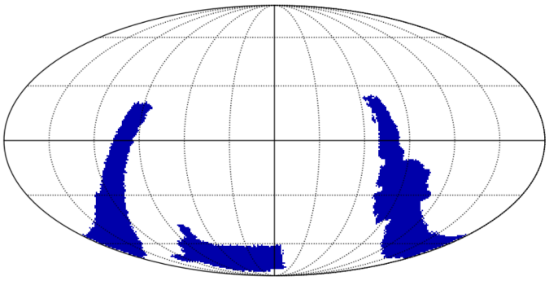
# OCVS VO Publication Description

v0.1 (Beta)

Olga Vozyakova
Pavol Jozef Šafárik University in Košice, Slovakia
olga.vozyakova@student.upjs.sk
16 February 2026

**Coverage**



The OGLE Collection of Variable Stars (OCVS) is re-published via IVOA-compliant services using the DaCHS suite.

The original data were obtained from https://www.astrouw.edu.pl/ogle/ogle4/OCVS/ (accessed December 2025).

All resource descriptors are available on GitHub:
https://github.com/olgavoz1971/dachs_upjs_vo/tree/main/ogle

# VO Protocols

OCVS lightcurves ("timeseries") are published via **TAP, SSAP, DataLink** and **Cone Search (SCS)**. The Cone Search service works on the table with observed objects.
The timeseries are also discoverable and accessible through the `ivoa.obscore` table.
Have I forgotten something?

## Services endpoints

**TAP**:
https://skvo.science.upjs.sk/tap

**SSAP**:
https://skvo.science.upjs.sk/ogle/q/ssa/ssap.xml

**SCS**:
https://skvo.science.upjs.sk/ogle/o/ogle-objects/scs.xml

# Publication from the client’s point of view

The data are accessible with any VO-compatible tool, for example, TOPCAT, SPLAT-VO or `pyvo`. If you found any that do not understand this data, please tell me.

## The Tables

### Observed objects

The original OCVS data are divided by sky field and variability class. I follow this division, but this first layer is hidden from external (ADQL) access to avoid confusing the user.

Instead, the client sees tables where stars are grouped solely by variability class. So, you can find the following tables:

`acepheids, cepehids, dsct, eclipsing, blap, heartbeat, miras, rotating, rrlyr, t2cep, transits, cv, m54` tables.

The last one (m54) comprises different types of variable stars, as found in the original data.

*All observed objects* are aggregated in the `objects_all` table. This table contains only basic parameters such as coordinates, periods, epochs, mean magnitudes, etc.
Some class-specific parameters are mapped somewhat arbitrarily. Namely: the depth of eclipsing binaries and transits is mapped to amplitude; the mean magnitude at maximum (for eclipsing binaries) is mapped to mean magnitude.

I preserve the original splitting by class + sky field (OCVS directory structure) in the `ssa_collection` column, e.g., OGLE-BLG-CEP, OGLE-BLG-LPV, OGLE-GAL-ACEP, OGLE-GD-RRLYR, OGLE-LMC-HB, OGLE-SMC-T2CEP, OGLE-M54, and so on.

The OCVS variability class is stored in the `ogle_vartype` column. Following IVOA recommendations, I also assign a SIMBAD [object type](object type) to each class, mapping them manually (I believe correctly) to the `ssa_targclass` column:

| ogle_vartype | subtype | ssa_targclass |
|---|---|---|
| ACep | pulsating mode | Ce* |
| Cep | pulsating mode | cC* |
| T2Cep | RVTau / WVir / pWVir / BLHer / NULL / pulsating mode | WV* |
| dSct | pulsating mode | dS* |
| RR Lyr | RRab / RRc / RRd / aRRd / NULL | RR* |

| | | |
|---|---|---|
| Mira | NULL | LP* |
| BLAP | NULL | Pu* |
| Ecl | C / NC / NULL | EB* |
| Ecl | CV | CV* |
| CV | NULL | CV* |
| Ell | ELL / NULL | El* |
| Hb | RG / MS | Pu*,El* |
| Irr | NULL | Ir* |
| SXPhe | NULL | SX* |
| Rot | NULL | Ro* |
| spotted | NULL | |
| SR | NULL | V* |
| Transit | NULL | V*,Pl? |
| v | NULL | V* |
| NULL | NULL | V* |

## Lightcurves

Technically, all measurements (1.7 billion – 1662458916, to be precise) in both filters are stored in one huge table, `lightcurves`. Properly indexed (I hope), it supports efficient queries by `object_id`, `passband`, or `obs-time` (MJD).

