Mars Image Viewer

User's Guide (v1.1.3)

Nicholas Toole

nttoole@jpl.nasa.gov

National Aeronautics and
Space Administration

Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California

This document has been reviewed and determined not to contain export controlled technical data.

Table of Contents

I. Introduction

A. Purpose

The Mars Image Viewer (*marsviewer*) is a multi-platform application designed to aid in quality control, browsing, and analysis of original science product images (Experiment Data Records, or EDRs) and derived image data products (Reduced Data Records, or RDRs) returned by in-situ missions.

The initial motivation behind the design was to separate the task of viewing an image set from the task of determining how to locate that set. By selecting an image designated as the primary lookup key, all products associated with that key image can be retrieved and viewed. The lookup and searching behavior is encapsulated in the *file finder* interface.

Many of the derived products (i.e. XYZ, range maps) are not simple viewable images, so *marsviewer* supplies a set of *image contents* which interpret the underlying data and provide a more meaningful visual representation. Using the *overlay* feature, this representation can then be blended with the original image to register features between the original and derived products.

B. Requirements

- Java Runtime Environment (JRE) 1.5 or later
- 128 MB of memory (512+ MB recommended)
- Java Advanced Imaging (JAI) library and JAI IIO Tools
- JAR/Class files containing any custom codec/plugins and utilities

Typically, *marsviewer* will be distributed with required JAR files (i.e. Vicar and PDS libraries) as licensing allows.

C. File Finders

Marsviewer offers an abstraction of the products' organization via a *file finder* interface. File finders serve as a way of separating the tasks of viewing a product set from that of locating the set. By selecting one of the available file finders and a root location (usually a directory), one can focus on selecting source images and let the associated products be loaded automatically. For example, the application "understands" the file structure and filename

conventions of the MER Operational Storage Server (MER OSS), helping the user to navigate this complex file system to find desired images. *Marsviewer* can also work with a flat file system, remote operations file systems, image archive file systems, and others.

Below is a list of currently implemented finders, each with a brief description of their behavior and what root location they expect.

- `MER_OSS` - For MER OSS directory structure. Specified directory should be the surface directory containing the subdirectory tactical/sol.
- `MER_FLAT` - For flat, just select a directory; all MER products should be in that single directory. Files must conform to the MER naming convention.
- `MER_FEI` - for FEI, select any directory and it will strip off the *_inst_type* extension and look for all the proper extensions.
- `MER_REMOTE_OSS` - Similar to MER_OSS finder, with the exception that some RDR images can be used as source images if the corresponding EDR is not found. This is a common case in remote systems.
- `MER_ARCHIVE` - for the PDS Archive structure, select root directory containing volume subdirectories (ex.'mer2op_0111').
- `MER_ARCHIVE_VOLUME` - simpler version of the MER_ARCHIVE finder which searches within a single archive volume. Select two directory levels below the MER_ARCHIVE (e.g. 'data').
- `MER_MOSAIC_OSS` - implementation of the mosaic finder for the MER OSS directory structure. Root directory should be the same as that for MER_OSS.
- `MER_MOSAIC_FLAT` - implementation of the mosaic finder for the MER for a single directory. Root directory should be directory contains all products.
- `PHX_OSS` - For Phoenix OSS directory structure. Specified directory should contain the ‘sol’ subdirectory.
- `PHX_FLAT` - For flat, just select a directory; all Phoenix products should be in that single directory. Files must conform to the Phoenix naming convention.
- `PHX_SCILAN` - Implementation of the Phoenix Science LAN directory structure. Specified directory should contain the ‘sol’ subdirectory.
- `MSL_ODS` - For MSL ODS directory structure. Root should be 'sol' directory within the OPGS directory structure.
- `MSL_ODS_DOY` - Implementation of the MSL ODS directory structure using Day Of Year formatting. Root should be the year directory.

- MSL_FLAT - For flat, just select a directory; all MSL products should be in that single directory. Files must conform to the MSL naming convention.
- MSL_ODS_W10N - Implementation of the MSL ODS data access via *webification* (w10n) service. Root should be a URL to w10n service that contains the sol directory.
- MSL_MOSAIC_ODS - Mosaics within the MSL ODS directory structure. Root should be 'sol' directory within the OPGS directory structure.
- MSL_MOSAIC_FLAT - MSL Mosaics within a single flat directory structure. Files must conform to the MSL naming convention for mosaic products.
- MSL_MMM_ODS - Implementation of the MMM specification for files within the SOAS file structure. Root should be 'sol' directory within the ODS directory structure.

Starting with marsviewer 3.1.0, the ability to create new file finders by combining existing finders was introduced. This opens the possibility of examining products across multiple finder types without switching between specific finders. Note: Combined finders must all share the same root location.

- MSL_CAM_MOSAIC_ODS - Combines MSL_ODS and MSL_MOSAIC_ODS file finders for searching EDR/RDR products as well as mosaics within the ODS directory structure.
- MSL_OPGS_MMM_ODS - Combines MSL_ODS and MSL_MMM_ODS file finders for searching OPGS and MMM products within the ODS directory structure.
- MSL_UBER_ODS – Combines MSL_ODS, MSL_MOSAIC_ODS, and MSL_MMM_ODS file finders for searching across OPGS and MMM products along with OPGS mosaics within the ODS directory structure.

Note that local file finders require the data to be stored on the machine from which you run *marsviewer*. You can however display the application remotely (i.e. using X-Windows or a desktop sharing application.) Remote file finders, (i.e. via webification) access data from a remote host.

D. Image Contents

Image contents are essentially a viewable representation of a derived product. They are classes that encapsulate the original image and image-processing render chain, along with the associated controls and attributes. Given the image product type and mission namespace, an image content factory constructs the appropriate image content instance.

Currently, supported product types include stereo disparity, XYZ, range, mask, arm reachability, radiometrically-corrected, surface normal, slope products, solar energy, preload, roughness images, and others. Undeclared product types are placed within a default image content class.

Image contents are usually identified by the product type id, which is retrievable from the product filename. Missions will often use similar yet different product type ids, so there is a two-step process for determining the correct content type. The first is to take the product type id and the mission namespace and lookup image content metadata that is bound to that pair. This metadata is then used to initialize the image content instance. This allows the same image content classes to be shared amongst different mission definitions.

In the 'Other Tools' section, you will find more information on the *Product Query Tool*, a simple command line application which reads the image configuration file and presents a human-readable listing of namespaces and associated product types.

For specific information on mission defined product types along with associated product ids and descriptions, please refer respective mission documentation (i.e. camera software interface specification).

II. Setting-Up and Starting *marsviewer*

A. Running the application (native launchers)

Native installations are available for supported platforms (i.e. Windows, Mac OSX). These install programs setup the environment with necessary executables and libraries. Below is information based on the platform:

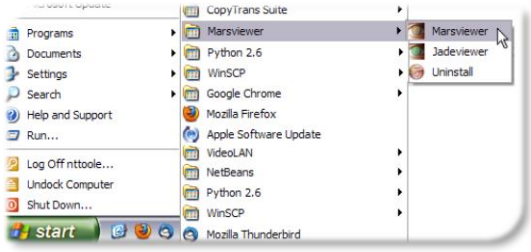
• **Windows** (XP/Vista/Win7):

Installation: Download *marsviewer-installer.exe*. Run the installation wizard, specifying the location to which the application suite will be installed. The default installation location is *%PROGFILES%\JPL\Marsviewer*. (where %*PROGFILES*% should be replaced with the suitable 'Program Files' directory of your system).

NOTE: If you received an error while attempting to write files to new directories, ensure that your account has appropriate permissions set for writing to that location. In Windows 7, you may try right clicking on the installer and selecting 'Run as Administrator'. This allows the installation to create new directories in otherwise protected locations (i.e. 'C:\Program Files\').

Execution: In that install directory you will find two executables (*marsviewer.exe* and *jadeviewer.exe*). Double click on either to run the respective application.

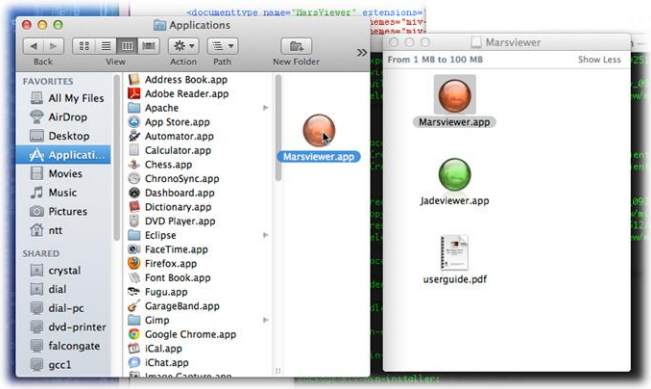
Shortcuts: During the install process, there is an option to install shortcuts. If enabled, these shortcuts can be found on your Desktop and/or the Start Menu.



Note: You can also edit the shortcut properties to provide command line arguments for the application (i.e. set default file finder type and location).

• **Mac OSX**:

Installation: The *Marsviewer* suite is also packaged as a .DMG file. Once this file is expanded, simply drop the *Marsviewer* and *Jadeviewer* apps into your **Applications** folder. Please refer to Apple documentation of installing bundled apps for more information.



Execution: To run, double-click the application icon.

