



MEMORANDUM

TO: NOP, Astros
FROM: R. Savalle, P. Martin
DATE: October 25, 2001
SUBJECT: Design for Moving Fields in QSO

QSO-015/Version 1.7/FINAL

Latest version available at:

http://www.cfht.hawaii.edu/LocalAccess/QSO/Memos/QSO-015-MovingFields_current.pdf

1 History

Version	Date	Comments
1.0	October 9, 2001	Initial version
1.1	October 16, 2001	Sent to PM for corrections
1.2	October 19, 2001	Corrections by PM
1.3	October 19, 2001	Incorporated corrections
1.4	October 22, 2001	New Overview Schema, sent to CV for corrections
1.5	October 23, 2001	Distributed before Design Review Meeting
1.6	October 25, 2001	Corrections before Design Review Meeting
1.7	October 29, 2001	Corrections after Design Review Meeting (additions prefixed with ❖)

2	<i>Contents</i>	
1	<i>History</i>	1
2	<i>Contents</i>	2
3	<i>Context</i>	3
4	<i>The Big Picture</i>	3
4.1	Constraints	3
4.2	Overview	3
5	<i>PH2 Modifications</i>	4
5.1	Program Constraints	4
5.2	New page: Moving Targets defined by Orbital Parameters	5
5.2.1	Orbital Elements	5
5.2.2	Screenshot of current design	6
5.2.3	Features	7
5.2.4	AstroRes format for Orbital Parameters (DRAFT)	8
5.3	New page: Moving Targets defined by Ephemeris	10
5.3.1	Features	10
5.4	OB form	11
5.5	OG Form	12
5.5.1	Validation	12
5.5.2	Implementation	13
6	<i>Database Organization</i>	13
6.1	Structure	13
6.1.1	Ph2 database	13
6.1.2	Operational Database	14
6.2	Queue Preparation and Queries Strategy	15
6.3	Observations	16
7	<i>QSO Tools and GUI implications</i>	16
8	<i>Issues in the QSO system</i>	16
8.1	Photometry	16
9	<i>TCS Modifications</i>	17
9.1	TCS	17
10	<i>NEO Modifications</i>	18
10.1	TCS agent	18
11	<i>Schedule</i>	18

3 Context

The possibility to observe moving fields, that is, fields containing objects with changing coordinates over short periods of time (e.g. TNO, comets), has been planned since the beginning of the QSO project. Due to time constraints, this has not been implemented yet. But, it is important to do so for the semester 2002A to at least identify all the issues with this particular observing mode with CFH12K before applying it during the MegaPrime era. This memo discusses the final design and implementation for handling moving objects within the QSO project. It also discusses the needs and implications on other projects, in particular NEO and TCS.

4 The Big Picture

4.1 Constraints

We propose to allow the observations of moving objects in QSO mode with the following restrictions

- Moving Objects must be defined by either
 - A name of a known object following MPC nomenclature rules
 - A set of orbital parameters
 - An ephemeris suitable for observation
 - Planets (but not the Moon nor planet satellites)

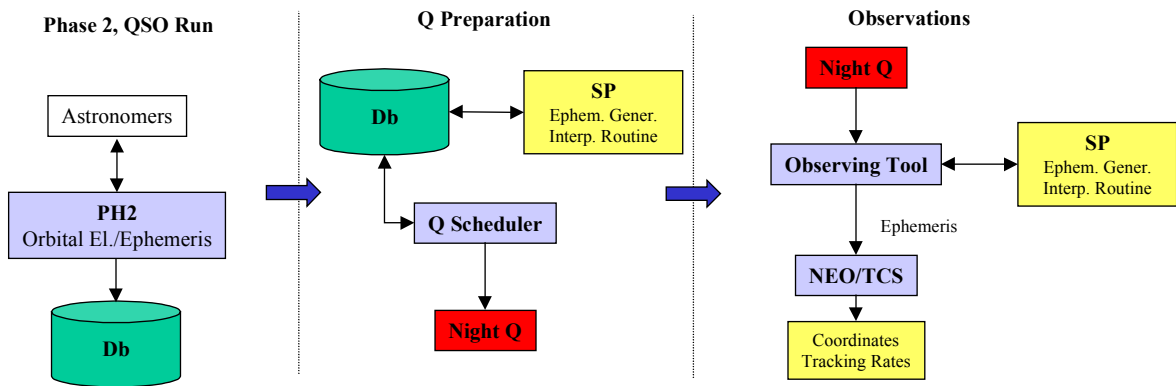
- At the observation time, the moving objects should not be faster than the maximum differential tracking rates allowed by the telescope without losing guiding accuracy.

