UNIVERSITY OF HAWAII AT MĀNOA

Institute for Astrononmy

Pan-STARRS Project Management System

Data Exchange Standard (2.02) for solar system object detections and orbits

A tool for Input/Output definition and control

Grant Award No.	**: F29601-02-1-0268**
Prepared For	**: panstarrs team**
Prepared By	**: A. Milani, L. Denneau, F. Pierfederici, R. Jedicke**
Document No.	**: PSDC-MOPS-DES-2.02**
Document Date	**: August 9, 2007**
Revision	**: 2.02**

DISTRIBUTION STATEMENT

Approved for Public Release – Distribution is Unlimited

Submitted By:

[Insert Signature Block of Authorized Developer Representative] Date

Approved By:

[Insert Signature Block of Customer Developer Representative] Date

Revision History

Revision Number	Release Date	Description
1.05	2006.07.25	First approved draft
1.06	2006.10.05	Revised version after first tests
2.00	2007.01.15	This major revision has the purpose of making this document exportable, rather than just for internal discussion. Most specific references to MOPS or Pan-STARRS have been removed. The terminology has been explicitly stated and basic concepts have been explicitly defined. This has the additional advantage of allowing a presentation of the export format in an object oriented way. The file names and formats are not just a low level I/O tool, but are at the service of a software structure connected to our conceptual understanding of what we are doing. This makes the data exchange standard easier to explain to the people who are not already been involved in our work and helps in the software writing and especially maintenance.
2.01	2007.4.24	Revised Orbit definition and format, to allow for the use of different coordinate system when the COM one is unstable; also implemented Orbit Computer Code and changed Tracklet_Pointer/Track_Pointer to comply with de facto conventions in use in the simulations. Discussions on the generation of automatic OIDs have been added in several places.
2.02	2007.7.28	Added radar astrometry data types; necessary to ingest in DES historic observations as stored at MPC (and JPL for radar only).

TBD / TBR Listing

Section No.	**Page No.**	**TBD/R No.**	**Description**
3.10	6	2006.10.05	missing IMAGE_METADATA definition and format

Contents

1 Purpose

This document defines a standard for data exchange of solar system object data (e.g., detections and orbits) in the form of flat text files. e.g., data exchange between Pan-STARRS, MPC, LSST, NEODyS and JPL; distribution of catalogs to astronomers; input to planetarium programs; requests for follow up.

This low level format will be later incorporated in a higher level structured format e.g., XML: there will be a parser combining the flat files described here into an XML structure and one unpacking the XML into files with appropriate names. In this way an announcement/job request can be sent either as a tarball with many files or as a single (possibly compressed) XML file.

2 Introduction

In this section we provide a cursory discussion of the logically and astronomically complex problem of identifying detections, fitting orbits, and verification of real solar system objects within astronomical data. For a more detailed discussion please consult the references.

To understand the definitions of objects and data types provided in the following sections we need to define our terminology on identification, orbit determination and discovery of solar system objects. This is complicated by the existence of discordant terminologies in the literature and because of the convergent interest on these topics by two communities: the one interested in solar system natural objects, and the one working on artificial satellites and space debris[1]. Although each terminology is acceptable, provided unambigous definitions are available in the published literature, we need to select a single name for each object and data type to avoid confusion in a data exchange standard (DES). These unambigously defined terms will always be used with capitalized initial to mark the distinction with possible informal uses of the same word.

The process of discovery (lowercase) of a solar system object begins with Detections, essentially one astrometric measurement at a given time. Detections belonging to the same observing night are assembled into Tracklets[2] by polynomial fitting in such a way that they may belong to the same Solar System Object (SSO). One Tracklet is almost never enough to compute an Orbit, the set of six heliocentric orbital elements at some epoch (with an absolute magnitude if photometry is available with the astrometry). At this stage of the processing there is no way to decide whether a Tracklet is *True* - formed only with Detections of one and the same SSO. There is also no way to decide if each individual Detection is *True* - belonging to a SSO rather than a variable star, a statistical fluke, etc.

The next stage in the processing is to assemble Tracklets into sets called Tracks that may be consistent with belonging to the same SSO (if all the Detections in a Track are *True* and belong to the same SSO then the Track is also *True*). This processing step is the most difficult and could be performed with different algorithms. e.g., by using a set of (one or more) Virtual Asteroids, objects with a hypothetical Orbit which could have to do with the data, or by using polynomial fitting with controlled coefficients [Kubica et al. 2007]. The Orbits utilized in this step are Preliminary; e.g., they may use only some subset of the Detections (or some quantities derived from the Detections, such as the Attributable, angular positions and velocities on the celestial sphere).

There is no way to confirm that a Track is True unless it is possible to find a Least Squares Orbit, an Orbit resulting from a fit of the Orbit to all the Detections in the Track according to the Least Squares Principle[3]. If a Least Squares Orbit is

[1] The terminology used internally by MOPS is partly derived from the space debris community rather than the asteroid terminology.

[2] In the discovery of TNO it is customary to use inter-night detections to form Tracklets over a time span of several days.

[3] The number of parameters being fit may be less than 6, e.g., 4 or 5; some Detections may be discarded as Outliers but the amount of information contained in the Detections needs to be such that the observation equations are overdetermined.

found then the following objects are also available: the Covariance, which is an intermediate product in all Least Squares algorithms and can be used to assess the uncertainty of the Orbit; the Residuals (observed minus computed) for each Detection; and the Metrics, the results of the application of statistical tests to the set of Residuals (the simplest example being the RMS of all Residuals, Outliers excepted). If the Last Squares Orbit has been found, and the Metrics satisfy some quality control criteria, the Track can be tentatively considered True and the join of each Detection and its Residual can be considered an Observation[4] of a SSO with at least some known properties. The information collected in this process is an Identification, consisting of a Confirmed Track with its Orbit, its Covariance, its Observations and its Metrics.

Given an Identification, the question arises on how to arrive at a formal conclusion for whether it is True and whether it represents a new Discovery of a SSO. The answers to these questions are difficult in practise because it is never sufficient to consider a single Identification in isolation from all others obtained by the same and by other surveys. Thus, it is necessary to apply some additional quality control to the Identification, e.g., , to find out whether the Orbit has been determined well enough to establish the nature of the object discovered. It is also necessary to assemble all Identifications in a structured list with some syntactical rules, generically indicated with the term Normalization[5], removing Identifications which are found to be duplicated, discordant and belonging to already known SSO.