• **Unix/Linux**:

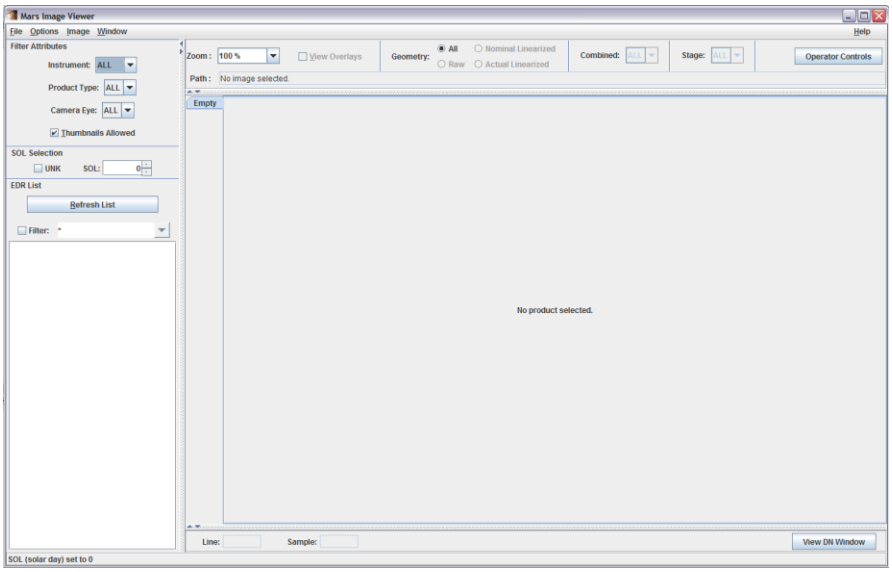
There is currently no native support for *marsviewer* applications on Unix and Linux platforms. Instead, configuration and launcher scripts are used via the command line. These files and the associated libraries are packaged in a .Tar and .Zip files.

Installation: Download the marsviewer.tar (or .zip) file. Create a deployment directory. Expand the archive into the new deployment directory.

Execution: Move into the deployment directory. Setup the environment by running the *use_miv* script. Finally, run the marsviewer or jadeviewer launcher.

```
> cd /apps/marsviewer/
> source use_miv.csh          # or use_miv.sh for Bourne shell
> marsviewer.csh &            # or marsviewer.sh for Bourne shell
> jadeviewer.csh &            # or marsviewer.sh for Bourne shell
```

Note: The launchers can be edited to include an initial file finder location and type if product repository is used regularly.



B. Running the application (command line using Java)

NOTE: This section is geared more towards a mission operations environment, where *marsviewer* has already been installed in the current setup. This would include all necessary JAR (Java archive files) having been set on the CLASSPATH environment variable.

As a Java application, *marsviewer* can be launched using the `java` command.

```
$ java jpl.mipl.mars.viewer.MarsImageViewer
```

Due to memory usage requirements, it is recommended that you include the Java system option -Xmx to set the heap usage limit. For example, to launch with a 512 megabyte heap:

```
$ java -Xmx512M jpl.mipl.mars.viewer.MarsImageViewer
```

Optional arguments can be used to initialize the file finder.

```
$ java jpl.mipl.mars.viewer.MarsImageViewer finderRoot
```

```
$ java jpl.mipl.mars.viewer.MarsImageViewer finderRoot finderType
```

The *finderRoot* parameter is a path to the file finder root location. The *finderType* is the identifier for the file finder implementation (see File Finder section above). If the parameters are not specified at the command line, they will need to be set within the application.

It is recommended that users or operations teams create and utilize a launcher script that combines the file finder parameters with the java command for simpler invocation. For example, the MIPL operations team uses a launcher script called '`marsviewer`' for the MER, Phoenix, and MSL missions (each mission with its own implementation). Launcher scripts have the added

benefit of automatically including configuration properties or overrides (discussed in *Advanced Topics::Configuration Options* section).

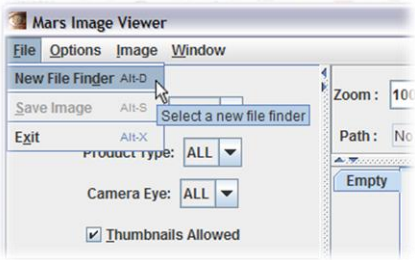
C. Version Information

The current version of the application can be found in the menu bar by selecting **Help → About**. The version information is useful when reporting any issues you may have to the development team.

D. Setting the File Finder

When starting *marsviewer* without both file finder arguments, the first step is to specify the active file finder instance of the session.

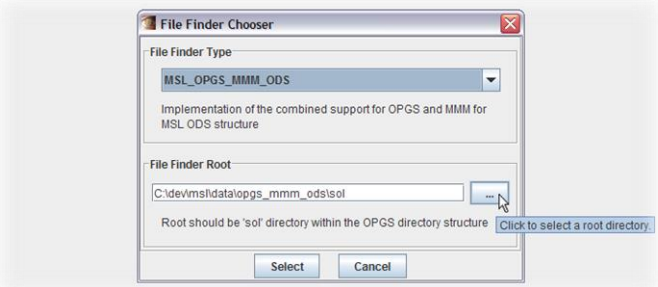
Goto the **File** menu, and select **New File Finder.** A dialog will popup with fields for the type and root directory.



Upon selecting a file finder type, you will then select a root location. You can either type or paste the value directly into the field, or select the '…' button to use a file chooser dialog. The file chooser button will only be enabled for file finders that crawl the local file-system. In the case of finders that access external data (i.e. via *webification*), you will be required to enter the root location into the field. Once both entries are set, click 'Select' to create the new file finder.

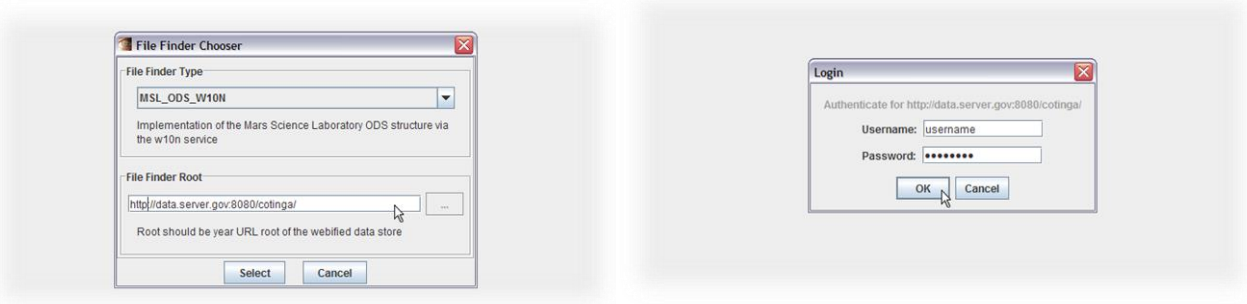
The specific root location for your product distribution falls outside the scope of this document and is dependent upon the mission's configuration and setup. If you are uncertain about the location, please either refer to the mission documentation or contact an individual cognizant of the data deployment. For flat file finders, typically the root location is the directory containing all of the data products.

Please note that descriptions of the finder and the expected root are displayed in the dialog to assist you with the expected structure and root location.



File Finder (local data): *Specify the file finder type followed by root location.*

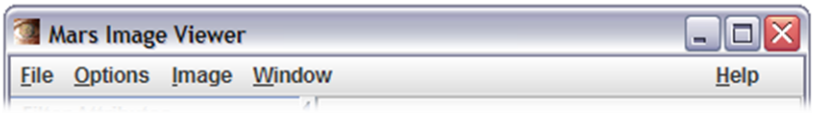
When using a remote access file finder (i.e. w10n), the use of the chooser is slightly different. Upon select a remote finder, the '…' chooser becomes disabled. The specific URL of the data access layer must be entered in the root field. If authentication is required to access the layer, then upon clicking *OK*, a Login dialog will popup requesting user credentials. Three attempts will be made to authenticate the user before quitting.



File Finder (remote data): *Left*: File finder for remote data and service URL.
Right: Authentication popup dialog.

E. Application Menu

Newer versions of *marsviewer (*3.0+) include a restructured application menu layout. The five menus are: **File, Options, Image, Window**, and **Help**. Below is a description of each menu item. More detailed information on particular sub-menus will be provided later.



- **File** Contains application level items allowing for the selection of a new file finder, saving the current displayed product to an image, and exiting the application.

- **Options** Gathers options and settings that control various components and behaviors across the application.

  The *Preferences* sub-menu gives the user control over when options will be cached across application invocations or reset to original default values. The *EDR* and *RDR Options* sub-menus provide controls to viewing and filtering search results.

- **Image** Consists of image-centric operations such as contrast stretch, rotation, scaling and metadata options.

- **Window** Consolidates most of the separate data dialogs and windows of the application. This includes the Operator Control, DN Display, Label View.

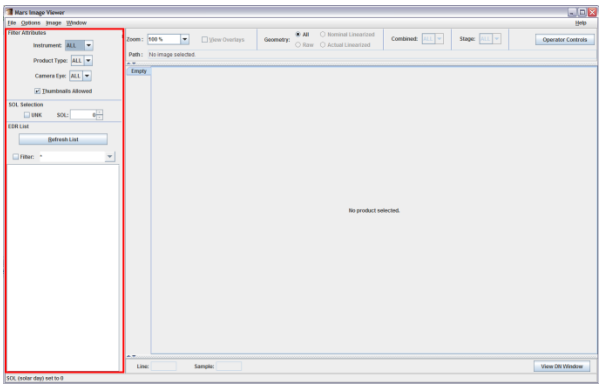
- **Help** Provides information on the application such as version and, if included, user documentation.

III. Querying EDRs

A. EDR Query Panel

Once the file finder has been setup, the next step is to query for available EDRs. Otherwise known as 'source' images, these are the original products that are used as a primary lookup key when querying for RDR's in the next step.

On the left side of the *marsviewer* window is the set of EDR panels. This includes *Filter Attributes*, *SOL Selection* (when applicable) and the *EDR List* panels:



- *Filter Attributes*: Filters the EDR result list by instrument name, product type, and camera eye ID. Can also include or remove thumbnail types.

