Users of this document are advised to use the ASCII version for copying and pasting text.

Using ISIS to read VIR cubes into ISIS

by Eric E. Palmer
Updated 28 January 2014

The goal of this tutorial is not to teach you ISIS, but to give a list of common commands that would be used when processing VIR data by ISIS. The USGS provides extensive support, tutorials and documentation at their website: http://isis.astrogeology.usgs.gov

You will likely want to consult the ISIS website to get additional details to their software.

Step 1: Install and/or update ISIS

Instructions are at the USGS ISIS website. They have an automated script that will install the program, the data and adjust your logon script to initialize the environment variables. If the script does not work, these functions must be done by hand, especially setting up the logon script.

http://isis.astrogeology.usgs.gov/documents/InstallGuide/index.html

Comment: You may not need to download the entire Dawn data library. You can use the ISIS web service (needed for spiceinit).

If you install the high resolution (Gaskell) DEM, your map projections will be more accurate.

If you have problems, you can search the forums for answers or post a question.
https://isis.astrogeology.usgs.gov/IsisSupport/

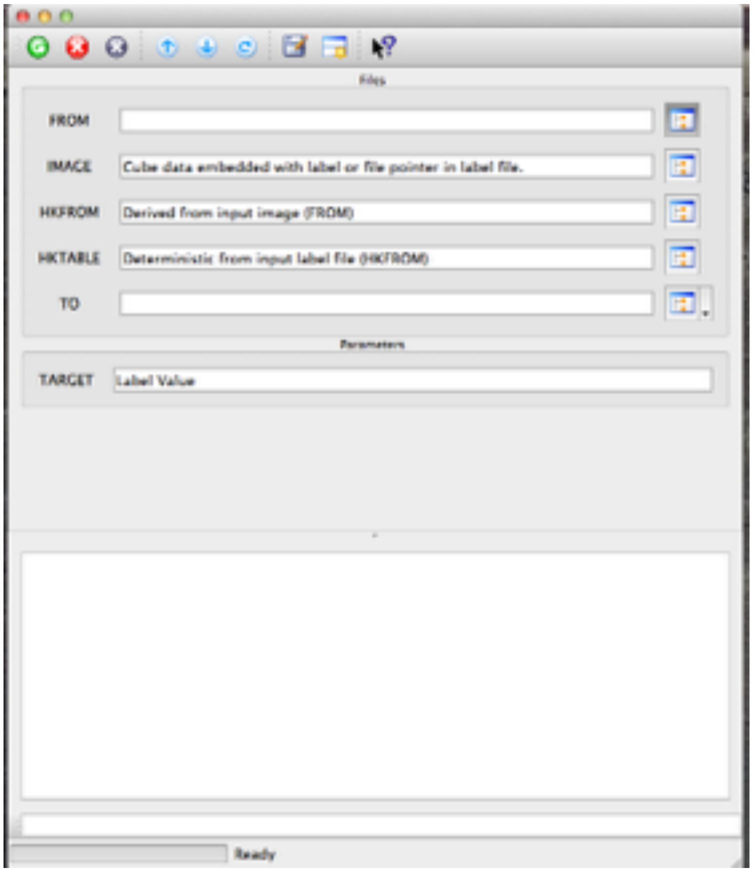
Note, if you get "command not found" that means your path variable does not have the ISIS programs in the path. This is normally set via a script run in your .login, .profile, .cshrc, etc. Look for details on the UNIX environment setup on the ISIS install webpage.

Step 2: Convert the QUB files into ISIS cubes

The Dawn VIR data can be downloaded here: http://sbn.psi.edu/pds/resource/dwnvvir.html
It is easiest to copy whatever data you need into a working directory or run the ISIS commands within the directories themselves.

Basic info about the files

filename.QUB - the actual data files (3D spectra)
filename.LBL - the label files that explain the file format and metadata
filename_HK.TAB - the house keeping data (when each sample was taken)
filename_HK.LBL - describes the columns of the house keeping data
filename_QQ.LBL - described the quality control file (not needed)
filename_QQ.QUB - a cube of quality control data (not needed)



FROM
IMAGE Cube data embedded with label or file pointer in label file.
HKFROM Derived from input image (FROM)
HKTABLE Deterministic from input label file (HKFROM)
TO
TARGET Label Value

Option 1: Run dawnvir2isis

This will assemble all the parts into a single file that you can use qview and other ISIS tools to view and use.

