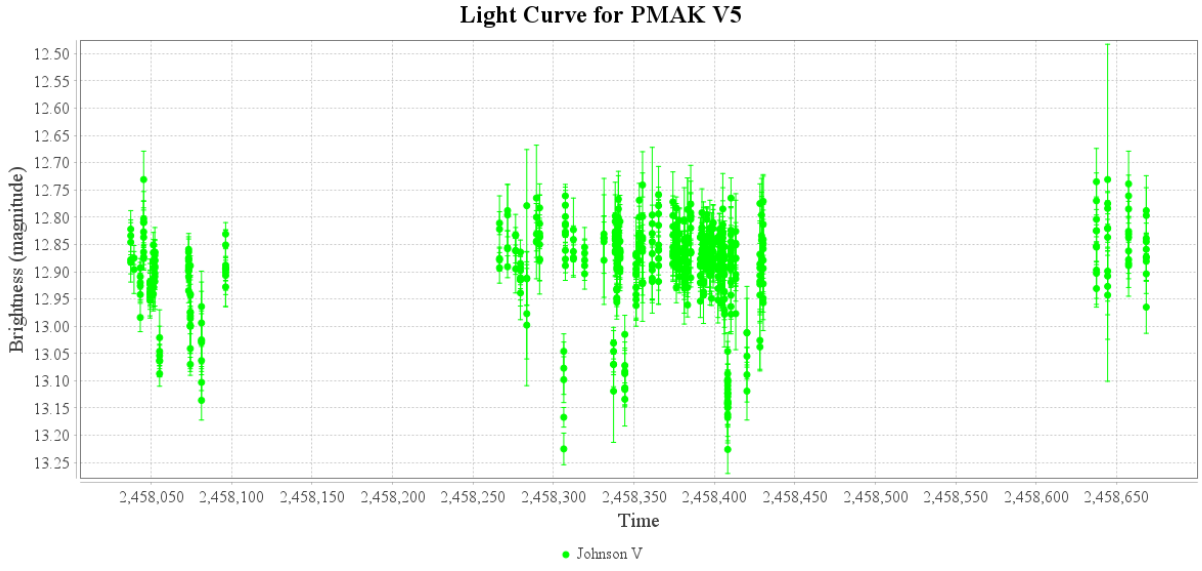
BJD_TDB Converted Plug-in

The plug-in converts loaded observations to the HJD_{TDB} time scale[1]. It recalculates the timestamps for both JD and HJD observations using the appropriate conversion algorithms.

By default, the time-conversion service provided by The Ohio State University is used (https://astroutils.astronomy.osu.edu/time/). If the service is unavailable (e.g., due to maintenance), the plug-in can fall back on a local Python (Flask) microservice, which must be explicitly configured and activated by the user (see Appendix).