- *SOL Selection*: For file finders that store products according to sol (or 'solar day') or day-of-year, this control is used to specify the particular sol of interest. The **UNK** checkbox is for selecting a specially designated directory within the structure that contains products with an unspecified sol value. Some file finders (like flat finders) do not require a sol. Once the criteria are all set, you will click the **Refresh List** button to perform the EDR query.

- *EDR List*: This panel displays the EDRs that were returned by the query. It includes the ability to filter the list by wildcard expressions (enable 'Filter' check-box, specify a wildcard expression, and hit *ENTER* key).

B. Usage

Most typically, you will simply select the SOL and click 'Refresh List'. This submits the query and should result in a displayed listing of EDR filenames. It should be noted that the list does not poll the file-system automatically. If you are using *marsviewer* while a product pipeline is executing, you will need to manually *Refresh* the listing periodically to view the new products.

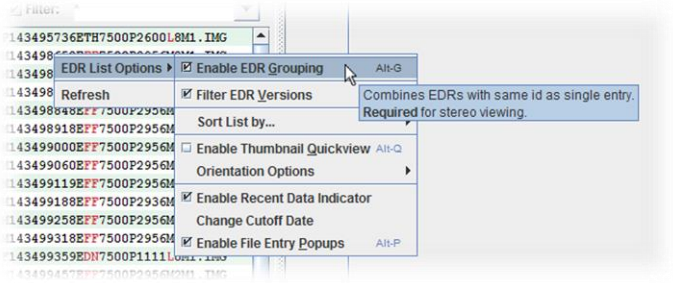
An entry in this list may refer to an individual EDR filename or an EDR Group.

C. EDR Grouping

EDR Grouping is a notion where all of the EDRs in the result set are partitioned and combined into groups based on a shared set of attributes. Most commonly, the group that is displayed is based upon the notion of stereo pairs, where left and right eye EDRs are combined. This common group also includes the full and thumbnail sizes. In the case of mosaics, another grouping may combine across different projection types.

One of the filenames in the group will be selected for the group name as displayed in the list. Syntax highlighting is used to indicate those characters in name that differ in at least one other group entry. For example, if there is a left and right eye pair, then the *eye* characters for that entry will be highlighted in red. Another example is for full-size and thumbnail images in a group, where the *size* characters will be in red. If an entry has no highlighted characters, this suggests that the group contains the single file entry.

To disable (or enable) EDR grouping, open the **EDR List Options** menu and click on the **Enable EDR Grouping** entry. (EDR List Options menu can be access in the menu bar **Options** menu, or by right-clicking the EDR list component).

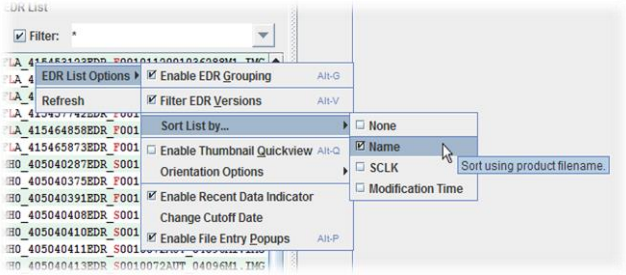


D. Options

Enable EDR Grouping – Already discussed in the previous section, this enables and disables EDR grouping.

Filter EDR Versions – Some mission pipelines are configured so that there can be only a single version of a source EDR. Others generate versions of EDRs and retain them all. This option filters out older versions so that only the highest version of the particular EDR is displayed. Otherwise, all EDRs, regardless of version, are displayed. By default, this option is enabled.

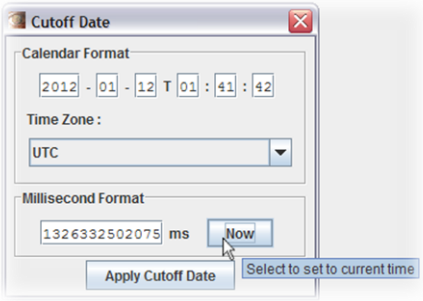
Sort List by… – The list of EDRs can be sorted according to various criteria. Currently supported options are*:* 1) **None** – uses default ordering; 2) **Name** – sorts entries alphabetically by filename; 3) **SCLK** – sorts entries according to SCLK (spacecraft/instrument clock); and 4) **Modification Time** – sorts entries according to latest modification time.



Thumbnail Quickview – Displays a thumbnail next to the EDR entry in the list (if thumbnail image is available). You can orient the thumbnail vertically or horizontally in relation to the text.

Recent Data Indicator – This feature will highlight newer entries in the list Files with modification time after this cutoff will be highlighted. By default, the cutoff is *marsviewer*'s start time.

The **Change Cutoff Time** option provides a dialog to specify a new cutoff time in *Calendar* format: YYYY-MM-DDThh:mm:ss, where 'YYYY' is year greater than 1970, 'MM' is the month 01-12, 'DD' is the day of month 01-31, 'hh' is the hour of day 00-23, 'mm' is minutes 00-59, and 'ss' is seconds 00-59. The time zone can also be specified using the *Time Zone* combo list. The user can also specify the date as the number of milliseconds since the epoch (00:00:00 GMT, Jan 1, 1970), as file modification dates are specified in Java. Note: Cutoff changes are not applied until user selects *Apply Cutoff Date* button.

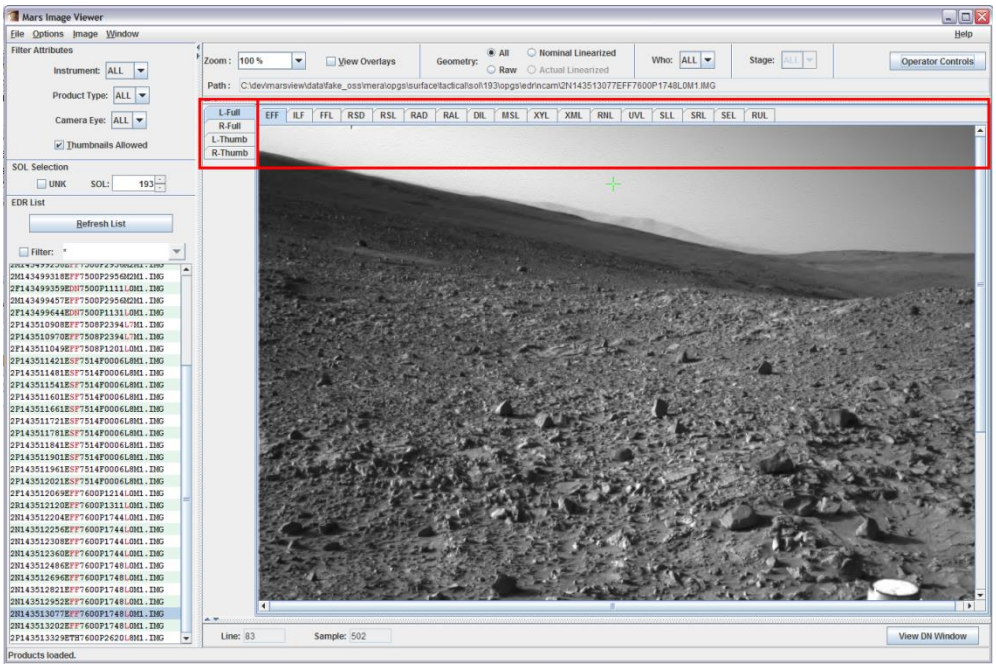


Entry Pop-ups – If enabled and you rest your mouse upon an EDR entry, you will see a popup with the full path of the EDRs that make the entry, as well as other metadata information. While enabled by default, it is recommended that remote users disable this feature to reduce repaint requests.

IV. Querying RDRs

A. RDR Query

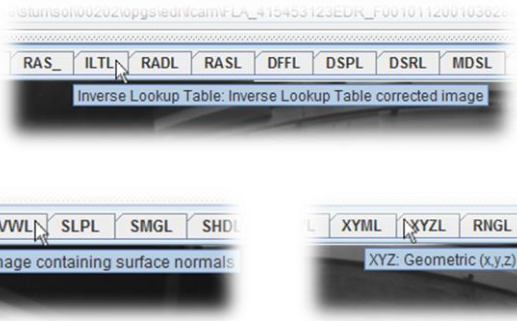
The RDR query is performed automatically by selecting an entry from the EDR list. If EDR grouping is enabled, then RDRs for all of the entries in the group will be queried and then partitioned.



In the example above, an EDR group containing four EDRs (fullsize-left, fullsize-right, thumbnail-left, thumbnail-right) was selected from the EDR listing. The resulting RDR panel contains two tabbed panes (see red bounding box). The first pane (along the left) includes a tab listing the partitions (L-Full, R-Full, L-Thumb, R-Thumb). Typically, users will be interested in the L-Full tab. The second pane, nested within (along the top), contains the various product tabs.

Focusing on the product pane, you can see from the image above a series of tabs with short titles. These titles are created by displaying the product IDs of the RDR image along with any other information that the file finder is configured to present (i.e. geometry, special processing character, etc). In the case of MSL, an extra character is included at the end indicating the geometry ID.

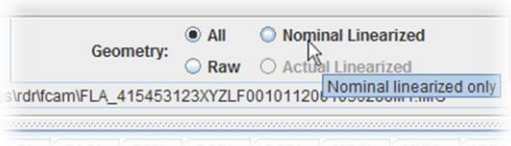
There can be a large number of tabs, and the meaning of the IDs may not be easily recalled. If you hover over a tab, you will see a popup message giving a description of the product type.