I consider the `lightcurves` table suitable for quite exotic queries, while most standard access is expected to go through the `ts_ssa` table. This table implements the [IVOA SSA protocol](), which is also recommended for publishing timeseries.
In the `ts_ssa` table, each row corresponds to one lightcurve (i.e. one object in one passband). The table contains a lot of columns, most of which are required by the SSA protocol. Among the practically useful ones are:

`accref` – the link to the lightcurve (e.g. https://skvo.science.upjs.sk/ogle/q/sdl/dlget?ID=ivo://astro.upjs/~?ogle/q/OGLE-BLG-ECL-116111-I)
`preview` – link to a low-resolution preview of the folded lightcurve
`ssa_targname` – object identifier (e.g. OGLE-BLG-ECL-115471)
`object_id` – the same; allows a straightforward join with the object tables
`ssa_targclass` – SIMBAD object type (e.g. “EB*”)
`ssa_timeExt` – time span between the first and the last measurement, in days (1187.914)
`ssa_length` – the number of measurements in this lightcurve

`ssa_collection` – reflects OCVS directory structure (e.g. OGLE-BLG-ECL)
`ssa_bandpass` – photometric band ("I" or "V")
`ssa_instrument` – ("Warsaw 1.3m Telescope")
`ssa_reference` – bibliographic reference for this curve (e.g.,2016AcA....66..405S)

A `DataLink` service is associated with this table.
The lightcurves are also discoverable through the `ivoa.obscore` table (see an example in the ADQL section).

## Access via TAP (ADQL queries)

**TAP** URL: https://skvo.science.upjs.sk/tap
The Table Access Protocol (TAP) is an IVOA standard that allows users to query database tables using ADQL (Astronomical Data Query Language).

Through the TAP service, you can select specific columns, filter by physical parameters, search by sky position, or join tables.

You can access TAP service programmatically (via `pyvo`), or from an application, for example, from Aladin or TOPCAT (See basic TOPCAT how-to in Appendix A)

### Examples

#### Download all RR Lyrae lightcurves with periods between 0.5 and 0.7 days in the I band:

```
SELECT ssa_targname, accref, o.period FROM ogle.ts_ssa AS t
JOIN ogle.rrlyr AS o USING (object_id)
WHERE o.period BETWEEN 0.5 AND 0.7 AND t.ssa_bandpass = 'I'
```

#### Download all previews of the lightcurves of all Anomalous Cepheids in the SMC:

```
SELECT preview FROM ogle.ts_ssa WHERE ssa_collection='OGLE-SMC-ACEP'
```

In TOPCAT, you can then "watch a movie" of the previews (see Appendix A).

#### Select ten eclipsing binary systems, classified as "Contact" with the longest periods:

```
SELECT TOP 10 t.accref, t.preview, o.*
FROM ogle.eclipsing AS o
JOIN ogle.ts_ssa AS t USING (object_id)
WHERE o.subtype='C' AND t.ssa_length > 100
ORDER BY o.period DESC
```

Retrieve the I-band lightcurve of ZZ Dor:

```
SELECT * FROM ogle.ts_ssa AS t
JOIN ogle.objects_all AS o USING (object_id)
WHERE 1=ivo_hasword(vsx, 'SW TUC') AND ssa_bandpass='I'
```

**Note**: It is preferable to use `ivo_hasword` instead of `vsx='ZZ Dor'` (when this function is supported by the service), because it is more flexible: it is case-insensitive and can handle columns containing multiple names, as in this example.

Using ObsCore: get 100 longest lightcurves of the pulsating variables (BLAP):

```
SELECT TOP 100 * FROM ivoa.obscore
WHERE dataproduct_type='timeseries' AND target_class='Pu*'
ORDER by t_xel DESC
```

## DataLink

Actually, access to previews and lightcurves is organised via the DataLink service: the `accref` and `preview` columns contain DataLink service references instead of static file links.