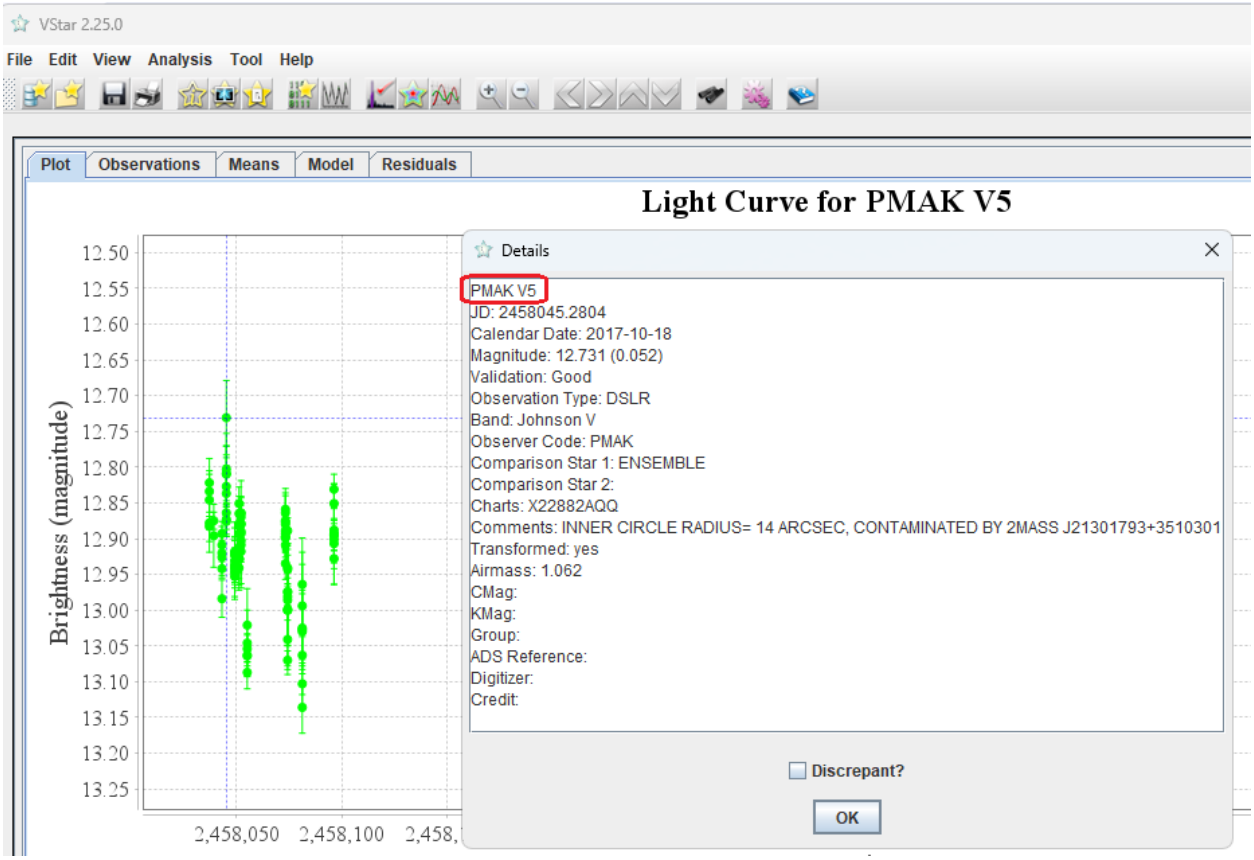
To install the *BJD_TDB Converter* plug-in, go to the *Tools* menu in VStar and select *Plug-in Manager*. Scroll through the list of available plug-ins, select *BJD_TDB Converter*, and click the *Install* button. After installation, restart VStar.

As an example, load PMAK V5 data in the Johnson V filter for JD between 2458037 and 2458670: go to the *File* menu, select *New Star from AAVSO Database*, enter 'PMAK V5' in the *Star* field, and set *Minimum JD* to 2458037 and *Maximum JD* to 2458670. Ensure that the 'Johnson V' checkbox is selected and all others are unchecked, then click OK.

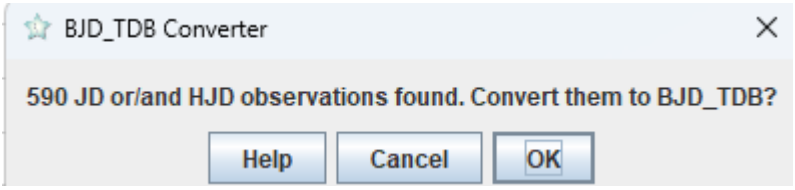


After loading the light curve, you can select a point and use the *View -> Observation Details* menu command to verify that the observations are in the JD time scale:

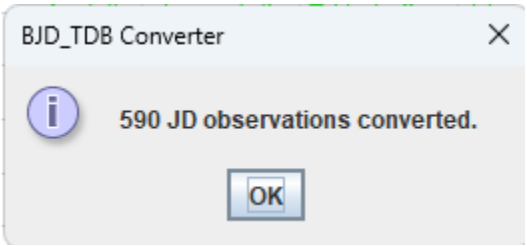
[1] You can find a comprehensive discussion of the various astronomical time scales and the significance of HJD_{TDB} here: https://ui.adsabs.harvard.edu/abs/2010PASP..122..935E/abstract



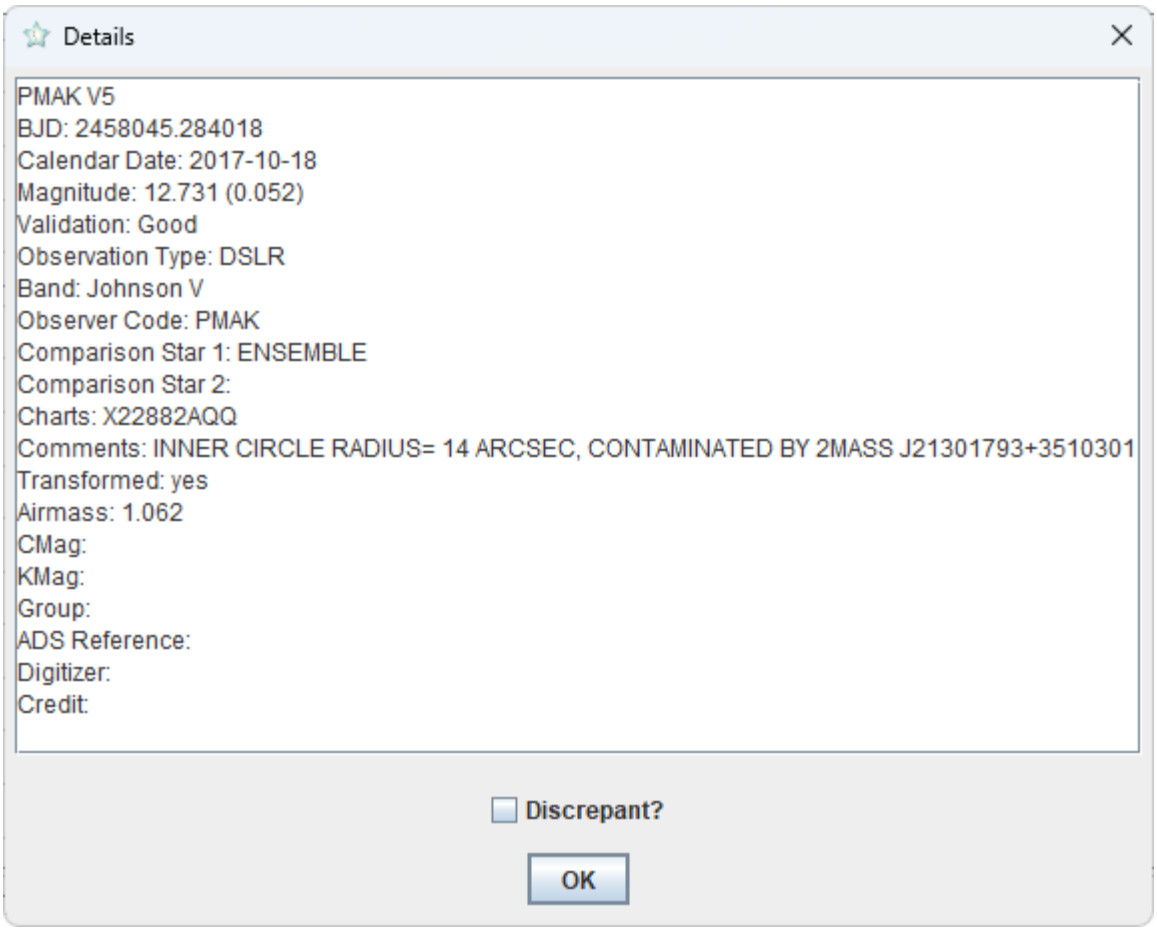
To convert the observation times to HJD_{TDB}, use the *Tools -> BJD_TDB Converter* command. The following dialog will appear:



Click OK. After a brief delay, an information dialog will appear:



Use *View->Observation Details* again. You should see that the observations are now in the BJD_{TDB} time scale:



Appendix. Using the local microservice

Prerequisites: Install Python 3.11. Then, install AstroPy and Flask packages. This can be done with the pip utils:

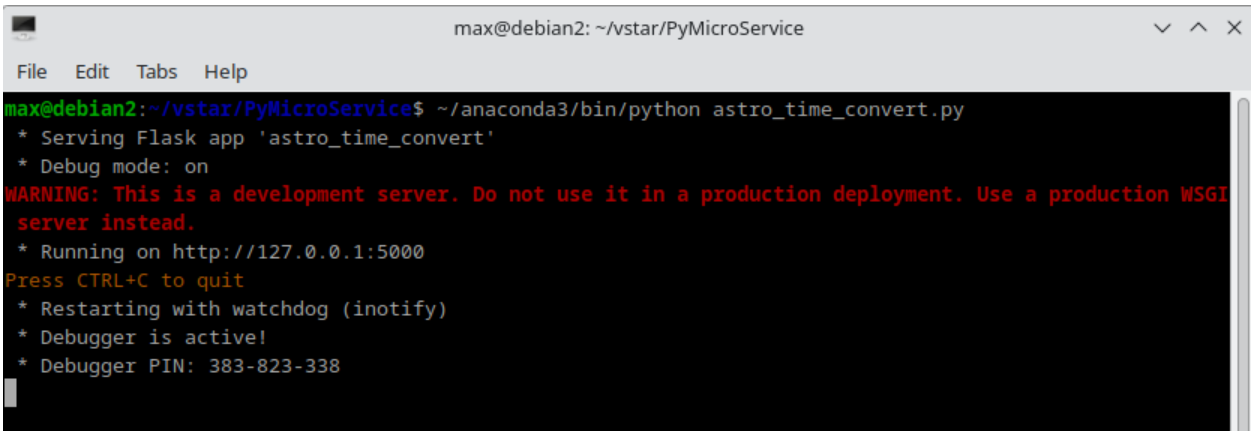
>pip install astropy

>pip install flask

The Python microservice ‘astro_time_convert.py’ is located in the ‘PyMicroService’ subfolder within the ‘vstar’ folder. Run it with the command:

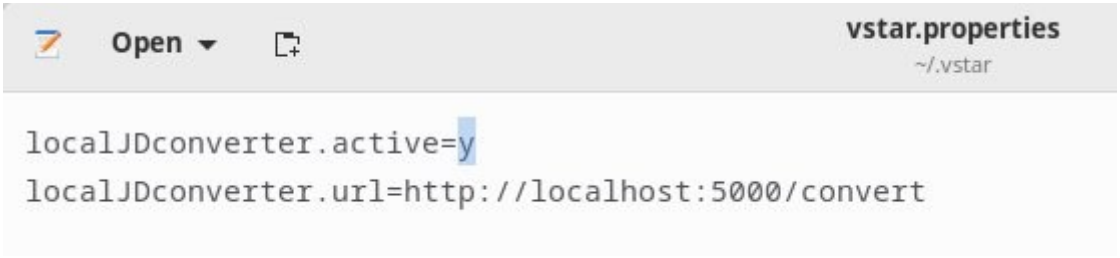
>python astro_time_convert.py

Once the microservice launches successfully, you should see something like the following:



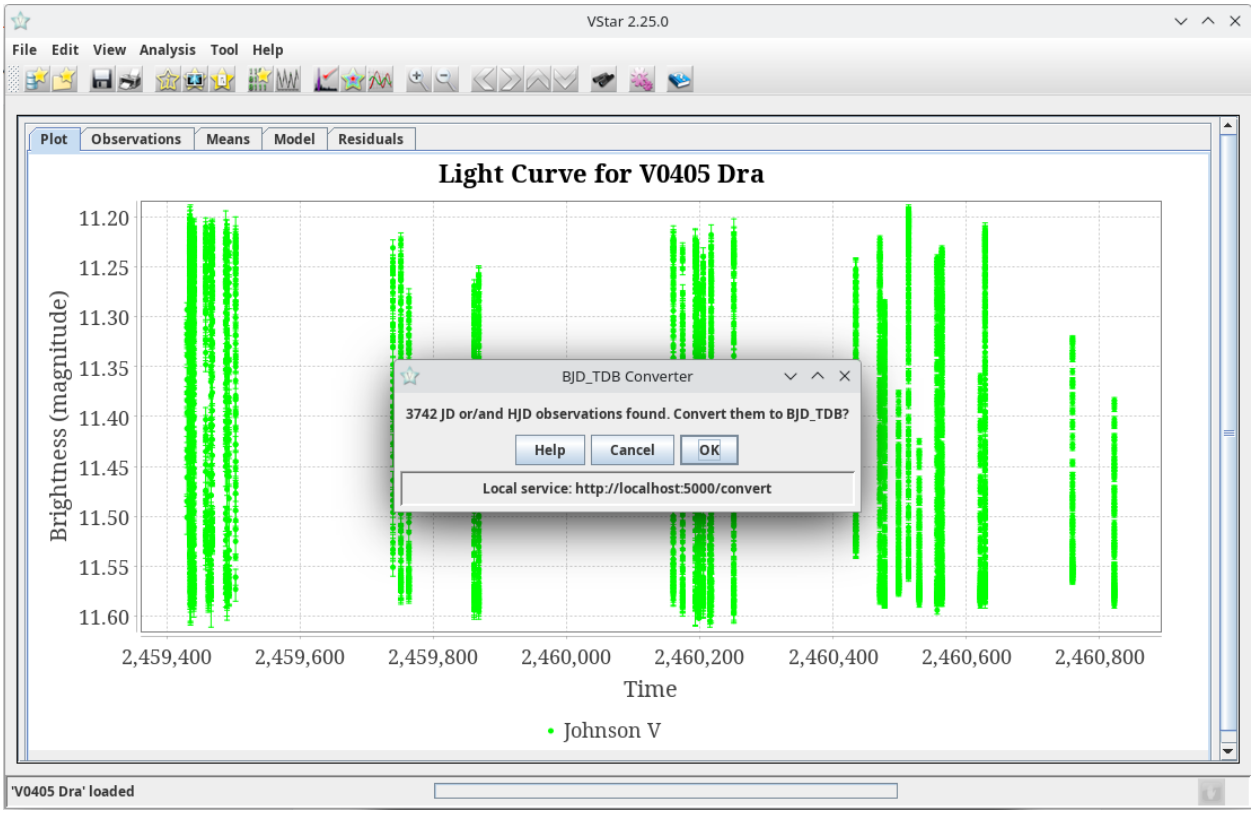
You may ignore the warning; it simply means that you are running a single-threaded internal Flask web server, which is perfectly sufficient for our needs.

In your home directory[2], locate the '.vstar' folder. It should contain a 'vstar.properties' file. In Windows, this folder and file should be created by the installer. If there is no such folder, create it. Then try to locate the 'vstar.properties' file in the VStar installation folder and copy it to the '.vstar' folder. Open .vstar/vstar.properties in a text editor and set the value of the 'localJDconverter.active' parameter to 'y':



To test the microservice, run VStar, load some data (for example, V405 Dra, all times, Johnson V). Invoke the [Tools] -> [BJD_TDB Converter] command and press OK.

[2] In Windows, this is typically C:\Users\<username>; in Linux, it is usually /home/<username>



In a few seconds, you should see the following message (number of observations may differ):

You can try deactivating the local service (via ‘vstar.properties’), restarting VStar, and attempting the conversion again (the Ohio State University service will be used).

Maksym Pyatnytskyy

Rev A, 2025-06-19