B. Subsetting Results

As more missions are added, there seems to be a larger set of RDR products to view. For example, in the case of MSL, there are new product types as well as existing types with different image geometries. As such, we provided some subsetting capabilities.

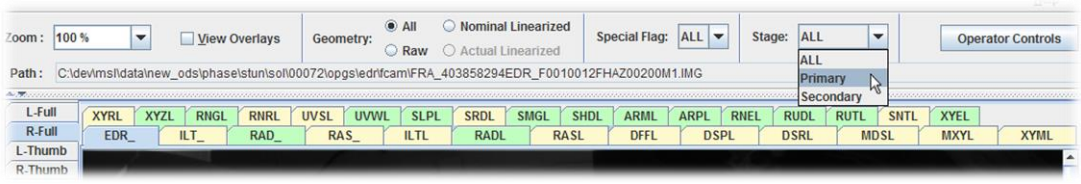
- *Geometry Filter*: There are two primary types of image geometries: *raw* and *linearized*. Raw geometry is the original image with any distortion that might be built into the camera. During processing, a linearization step is applied to create a linear product. Most of the downstream (stereo-derived) products are typically derived from the linearized version. Linearized products can then be divided into Nominal and Actual linearization types. By selecting a particular Geometry in the geometry filter panel, the resulting RDR panel will display only products of that geometry.



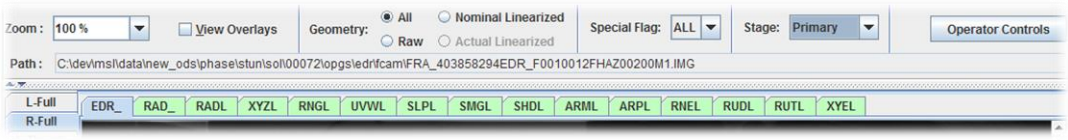
- *Secondary Filter*: Some file finders provide a secondary filter criterion. The particular details are dependent upon the mission itself. Phoenix does not provide one. MER uses the '*Who*' field of the product as a filter (indicating which group generated the product). In the case of MSL, this filter is used to select one of the available Special Processing flags that are present in the filename.

Secondary Filter *Left*: MSL uses 'special flag' field. *Right*: MER uses 'who' field.

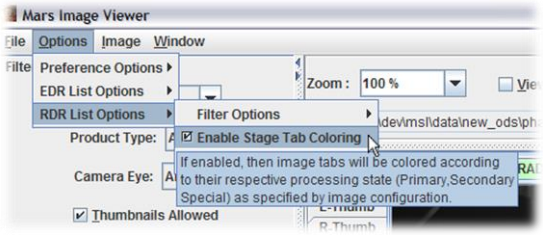
- *Stage Filter*: The MSL mission introduced the notion of a stage, which indicates a level associated to products generated in the processing pipeline. There are three stages: *primary*, *secondary*, and *special*. Primary stage products are those that are typically worth inspecting regularly (i.e. XYZ, range, slope). Secondary products are those that are generally intermediate processes and are viewed less regularly (masks, rover-frame products). Finally, special products are those that warrant inspection as they are not part of the general pipeline.



When enabled, stage coding will also color the product tabs according to their stage. **Green** indicates *primary*, **Yellow** indicates *secondary*, and **Red/Pink** indicates *special*. In the example above, you will see a blue tab. This merely indicates the active tab (active color may depend on your system's native user interface).

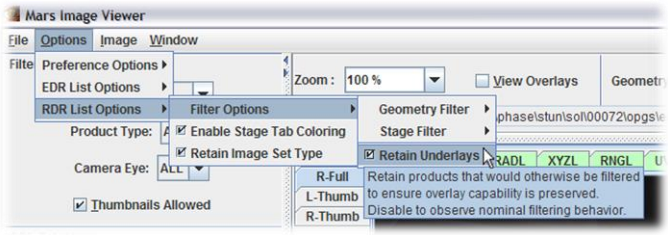


Stage tab color can be activated and deactivated via the RDR List options panel. (**Options → RDR List Options → Enable Stage Tab Coloring**).



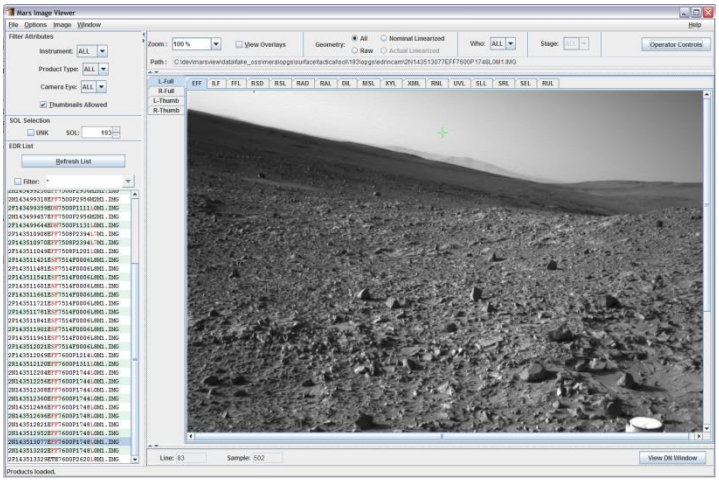
It should be noted that stage assignments are defined by the file finder, and that many file finders do not provide any stage information. In these cases, the tab coloring (and filtering) will not be applied to the product tabs, even if tab coloring is enabled.

When filtering by stage, there is a possibility that overlay behavior can be affected. This is because when applying a particular stage filter (i.e. Special), you will have filtered out the underlying image required for overlays (i.e. the underlying B/W image is most often a *primary*-stage product). To deal with this issue, an application option was added that instructs the filtering mechanism to retain the appropriate underlying image product, no matter what it's stage may be. This option can be access via (**Options → RDR List Options → Filter Options** -> **Retain Underlays**).


Mars Image Viewer
File
Options
Image
Window
Preference Options
EDR List Options
RDR List Options
Filter Options
Enable Stage Tab Coloring
Retain Image Set Type
Geometry Filter
Stage Filter
Retain Underlays
Zoom :
100 %
View Overlays
Product Type:
Camera Eye:
Thumbnails Allowed
R-Full
L-Thumb
R-Thumb
XYZL
RNGL
Retain products that would otherwise be filtered
to ensure overlay capability is preserved.
Disable to observe nominal filtering behavior.

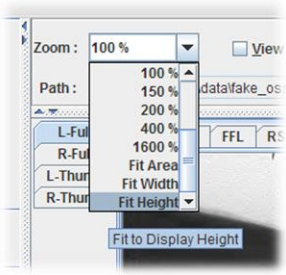
V. Viewing Products: Basic Controls

Let us start with a non-specialized image product that requires no extra visualization. The original EDR, for example, is most commonly a single-band (black and white) image.



Basic Operations:

A) Zoom: Image can be zoomed in or out by selecting a preset value from the **Zoom** combo box or by entering a value in the field. There are special zoom entries that will scale the current image into the viewable display, including fitting the width, height, or area (minimum of the width and height). Maximum zoom is set at 16x normally.

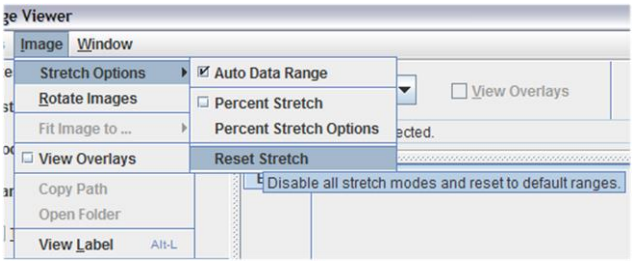


B) Stretch / Contrast Enhancement: For common images, you can apply a stretch operation to the image, which will affect the resulting brightness and contrast. This sets a data range that will be mapped to the min and max pixel value. Stretch can be applied by either selecting a stretch mode from the **Stretch Options** menu or by adjusting the range in the operator control manually (discussed in *Operator Controls* later).

The basic stretch modes are found by selecting from the menu bar **Options → Stretch Options**. The three available modes are i) **Auto Data Range**; ii) **Percent Stretch**; and iii) **Reset Stretch**.

The auto range option scans the image for the lowest and highest pixel values and sets the range to those extrema. The percent stretch applies a similar operation but first throws away a

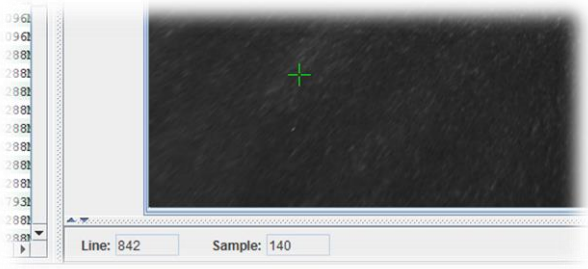
percentage of minimum and maximum outlier values based on the image histogram. The min and max percentage is controlled via the dialog that pops up when **Percent Stretch Options** is selected. **Reset Stretch** will reset the stretch mode and set the min and max ranges back to their respective default values.



While auto data range is useful for a quick contrast adjustment, percent stretch will often provide better results and therefore is recommended. You can see an example of stretching in *Section VI-A* below.

C) Rotation: The images can be rotated using the rotation dialog control. Dialog can be opened by selecting either from the menu bar **Options → Rotate Images**; or by right clicking on the image and selecting **Rotate Images** from the pop-up menu. Angles can be specified in degrees or radians. The rotation applies to all the images in the set.

D) Image Mouse Coordinate Reporting: Using the mouse, try panning over the image. A special cursor is provided in place of the system mouse cursor. The Line and Sample boxes (on the lower left of the RDR pane) report the line and sample values of the pixel the mouse is currently over.

