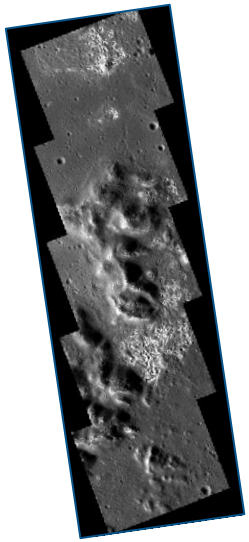
USGS
science for a changing world

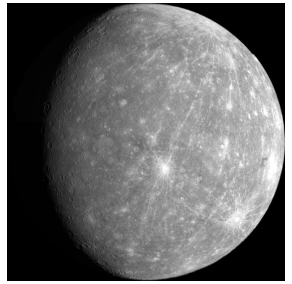
ASTROGEOLOGY SCIENCE CENTER



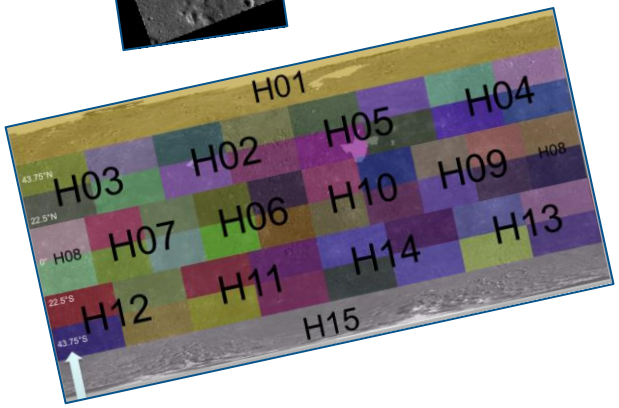
MESSENGER MDIS Data Users' Workshop

46th LPSC

March 15, 2015



Presented by



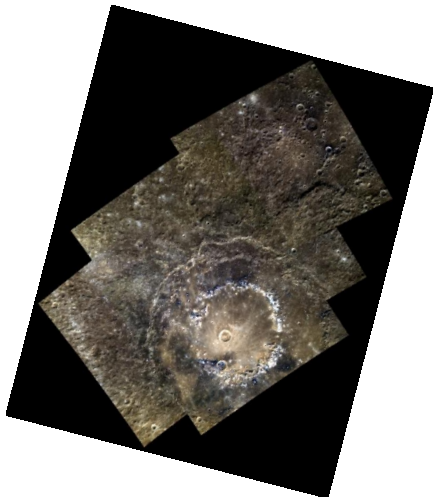
Kris Becker

Computer Scientist, kbecker@usgs.gov

&

Tammy Becker

Cartographer, tbecker@usgs.gov



ISIS3 Tutorial and MESSENGER MDIS Data Users' Workshop

- **Objective:** Provide an introduction to ISIS3 and demonstrate its use in processing of MESSENGER MDIS data

  - ISIS3 Fundamentals
  - Standard Processing Concepts and Tools
  - Cartographic Map Projections
  - Hands-On Lessons
    - Create a Monochrome Map Mosaic (EDRs-WAC)
    - Create a 3-Color Map Mosaic (EDRs-WAC)
    - Color processing of an 8-Wavelength Observation
    - Working with PDS Map Products
      - Monochrome MDIS BDR Mosaic
      - Color MDIS MDR Mosaic
    - Working with Very High Resolution Data (NAC)
  - Export and Application Support for ISIS3 Products

ISIS - Integrated Software for Imagers and Spectrometers

- Over 300 image processing applications
- Strong emphasis on geometric functionality
  - Photogrammetry / Camera models
  - Cartography / Map projections
  - Adjust/Improve Instrument Position and Orientation
  - Generate Digital Map Mosaics
- In use for over 30 years
  - Software generations: PICS, ISIS2, ISIS3
- Support for over 55 NASA/ESA instruments
  - Framing Cameras (e.g, MDIS Narrow Angle and Wide Angle cameras)
  - Line Scan Cameras (e.g., THEMIS-IR, HiRISE)
  - Push Frame Cameras (e.g., LROC-WAC)
  - Special Cases
    - Radar Instruments (e.g., LROC-MiniRF)
    - Spot Instruments (e.g., Cassini-VIMS)