Run the GUI version of the command by entering at the command line:

```
dawnvir2isis
```

Just click the box on the right side of the FROM line. This will let you select the filename.LBL file that you need. ISIS extracts the name of the image file and housekeeping files from the label. You can specifically select different files if needed (shouldn't be necessary for this data set). Then, set the name of the output cube.

If you want to use just the command line, type:

```
dawnvir2isis from=VIR_IR_1B_1_366390345_1.LBL to=testcube
```

Option 2: Run pds2isis

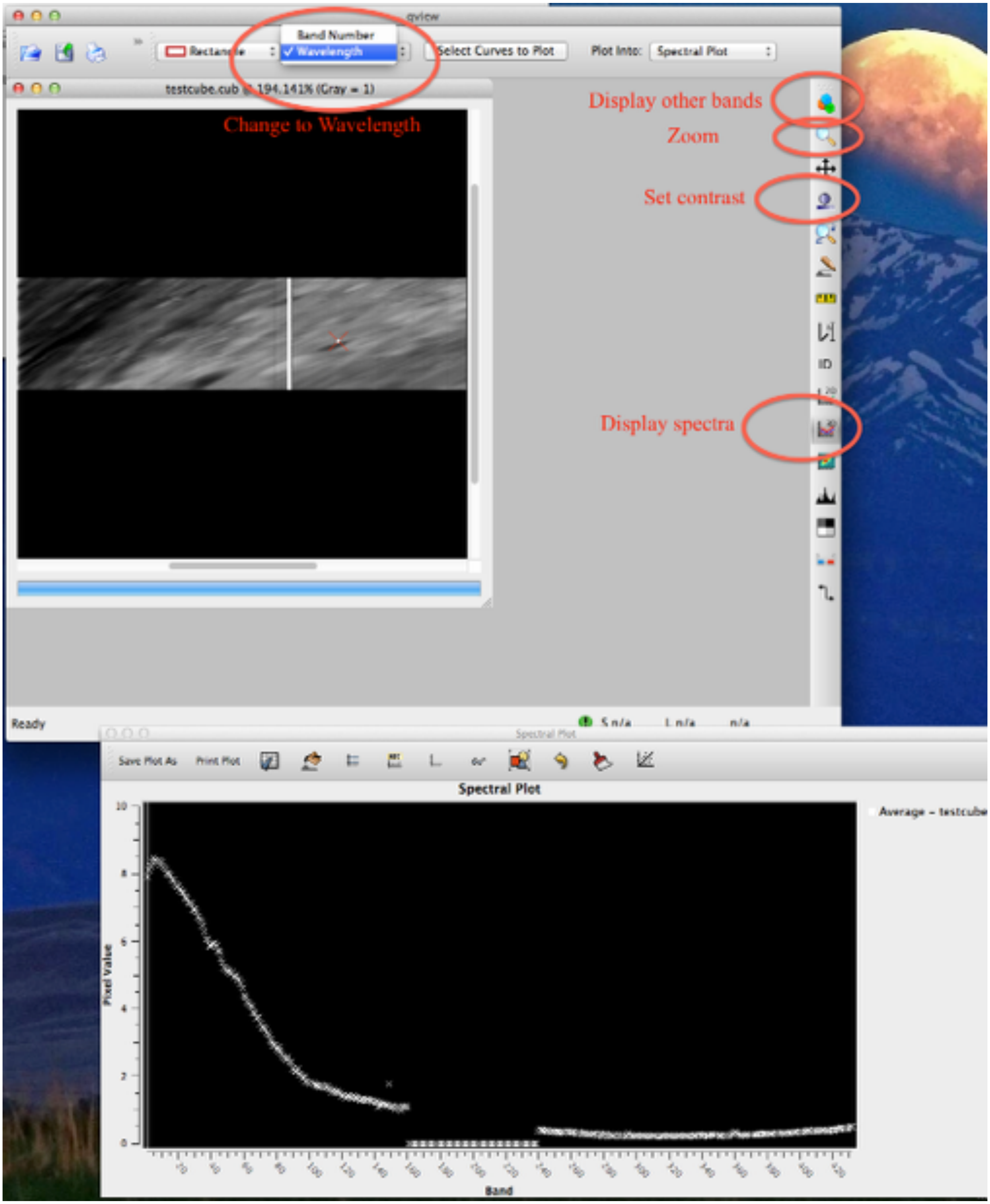
pds2isis is a more universal program to read in data from the PDS. It does not maintain all of the camera specific information, so some features of the data will be lost (such as geometry data and wavelength definitions).