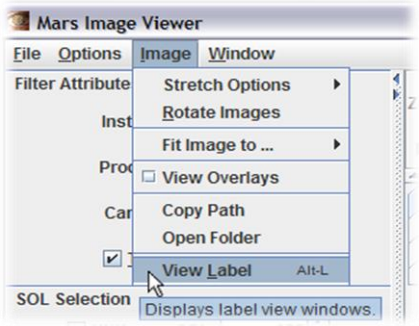


You can lock the special cursor onto a particular coordinate by left-clicking on the location while holding down either the *Control* key (*Control+Left Mouse Button*) or *Alt* (*Alt+Left Mouse Button*) key. At this point, the cursor stays in that position but your regular mouse cursor is available for you to use as normal. Do the same operation again to remove the locked location.

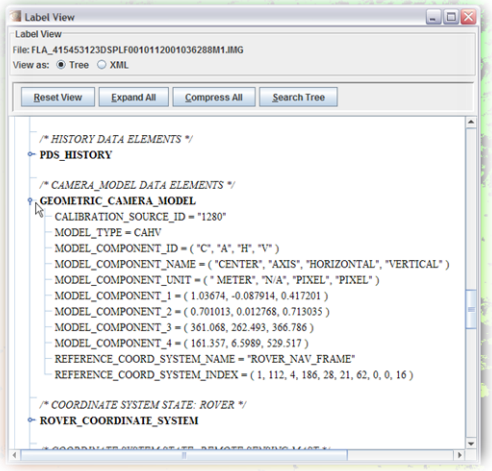
Often when locking the cursor, you may wish to adjust the location only slightly. Manual cursor adjustment is supported by using the arrow keys while holding down the *Alt* key (*Alt+Arrow*).

Sometimes, the default cursor may be difficult to see on the image (for example if the green cursor is over a green part of the image it can be nearly impossible to find.) There are a couple of ways of handling this. The first is to enable the default mouse cursor along with the special cross-hair cursor by holding *Control* and *Shift* down while left-clicking (*Control+Shift+Left Mouse Button*). Do the same to remove the system cursor. The second is to actually change the mouse icon to another pattern. You can iterate across different cursor images by right-clicking on the image and selecting **Next Mouse Icon** or by holding Shift down while left-clicking (*Shift+Left Mouse Button*).

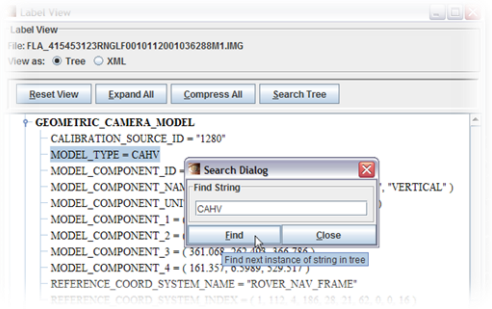
E) Label View: Most images and derived products have a label that is viewable via the Label view dialog. To inspect the label, open the Label dialog by selecting from the menu bar **File → View Label**.



The label view dialog present two views of the label: Tree and XML. The Tree view is an expandable, searchable tree component, while the XML view is the raw XML label.

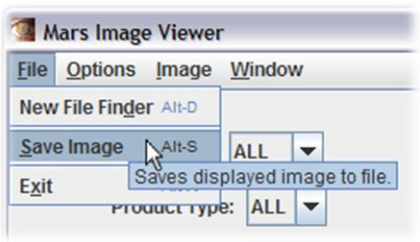


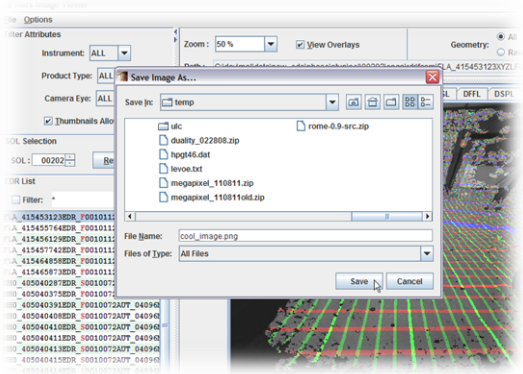
To search the label, select ‘Search Tree’ and enter the target string. Click Find to find the next occurrence from the current location.



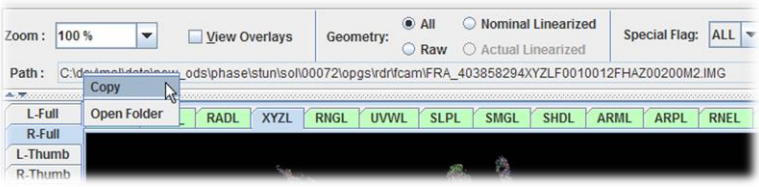
F) Saving Images: Images can be saved on your local store using a standard image format. This will include stretches, rotations, zooms, as well as some advanced features (i.e. overlay) that will be covered in the next section. It should be noted that you are not saving products, but rather the interpreted or visualized images of those products. Some common image formats are JPEG and

PNG. The extension of the saved filename will be used to infer the format, so please stick to common image extensions.

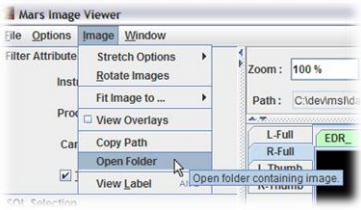


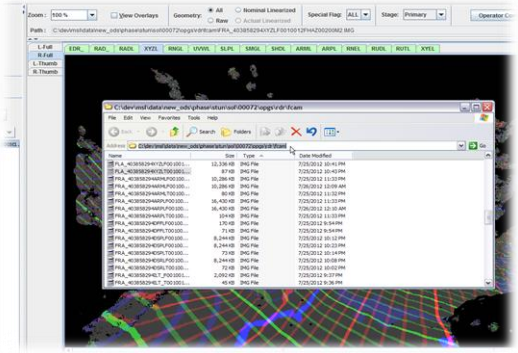


G) File Utility Commands: The application offers a few file based utilities that may be useful. The first, **Copy**, provides a mechanism for copying the location of the currently active product to your system's clipboard. You can copy the entire path of the product by right-clicking on the *Path* label and selecting **Copy**, or by accessing the same option via the application menu or product pane popup menu. To paste the copied value, simply use your system's keyboard shortcut (i.e. Ctrl+V on Windows, Command+V in Mac, etc).



Another utility, **Open Folder**, provides quick access to the location that *contains* the product. If the product is on the local file system, then an attempt will be made to open the system file browser at the appropriate directory. If the file is remote and its location a URL, then your default browser will be opened to the path portion of the URL.

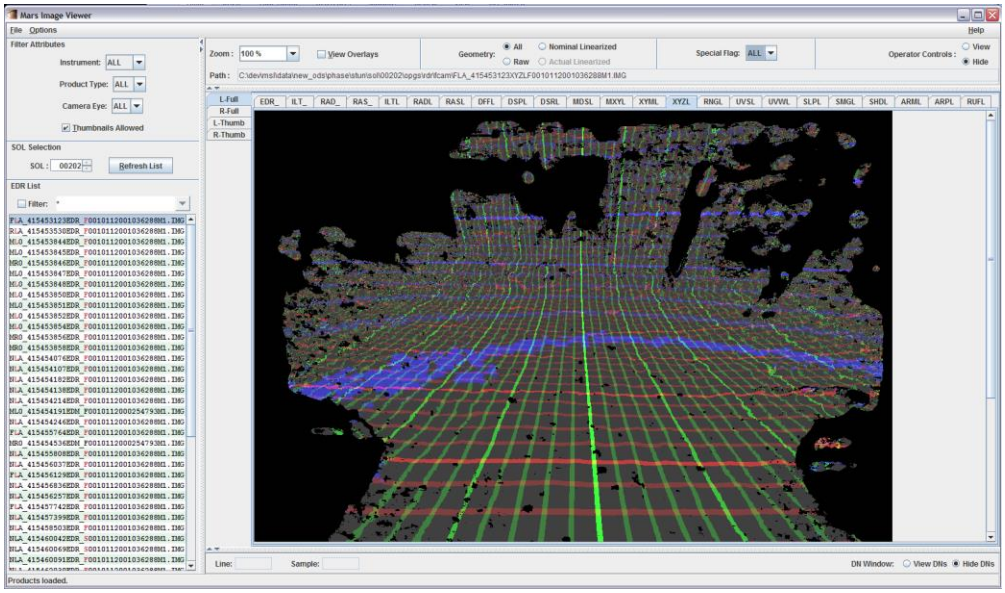




VI. Viewing Products: Image Content Controls

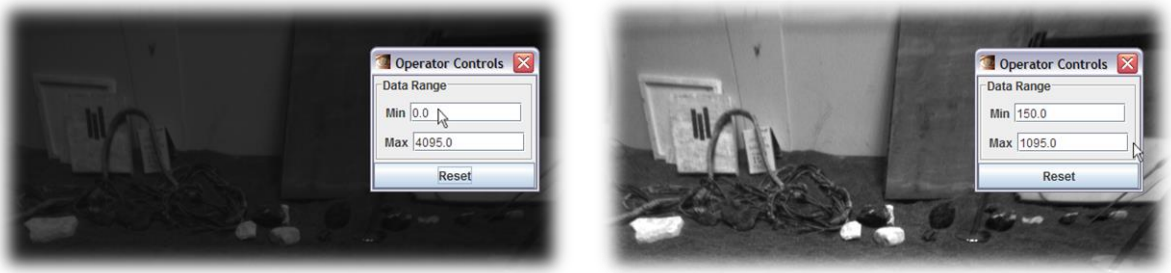
This section moves on to the image content controls, which relate more to the image contents that interpret the underlying product values rather than generic images.