Mission Instruments Supported by ISIS3

Moon
- Lunar Orbiter III, IV, &, V (Medium and HiRes)
- Clementine UVVIS, NIR, HIRES, & LWIR
- Apollo Metric 15/16/17
- Apollo Panoramic 15/16/17
- Lunar Reconnaissance Orbiter NACL, NACR, WAC (VIS & UV), MiniRF
- Chandrayaan-1 M3, MiniRF
- Kaguya MI (VIS & NIR)

Mercury
- Mariner 10 (A & B)
- MESSENGER MDIS (NAC & WAC)

Asteroids
- Dawn FC (1 & 2), VIR
- Near Earth Asteroid Rendezvous Shoemaker MSI
- Hayabusa AMICA
- Upcoming – OSIRIS-REx OCAMS

Multiple Target Bodies
- Voyager 1 & 2 (NAC & WAC)

Mars
- Mars Global Surveyor MOC (NAC & WAC)
- Mars Odyssey THEMIS (VIS & IR)
- Mars Express HRSC
- Mars Reconnaissance Orbiter HiRISE, CTX, MARCI, CRISM
- Viking Orbiter 1 & 2 (A & B)

Jovian
- Galileo SSI

Saturnian
- Cassini ISS (NAC & WAC), VIMS

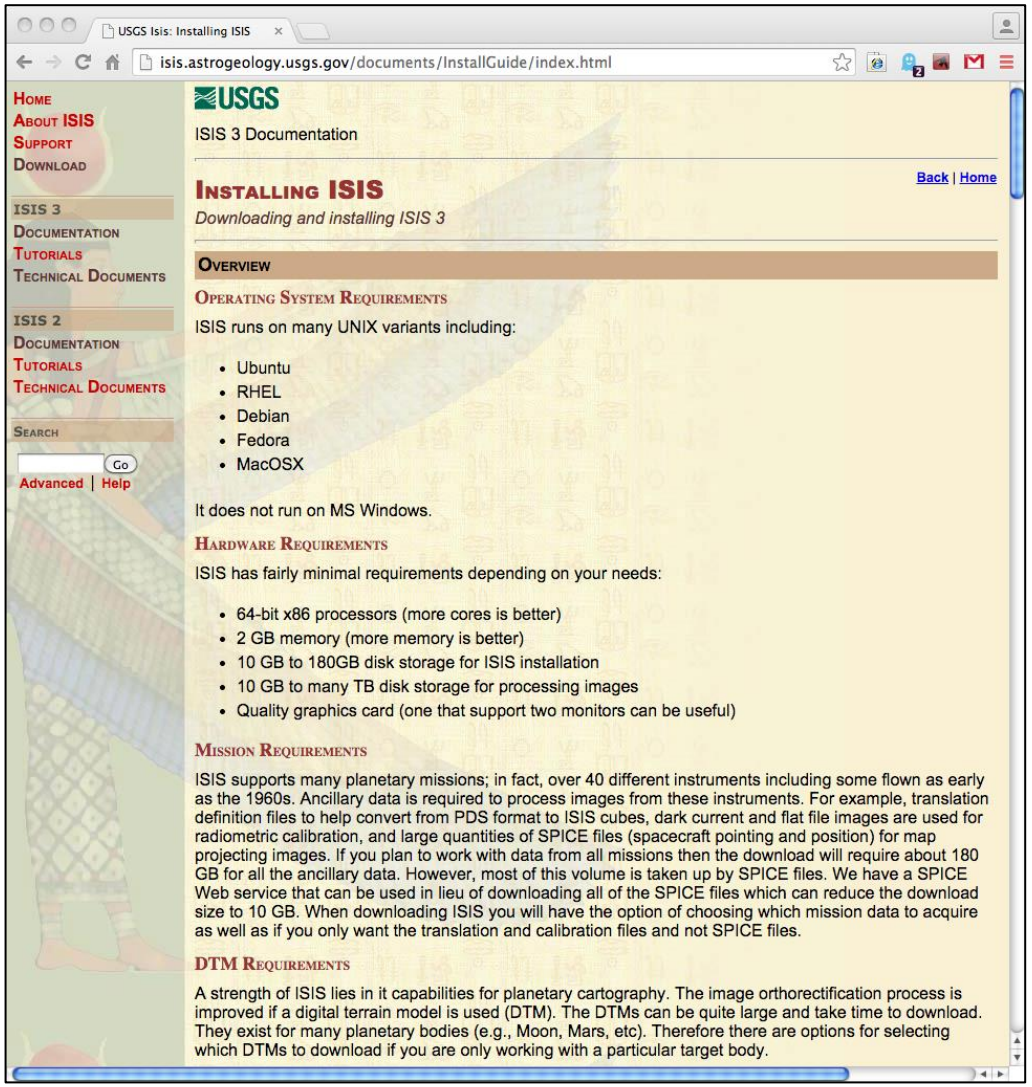
Other
- Ideal Camera (Special ISIS Perfect Virtual Instrument – Distortion-Free!)