The command line would be:

```
        pds2isis from=VIR_IR_1B_1_366390345_1.LBL to=testcube
```

Step 3: View the data

The program "qview" will show you the data. Use it for your analysis.



Change to Wavelength

Display other bands

Zoom

Set contrast

Display spectra

You can either just type qview and then open the file using the GUI (via menus), or you can do qview <filenname> to have it loaded when it starts.

Example: qview testcube.cub

The menu bar on the right lets you change which mode the program is in (i.e. it changes what happens when you click somewhere).

Note: Some of the released VIR data are calibration cubes and do not contain images of Vesta. If you end up trying one of these, you'll get something that doesn't make sense.

Step 4: Add geometry data to the ISIS cube

During the mission, the navigation team provided detailed files that described the spacecraft and object's position and pointing known as SPICE kernels. You can download them all from the USGS ISIS, which can be a large amount of file space, or you can get only the portion of the data you need at runtime (but you must be connected to the web).

Run the command

```
        spiceinit
```

Once you select the cube you want, the program will add the SPICE kernel info to the cube.

You can start with using the default shape model just to get started, but you will want to use a higher resolution shape model for better coordinates and map projections.

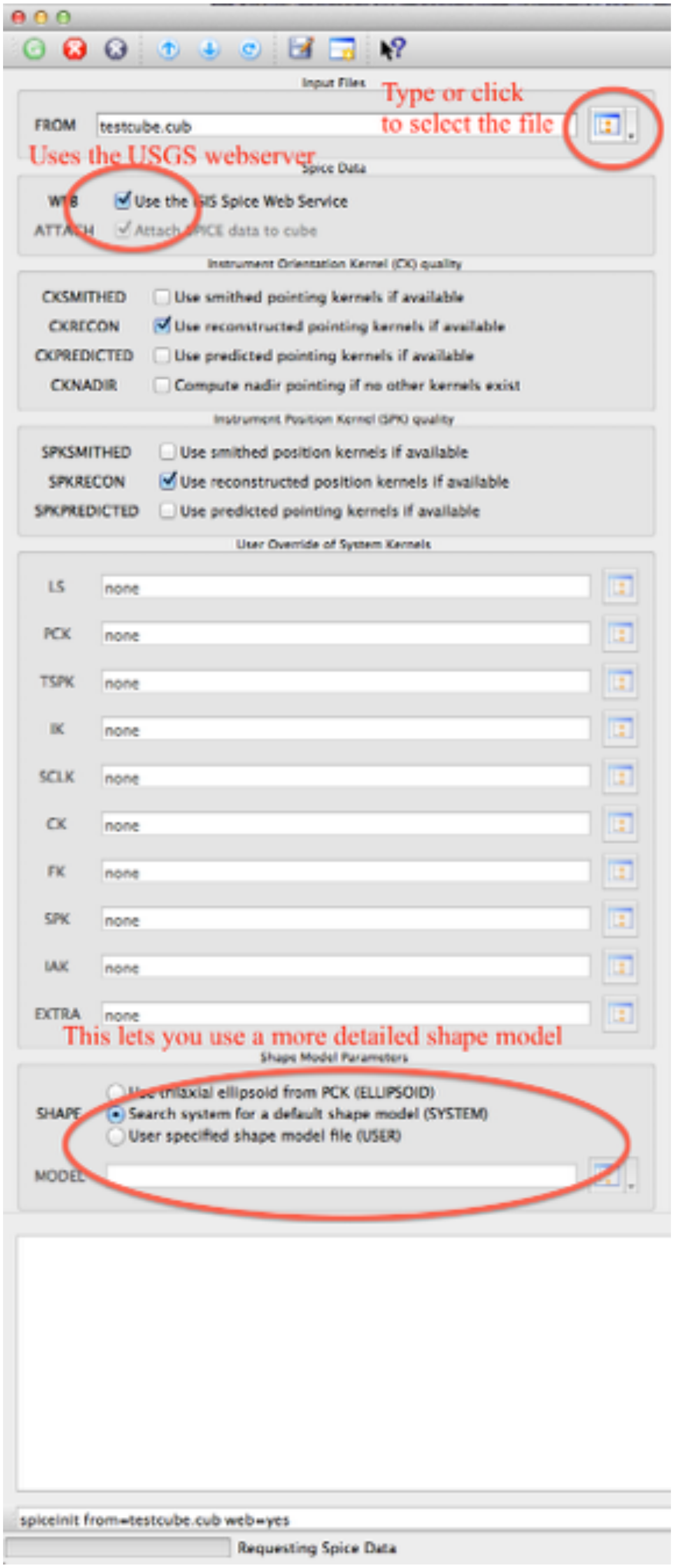
The command line is:

```
spiceinit from=testcube.cub ckpredicted=true
```