- Photometry of Moving Objects may be restricted (See 8.1)

4.2 Overview

The general idea behind queue observing of moving fields is displayed in the figure below. Details are given in the following sections. The main differences with the normal QSO mode are:

1. Strong interaction with the PI during an observing run for refining coordinates/orbital elements
2. Accurate coordinates are calculated by the QSO system during queue preparation and by TCS (including tracking rates) during the observations.



5 PH2 Modifications

Several modifications must be implemented within the Phase 2 Tool (PH2) to allow users to specify the necessary information for observing moving targets. A section in PH2 will be developed for this purpose, and will include the typical forms common to PH2.

The *validation* of the entered data remains a key point in PH2. We will see that moving objects cause non-trivial validation issues, most of which happen at the OG definition level.

5.1 Program Constraints

- New option on the program constraint page: **“Does you program include moving fields?”** This flag is then used to display/allow moving object PH2 forms and features.

5.2 New page: Moving Targets defined by Orbital Parameters

5.2.1 Orbital Elements

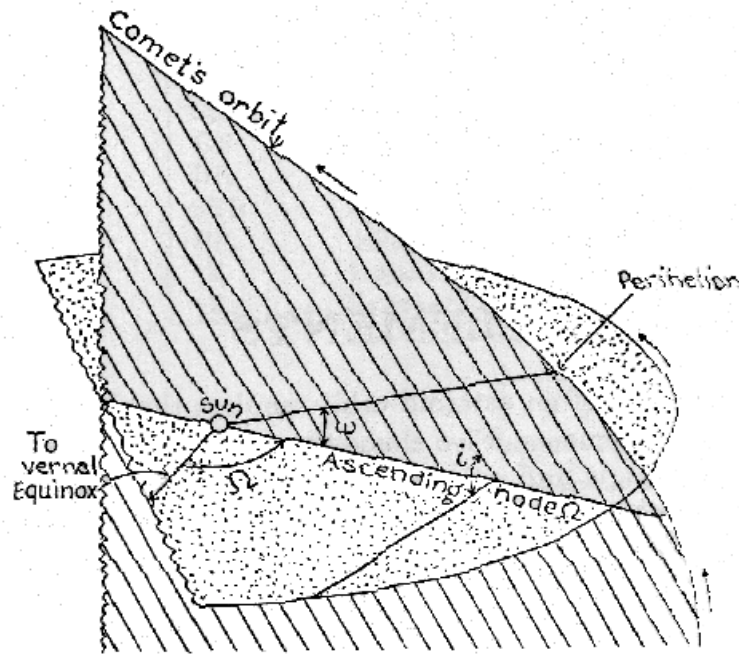


Figure 1: Orbital Elements

In order to define a moving object we need 6 numbers called the Orbital Elements.

□ The first 5 elements define the elliptic orbit:

1. Semi-major axis or the orbit a
2. Eccentricity e
3. Inclination i
4. Longitude of Ascending Node Ω
5. Longitude of Perihelion Π
 - NB: Some authors (and the figure above) use the *argument* of perihelion ω : $\omega = \Pi - \Omega$

Since those numbers are defined in relation with the vernal equinox γ , which moves because of the precession of the equinox, they are defined in J2000.0.

□ Two other parameters are needed to locate the object on its orbit:

1. Mean longitude L (defines the position of the object at the Epoch of Osculation)
 - NB: Some authors use the mean anomaly M : $M = L - \Pi$
2. Epoch of osculation: the date for which L is given

5.2.2 Screenshot of current design

RUNID: 01BE99 PI: rsavalle Rows OT1 to OT5 of 7 Moving Targets defined by Ort

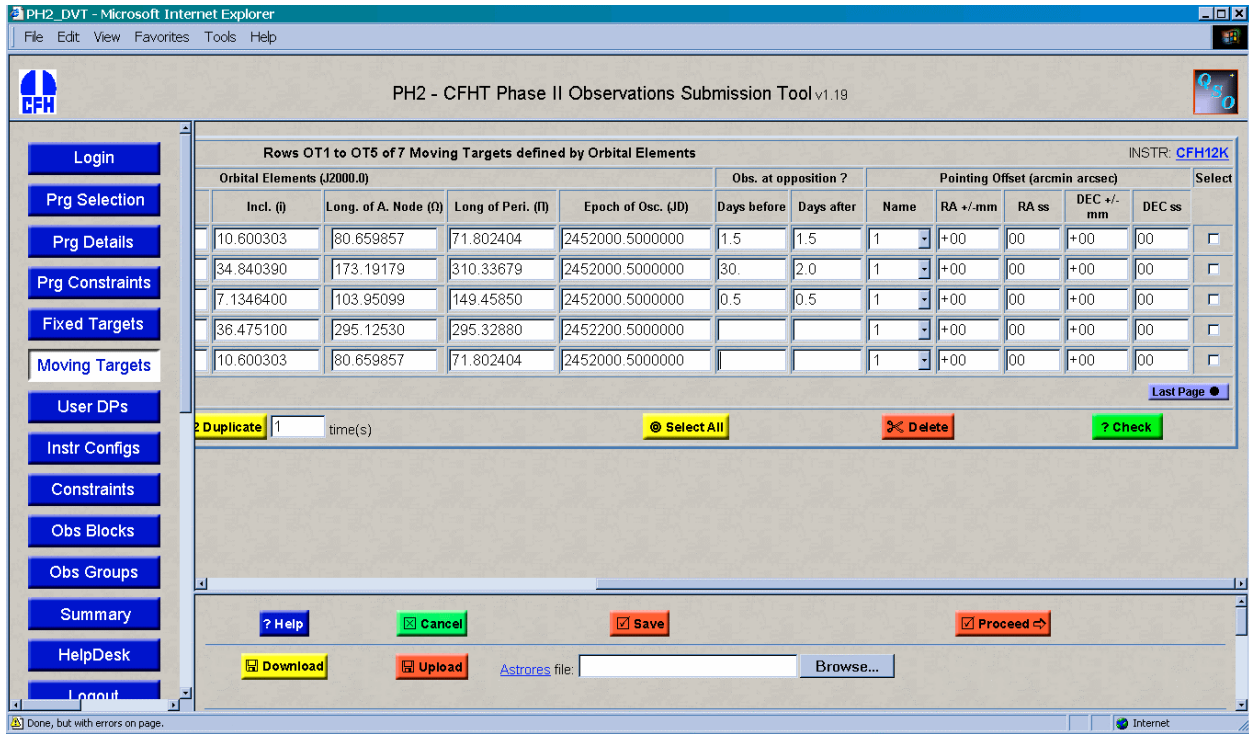
Label	Name	Moving Object Resolver		Orbital Elements (J2000.0)						
		Launch	Grab	Mean Long (L)	SM Axis (a)	Eccent. (e)	Incl. (i)	Long. of A. Node (Ω)	Long of Peri. (Π)	Epoch
OT1	Ceres2	MOR	Grab ⇌	80.477333	2.7678871	0.0760410	10.600303	80.659857	71.802404	245200
OT2	2 Pallas (MPC)	MOR	Grab ⇌	90.123650	2.7733539	0.2298490	34.840390	173.19179	310.33679	245200
OT3	4 Vesta (MPC)	MOR	Grab ⇌	104.89803	2.3607751	0.0897529	7.1346400	103.95099	149.45850	245200
OT4	C/2001 A2 (LINE)	MOR	Grab ⇌	0.0000000	175.08810	0.9993370	36.475100	295.12530	295.32880	245220
OT5	Ceres2	MOR	Grab ⇌	80.477333	2.7678871	0.0760410	10.600303	80.659857	71.802404	245200

OT1 OT5

Add 1 row(s) **x2 Duplicate** 1 time(s)

? Help Cancel Save Proceed ⇌

Download Upload Astrores file: Browse...



5.2.3 Features

- Allows PI to enter targets defined by orbital parameters.
 - Name. Should be compliant with MPC nomenclature to allow the MOR and AUTO feature described below.
 - “MOR” feature: links to the Moving Object Resolver (MOR): allows to populate the elements from the name, using a table of moving objects in the ph2 database kept in sync with data available from Lowell Observatory ASTORB.DAT file¹ or the MPC Minor Planet Center Orbit Database (MPCORB)².
 - “AUTO” checkbox: OK to fetch the latest elements to from our updated moving object table during Q preparation and observation. The object name must be compliant with the MPC nomenclature.
 - ❖ Need for displaying the auto-resolved object name separately from the user name.
 - ❖ Possible to make entry boxes read-only if object auto-resolved ?
 - Six Orbital Elements as described above + Epoch of osculation [Can be preset by the MOR]

¹ <ftp://ftp.lowell.edu/pub/elgb/astorb.html> <ftp://ftp.lowell.edu/pub/elgb/astorb.dat.gz> updated daily

² <ftp://cfa-ftp.harvard.edu/pub/MPCORB/> updated daily

- [OPTIONAL] Most objects being observed at opposition (when the object as seen from the Earth is opposite from the Sun, thus having a “full” phase and being brighter) we allow the PI to force observation of a moving object around opposition. 2 parameters define a time window for observation around opposition:
 1. Days before opposition (maximum **TBD**)
 2. Days after opposition (maximum **TBD**)
 - Pointing. As with Fixed Target, we allow the PI to position the object accurately on the CCD mosaic.
- Row manipulation (as in existing forms)
 - ⇒ Allows access to refine parameters. Currently OK if object does not belong to a validated observation. We need to allow copy object in all cases (currently not possible!)
 - Validation rules:
 - Orbital elements:
 - Angles \perp , \perp , Ω , Π are checked to be between 0 and 360 degrees
 - Eccentricity e checked to be > 0
NB: elliptic objects like asteroids have $0 < e < 1$ but hyperbolic comets have $e > 1.0$!
 - ❖ NB: no limits on Epoch of osculation
 - Objects can be downloaded and uploaded using the XML-compliant Astrores format.
 - Supply programs to convert popular orbital elements format into AstroRes ?
 - MPC 1 line
 - MPC 8 lines
 - Others?

5.2.4 *AstroRes format for Orbital Parameters (DRAFT)*


```

<?xml version = "1.0"?>
<!DOCTYPE ASTRO SYSTEM "http://vizier.u-strasbg.fr/xml/astrores.dtd">
<ASTRO ID="v0.8" xmlns:ASTRO="http://vizier.u-strasbg.fr/doc/astrores.htx">
  <TABLE ID="Table">
    <NAME>Moving Targets defined by Orbital Elements</NAME>
    <TITLE>Moving Targets defined by Orbital Elements for CFHT QSO</TITLE>

    <!-- Definition of each field -->
    <FIELD name="NAME" datatype="A" width="20">
      <DESCRIPTION>Name of target</DESCRIPTION>
    </FIELD>
    <FIELD name="LONG_MIN" datatype="D" width="9" unit="deg">
      <DESCRIPTION>Mean Longitude</DESCRIPTION>
    </FIELD>
    <FIELD name="SMAJ_AXIS" datatype="D" width="9" unit="AU">
      <DESCRIPTION>Semi-major axis</DESCRIPTION>
    </FIELD>
    <FIELD name="ECCENT" datatype="D" width="9" >
      <DESCRIPTION>Eccentricity</DESCRIPTION>
    </FIELD>
    <FIELD name="INCLIN" datatype="D" width="9" unit="deg">
      <DESCRIPTION>Inclination</DESCRIPTION>
    </FIELD>
    <FIELD name="LONG_ASCN" datatype="D" width="9" unit="deg">
      <DESCRIPTION>Longitude of ascending node</DESCRIPTION>
    </FIELD>
    <FIELD name="LONG_PERI" datatype="D" width="9" unit="deg">
      <DESCRIPTION>Longitude of perihelion</DESCRIPTION>
    </FIELD>
    <FIELD name="JDATE_OSC" datatype="D" width="15" precision="7" unit="d" format="JD">
      <DESCRIPTION>Julian date for orbital elements</DESCRIPTION>
    </FIELD>
    <FIELD name="ELONG_MIN" datatype="D" width="9" unit="deg">
      <DESCRIPTION>Minimum Elongation</DESCRIPTION>
    </FIELD>
    <FIELD name="ELONG_MAX" datatype="D" width="9" unit="deg">
      <DESCRIPTION>Maximum Elongation</DESCRIPTION>
    </FIELD>
    <FIELD name="POINT" datatype="A" width="5">
      <DESCRIPTION>Pointing name</DESCRIPTION>
    </FIELD>

    <!-- Data table -->
    <DATA><CSV headlines="4" colsep="|">
    <![CDATA[
NAME          |LONG_MEAN|SMAJ_AXIS|ECCENT  |INCLIN  |LONG_ASCN|LONG_PERI|JDATE_OSC  |ELONG_MIN|ELONG_MAX|POINT|
12345678901234567890|123456789|123456789|123456789|123456789|123456789|123456789|123456789012345|123456789|123456789|12345|
-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
]]></CSV></DATA>
  </TABLE>
</ASTRO>

```

Figure 2: Astrores Format for Orbital Elements

5.3 New page: Moving Targets defined by Ephemeris

Newly discovered moving objects are not yet known well enough to have orbital parameters. This is the responsibility of the PI to provide, and possibly update, ephemeris for those types of objects.