DataLink allows a client to access related resources, for example, the actual data file, a preview, or other auxiliary products.

To explore how this is organised for a particular lightcurve, you can either: open the direct link to the `dlmeta` service in a browser, something like
https://skvo.science.upjs.sk/ogle/q/sdl/dlmeta?ID=ivo://astro.upjs/~?ogle/q/OGLE-SMC-CEP-2353-I (for OGLE-SMC-CEP-2353 in the 'I' band)
or use TOPCAT (see Appendix A).

## SSAP

Our SSAP URL for OCVS is https://skvo.science.upjs.sk/ogle/q/ssa/ssap.xml
*(If you open this link directly in a browser, it shows an error. This is normal: the SSAP service expects a proper query with parameters.)*

The Simple Spectral Access Protocol (SSAP) allows users to discover and access spectra or timeseries by celestial coordinates. It is simpler than TAP but less flexible for complex queries.

SSAP is implemented in several VO clients, for example, TOPCAT and SPLAT-VO.
See Appendix A for an example of usage.

# From the publisher’s point of view

## Resource Descriptors

All resource descriptors (RDs) for the OCVS publication via [DaCHS]() can be found on my GitHub: https://github.com/olgavoz1971/dachs_upjs_vo/tree/main/ogle.

The main idea behind the multiplicity of RDs is: we ingest the data “as is”, and then make it appropriate for VO-publishing, mostly via database views – both materialized and not.

This allows us (as I hope) to update data without superfluous effort.

Roughly, the RDs fall into the following categories:

**Auxiliary RDs** (`meta.rd, aux.rd, phot.rd`). These contain common elements used by multiple resources: streams, mixins, and shared metadata.
`meta.rd` and `aux.rd` should not be imported directly.
`photosys.rd` should be imported, as it defines the photometric system used in the publication.

**Original data RDs** are divided by sky field (following the OCVS directory structure): `blg, lmc, smc, gd,gal`, and `misc` (miscellaneous for the rest). These should be imported first.
Another RD for original data is `lc.rd,` which handles the ingestion of the original lightcurves. It stands out because it processes a really huge amount of data (1.7 billion rows) and should be treated with caution. Lightcurve ingestion operates in “updating” mode, so this can easily lead to duplicate data.

**RDs for client visible data**: `o.rd` and `q.rd`. These create the views and describe the services that end users will see.
`o.rd` handles observed objects and Cone Search; `q.rd` builds SSAP timeseries view and related services.
These RDs should be imported last.

The right way for importing is:

1) `phot.rd`
2) `lc.rd` and sky-fields RDs (aggregated in `import_fields.sh`)
3) `o.rd, q.rd`
4) `limits o.rd, q.rd`
5) `pub o.rd, q.rd, lc.rd`

# Table relations

## OCVS original layer

Usually, each original sky-field/variable directory contains two types of files with information about observed objects:

- `iden.dat` - with names, coordinated and cross-match fields;
- a set of files with variable star parameters, such as periods, epochs, lightcurve parameters, classification, etc

I reflect this division by having two tables corresponding to each OCVS directory: `ident_*` and `param_*`. Their names include the sky-field/class specification; for example, `ident_blg_cep` and `param_blg_cep` for Galactic bulge Cepheids. These tables are hidden for external (ADQL) queries.

Tables from this layer are described in the “field-RDs”: `blg, gd, gal, lmc, smc` and `misc.`