For example, let's assume the user has selected the Product "XYZL", which is a linearized XYZ product for MSL.



A. Operator Controls

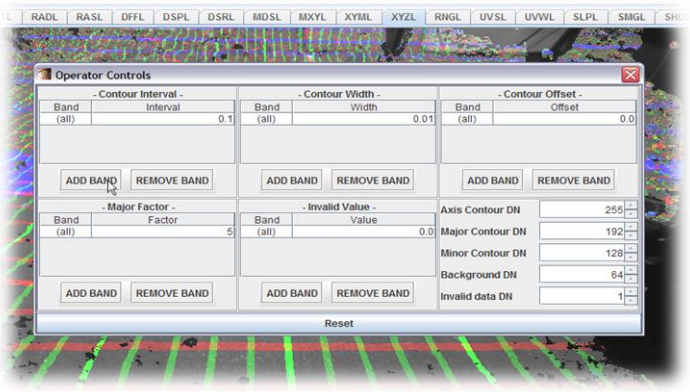
Various image contents have control parameters that affect the interpretation of the underlying data. The most common and basic control is the data range control which affects manual stretching. You read about the stretch modes in the previous section that provide an automatic method for setting this data range, but with the data range control you are able to set these directly. The data range control affects most common images as well as continuously-valued products such as slopes.



Data Range Operator Control; *Left*: Original data range with darker image.
Right: Adjusted data range which produces brighter image with more contrast.

However, in this example, we are working with an XYZ image. The original file content is three-banded float image where each pixel represents an (x,y,z) coordinate. Were you to view this product as an image directly, you would in all likelihood see a dark, noisy image. However, using the XYZ image content to interpret the data, the result is a series of contour lines along the different axes (where red lines indicates areas of constant X, green lines indicate constant Y, and blue lines indicate constant Z). Let's took a look at the associated content control.

On the top-left corner of the application window, you will see a panel labeled **Operator Controls** with a **View** and **Hide** option. By selecting **View**, you will see the control dialog.



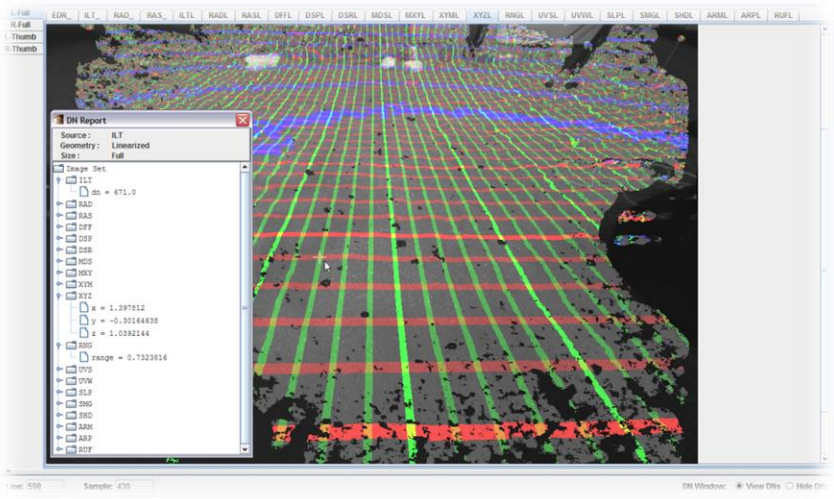
As you can see, this is a more complex dialog that affects the contour intervals, width, and other parameters. By editing and committing a change, the resulting image will be redrawn. For XYZ and range contours, a good rule of thumb is to set the contour width to 1/10 of the contour interval.

The specifics of all the various controls are not covered in this document, but are generally straightforward. It is still recommended that you familiarize yourself with the other product types and their controls.

B. DN Display Window

With a set of co-registered image products, we are able to take the pixel coordinate of any of those images and report the underlying data values for all images in the set at that same pixel location. The *DN Window* provides a view into these values.

Based on the geometry and size of the current image, the control polls the complete RDR result set for all other images with the same geometry and size. It then looks for what it considers to be the source image (usually the original EDR or a radiometrically-corrected RDR) and performs the co-registration lookup via that image.



In this example, we are look at the pixel at location (430, 598). The values at the location for the following products are: the ILT (inverse lookup table) = 671; XYZ = (1.397812, -0.30164638,1.0392144), and RNG (range) = 0.7323816.

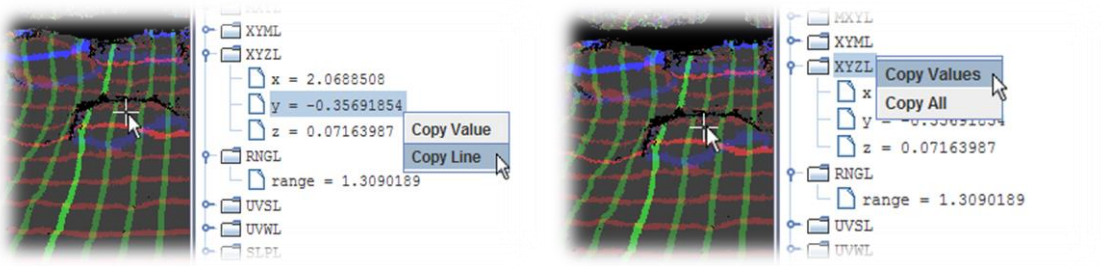
DN Display: Copying Values

As of version 3.0, *marsviewer* supports copying values from the DN window to the system clipboard. It should be noted that the cursor should be planted onto a particular position, otherwise you are likely to copy '*N/A'* values.

There are two classes of nodes which can be copied: 1) value nodes (leaf); and 2) product nodes (parent).

For a value node, there are two copy modes: 1) *Copy Value* – copies only the right-hand side of the '*name = value'* entry (i.e. 'value' only); 2) *Copy Line* – copies the entire string of '*name = value'*.

For a product node, there are two copy modes: 1) *Copy Values* - creates a comma-separated string of the right-hand side of underlying value nodes; and 2) Copy All – creates a multi-lined string include the product type and then a series of lines for the children value nodes.



DN Copy *Left*: Copying from a value node. *Right*: Copying from a product node.

In the above example on the left, copying is performed on a value node. Selecting '*Copy Line'* results in the string '`y = -0.35691854`'. '*Copy Value'* results in '`-0.35691854`'.

In the example on the right, copying is performed on a product node. Selecting '*Copy Value'* results in the string '`2.0688508,-0.35691854,0.07163987`'. '*Copy All'* results in the following as a single, yet multi-lined, string:

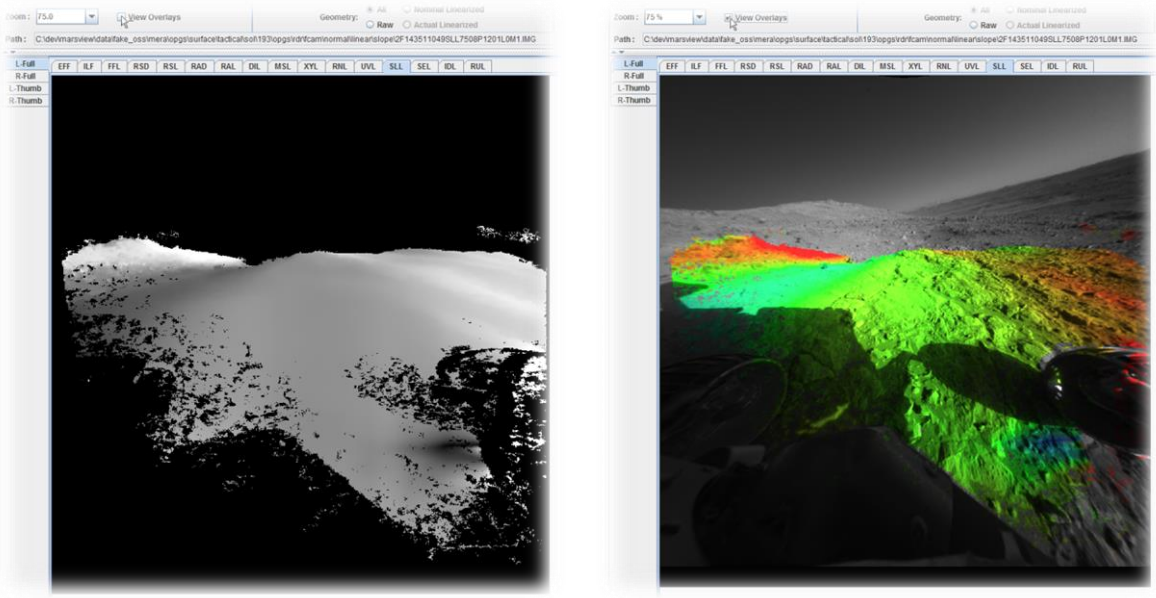
```
XYZL
x = 2.0688508
y = -0.35691854
z = 0.07163987
```