3 Data types

The are two generic data types: basic and composite. The composite data types are built out of basic data types.

Each data object belonging to either a basic or a composite type needs to have a unique object identifier (OID). This corresponds to the *Object Oriented Data Model* of database theory[6].

The software implementation of the data types is not in any way constrained by this standard although some of the definitions given in this document are highly suggestive of objects and classes. The implementation is language dependent so that the same objects represented in Fortran 95, C++ and SQL may be difficult to recognize.

The following sections provide the semantic description of each data type, its definition in terms of records formed by other data types (starting from the basic types), and the format used for import/export. Note that the format is not specified in too much detail: character strings and floating point numbers can be read in free format. There could be some restrictions to accepted formats due to the implementation of the standard in a specific software, e.g., character strings and lists may have a maximum length, but this is not part of the standard.

3.1 Basic Data Types

There are 12 basic data types which can be exported/imported:

1. DETECTION
2. RADAR_ASTROMETRY
3. ORBIT
4. COVARIANCE

[4]The Outliers removed in the Least Squares process may not be Observations, or may be Observations but of lower quality.

[5]There are several different definitions of Normalization, for details see Section 6.

[6]Two different objects could be identical in their content and still be distinct as specified by their different OID. This can be avoided with a suitable database normalization, in which duplicates are removed, as discussed in Section 6.

5. RESIDUAL
6. RADAR_RESIDUAL
7. METRICS
8. IDENT_HEADER
9. TRACKLET_POINTER
10. TRACK_POINTER
11. EPHEMERIS
12. IMAGE_METADATA

3.2 Composite data types

The seven composite data types are:

1. TRACKLET
2. OBSERVATION
3. RADAR_OBSERVATION
4. TRACK
5. SOLUTION
6. IDENTIFICATION
7. SCHEDULE.

Composite data objects are obtained by combining the basic ones with some syntactical rules. The notation Z={A, B, C} represents the RECORDOF operator [Ullman 1991, Section 2.7] for which the data object Z consists of the data objects A, B and C. We use Z={As} to indicate that Z consists of many instances of the data object A. In this way it is comparatively easy to define, and also to input/output, a new composite data type.

A survey's data products are generated as data types with some syntactical rules. The data exchange itself occurs in the exchange of flat data files as specified herein.

3.3 File Format

Data exchange is managed through the transfer of flat files. The input request must be organized as follows:

[jobname].in.manifest

[jobname].in.[objectType1]

[jobname].in.[objectType2]

....

[jobname].in.[objectTypeN]

The corresponding output answer should be organized as follows:

[jobname].out.manifest

[jobname].out.[objectType1]

[jobname].out.[objectType2]

....

[jobname].out.[objectTypeN]

Composite data types are created using multiple files sharing the same prefix [jobname] with different suffixes representing the basic type: [jobname].[in|out|nor].[suffix]. The 'jobname' may be any arbitrary string of letters and numbers and may include a limited number of other characters (e.g., '=', '+', '-'). The 'in' and 'out' qualifiers specify whether the file is an input or output file; the 'nor' qualifier is used for the output of a normalization procedure. The suffix specifies the data type contained in the file. To simplify data transmission and verification, all job transmissions also include a manifest file ([jobname].[in|out].manifest) containing a list of all the files enclosed in the job (including the manifest file itself). A single request or response in the data exchange may be accomplished with files contained in a tarball that may contain multiple and different composite data types distinguished by the jobname.

Input and output files may be combined in a single tarball. This file format and structure allows multiple requests, even of different natures, and a mixture of requests and answers without ambiguity.

Each file may contain multiple objects of the same type preceded by lines of column headings (for single line formats) or explaining the content (for multiline formats). In each line the only separator used is a string of blanks (no tabs). Optional arguments are allowed only in multiline formats with keys (described in more detail below). Comment lines indicated with an exclamation mark (!) in the first column may be safely skipped on input. Heading lines are preceded by double exclamation marks (!!) and may also be skipped on input or used as a double check.

Note that this simple format allows merging of two lists of the same basic data types by simply appending the files. Lists of composite data types may also be merged by appending each of the output files, taking care to preserve the ordering[7].

This simple file format allows easy read/write access from FORTRAN, Perl and C but also makes them amenable to UNIX operating system commands e.g., grep, sort, sed. Interface routines to handle I/O for these data from programs in Fortran 77/95 and from C are provided as free software (e.g., , the Fortran will be distributed with OrbFit; the C and Perl version from Pan-STARRS/MOPS.).

4 Basic Data Types

Note that each basic data type is exported in a specific file, with its own format.

[7] This does not imply that appending the files amounts to what needs to be done for a complete merge of two structured databases of objects. To preserve some syntactical rules, e.g., , to avoid duplications and discordancies some normalization procedure may be necessary. See Section 6.

4.1 Detections

A *DETECTION* is the astrometry and photometry of an image 'source'. It may or may not correspond to a solar system moving object, e.g., it could be a statistical background fluctuation above the threshold S/N level or a variable but not moving object. It must contain metadata allowing the estimation of the astrometric error according to some error model.

DETECTION={OID, TIME, OBS_TYPE, RA, DEC, APPMAG, FILTER, OBSERVATORY, RMS_RA,RMS_DEC, RMS_MAG, S2N, Secret_name}

FORMAT: SINGLE LINE

FILENAME:

[jobname].[in|out].detection (for individual detections),

[jobname].[in|out].tracklet (detections in tracklets) or

[jobname].[in|out].orphan (for low signal to noise individual detections).