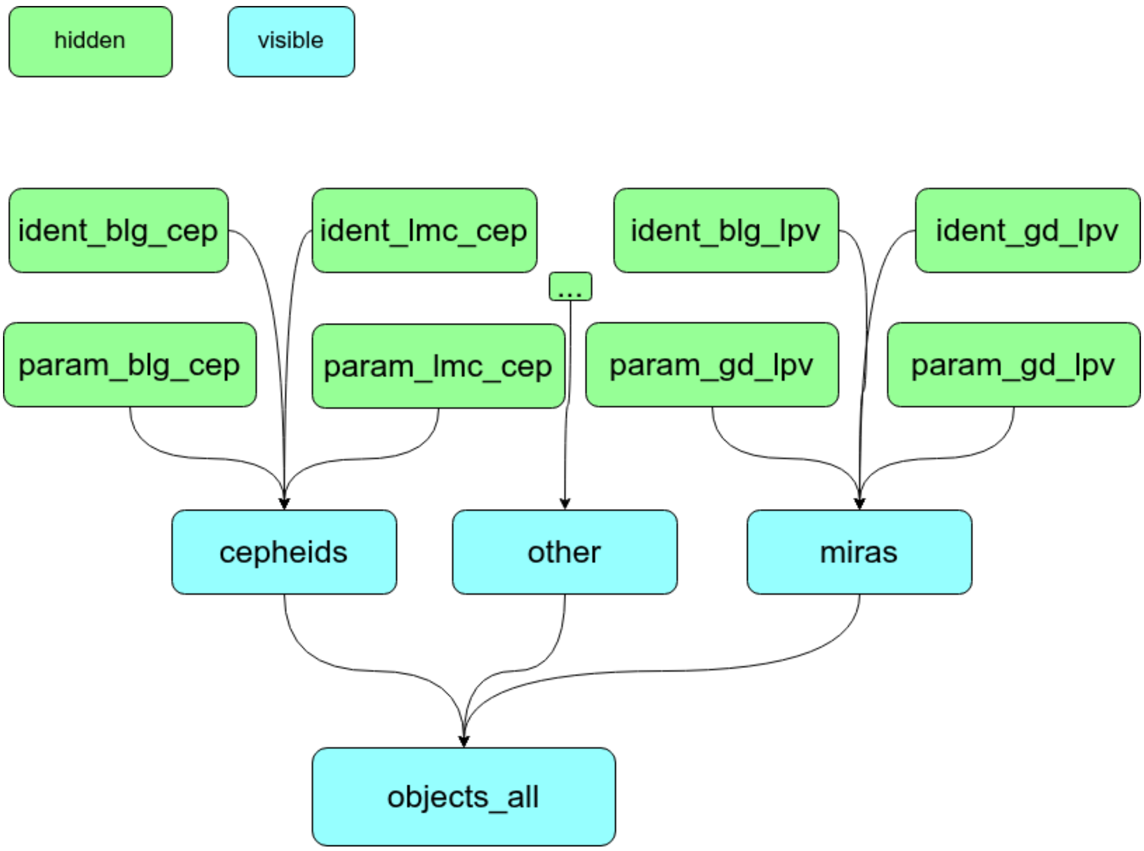
## Observed objects layer

In the `o.rd` data from field-RDs is aggregated into tables (actually organised as Views) specified by variable star class:
`blap, acepheids, cepheids, dsct, eclipsing, heartbeat, miras, rotating, rrlyr, t2cep, transits, cv and m54.`

Again, `m54` is a collection of stars from M54 and comprises different types of variable stars. These tables contain a slightly different set of columns, depending on the particular features of each variable star class.

All objects are unified into the `objects_all` table (also organised as a view), which stores only the basic common columns from the variable-class tables.

Original division by the sky-field and variability class is preserved in the `ssa_collection` column and is present in almost all tables with objects and lightcurves.




### Lightcurves

All measurements are stored exclusively in the database, in the `lightcurves` table. This data was imported from the original files via the `lc.rd`.

Each row contains observation time (`mjd`), magnitude (as `phot`), error, band and `ogle_phase` column (values 0–4) to help users perform photometric transformations or filtering, if they find it useful:
2 = OGLE II
3 = OGLE III
4 = OGLE IV
0 = not specified

### Timeseries in IVOA SSA format

The `ts_ssa` table is built in the resource descriptor `q.rd`. Corresponding data is also added to the `ivoa.obscore` table.

## References

I found related papers in each directory of the OCVS archive: [https://www.astrouw.edu.pl/ogle/ogle4/OCVS/](https://www.astrouw.edu.pl/ogle/ogle4/OCVS/) and placed the bibcode of the first paper (if there is more than one) as a reference for the corresponding light curve.

I also added, to some extent, an arbitrary list of creators to the general description (visible in the left panel of our DaCHS pages, e.g. https://skvo.science.upjs.sk/ogle/o/ogle-objects/info ) under the “Creator” tab:

(Soszyński, I.; Udalski, A.; Szymański, M.K.; Pietrukowicz, P.; Borowicz, J.; Glowacki, M.; Hamanowicz, A.; Iwanek, P.; Kołaczek-Szymański, P.A.; Mróz, M.; Pawlak, M.; Ratajczak, M.; Skowron, J.; Wrona, M.).