Because the encounter with Vesta is over, the best SPICE kernels have been generated, thus you select the CKRECON and SPKRECON options. You can select specific kernels if improvements have been made that haven't been incorporated into the USGS's system.

The WEB allows you to use access the SPICE kernels from the USGS webserver so that you only download the ones you need, which is a huge savings of download and file space. I've had mixed results. To use their server, use:

```
spiceinit from=testcube.cub WEB=true
```



Input Files

FROM testcube.cub

Type or click to select the file

Uses the USGS webserver

Spice Data

WEB ☑ Use the ISIS Spice Web Service

ATTACH ☑ Attach SPICE data to cube

Instrument Orientation Kernel (CK) quality

CKSMITHED □ Use smithed pointing kernels if available

CKRECON ☑ Use reconstructed pointing kernels if available

CKPREDICTED □ Use predicted pointing kernels if available

CKNADIR □ Compute nadir pointing if no other kernels exist

Instrument Position Kernel (SPK) quality

SPKSMITHED □ Use smithed position kernels if available

SPKRECON ☑ Use reconstructed position kernels if available

SPKPREDICTED □ Use predicted pointing kernels if available

User Override of System Kernels

LS	none
PCK	none
TSPK	none
IK	none
SCLK	none
CK	none
FK	none
SPK	none
IAK	none
EXTRA	none

This lets you use a more detailed shape model

Shape Model Parameters

SHAPE
○ Use triaxial ellipsoid from PCK (ELLIPSOID)
● Search system for a default shape model (SYSTEM)
○ User specified shape model file (USER)

MODEL

```
spiceinit from=testcube.cub web=yes
```

Requesting Spice Data

Step 5: Run qview again

Now that the geometry has been added to the spectral cube, you can see the latitude and longitude of your mouse, pixel resolution, distance between two points, area of objects, etc -- all options in qview. Refer to the ISIS documentation for more details.

Step 6: Do other things with the data

Get the photometric back planes from a cube. phocube generates a new cube that has numerous different back planes of very useful data. You can get emission and incidence angles based on a smooth surface, or you can select localemission and localincidence to get it based on the surface topography.

```
phocube from=testcube.cub+35 to=testphoto.cub emission=no incidence=no
localemission=yes localincidence=yes
```

Get basic information about the image, referenced by the camera. This will provide average i, e, phase, min/max lat/lon, observation time, etc.

```
caminfo from=testcube.cub to=info.txt
```

Output an image. This command will generate png images (or other formats) from the cube you want. The +35 tells ISIS which band to use.

```
isis2std from=testcube.cub+35 to=out.png
```

Map project a cube. You can rotate, shift and bend the image so that it conforms to the shape model. This is very useful if you want north up.

```
cam2map from=testcube.cub to=testmap.cub
```

The USGS provides documentation: http://isis.astrogeology.usgs.gov/UserDocs/index.html
and tutorials: http://isis.astrogeology.usgs.gov/IsisWorkshop/index.php/IsisWorkshop

You get something like this, map projected with north up: