Flat Pointer Tool Description

While installing a newly acquired FlatMan light source, I researched the Software Bisque forums for general information on flats light source usage as well as known problems. I came across a procedure written up by Tom Bisque. He described how to make a Sky Database object that represented the position of the light source. Once created, the user could use “Find” and “Slew” to locate that named reference point both inside TSX and using the automation (e.g. scripting) interfaces. The procedure worked well enough, but was rife with opportunities for frustration if one didn’t both follow the instructions precisely or know how to deal with some text editor idiosyncrasies in Windows. The procedure also didn’t address dealing with negative altitudes, which was, of course, my case. So, I thought, why not automate it? Well, automate at least as much as the TSX automation interfaces would allow.

So, I built the Flat Pointer tool to assist a TSX user in generating a custom SDB Reference Object based on Tom’s process. Basically, the code instructs the user to point the scope at the target using the hand paddle, whereupon the program downloads the az/alt coordinates. Or, the coordinates can be entered directly as well. The app then generates a SDB source text file and copies it to the “SDBs” folder in the TSX directory. There are no automation interfaces for loading and compiling SDB source files into database objects, so step-by-step instructions are displayed to walk the user through the process in TSX. Once completed, there will be a Reference Object created in TSX called “My Flat Field” with the given coordinates.

As a bonus exercise, I wrote a simple executable that finds and slews the mount to “My Flat Field” then turns off tracking. This executable can either be run stand-alone, or launched from an automation app like CCD Auto Pilot prior to taking flats. Accordingly, I built into Flat Pointer a button command that copies that executable into the Script folder of CCDAP just to make it even easier to find.

This general description is under the filename FlatPointerDescription.docx in:

https://www.bisque.com/sc/members/rmcalister/files/Software/default.aspx

The executable is under the name FlatPointer\_Exe.zip. See instructions below for installation.

https://www.bisque.com/sc/members/rmcalister/files/Software/Applications/default.aspx

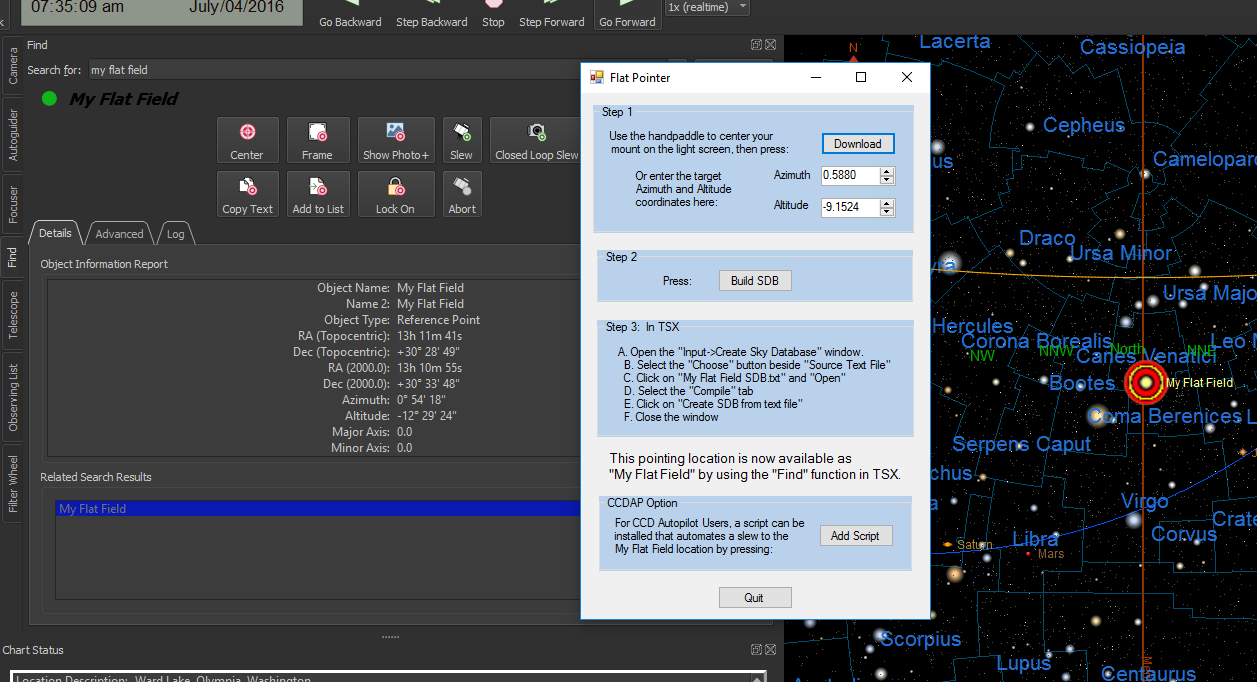
The source is under the name FlatPointer\_Src.zip as a MS Visual Studio project:

https://www.bisque.com/sc/members/rmcalister/files/Software/Source/default.aspx

Details are below for anyone interested in trying it out.

* Rick

Overview:



Requirements:

Flat Pointer is a Windows Forms executable, written in Visual Basic. The application runs as an uncertified, standalone application under Windows 7, 8 and 10. The application requires TSX Pro.

Installation:

Download the FlatPointer\_Exe.zip and open. Run the "Setup" application. Upon completion, an application icon will have been added to the start menu under the category "TSX Toolkit" with the name   
“Flat Pointer". This application can be pinned to the Start if desired. During operation, Flat Pointer will expect that TSX is loaded in the default location (User\Documents\Software Bisque etc.), and, if used, that CCDAP also uses the default file directory structure.

Operation:

Upon command, Flat Pointer will download and store the current Az/Alt position of the mount. A location can be entered manually as well. With another command, Flat Pointer will generate and copy a Sky Database (SDB) source file into the SDBs directory. As no automation interfaces exist for the final step, the form merely directs the user through the sequence of TSX actions to be performed to compile that source file into a custom object. Lastly a command is provided to copy a simple Windows app into the CCDAP script directory. This app will run a find on the newly created object, slew the scope there and turn off tracking. Note, that the user must turn off Slew Limits if the target is at a negative altitude (e.g. floor-mounted FlatMan).

Support:

This application was written as freeware for the public domain and as such is unsupported. The developer wishes you his best and hopes everything works out, but recommends learning Visual Basic (it's really not hard and the tools are free from Microsoft) if you find a problem or want to add features. The source is supplied as a Visual Studio project.