C. Overlays

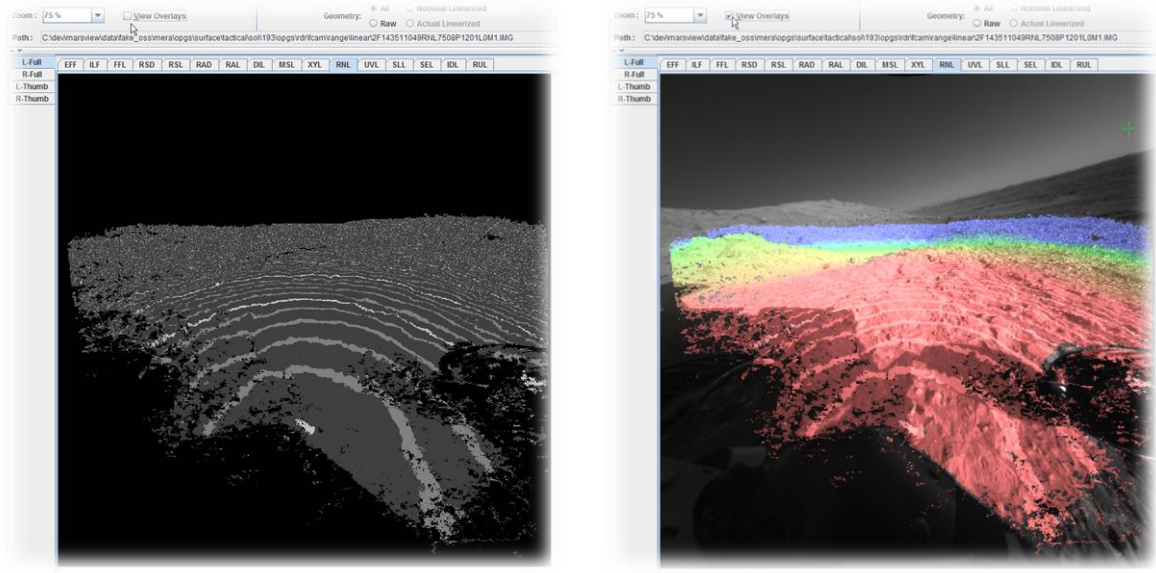
Perhaps the greatest benefit of co-registered image products is the afforded ability to stack one image over another to get a visual representation of the relation between the two. Specifically, you can overlay an RDR product over the original (or linearized source) image.

Beyond matching product data to image data pixel by pixel, there can also be the opportunity to make the product data more readable. For example, what might be a grayscale product image can be assigned a series of hues to indicate which levels are considered 'good' vs 'bad', and levels in-between. By convention, a rainbow is used where blue indicates 'best' values and red the 'worst', with other colors (green, yellow, orange) in between. The meaning of 'good' and 'bad' depend on the product type. For example, a high solar energy value is considered good while a low slope value is considered good (easier traversal). The thresholds for good and bad are generally set by the data range in the *Operator Control* dialog.

If you find that you need to stretch the underlying image (black and white) in order to see the overlay better, the best option is to use *Percent Stretch* or *Auto Data Range*. However, you can also select the underlying image from the product pane, and set its data range explicitly, and go back to the RDR overlay.



Overlay with Slope Product; *Left*: Slope product image. Right: With *Overlay* enabled, the same product. Blue considered flat, green safe slopes, and red considered dangerous slopes.



Overlay with Range Product; *Left*: Range product image. Right: With *Overlay* enabled, the same contours but with hue added to indicate distance from origin. Red at origin, blue away.

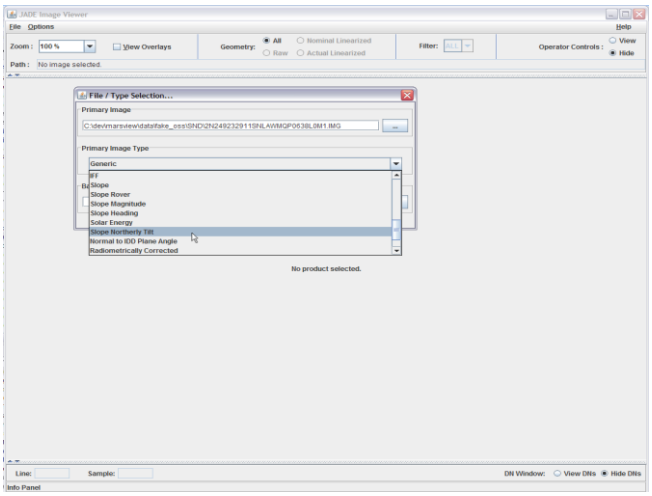
VII. Other Tools

A. Jade Image Viewer

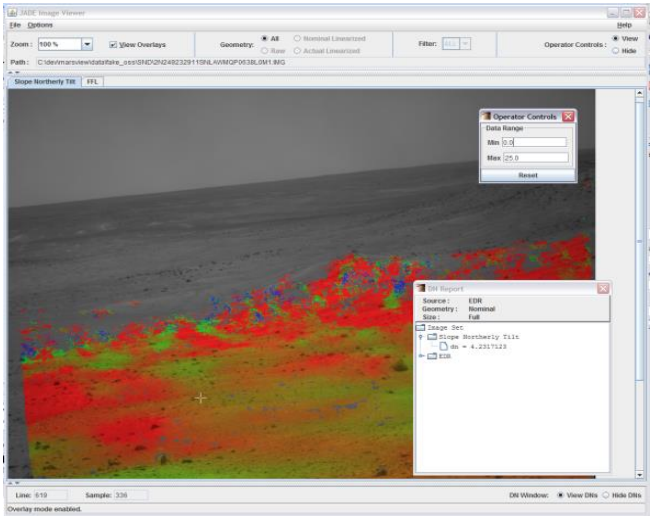
The Jade Image Viewer (*jadeviewer*) is a stand-alone image viewing application derived from the Mars Image Viewer.

The difference lies in where *marsviewer* searches for and displays products, *jadeviewer* only displays them. This affords the ability to apply functionality present in *marsviewer* to a less structured image set or a single product.

Because *jadeviewer* does not rely on the file finder framework, it cannot infer the product type from the filename. As such, when loading a product, the user must specify the product type explicitly.



If a background image is specified, then the overlay feature is enabled and can be utilized. Along with basic operations like zoom, stretch, and rotate, note that operator controls, DN Window reporting and label views are also available.



Usage examples for *jadeviewer:*

1) Start with no initial image product:

```
$ java jpl.mipl.jade.viewer.ImageViewer
```

2) Start with an initial image product and type ID (notice the product type after the filename):

```
$ java jpl.mipl.jade.viewer.ImageViewer /mer/prods/2N249232911SNLAWMQP0638L0M1.IMG SLP

$ java jpl.mipl.jade.viewer.ImageViewer /mer/prods/2N249232911RUTAWMQP0638L0M1.IMG
MER:RUT
```

If image content associated with product ID is unique across mission namespaces, then the product ID alone is sufficient. In the first example, *SLP* refers to same slope image content in MER, Phoenix and MSL, hence there is no conflict.

However, if there is a collision between mission namespaces, then the appropriate mission namespace must be prepended in the format as shown in the second example. The RUT image contents between MER and MSL differ (MER: a generic roughness thumbnail; MSL: an arm DRT roughness), thus the product type ID must be identified as 'MER:RUT' to indicate the MER version is being specified.

3) Start with an initial image product and background image (background type is usually assumed to be EDR but can be specified as fourth argument if desired):

```
$ java jpl.mipl.jade.viewer.ImageViewer /mer/prods/2N249232911SNLAWMQP0638L0M1.IMG SLP
/mer/edrs/2N249232911FFLAWMQP0638L0M1.IMG

$ java jpl.mipl.jade.viewer.ImageViewer /mer/prods/2N249232911SNLAWMQP0638L0M1.IMG SLP
/mer/edrs/2N249232911FFLAWMQP0638L0M1.IMG FFL
```

B. Simple Overlayer

The *SimpleOverlayer* application is a command line tool that can be used to generate overlay images without a graphical user interface like *marsviewer* or *jadeviewer*.

From the application usage message:

```
    Simple overlay program that accepts an image with type
    and associated background image with same geometry,
    constructs an overlay product based on type,
    and writes results to file.

  Usage:

    java jpl.mipl.jade.SimpleOverlayer [parameters]
```

```
Parameters:

   + File Specification (Required):
      - Input. Specify the following options:
        (1)  -i  image_file   }- overlay image file
        (2)  -t  image_type   }- overlay image type
        (3)  -b  bg_file      }- background image file

      - Ouput.  Specify filename to which result is written:
        (1)  -o output_file      }- output file name

   + Optional flags:
        (1)  -v                  }- enables verbose output
        (2)  -h                  }- prints this message
        (3)  -c                  }- enables operator control
        (4)  -p type:name=value  }- sets operator setting
        (5)  -z scaleValue       }- Sets scale ('1.0' = full)
        (6)  -g bg_type          }- background image type (default: EDR)
```

Example:

```
java jpl.mipl.jade.SimpleOverlayer –i /mission/prods/2N249232911SNLAWMQP0638L0M1.IMG -
t SNL -b /mission/edrs/2N249232911FFLAWMQP0638L0M1.IMG -o /user/home/snlOverlay.jpg -g
FFL -p FFL:dataRangeMin=300.0
```

C. Product Query Tool

Marsviewer, *Jadeview*, and *SimpleOverlayer* all use a common image configuration framework that manages mission products and their associated image content handlers. The configuration file is an XML document and can sometimes be embedded in a Java ARchive (JAR) file. Thus, it may be a burden to determine the product types that are supported.

The *ProductQueryTool* is a command line application that provides a more readable interface for the image configuration file. With it you can query product information for a particular mission namespace or across all namespaces. The product type, associated image content name, and a description can be displayed.

Note: By convention, if a product type is not recognized, it is treated as a standard single or triple banded image. This is the case for EDRs across missions.