5.3.1 Features

- Allows PI to enter targets defined by ephemeris:
 - For the object
 - Name

- NB: No auto flag, we do not plan to generate ephemeris but rely on PI interaction.
- NB: No min/max elongation, time constraint has to be specified in the OG form.
- Pointing
- Ephemeris lines, for each line:
 - Date (Julian Day)
 - RA and DEC: the position of the object at the date.
 - Epoch: Even if the coordinates (RA, DEC) are the position at the date, they are usually expressed in Epoch = J2000.0.
- ❖ OK to make it mandatory for the users to use the J2000.0 epoch.
- Row manipulation (as in existing forms)
 - ⇒ Allows access to refine parameters. Currently OK if object does not belong to a validated observation. We need to allow copy object in all cases (currently not possible!)
- Validation rules
 - ❖ The ephemeris must be a continuous table, and sorted by time upon saving (NB: this removes the need for an “INSERT” feature in the table).
- Ephemeris can be downloaded and uploaded using the XML-compliant Astrores format.
- Supply user programs to convert popular ephemeris format into AstroRes format?
 - MPC ephemeris format
 - Others ?

5.4 OB form

- Selection of OT and ET in targets to build OB
- New tracking option:
 - “**Non-Sidereal**” = Guiding on Guide Star + apply rates to simulate guiding on moving object.
 - ⇒ Total of 3 guiding options for an OB (currently 1st two are available)
 1. Sidereal, Guiding
 2. Sidereal, No Guiding
 3. Non-Sidereal
- Validation

- Non-sidereal guiding only allowed for OBs containing moving objects.
- ❖ Accurate pointing is not offered for OBs containing moving objects.

5.5 OG Form

The OG form allows setting a time constraint for the observation. As we have seen, moving objects can have “implied” time constraints:

1. When they have a min/max elongation defined (observation around opposition)
2. Because the ephemeris given is only for part of the semester

⇒ Conflicts can happen. Detecting those conditions during the validation of the OG s is probably one of the most complex issue for the Phase 2 tool.

5.5.1 Validation

□ Validation for OGs containing OT objects:

1. OG contains one or more OT object(s) with min/max elongation and *has time constraints*
 ⇒ OG Time constraint should be compatible with the dates of opposition +/- min/max elongation of the objects. In particular an OG of type “Sequence” with OBs containing OT objects with min/max elongation defined and with very different dates of opposition cannot be built.

⇒ Algorithm:

- For all OT objects, compute date of opposition, add min and max elongation ⇒ define observability window
- If SOB, several OT objects, check windows overlap ⇒ define intersection
- Check that OG window overlap intersection

□ Validation for OGs containing ET objects:

1. OG contains one or more ET objects and *does not have time constraints*
 ⇒ Check that there are enough ephemeris lines to allow correct interpolation during the whole semester. **How TBD.**
2. OG contains one or more ET objects and *has time constraints*
 ⇒ Check that the ephemeris given for the object is compatible with the time constraint

□ [VERY OPTIONAL NOW] New OG type “xsequence”. OG contains several possible OBs, only one needs to be observed to validate the OG

- Could be useful if PI has several moving objects/fields to observe but does not mind which are observed, wants to maximize chances of visibility by providing targets spanning the semester.
- **Pros:** can release time constraints, which are an issue in Q mode.
- **Cons:** all SPs using OG type must be edited to allow for new scheme.

- ❑ A general issue with time constraints: time constraints at the OG level should be checked to be within the Q Runs (the dates of the Q Runs are known before opening the Phase 2 Tool). NB: Other checks could also be implemented: ex: monitoring OG with period of 14 days requesting dark sky condition is not feasible...

5.5.2 Implementation

Because of the need to compute ephemeris and/or opposition dates for validation purposes, it is probable that some the validation of the OG form, unlike other forms, need to be done on the server side (vs. in JavaScript). Therefore the validation step is likely to require another connection with the server, and will become longer if lots of computations are needed.