The items contained within a *DETECTION* are:

- OID - a unique identifier that has a different interpretation depending upon the case: for raw detections and orphans, the OID is a DETECTION_OID and must be unique for each record.
- TIME - UTC Modified Julian Date (MJD) in days of the observations.
- OBS_TYPE - the type of observation (=O for an optical observation[8].
- RA, DEC - right ascension and declination in degrees
- APPMAG -the apparent magnitude in the specified FILTER
- FILTER - the image measurement band (R, V, I, etc.).
- OBSERVATORY - the 3 character observatory identifier.
- RMS_RA, RMS_DEC[9] - the estimated accuracy in arcsec in RA and RA*$\cos$(DEC) for DEC respectively (two equal values imply a circle on the tangent plane at RA, DEC to the celestial sphere)[10].
- RMS_MAG - (in magnitudes) the formal uncertainty in the magnitude. It must never be used for weighting due to lightcurve variations.
- S2N - the signal to noise ratio of the source.
- Secret_name - an independent multiple character ASCII identifier for the detection. e.g., it may be the name of the object in a solar system model used to generate the detection. FALSE is used for a synthetic false detection. For real data the name of the object could be inserted after a firm attribution. The string NS (for Not Synthetic) indicates real data. It is possible, in case the detection is confirmed to belong to a real object, to put there an official IAU designation (without blanks), e.g., 2004VD17, (99942).

[8] A radar observation is a different data type. A DES format for radar data will be defined in a later version of the DES.

[9] A correlation between RA and DEC may result from the astrometric solution. This problem will be severe only for strongly trailed images.

[10] It has been proposed to add to the standard also RMS_TIME - the estimated uncertainty in the TIME (especially relevant for fast moving objects). This has not yet been implemented.

4.2 Radar Astrometry

Radar Astrometry comes as either a range or a range-rate measurement. It is essentially different from optical astrometry not just because of the different observable, but because the individual data points are 1-dimensional, that is, at a given time there could be either range, or range-rate only, or both. Radar Astrometry cannot be obtained unless there is already a known orbit, giving a small uncertainty in the predicted angular position (otherwise the narrow radar beam could not pointed properly). Thus each observation contains a single designation of the obejct being observed, and the identification problem does not apply.

RADAR_ASTROMETRY= {NUMBER, NAME, DATE, TIME, VALUE, RMS, UNIT, SURF, FREQ, TRANSM, RECEIV}

FORMAT: SINGLE LINE (Warning: in the JPL format the NUMBER may be missing and be replaced by blanks, thus it is not possible to read in free format).

FILENAME: [jobname].[in|out].radar_astrometry

- NUMBER the official MPC number of a numbered asteroid; left blank if the asteroid is not numbered (Problem: radar observed comets).
- NAME the official MPC name (for a numbered asteroid) or the official MPC designation.
- DATE year/month/day
- TIME of the day, hour/minute/second (UTC). Note the UTC time is always rounded to 1 minute[11].
- VALUE of the observed quantity.
- RMS standard deviation, as estimated in the data reduction purposes. process[12].
- UNIT: either Herz (for range-rate, that is Doppler, measurement) or microseconds of 2-way light travel time.
- SURF a code indicating whether the observations have been reduced to the asteroid Center of Mass (code=COM) or surface bounce (code=PP).
- FREQ carrier frequency in MHz.
- TRANSM name of transmitting radar antenna
- RECEIV name of receiving radar antenna; could be different from TRANSM (bistatic mode) or be the same.

NOTE: this format is the one used by JPL, the primary source of radar astrometry data. It does comply with all the requirements of our standard, nevertheless we have chosen not to reformat the JPL records (to remove blank fields etc.).

[11] In fact, the radar astrometry is always a normal point representing data taken over an extended period. This is why the time of the normal points is rounded.

[12] This RMS estimated by the observers is always taken as estimate of uncertainty for weighing; no biases and no correlations are assumed.

4.3 Orbits

An *ORBIT* is a set of orbital elements defining an orbit including epoch and absolute magnitude.

ORBIT= {ID_OID, COO, ELEMS, H, t_0, INDEX, N_PAR, MOID, COMPCOD}

FORMAT: SINGLE LINE

FILENAME: [jobname].[in|out|nor].orbit

The main item is the 6-dimensional real vector ELEMS, which describes the orbit initial conditions by using a set of coordinates specified in the string COO. The most used coordinates are COM (COMetary) q, e, I, Omega, argperi, t_p and COT (COmetary True anomaly), with the last coordinate replaced by true_an, where

- q - perihelion in AU
- a - semi-major axis in AU
- e - eccentricity (a pure number)
- I - inclination in degrees
- Omega - longitude of node in degrees
- argperi - argument of pericenter in degrees
- t_p - time of perihelion in MJD TDT in days.
- true_an - true anomaly in degrees.

Other coordinates, such as KEPlerian, CARtesian, ATTtributable, EQUinoctal are allowed, but cometary type coordinates (either COM or COT) are used as default. The other data defining an orbit are

- ID_OID - the identification code (see below under Identification).
- H - absolute magnitude in the V band[13]
- G - light curve slope parameter (defaults to 0.15 if otherwise unavailable).
- t_0 - epoch time in MJD TDT in days.
- INDEX - a whole number indicating the order in the list of Solutions for the same identification (defaults to 1). i.e., there may be multiple orbit solutions for a list of detections.
- N_PAR - the number of fit orbital parameters. It may be 4 or 5 for constrained solutions, 6 for a nominal orbit, and 0 for preliminary orbits. *Note that a Preliminary Orbit provides almost all the information contained in the ORBIT data type while a Least Squares Orbit has more information requiring the composite IDENTIFICATION data type.*
- MOID - Minimum Orbital Intersection Distance with respect to the Earth in AU. If not available, the value -1.0 is used to indicate it has not been computed[14].

[13] It has been proposed to add the opposition effect parameter G, which is indeed available and helps in fitting H for many well observed objects. However, this has not yet been implemented.

[14] The MOID needs to be computed as a signed value [Gronchi and Tommei 2007], but the absolute value is used to comply with tradition.

- COMPCOD a code designating the software system used to compute the orbit, e.g., MOPS, OSS. Information useful to specify which part of the code has done the job may be appended, e.g., MOPS-LT, OSS3.

The header line which begins with '!!' may be used to specify the meaning of the columns, including the six coordinates of the ELEMS vector, for the majority of the records in the file; however, some lines may use different coordinates, as specified in the COO entry on each record. The reason for allowing the use of different coordinates in the same output file is that there can be cases where a given coordinate system is undefined (e.g., KEP for $e \geq 1$), has a mathematical singularity (e.g., KEP, COM, COT for $e = 0$ and for $I = 0$), is unsuitable to represent the result because of strong nonlinearity (e.g., KEP, COM, EQU for objects observed over very short arcs)[15].