Pluto
- Upcoming – New Horizons MVIC, LORRI, LEISA

Fifty-five instruments in all!!

Current ISIS Status

- ISIS 3.4.8 (Released Dec 2014)
- ISIS 3.4.9 (Scheduled Release: End of March 2015)
- UNIX-based Supported Platform OSes
  - Mac OSX 10.6, 10.8 and higher (32 & 64 bit Intel)
  - Debian 7 (64 bit)
  - Fedora 18 (64 bit)
  - Redhat Enterprise 6.5 (64 bit) (via SL 6.5)
  - Scientific Linux (SL) 6.5 (64 bit)
  - Ubuntu 12.04 LTS (64 bit)
- Download via Internet
  - Full distribution >200GB
  - Selective download using **rsync** utility
  - Java client installer



ISIS3 Documentation, Support and User Guides

- **General Information**
  - http://isis.astrogeology.usgs.gov
- **Installation Guide**
  - http://isis.astrogeology.usgs.gov/documents/InstallGuide
- **Table of ISIS Applications**
  - http://isis.astrogeology.usgs.gov/Application
    - The ISIS3 Software Manual is organized by
      - Functional Category & Mission Specific Programs
- **User Support Forum**
  - http://isis.astrogeology.usgs.gov/IsisSupport
- **Online Workshops**
  - http://isis.astrogeology.usgs.gov/IsisWorkshop
- **Isis Command Line Options**
  - http://isis.astrogeology.usgs.gov/documents/CommandLine/CommandLine.html
- **Unix/Linux Command Reference Cheat-Sheet**
  - http://www.cheat-sheets.org/saved-copy/fwunixref.pdf

ISIS User Preferences

Examples of IsisPreferences settings to change

- HistoryPath
  - Change location of the application parameter history files to default to the user's current directory
    - HistoryPath = ./
- ProgressBar
  - Show progress percent for all applications
    - ProgressBar = On
- Cube Customization defaults
  - Change the gigabyte size allowed for cubes
    - MaximumSize = 55
- Mac Browser Options
  - To use default browser for Mac OSX
    - GuiHelpBrowser = open

```
##################################################
# Customize elements of the user interface
#
# ProgressBarPercent = 1 | 2 | 5 | 10
# ProgressBar = On | Off
# GuiStyle = windows | motif | cde | motifplus |
#            platinum | sgi | kde | aqua
# GuiHelpBrowser = { your preferred browser path }
# GuiFontName = helvitica|times|charter | any legal font
# GuiFontSize = 10 | 12 | 14 | any font point size
# HistoryPath = {your preferred location for the
application .par files}
# HistoryRecording = On | Off
##################################################

Group=UserInterface
  ProgressBarPercent = 10
  ProgressBar        = On
  GuiStyle           = windows
  GuiHelpBrowser     = firefox
  GuiFontName        = helvitica
  GuiFontSize        = 10
  GuiWidth           = 460
  GuiHeight          = 600
  HistoryPath        = $HOME/.Isis/history
  HistoryRecording   = On
EndGroup

    …
```

Instructions to over-ride system-wide defaults:

1. Create a "hidden" .Isis directory in your $HOME area
2. Copy the IsisPreferences file to this directory
   > cp $ISISROOT/IsisPreferences $HOME/.Isis/IsisPreferences
3. View and edit your copy of the IsisPreferences file

For more information on Isis Preferences

http://isis.astrogeology.usgs.gov/documents/PreferenceDictionary/PreferenceDictionary.html
http://isis.astrogeology.usgs.gov/documents/EnvironmentAndPreferencesSetup/EnvironmentAndPreferencesSetup.html

