and data selection purposes**. From a general point of view, the atmospheric parameters should not jeopardize the possibility of observing; nevertheless, limits on their values could be defined in the night observation proposal.

Table 20: selection of Atmosphere Characterisation parameters needed for data quality check

Parameter	Units	Data from:
Atmospheric extinction profile	**TBD**	**LIDAR + Analysis Software**
Sky brightness	**mag/arcsec^2**	**Sky Quality Meter**
Night Sky Background level	**counts/sec**	**UVSiPM**
Diffuse Night Sky Background	**ph/(m^2 ns sr) or ph/(m^2 ns deg^2)**	**UVSiPM + Analysis Software**

16.2. Observations during partial/full covered sky or severe dust extinction (Calima)

Observations of backup sources or targets (TBD) during bad weather or unfavourable atmospheric conditions, which do not represent a danger for the whole array (i.e., partial or total cloud coverage, low light transmission due to the presence of dust, etc.), can be performed if the ENV limits on temperature, humidity, wind, etc. are nominal.

Measurements of cloudiness from the ASC and atmospheric extinction from the LIDAR should be evaluated in order to stop (or not even start) the scheduled observations for the current night and change to an alternative source/target.

The external conditions which shall influence the decision to continue with the so-called “unsuitable weather observations” are:

- Cloud coverage:

- If the global cloudiness from the ASC is > 50% (TBD), stop or avoid starting nominal observations. Check weather forecasting, satellite images, and the All-sky camera (ASC) in order to evaluate whether it would be possible to start/continue with nominal observations in a relative short time. If not, switch to an alternative source/target if the ENV monitor values are still within the safe limits.

- Atmospheric extinction: during cloudy conditions or presence of particles in the atmosphere due to Calima, check the LIDAR transmission up to a distance of TBD km:
  - If transmission is above 0.5, keep observing.
  - If transmission is below 0.5, stop observations. Check if unsuitable-weather-observations are possible.

It is the responsibility of the **Operator/Astronomy-on-duty** to be aware about the availability of backup sources and unsuitable-weather-condition targets, which should have been established in advance during the long-term schedule preparation.

17. Annex B - Avro schema for reading points

A **reading point** can be a monitoring point or the data acquired from assemblies. This definition covers in an uniform manner many data models reported in this document.

The following Avro schema reports the generic model of a reading point.

Each reading point is unequivocally identified by assembly, name, assembly_serial_number.

```
{
  "type": "record",
  "name": "ReadingPoint",
  "namespace": "astri.mon.kafka",
  "fields": [
     {
        "name": "assembly",
        "doc": "Name of the Software Component where this property is exposed,
should correspond to the monitoring data source it is serving e.g.
'PowerSupply1'",
        "type": "string"
     },
     {
        "name": "name",
        "doc": "The display name of the property, e.g. 'Current'",
        "type": "string"
     },
     {
        "name": "assembly_serial_number",
        "doc": "",
        "type": "string"
     },
     {
        "name": "timestamp",
        "doc": "Source time (when the property is received by the server)",
        "type": "long"
     },
     {
        "name": "source_timestamp",
        "doc": "Source time (when the property is produced by the device)",
        "type": ["null", "long"]
     },
     {
        "name": "units",
```

```
        "doc": "A string representing the units (normally base SI units or
combinations of SI units) of the quantity represented by the property. E.g.
'A'",
        "type": "string"
      },
      {
        "name": "archive_suppress",
        "doc": "FALSE if this value will be archived",
        "type": "boolean"
      },
      {
        "name": "env_id",
        "doc": "A string representing the env. e.g. sln, teide, testbed ...",
        "type": "string"
      },
      {
        "name": "data",
        "doc": "Contains time-series data that are used to monitor the status or
quality of hardware, algorithms, or other data products. These typically update
periodically during the operation of the array, or during data processing, at a
rate typically much slower than event data and faster than the length of a
typical Observation Block. Examples would be slow-control information like
tracking positions, weather monitoring data, or the status or quality-control
data of a particular hardware or software component",
        "type": {
          "type": "array",
          "items": ["double", "int", "long", "string", "boolean"]
        }
      }
   ]
}
```