4.4 Covariance

The *COVARIANCE* data type is a representation of the uncertainty of the orbit based on the Gaussian formalism. It contains the covariance and normal matrix and other information may be optionally supplied e.g., variance of the MOID and absolute magnitude.

COVARIANCE={ID_OID, INDEX, COVAR_MATR, NORMAL_MATR, [COV_MOID_MAG], [RMS], [EIG], [WEAK]}

FORMAT: MULTI LINE.

FILENAME: [jobname].[in|out].covariance.

The compulsory data are:

- ID_ OID - a unique identifier for the corresponding object
- INDEX - the index for the corresponding orbit in the case of multiple orbital solutions
- COVAR_MATR - 6x6 matrix given in lower triangular form (only elements Γ_{ik} with $i \geq k$ are provided with index i increasing first)
- NORMAL_MATR - 6x6 normalization matrix given in lower triangular form (only elements C_{ik} with $i \geq k$ are written with index i increasing first).

One of the reasons for using a multi-line format is the possibility of adding optional data (contained in records beginning with !):

- COV_MOID_MAG - the covariance 2×2 matrix of the signed MOID[Gronchi and Tommei 2007] and of the absolute magnitude.
- RMS - 6-vector standard deviations for each element.
- EIG - 6-vector eigenvalues of the covariance matrix
- WEAK - 6-vector eigenvector corresponding to the largest eigenvalue

Of those optional data above, some may become necessary for some purposes. e.g., COVV_MOID_MAG is essential to assess whether a discovered object is a Potentially Hazardous Asteroid. WEAK defines the weak direction, that is the

[15] A DATE in which the orbit has been computed could be added for some administrative purpose, but this is not yet implemented.

direction along which there has been no differential correction in a Line OF Variations 5-parameter solution. RMS is the most common way to assess the uncertainty, although it contains less information that the full Covariance Matrix [16].

The COVARIANCE is provided as a multiline format using keywords specfied in the header lines, e.g., , as in the OrbFit ML format the keywords are COV, NOR, !CMO !RMS, !EIG, !WEA (optionals with comment character). Given that multi-line formats with keywords are cumbersome to handle (no operation beyond a simple append can be performed with the operating system tools), this data type is the only one written in this way. It is possible to envisage a single-line format (without optional arguments) which would in fact be used as machine readable.

4.5 Residuals

The *RESIDUAL* data type contains information on how well a specific detection fits with one specific orbit solution. Residuals O-C, weights, rejection flags and χ^2 values used to test for outliers are essential ingredients.

RESIDUAL={OID, TIME, OBSERVATORY, RES_RA, RES_DEC, RES_MAG, WEIGHT_RA, WEIGHT_DEC, WEIGHT_APMAG, CHI, SEL_AST, SEL_MAG, [Secret_name] }

FORMAT: SINGLE LINE.

FILENAME: [jobname].[in|out].residual.

- OID - the DETECTION identifier. The value must coincide with that of the corresponding Detection (see the composite data type description for Observation).
- TIME - observation time in UTC MJD in days. The value must coincide with that of the corresponding Detection .
- OBSERVATORY code for the observing site/program (assigned by MPC). The value must coincide with that of the corresponding Detection.
- RES_RA - the residual in RA multiplied by COS(DEC) in arcsec.
- RES_DEC - the residual in declination in arcsec.
- RES_MAG - the residual in apparent magnitude.
- WEIGHT_RA - the RA RMS in arcsec used for weighting in the differential orbit correction (not necessarily the same as the RMS_RA of the detection. It is based upon an error model ([Carpino et al., 2003]) not *a priori* accuracy). The WEIGHT_RA refers to the residual in RA*COS(DEC).
- WEIGHT_DEC - the declination RMS in arcsec used for weighting in the differential orbit correction (not necessarily the same as the RMS_DEC of the detection. It is based upon an error model ([Carpino et al., 2003]) not *a priori* accuracy).
- WEIGHT_APMAG - the magnitude RMS in magnitudes for weighting in the differential orbit correction (not necessarily the same as the RMS_MAG of the detection. It is based upon an error model ([Carpino et al., 2003]) not *a priori* accuracy).
- CHI - the parameter of the statistical test to decide if the residuals are to be rejected in the orbit fit
- SEL_AST - integer indicating whether the detection was used in the orbit fitting. It is > 0 if used, otherwise zero. Values > 1 may indicate some special usage, such as selection for Preliminary Orbit.

[16] $RMS_i = \sqrt{\Gamma_{ii}}$.

- SEL_MAG - integer indicating if the APPMAG was included in the absolute magnitude fit. It is > 0 if used, zero otherwise.
- Secret_name - (optional) an ASCII identifier for the object. It may be either the real name or an identifier for a synthetic detection. The value must coincide with that of the corresponding Detection. The purpose is to assist visual inspection of Residuals, in particular when there are Outliers and/or false Tracklets.

NOTE: OrbFit RES_RA in output is multiplied by COS(DEC) while WEIGHT_RA is not.

NOTE: correlation between RA and DEC should be used if present in the astrometric solution,; this is possible only if this information is contained in the detection format.

4.6 Radar Residuals

The *RADAR_RESIDUAL* data type contains information on how well a specific detection fits with one specific orbit solution. Residuals O-C, weights, rejection flags and χ^2 values used to test for outliers are essential ingredients.

RADAR_RESIDUAL={DESIGNATION, TIME, OBSERVATORY, TYPE, VALUE, WEIGHT, CHI, SEL_AST }

FORMAT: SINGLE LINE.

FILENAME: [jobname].[in|out].radar_residual.