Executing ISIS3 Applications

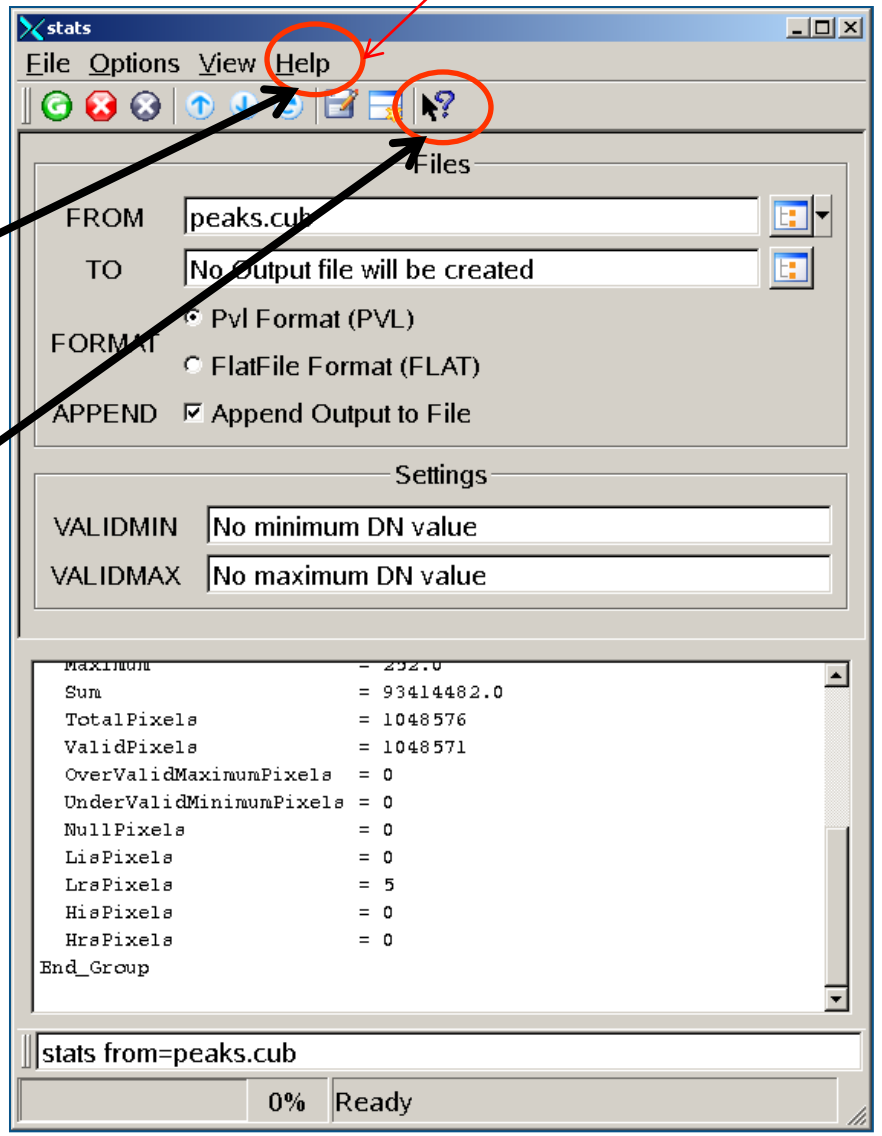
Application GUI Interface

Click the Help drop down menu

- The **"About this program"** opens a browser with detailed program information.