6 Database Organization

6.1 Structure

6.1.1 Ph2 database

In the ph2 database design, targets can be of 3 types:

- ❑ FT Object = Fixed Target (stored in table `ph2..target_f`)
- ❑ OT Object = moving Target defined by Orbital elements (stored in table `ph2..target_o`)
- ❑ ET Object = moving Target defined by Ephemeris (stored in table `ph2..target_e`)

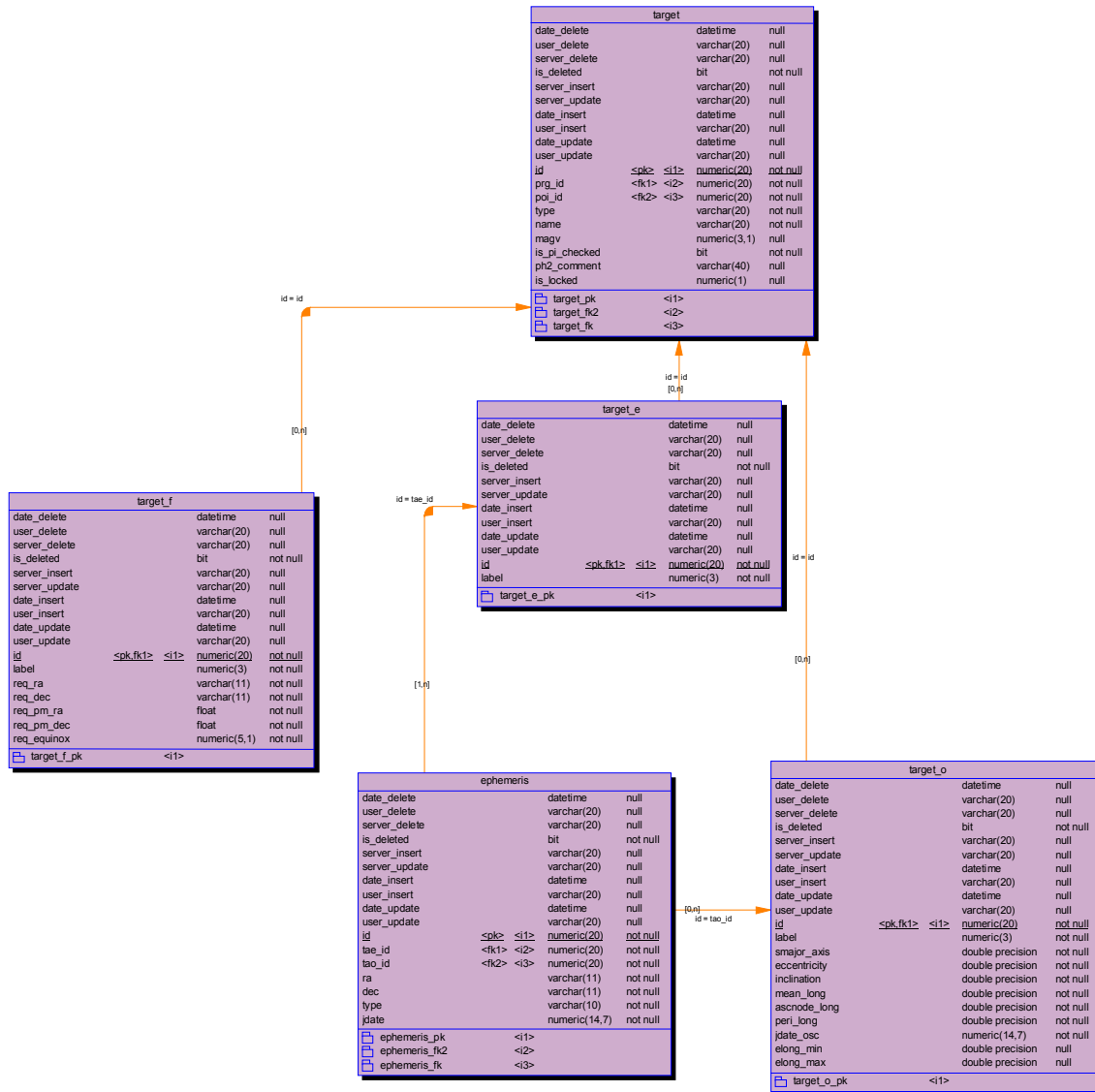


Figure 3: Tables for storing targets in the ph2 database

The `ephemeris` table is designed to hold the ephemeris entered for ET objects and also has provision to hold ephemeris produced from OT objects.