- DESIGNATION - the object identifier (either number or MPC designation).
- TIME - observation time in UTC MJD in days. The value must coincide with that of the corresponding Radar_Astrometry.
- OBSERVATORY code for the observing site/program (assigned by MPC). The value must coincide with that of the corresponding Radar_Astrometry.
- TYPE is R for range, V for range-rate.
- VALUE the residual is expressed in Km for range, Km/day for range-rate.
- WEIGHT used in the fit, is the conversion in the same units used for VALUE of the RMS appearing in the corresponding Radar_Astrometry.
- CHI - the parameter of the statistical test to decide if the residuals are to be rejected in the orbit fit
- SEL_RAD - integer indicating whether the radar data point was used in the orbit fitting. It is 1 if used, otherwise zero.

4.7 Metrics

METRICS are the output of statistical quality control applied to the residuals of a specific orbit solution. Must contain at least the RMS of astrometric and photometric residuals, number of outliers rejected, and other quantities describing systematic signatures in the residuals.

METRICS={ID_OID, INDEX, N_PAR, N_REJ, RMS, RMSH, [MAX_REJ, ZSIGN_RA, CURV_RA, SPAN_RA, BIAS_RA, ZSIGN_DEC, CURV_DEC, SPAN_DEC, BIAS_DEC]}

FORMAT: SINGLE LINE.

FILENAME: [jobname].[in|out].metrics.

- ID_OID - the object identified. It must be the same as in the ORBIT and IDENT_HEADER.
- INDEX - the orbit index in the case of multiple orbits for an object. It must be the same as in the ORBIT and IDENT_HEADER.
- N_PAR - number of free parameters in the fit (may be 4,5,6)
- N_REJ - number of outlier rejections (which do not contribute to the RMS of the fit).
- RMS - weighted RMS of the astrometric residuals,
- RMSH - weighted RMS of the photometric residuals.
- MAX_REJ - fraction of observations discarded from a single night (e.g., 1.0 means the entire night of detections has been discarded)
- ZSIGN_RA non-dimensional third derivative signature present in the residuals in RA normalized by dividing by the RMS_RA
- CURV_RA - non-dimensional curvature present in the residuals in RA normalized by dividing by the RMS_RA
- SPAN_RA - non-dimensional linear trend in the RA residuals normalized by dividing by the RMS_RA
- BIAS_RA - non-dimensional constant term in the RA residuals normalized by dividing by the RMS_RA
- ZSIGN_DEC non-dimensional third derivative signature present in the residuals in DEC normalized by dividing by the RMS_DEC
- CURV_DEC - non-dimensional curvature present in the residuals in DEC normalized by dividing by the RMS_DEC
- SPAN_DEC - non-dimensional linear trend in the DEC residuals normalized by dividing by the RMS_DEC
- BIAS_DEC - non-dimensional constant term in the DEC residuals normalized by dividing by RMS_DEC

4.8 Identification headers

IDENT_HEADER is a summary of the properties of a proposed or confirmed identification. It may be used as a basic data type to be included both in a TRACK and in an IDENTIFICATION. It must contain a list of the included tracklets and counters/parameters describing either the results obtained, e.g., , the number of Solutions, or the effort to be applied in the attempt to compute an orbit, e.g., , the maximum RMS of a Preliminary Orbit to be used as starting point for a Least Squares Fit.

IDENT_HEADER={ID_OID, NID, TRACKLET_OIDs, OP_CODE, N_OBS, N_RADAR N_SOLUTIONS, N_NIGHTS, ARC_TYPE [PARAMETERS]}

FORMAT: SINGLE LINE.

FILENAME:

[jobname].out.ident_header when it belongs to an IDENTIFICATION confirmed by an orbit computation

[jobname].in.request when it belongs to a TRACK.

- ID_OID - unique identifier for the identification (track).
- NID - number of Tracklets in the track.
- TRACKLET_OIDs - NID length list of Tracklet object identifiers that form this Identification (Track).
- OP_CODE - character code for a claim/request on this identification such as: DISCOVERY, COMPLETION, KNOWN_OBJ, WRONG_ID, POSSIBLE_ID, REQUEST_OBSERVATION, REQUEST_PRELIM, REQUEST_ORBIT, REQUEST_PRECOVERY (these OP_CODEs are explained below.).
- N_OBS - number of optical astrometry observations in the Track (including the ones rejected in the fit)
- N_RADAR - number of Radar_Astrometry data in the fit, 0 if there are none (as in most cases).
- N_SOLUTIONS - number of alternate Solutions for which the orbits/covariance/metrics are returned
- N_NIGHTS - number of separate nights of observations
- ARC_TYPE - the arc type [Milani et al. 2006]
- PARAMETERS - a set of 4 real parameters, used in different ways depending upon the context (see below under Track, Identification, and Orphan Request).

When the OP_CODE is WRONG_ID the IDENT_HEADER is sent alone (the data are replaced by those of the new identifications which have contradicted the removed one). More details are provided in the description of the composites TRACK and IDENTIFICATION data type in section 5.4.

4.9 Pointers from Detections to Tracklets

TRACKLET_POINTER is the output of an algorithm that links detections into tracklets.

TRACKLET_POINTER={TRACKLET_OID, DETECTION_OID }

FORMAT: SINGLE LINE

FILENAME: [jobname].[in|out].tracklet_pointer.

- TRACKLET_OID - tracklet identifier to which the detection belongs,
- DETECTION_OID - detection identifier.

Note that *a priori* the relationship between Detections and Tracklets is neither univalued nor one-to-one, although it is a goal of *tracklet management* to reduce the set of Tracklets in such a way that the relationship becomes univalued as much as possible.

Using TRACKLET_POINTER and DETECTION files it is easy to generate a TRACKLETs file and an ORPHANs file, with the optional generation of an OVERLAPPING_TRACKLETS file (for tracklet management).

4.10 Pointers from Tracklets to Tracks

TRACK_POINTER is the output of an algorithm that links tracklets into identifications.

TRACK_POINTER ={ TRACK_OID, TRACKLET_OID }

FORMAT: SINGLE LINE.

FILENAME: [jobname].[in|out].pointer_identification.

- TRACK_OID - track identifier to which the tracklet belongs,
- TRACKLET_OID - tracklet identifier.

Note that *a priori* the relationship between tracklets and identifications is neither univalued nor injective, although it is a goal of *identification management* to reduce the set of identifications in such a way that the relationship becomes univalued as much as possible.