Use the "What's This?" tool to find out more about the parameters

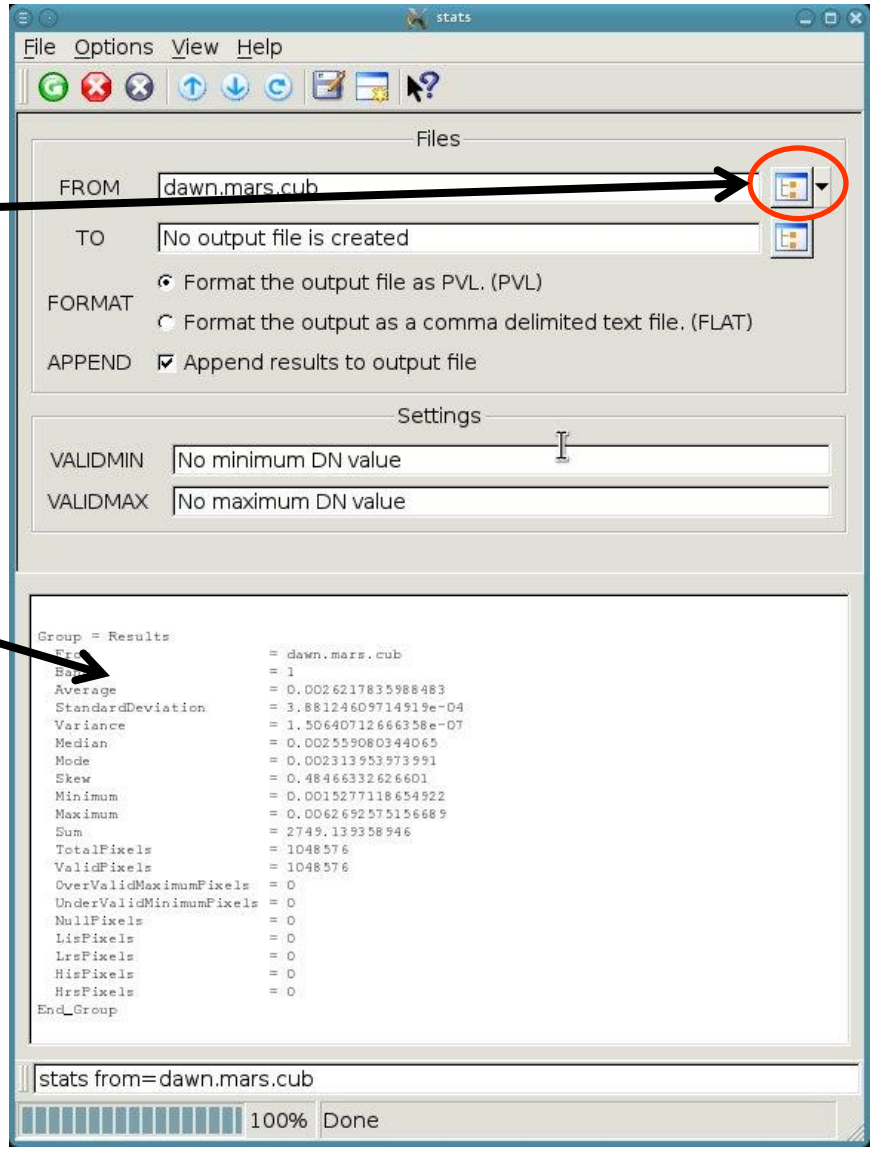
- FROM
- TO
- FORMAT
- APPEND



Executing ISIS3 Applications

Application GUI Interface

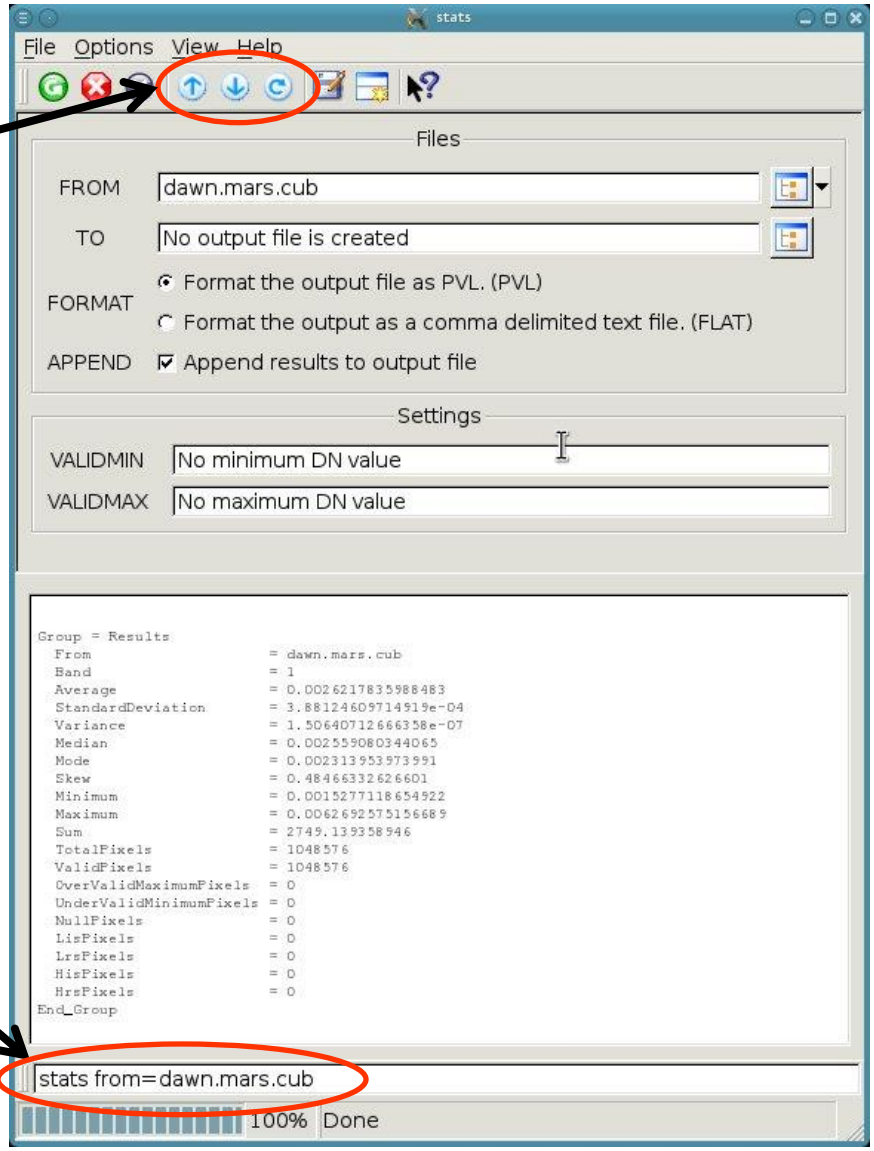
- Click File Selection Button 
  - Select Input filename to load FROM
- Click Run Button 
- View the Results Log Area
  - Program output placed here
  - Clear log 



Executing ISIS3 Applications

Application GUI Interface

- Parameter History
  - Recovery of previous execution commands
  - Use the History buttons to recall & cycle through the run commands
- Command line
  - The text box shows the terminal window command to run program outside of GUI
- Exit the program



Executing ISIS3 Applications

Command Line Options

```
> stats   from=input.cub   -help
FROM     = Null
TO       = Null
FORMAT   = (*PVL, FLAT)
APPEND   = TRUE
VALIDMIN = Null
VALIDMAX = Null

> stats from=input.cub    –webhelp

>  stats  from=input.cub    –gui

> stats –batchlist=file.lis from=\$1
```

- Run the stats program from the command line
- Append the ISIS –help option to see all parameters for this program
  - Valid parameter values with "*" are the Default settings
- The ISIS –webhelp option opens the on-line documentation for this program
- Use the –gui option to open the application with parameters already filled in from the command line
- Batch processing achieved through the command line option **[-batchlist]**
  - External to the GUI, -batchlist is only associated with the command line