6.1.2 Operational Database

In the operational database, the targets coordinates used in queries are in a separate table `op..xtarget`. This table contains the coordinates, including the pointing offset chosen by the PI. Currently, the coordinates are transferred from `ph2..target_f` to `op..xtarget` on a daily basis with the SP

compute_current_radec. For other target types, this transfer is not yet implemented.

xtarget		
date_delete	datetime	null
user_delete	varchar(20)	null
server_delete	varchar(20)	null
is_deleted	bit	not null
server_insert	varchar(20)	null
server_update	varchar(20)	null
date_insert	datetime	null
user_insert	varchar(20)	null
date_update	datetime	null
user_update	varchar(20)	null
id	<pk,fk1> <i1>	numeric(20) not null
req_equinox		numeric(5,1) null
app_dec	varchar(11)	null
app_ra	varchar(11)	null
app_dec_deg	float	null
app_ra_deg	float	null
date_app	datetime	null
xtarget_pk	<i1>	

Figure 4: Table for storing current target coordinates in the op database

❑ Modifications needed

- Add a field “opposition date” in op..xtarget (NULL if type=target_f or target_e)

❑ **Ephemeris Generation** for OT objects

Continual computation of the moving target positions by compute_current_radec requires an **ephemeris generator**. We have identified a candidate C program available locally and maintained by C.Veillet, which can be included in the system as a ESP – Extended Stored Procedure (we have experience doing that for slalib routines, datalogger access etc.). A dedicated server could also be built using the Sybase Open/Server interface. Basically, in order to achieve the integration, the main() function of the program should be converted into a function. That function is passed the orbital parameters, a range of date and a period, and returns (via pointers) an array containing the computed coordinates and the elongation of the object for the dates given. It also returns the date of opposition for the object.

❑ **Ephemeris interpolation** for ET objects

For objects defined by ephemeris we need an **interpolation routine** to generate an ephemeris for a precise date, from the ephemeris given for the semester. This interpolation routine would be used for queries and to send the coordinates to the new interpolation routine of the TCS (See 9).

6.2 Queue Preparation and Queries Strategy

During a Q preparation query (“Main Query”), done through the queue preparation tool, a moving object is considered observable if all these conditions are met:

- ❑ Its coordinates for the date of the Q are within the coordinate range of the query.
- ❑ Its elongation for the date of the Q is within the elongation range defined by the PI. Since the elongation is defined in dates before/after opposition, this translates to a time constraint.

In addition, usual criteria for selecting the OB (i.e. time constraints) still apply.

We plan to use the table op..xtarget to maintain the current target coordinates (refreshed every few

hours). Most Queues being designed for the current night, most queries will rely on the current target coordinates (furthermore we plan to do more real-time queries as the selection system is automated). For queries further in the future similar tables can be built.

For now we propose to build a temporary table “on the fly” for the duration of the query (to compute candidate moving object coordinates for the Q date). Eventually we could speed-up queries by maintaining tables for 1,2... days in the future on a daily basis. During Q preparation, we do not need a very accurate position of the object.

6.3 Observations

Contrarily to the Q preparation phase, during the observations it is necessary to generate precise target coordinates just before sending them to the TCS. A stored procedure will have to be called prior to sending the coordinates (as we do for computing the iteration).

- For OT objects the **ephemeris generator** will be called
- For ET objects the **interpolation routine** will be called

Triplets of coordinates are then sent to TCS (see 9.) The TCS calculates then the “now” coordinates and the tracking rates.

7 *QSO Tools and GUI implications*

- The QSO Tools, including TMT, uses Stored Procedures that usually return targets coordinates from the `xtarget` table. Therefore coordinates displayed in the QSO Tools are the current coordinates.
- Observing tool: SP call before sending coordinates, TCS limits check (See 9).

8 *Issues in the QSO system*

8.1 Photometry

During the Phase 2, when the PI checks the photometry flag for the OB it means that:

- Either the OB has to be acquired in photometric conditions
 - Or a Photometric OB (“Q98” OB) with the same targets and filter within the BVRI set, smaller exp times, has to be observed in photometric conditions during the semester.
- ⇒ What is the meaning of that photometric flag for an OB containing a moving object? If we keep the current meaning, then if the OB is not observed during photometric conditions should we:
1. Not offer photometry for OBs containing moving objects.
 2. Keep the moving target for the photometry OB. But the photometry is only useful with the same star field as the real observation.
 3. Keep the position of the moving target. But the Photometry OB can be done only after the science OB.

4. Observe OBs containing moving objects and requiring photometric conditions *only in photometric conditions*, thus removing the need of a Photometric OB for those science OBs. This might not be possible for an OB containing a moving object, which has implied time constraints (observation around opposition). So this solution cannot lead to a guaranteed photometric observation !