It is not necessary for Tracks to have a formal OID, because they are not inserted in a formal database (given the sometimes large overhead, that is, most of them are false and are discarded in the following stage of processing). However, some identifier is needed to be written in the file, and it is easier to maintain consistency if it is formed in some automated way, like A=B=C for the Track consisting of Tracklets A, B and C; note that to have a uniquely determined automatic Track OID, the tracklet OIDs A,B,C need to be sorted[17]. This is not formally mandated by the standard, but is recommended.

4.11 Ephemerides

EPHEMERIS={ID_OID, INDEX, TIME, ATTRIBUTABLE, APPMAG, RMSVAL, CORREL, RMSMAG)

FORMAT: SINGLE LINE

FILENAME: [jobname].[in|out].ephemeris.

- ID_OID - object identifier
- INDEX - Solution number
- TIME - time of the observations is MJD UTC in days.
- ATTRIBUTABLE={RA, DEC, RADOT, DECDOT} with RA and DEC (the position) in deg and RADOT, DECDOT (the angular velocities) in deg/day.
- APPMAG - apparent V magnitude
- RMSVAL={RMS_RA, RMS_DEC, RMS_RADOT, RMS_DECDOT} - 4-vector of standard deviations for the 4 components of ATTRIBUTABLE.
- CORREL - the correlation matrix. A lower triangular 4x4 matrix, with 6 numbers between -1 and +1 and row index growing first.

[17]If the tracklet OIDs are strings, they have to be sorted in lexicographic order. This is more consistent, since the automatic track OIDs can also be sorted in lexicographic order. However, as long as the tracklet OIDs are allinteger numbers, numeric sorting is also possible.

- RMSMAG - formal standard deviation in the predicted apparent magnitude (WARNING: the observer needs to leave margins for lightcurve).

An *EPHEMERIS* is a prediction for the astrometry and photometry of an object for which an orbit is available. It also contains a description of the uncertainty through the covariance formalism described above (section 4.4).

NOTE: beware of the mixing of scalar, vectorial and matrix variables.

4.12 Image meta-data

IMAGE_METADATA is image specific data (as opposed to detection, or source, specific data), including but not limited to: image time, center, corners, filter, limiting magnitude, exposure duration, seeing and other meteorological data, estimate of the fill factor, etc.

IMAGE_METADATA (Not specified in this standard)

FORMAT: SINGLE LINE

FILENAME: [jobname].[in|out].image_metadata

5 Composite Data Types

Composite data types are obtained by combining basic types.

5.1 Tracklets

TRACKLET is a set of detections that may belong to the same moving object because of proximity and approximate alignment in the images (not because an orbit has been fit to them). The expression Very Short Arc (VSA) is also used [Milani et al. 2004]. A single file may contain many different tracklets with sequential lines of detections belonging to successive tracklets.

TRACKLET={DETECTIONs}

FORMAT: SINGLE LINE

FILENAME: [jobname].[in|out].tracklet

A Tracklet is a different object from its component detections and has a different OID. If the DETECTION OID is the same in multiple lines it is the TRACKLET_OID unique to the tracklet[18]. The tracklet ends when the OID changes.

As for Tracks, it might be useful to define an automated Tracklet OID, which would be uniquely determined by the content of the Tracklet; however, to be sure of the uniqueness, even for large number densities per square degree, the coded string would have to be quite long.

NOTE: a single Detection has a DETECTION_OID; if it is necessary to fit an orbit with an individual Detection (an Orphan) then it is necessary for consistency to compose a Tracklet containing a single Detection, with a TRACKLET_OID which is NOT the same as the DETECTION_OID. If this rule was violated confusion might arise with another Tracklet having the same OID.

[18] OrbFit currently implements a 9 character alphanumeric limit on the TRACKLET_OID.

5.2 Observations

OBSERVATION consists of detections for which the hypothesis that it belongs to a moving object has passed some quality control criteria:

OBSERVATION={DETECTION, RESIDUAL}

FILENAME:

[jobname].tracklet (containing DETECTIONs) AND

[jobname].residual (containing RESIDUALs).

The two files must have the same number of records for each Identification and the ordering must be preserved. To avoid ordering problems Observations must be reported in increasing time order. The case of two Observations at the same time should not happen[19].

A complicating issue is that there is not a one-to-one correspondence between all Detections, all Observations and all Detections included in Tracklets. This is due to the possibility of two Tracklets having some Detections in common, and also to the possibility of False tracklets containing individual Detections to be removed as Outliers.

5.3 Radar observations

RADAR_OBSERVATION={RADAR_ASTROMETRY, RADAR_RESIDUAL}

FILENAME:

[jobname].radar_astrometry AND [jobname].radar_residual.

The .radar_ astrometry file contains the JPL format; the two files must have the same number of records for each Asteroid and the ordering must be preserved. To avoid ordering problems Radar Observations must be reported in increasing time order.

5.4 Identifications

Identification is a way to link together observations believed to belong to the same moving object. However, this can be done only through the intermediary of tracklets. Thus the essential element of an identification is an IDENT_HEADER providing a list of tracklets.

Unfortunately, the word identification is used in slightly different meaning by different authors. Thus we need to give very specific definitions, each corresponding to a different data type.

5.4.1 Proposed Identifications

A Proposed Identification or Track (terminology used internally by Pan-STARRS) consists only of a list of Tracklets.

TRACK= {IDENT_HEADER, TRACKLETs}

[19]If there are two Observations at the same time, and they are not simply duplicates, then they are discordant and one of the two should be an Outlier.

with N_SOLUTIONS=0 in the IDENT_HEADER[20].

5.4.2 Preliminary Orbit Identifications

When a Preliminary Orbit is calculated only the data type ORBIT is available, together with the IDENT_HEADER.

5.4.3 Least Squares Orbit Identifications

If a Least Squares Orbit has been calculated then it includes significant ancillary information forming a *SOLUTION*, with Covariance, Residuals, and Metrics:

SOLUTION={ORBIT, COVARIANCE, METRIC, OBSERVATIONs}

Solutions are called Derived Objects by Pan-STARRS. Unfortunately, it is not possible to exclude that for one proposed identification (that is, starting from a TRACK data type in input) multiple Solutions can be identified for each TRACK. This happens with double Least Squares Orbits in the 'Sweet Spots' at small solar elongation (starting differential corrections from double preliminary orbits, [Boattini et al. 2006]) and with multiple line-of-variation solutions [Milani et al. 2005a] when the orbit is only weakly determined. We use the data type IDENTIFICATION to indicate a confirmed Identification, with a positive number of SOLUTIONS.