From the application usage message:

```
ProductQueryTool Usage:

  $ java jpl.mipl.mars.viewer.image.config.ProductQueryTool
  Displays product information for all mission namespaces

  $ java jpl.mipl.mars.viewer.image.config.ProductQueryTool –missions
  Displays supported mission namespaces
```

```
  $ java jpl.mipl.mars.viewer.image.config.ProductQueryTool MISSION:
  Displays product information for the specified mission namespace
  Note the required colon.

  $ java jpl.mipl.mars.viewer.image.config.ProductQueryTool PRODUCT
  Displays information for the specified product across namespaces

  $ java jpl.mipl.mars.viewer.image.config.ProductQueryTool MISSION:PRODUCT
  Displays information for the specified product and namespace

  $ java jpl.mipl.mars.viewer.image.config.ProductQueryTool -help
  Displays this usage message
```

Example: To list the product types defined in the MSL namespace:

```
java jpl.mipl.mars.viewer.image.config.ProductQueryTool –mission
```

Output:

```
Mission namespaces:
  DEFAULT:
  MER:
  MSL:
  PHX:
```

Example: To list the product types defined in the *MSL* namespace:

```
java jpl.mipl.mars.viewer.image.config.ProductQueryTool MSL:
```

VIII. Advanced Topics

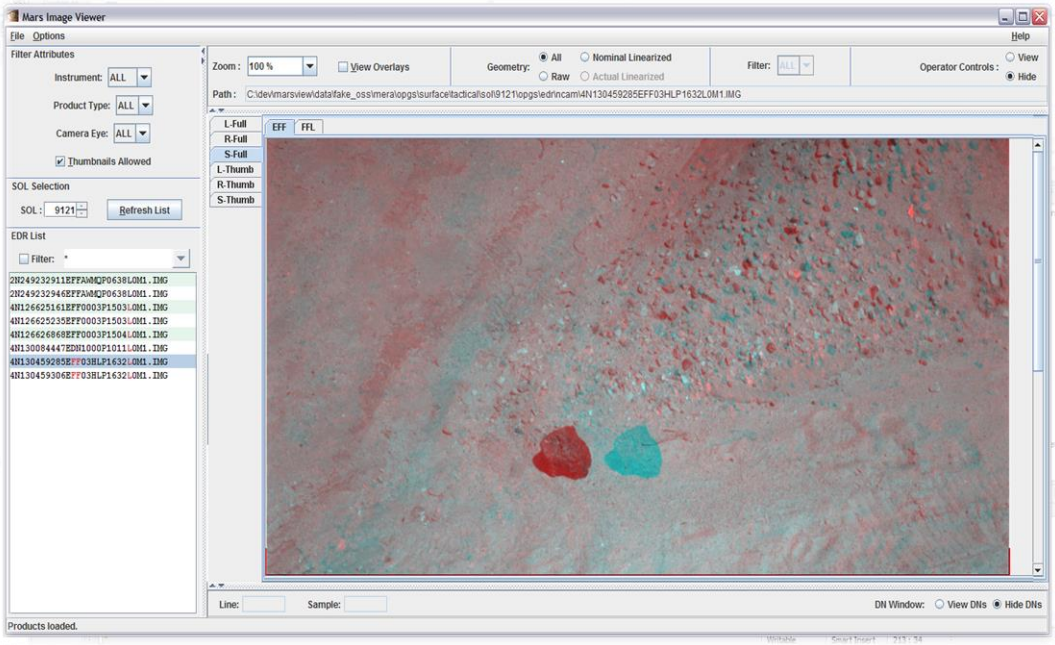
"Don't touch that please! Your primitive intellect wouldn't understand alloys and compositions and things with…molecular structures!" ~ Ash, *Army of Darkness*

A. Stereo Viewing

A feature that was added to *marsviewer* later in its development was a stereo display capability, that is, you can see stereo pairs in stereo! Support for software and hardware stereo was added through the integration of the Jadis toolkit. Jadis (Java Advanced Display Infrastracture for Stereo) provides a common interface for displaying Swing GUI components in stereo using either specialized stereo display hardware (e.g. liquid crystal shutter or polarized glasses) or anaglyph display (red/blue glasses) on standard computer displays.

By default, stereo mode is disabled. You can enable stereo by using the configuration option `miv.view.mode` discussed in the next section. Once enabled, you can specify the type of stereo mode using the `miv.stereo.mode` option. Most systems will support *anaglyph* mode, but only systems configured with the appropriate hardware and drivers will support *hardware/gl* mode (which incorporates shutter or polarized glasses).

Stereo requires EDR Grouping to be enabled to facilitate stereo pairing. When you load a set of RDRs, you will notice that there will be new entries on the left-tabbed pane. Typically these will be 'S-Full' and 'S-Thumb', representing stereo-fullsized and stereo-thumbnails, respectively. By selecting a product in one of these tabs, you will see a stereo view of the left and right products. Operator controls should allow you access to the individual controls of the image content as associated with each eye.



1) Running *marsviewer* with Anaglyph stereo:

```
$ java -Xmx512M -Dmiv.view.mode=stereo -Dmiv.stereo.mode=anaglyph \
       jpl.mipl.mars.viewer.MarsImageViewer
```

2) Running *marsviewer* with Hardware/GL stereo (requires necessary hardware and drivers):

```
$ java -Xmx512M -Dmiv.view.mode=stereo -Dmiv.stereo.mode=gl \
       jpl.mipl.mars.viewer.MarsImageViewer
```

B. Configuration Options

Marsviewer uses a set of application properties to guide configuration behavior. While these properties have defaults values, they can be overridden using the Java property syntax at start-up.

$ java –D*name1=value1* –D*name2=value2* jpl.mipl.mars…

Next is a list of *marsviewer* properties and their descriptions:

`miv.ui.lnf`	Property that determines which look and feel to use with UI. If set to 'native', then the UI associated with the native window system will be used. Otherwise, or in the case of an error in the option specification, the cross-platform UI will be used.

miv.view.mode	Property that determines which type of frame to use when constructing the application. If not defined, the standard frame is used. Values can be 'stereo' or 'mode_standard'. Default is the latter.
miv.stereo.mode	Property that determines which stereo mode to use when constructing a stereo application. If not defined, the anaglyph is used. Values can be 'anaglpyh' or 'hardware'
miv.image.config	Property name for overriding the image content and product type configuration file. The default location (relative to Jar file) is /jpl/mipl/mars/viewer/image/config/image_config.xml
file.finder.config	Property name for overriding the file finder configuration file. The default location (relative to Jar file) is /jpl/mipl/mars/finder/config/filefinder.xml
miv.group.ids.enable	Property name to specify whether EDR grouping is enabled by default.
miv.imageloader.use	Property that controls how application will choose between codec (*fileload*) and plugin (*imageread*) for image loading operation. Legal values are 'codec' and 'plugin'. Default is 'codec'
miv.edr.sort.mode	Property that controls the initial EDR list sort mode for the application. Legal values are 'modtime' (file modification time), 'sclk' (spacecraft/instrument clock) and 'none' (no sorting). Default is 'none'.
miv.idd.always.label	Property that requires the label of IDD reachability (usually on MER or Phoenix) products to be queried for the band names. If not set, then the band names are inferred from product type. Note: Enabling this option will introduce a delay.
miv.arm.always.label	Property that requires the label of Arm reachability (usually on MSL) products to be queried for the band names. If not set, then the band names are inferred from product type. Note: Enabling this option will introduce a delay.
miv.disable.ee	Property that takes a boolean value to indicate that any easter egg features should be disabled. Default set to false.

jade.repaint.mode	Property that controls the repaint mode for Jade displays. Legal values are 'immediate', 'cache', and 'deferred'. The default setting for this is *CACHED* mode.
miv.edr.sort.mode	Property that controls the initial sort mode of the EDR list panel. Current supported values are 'name' (filename), 'modtime' (file modification time), 'sclk' (product SCLK), or 'none' (no special sorting applied).
miv.popupmenu.lightweight	Property that, when assigned to true, allows for pop-up menus to use lightweight components. Note: This was disabled by default to resolve a repaint issue where the menu is painted over by the underlying image.
miv.disable.prefs	Property that, when assigned to true, disables option preferences and uses default values for application state.
miv.documentation.location	Property that overrides the default documentation location. Expected value is a formatted URL which will refer to user documentation.
miv.documentation.disabled	Property that, when assigned to true, disables any attempt to load user documentation.
miv.https.check.disabled	Property that, when assigned to true, configures HTTP secure client to accept connections over https to servers without stringent checks.

Phoenix-specific: Unlike MER and MSL, Phoenix stereo pairs rarely share the same SCLK (spacecraft clock time) in the left and right filenames. Instead, Phoenix uses a shared group id that can be found in the image label (inefficient) or by performing a database lookup (much faster). The database lookup module requires values for the following properties to set the database parameters:

phx.db.url	Database location via JDBC URL (i.e. 'jdbc:mysql://miplphx3:3306/phoenix_ops').
phx.db.driver	JDBC driver implementation (i.e. 'com.mysql.jdbc.Driver')
phx.db.user	Database username (i.e. 'phxrt')
phx.db.pass	Database user password
phx.db.defaults	Boolean value indicating default values be used (applies to driver, username, and password)

MSL-specific: Some properties that affect MSL file finders are listed here.

`miv.msl.ods.failonerror`	Property value that, when set to true, treats missing directories as an error (potentially with message dialog) rather than a warning printed to standard error.
`miv.msl.filter.opgs.mmm`	For MSL file finders that combine OPGS and MMM generated products, this property, when set to true, will remove OPGS versions of MMM products from results sets
`miv.mslmosaic.ods.failonerror`	Property value that, when set to true, treats missing directories as an error (potentially with message dialog) rather than a warning printed to standard error.

JAI Properties: Some properties can be set to control the behavior of the Java Advanced Imaging library. See JAI documentation for more information. Common properties include:

`com.sun.media.jai.disableMediaLib:` JAI can be run without its native acceleration layer without loss of functionality. This may be accomplished by setting this system property to true. Note that when running without the native layer unless the aforementioned property is set to true warning messages will be emitted to indicate the inability to load the native wrapper or libraries.