❖ At the design review meeting it was agreed to start with the simpler solution 4. (observe OBs containing moving objects and requiring photometric conditions *only in photometric conditions*)

9 TCS Modifications

9.1 TCS

Currently TCS has differential tracking but only accepts a rate (speed vector) which can describe a rectilinear movement. As the operations shows³, for many object this approximation would not be accurate enough for 2 hours.

An OB having a max I-time of 2 hours and guiding instructions being sent to TCS at the beginning of each OB, the non-sidereal guiding scheme has to be modified.

After discussion with the TCS team, we propose to send 3 (❖ maybe 5) ephemeris lines, allowing TCS to generate a parabolic or spline (**TBD**) arc suitable to follow a moving object for 2 hours. Those ephemeris lines are generated by adding the PI-specified **pointing offset** to the position of the object computed by the QSO system for the date of the observation.

NB: This interpolation routine is simpler than the one that we use in the DB, as it only accepts 3 points already computed to fit the observation window.

Issues:

- ❑ Requirements of best time gap between those sets of coordinates for TCS. Example:
 1. Start of observation
 2. Start of observation + 1 hour
 3. Start of observation + 2 hours
- ❑ The following conditions related to TCS hardware limits in non-sidereal guiding mode must be checked by the QSO system before observations, but should also be detected earlier during Q preparation and communicated to the PI for possible correction. **If the telescope cannot track a moving object in non-sidereal mode for the duration of an OB, that OB should not be started.**
 1. Maximum telescope differential rate is roughly the rate of the Moon. Fast objects could go faster, so **the moving object speed must be checked before being sent to the telescope.**
 2. Maximum bonnette guider probe excursion. For a particular object, how long can non-sidereal guiding last before the Guide Start must be reacquired? **Moving object speed * exposure time**

³ Moving Objects observers are known to re-send rates during an observation to continue following an object

should be less than maximum guider probe excursion (about 600 arcsecs). If we continue to select guide stars automatically as for Fixed Objects, we don't know the exact position of the guider probe in advance, so an average value could be used. Guide Star selection could also be done in advance for those objects (manpower issue for the QSO team ?).

- ❑ A non-trivial problem for TCS: knowing the rate and direction of the OG, selection of the most appropriate Guide Star.

10 NEO Modifications

10.1 TCS agent

- ❑ Accept new way of guiding, and way of sending 3 sets of values suitable for TCS for computation of the differential tracking.

Currently for fixed targets we have

```
tcoords RA DEC EQ NAME
```

Proposal for moving targets:

```
tcoordsm TIME1 RA1 DEC1 EQ1, TIME2 RA2 DEC2 EQ2, TIME3 RA3 DEC3 EQ3 NAME
```

Implies differential guiding if next command is `gcoords select`

- ❖ Assuming that points are regularly spaced, need to give only the date of the 1st and the last ones. The proposal becomes:

```
tcoordsm JJ1 JJ2 RA1 DEC1 EQ1, RA2 DEC2 EQ2, RA3 DE3 EQ3 NAME
```

NB: If we settle for accepting and computing only coordinates in J2000.0 the parameters EQ_n n=1..3 or 1..5 will always be 2000.0.

11 Schedule

	Design	Implementation	Test	Production
PH2	DONE	NOW-Nov 2001	15 Dec 2001	01 Jan 2002
Ph2 DB	DONE	DONE	15 Dec 2001	01 Jan 2002
Op DB	Nov 2001	Dec 2001	01 Jan 2002	01 Feb 2002
QSO Tools	Nov 2001	Nov-Dec 2001	01 Jan 2002	01 Feb 2002
TCS⁴	Nov 2001?	Nov-Dec 2001?	15 Jan 2002	01 Feb 2002
TCS Agent	Nov 2001?	Nov-Dec 2001?	15 Jan 2002	01 Feb 2002

⁴ NB: TCS Team Project Engineer has vacation planned in Dec 2002

We will need an integration test campaign at the beginning of the semester 2002A, similar to the QSO test campaign of Jan 2001.

⇒ Objective 1 (required)

- Define moving objects in PH2 DB
- Prepare Q with moving objects
- Send Q observations of moving objects to “fake” director session

⇒ Objective 2 (recommended)

- Real observation of some bright asteroids/comets with different rates.
- ½-1 E-night QSO+TCS to test TCS agent + TCS new tracking mode and validate a few images. There is no CFH12K time in the second part of January. D05 on Jan 7 is likely to be too early:

We recommend that the first Q run of 2002A begin with 0.5-1 E-night and ask Astronomers to think about potentially interesting bright moving objects for that E-night.

C/2000 WM1 Linear ?