IDENTIFICATION={IDENT_HEADER, SOLUTIONs}

The OP_CODE should be one of the following: DISCOVERY, COMPLETION, KNOWN_OBJ, POSSIBLE_ID, TSA_JOIN.

WARNING: it might appear useless to output the tracklet [jobname].out.tracklet file from an orbit computation since it only contains information contained in the file [jobname].in.tracklet. However, the order in which the records are written contains essential information. The tracklets in input could each be used in different identifications. Two tracklets may have an observation in common. Thus, the list of records contained in the .in.tracklets file and those in the .out.tracklets files are totally different. The section of the .out.tracklet file and of the .out.residual file corresponding to the same SOLUTION must have the same number of lines in the same order. A cross check is provided by duplicating the TRACKLET_OID, TIME, and OBSERVATORY.

NOTE: the PARAMETERS field in the IDENT_HEADER is used to pass information between programs and their meaning depends upon the context. See Section 7.

5.5 Schedule of observations

A *SCHEDULE* allows publication of a set of fields that were (or might be) acquired.

SCHEDULE=IMAGE_METADATAs

FILENAME: [jobname].[in|out].schedule.

6 Exported Data Products

Exported data products may be of the following categories:

[20]The choice of the Track OID has been discussed in Section 4.10.

1. DISCOVERY
2. PRIORITY
3. ATTRIB_KNOWN
4. ID_CORRECTION
5. UNIDENTIFIED
6. FOLLOW_UP
7. SKY_COVERAGE

These categories are described in detail in the following sub-sections.

6.1 Normalization

To understand the output data products of a survey it is not enough to consider each individual "discovery" in isolation. The output shall consist of list of composite data types, but these lists have to satisfy syntactical rules which are meaningful only when considering the output globally. These syntactic rules are generically indicated with the word *normalization*. As the simplest example, a set of tracklets is *weakly normalized* if there are no duplications (two Tracklets with the same detections) and *strongly normalized* if there are no *Discordant Tracklets* with some (not all) Detections in common. Which of the two is adopted depends upon the requirement for publication. The process by which these normalizations are performed is *Tracklet Management*.

For Identifications there are three levels of normalization. *Weak Normalization* removes duplicates, that is Identifications with the same list of Tracklets, possibly leaving multiple Solutions if the Orbits are not compatible (as measured by using the Covariances). *Medium Normalization* as described in [Milani et al. 2005b, Milani et al. 2006] also removes Discordant Identifications (sharing some but not all Tracklets). *Strong Normalization* requires also that for each Identification there is only one acceptable Solution.

6.2 Discovery claims

The definition of Discovery is the subject of a formal discussion within Division III of the IAU. For the moment, we adopt the proposed definition of [Milani et al. 2006].

DISCOVERY CLAIM: a list of Identifications with Strong Normalization (neither discordancies nor multiple Solutions are allowed). Only nominal least squares orbits with 6 parameters are allowed. The Observations must form Arcs of Type > 2. In most cases this requirement corresponds to obtaining at least 3 tracklets at opposition and 4 at the sweet spots.

In order that an IDENTIFICATION be claimed as a discovery it must satisfy the following constraints: in the IDENT_HEADER, OP_CODE=DISCOVERY, ARC_TYPE>2, N_SOL$=1$, single ORBIT, with N_PAR$=6$ and only one METRIC with "good" values that pass some Quality Assurance standard (yet to be adopted).

6.3 Priority claim

PRIORITY CLAIM: a list of Identifications which is *weakly normalized* by removing only duplications and Identifications with tracklets in common which are inferior (linking less nights). Discordancies (between identifications with the same number of nights) and alternate orbit solutions (as in the sweet spots) are allowed and each one qualifies for Priority if and when the Discovery is completed by follow up observations. In practice, most of the Priority claims will be for observations with Arcs of Type 2 (mostly 2-nighters, plus 3-nighters TNO) but also for discordant Type 3 observations at opposition and Type 3 observations with double solutions at the sweet spots [Boattini et al. 2006].

The data to present an incomplete discovery, to be completed by others, are of type IDENTIFICATION with, in the IDENT_HEADER, OP_CODE=POSSIBLE_ID or RESERVE_DISC[21].

6.4 Attribution to previously known object

Observations of known objects: a list of Tracklets with Attribution [Milani et al. 2001] to some already Discovered objects. In some cases this completes a Priority Claim to a Discovery claim and this has to be specified.

The data to propose an attribution to a known object are of type IDENTIFICATION with OP_CODE=KNOWN_OBJ or COMPLETION (of a discovery) in the IDENT_HEADER.

6.5 Correction of a previous identification

WRONG_ID: a list of previous Identifications later found to be incorrect. This happens as a result of new observations that result in an Identification which is discordant and Superior (e.g., more tracklets identified) to one already submitted.

The corrections to previous identifications are of type IDENT_HEAD, with OP_CODE=WRONG_ID.

6.6 Unidentified tracklets

Unidentified tracklets: a list of tracklets, or identifications among tracklets of the same night, or 2-night ids of distant objects, for which no identification with other tracklets has been possible and the Arc Type is still 1 (no significant curvature and no meaningful orbit).

The unidentified tracklets file is of type TRACKLETs containing lines of DETECTIONS with changes in TRACKLET_OID indicating a new tracklet. An IDENT_HEADER file is provided with one record for each identification and OP_CODE=TSA_JOIN, ARC_TYPE= 1.

6.7 Requests for observations

For interesting objects which have an incomplete orbit a request for follow-up could be sent to an observer or broadcast to a wider audience (e.g., , to MPML). As an alternative, the future PS-DyS site could be used to provide the same data with a user-friendly web interface.

The composite data type contains the basic data to uniquely select an object and the predictions for a time series of potential observing times.