Please write to me so that I can correct it.

# Appendix A: Simple example of OCVS data access via TOPCAT

## TAP

In TOPCAT, open **VO** –> “**Table Access Protocol (TAP) Query”.**

Then, in the “**Select Service**” tab, at the bottom, you will see the “**TAP URL**” entry. Type there: `https://skvo.science.upjs.sk/tap`, and click the “**Use Service**” button.

In the tab that opens, you'll see a tree of tables. OCVS-related tables are located in the "ogle" branch. Table descriptions are displayed in the right panel under the "**Table**" tab. A general description of the entire OCVS publication will be in the “**Schema**” tab.

If you don’t like descriptions of the Schema or any table, please tell me.

### Examples

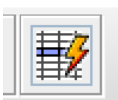
#### Watch a “preview movie”

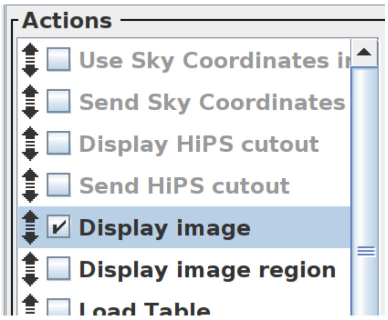
To download previews of the lightcurves of all Anomalous Cepheids in SMC, use the query:

```
SELECT preview FROM ogle.ts_ssa WHERE ssa_collection='OGLE-SMC-ACEP'
```

Type the query in the “**ADQL Text”** entry.

In the **Control Window**:
Select the table with the result of the last query
Hit the **Activation Action**
Check the “**Display Image**” option






In “**Configuration**”, select `Image Location: preview`


Click on the “**movie**” icon: 

# DataLink

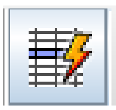
Execute ADQL query on the `ts_ssa` or `obscore` table, as described above, for example:

```
SELECT t.* from ogle.ts_ssa AS t
JOIN ogle.objects_all AS o USING(object_id)
WHERE 1=gavo_match('V6229 Sgr', vsx) AND ssa_bandpass='I'
```

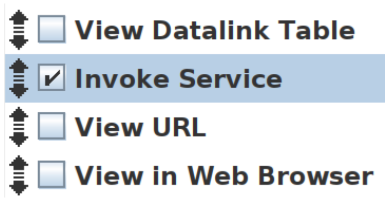
or

```
SELECT top 10 * from ivoa.obscore where
dataproduct_type='timeseries' AND target_class='RR*'
```

Select the table with the results.
Open “**Activation Action**”

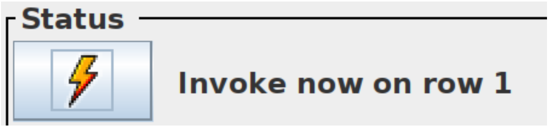


Select **Invoke Service**



**Action**: `View DataLink Table`


Click **Invoke** – and enjoy!




## SSAP

The URL of the SSAP services is
https://skvo.science.upjs.sk/ogle/q/ssa/ssap.xml

In TOPCAT: **VO –> "Simple Spectral Access (SSA) Query"**

The first line of the bottom panel, enter **SSA URL**

Try: for example:
**Object Name**: `SMC`;
Click "**Resolve**";
**Diameter**: 10 arcmin(!);
Click "**OK**"

The result table, "`SMC-SSA-10m`", will contain about 1160 rows from the `ts_ssa` table, but with slightly different column names.

## Simple Cone Search (SCS)

In TOPCAT:
**VO –> "Cone Search"**

In the first line of the bottom panel, enter the
"**Cone URL**": https://skvo.science.upjs.sk/ogle/o/ogle-objects/scs.xml

Then try, for example:
**Object name**: `LMC`;
"**Resolve**";
**Radius**: 10 arcmin(!);
"**OK**"
The result table, "`LMC-Cone-10m`", will contain roughly 900 rows from the `objects_all` table with the basic parameters of variable stars.