REQUEST_OBS={IDENT_HEAD, EPHEMERs}

with the OP_CODE=REQUEST_OBSERVATION. File names should be [jobname].in.request_obs, [jobname].in.ephemeris.

[21] This applies only if the proposed IAU rules 11 and 12 for discovery credit, which allow the option of *reserving a discovery* without supplying a complete data set qualifying as a discovery, are approved and implemented.

7 Transactions

This is a list of other mostly internal cases of data exchange currently in use in the the exchange of data between MOPS, OrbFit and LSST software. They could be useful for external data exchange when providing and/or receiving requests from users or collaborators.

7.1 Preliminary/final orbit request and return

ORBIT_REQUEST={IDENT_HEADER, TRACKLETs}

where each orbit request should correspond to ≥ 3 tracklets. In IDENT_HEADER the OP_CODE may be either REQ_PRELIM (for preliminary orbit only) or REQ_ORBIT (which also attempts a differential correction).

The PARAMETERS of the IDENT_HEADER can be used to specify control parameters.

PARAMETERS= RMS_PRELIM, RMS_ORBIT, MAXX_REJ, MAX_CSB

This forces values for the controls RMS $<$ RMS_PRELIM for the preliminary orbit, RMS $<$ RMS_ORBIT for the least squares orbit, MAX_REJ $<$ MAXX_REJ for the maximum rate of rejection, MAX(CURV_RA, CURV_DEC, SPAN_RA, SPAN_DEC, BIAS_RA, BIAS_DEC) $<$ MAX_CSB to control Curvature, Span and Bias (see Section 4.7).

The return data are of type IDENTIFICATION for REQUEST_ORBIT; the PARAMETERS in the IDENT_HEADER contain the RMS of the fit for the SOLUTIONs. For REQUEST_PRELIM the output contains only the IDENT_HEADER (with the RMS of the 2-body Solution(s) obtained as Preliminary Orbit(a) in the PARAMETERS) and the ORBIT file.

If the orbit computation is succesful the OP_CODE of the .out.ident_header file is either PRELIM or ORBIT; if the orbit computation is not succesful the values of either REQUEST_PRELIM or REQUEST_ORBIT from the input file are copied to the output file. This allows the option of using the .out.ident_header file as a .in.request file without modification to attempt a more robust orbit computation in a second iteration.

7.2 Orphan requests

An orphaned detection is a detection that is not within a tracklet. This may be due to e.g., the CCD mosaic missing a detection (fill factor < 1) or the detection being of too low S/N to pass the threshold for inclusion in a tracklet. The purpose of an 'orphan request' is to identify and return orphan DETECTIONs. The same format is used to request a precovery DETECTION.

PRECOVERY_REQUEST={IDENTIFICATION}

where in the IDENT_HEADER the OP_CODE should be REQUEST_PRECOVERY. The PARAMETERS field in the IDENT_HEADER is compulsory and is the of the form:

PARAMETERS={CHI_TRACKLET, S2N_TRACKLET, CHI_SINGLE, S2N_SINGLE}

where the four real numbers are the chi value (square root of the identification penalty parameter K) and the signal to noise ratio [Milani et al. 2001] for each of the two cases (TRACKLET or SINGLE detection). If only tracklets are requested then the two values referring to singles must be zero and viceversa.

The IDENT_HEADER file is called [jobname].precovery_request.

The response is

PRECOVERY={IDENT_HEADER, TRACKLETs}

where IDENT_HEADER is updated to describe the results[22]. Note that for singles the answer contains tracklets with only one observation but the OID must be a TRACKLET_OID (i.e., a 1-detection tracklet).

References

[Carpino et al., 2003] Carpino, M., Milani, A., Chesley, S. R., 2003. Error Statistics of Asteroid Optical Astrometric Observations. *Icarus* **166**, 248–270.

[Gronchi and Tommei 2007] Gronchi, G.F. and Tommei, G., 2007. On the uncertainty of the minimal distance between two confocal Keplerian orbits, Discrete Continous Dynamical Systems, Series B, **7**, 755–778.

[Kubica et al. 2007] Kubica, J., L. Denneau, T. Grav, J. Heasley, R. Jedicke, J. Masiero, A. Milani, A. Moore, D. Tholen, R.J. Wainscoat, 2007. Efficient intra- and inter-night linking of asteroid detections using kd-trees. Icarus, in press.

[Milani et al. 2001] Milani, A., Sansaturio, M. E., Chesley, S. R., 2001. The Asteroid Identification Problem IV: Attributions. *Icarus* **151**, 150–159.

[Milani et al. 2004] Milani, A., Gronchi, G. F., de' Michieli Vitturi, M., Knežević, Z., 2004. Orbit Determination with Very Short Arcs. I Admissible Regions. CMDA 90, 59–87.

[Milani et al. 2005a] Milani, A., Sansaturio, M.E., Tommei, G., Arratia, O. and Chesley, S.R. 2005a, Multiple solutions for asteroid orbits: computational procedure and applications, *Astron. Astrophys.* 431, 729–746.

[Milani et al. 2005b] Milani, A., Gronchi, G.F., Knežević, Z., Sansaturio, M.E. and Arratia, O. 2005, Orbit Determination with Very Short Arcs. II Identifications, *Icarus*, **79**, 350–374.

[Milani et al. 2006] Milani A. et al. 2006, Unbiased orbit determination for the next generation asteroid/comet surveys, in *Asteroids Comets Meteors 2005*, D. Lazzaro et al., eds., Cambridge University Press, pp. 367–380.

[Boattini et al. 2006] Boattini, A., Milani, A., Gonchi, G.F., Spahr, T, and Valsecchi, G.B. 2006, Low solar elongation searches for NEOs: a deep sky test and its implications for survey strategies, in *IAUS 236 Proceedings*, Milani et al. eds., in press.

[Milani et al. 2006] Milani, A., Gronchi, G.F. and Knežević, Z. 2006, New Definition of Discovery for Solar System Objects *EMP*, 100, 83-116.

[Ullman 1991] Ullman, J. D. 1991, *Principles of Databases and Knowledge-Base Systems, Vol. I*, Computer Scienvce Press.

[22]The specifications for this transaction are not yet complete, because no simulation has been compelted yet.