  - Not accessible to applications within the GUI
  - Reads single to multiple rows and columns of text and passes to the application parameters

ISIS command options and their uses
http://isis.astrogeology.usgs.gov/documents/CommandLine/CommandLine.html

ISIS3 Application User Documentation

Selecting a **'bolded/underlined'** term will display the **Glossary**

USGS

Isis 3 Application Documentation

Standard View | TOC | Home

CAMPT

Computes geometric and photometric information at a given pixel location

Description
Categories
Groups
History

Description

Campt computes geometric and photometric information at a given pixel location in the input image **cube**. The program computes spacecraft and instrument related information, and other types of coordinates as described later in this document. The user will have a choice of coordinates in which to output the information as well as a choice in the output format of the information acquired.

Note the input image cube has preliminary requirements:

- The input image requires **SPICE** information (see ***Spiceinit***).
- The input image cube must be a **Level0** or a **Level1** ISIS cube.
- To use a **Level2** file as the input image (see ***Mappt***).

The point of interest in the image can be entered as **Latitude**/**Longitude** coordinates or **Line**/**Sample** coordinates. Keep in mind that the input **Latitude** and **Longitude** values entered will be interpreted as **Universal Coordinates** (ISIS default) regardless of the target body. In the output, all positions are in **Body-Fixed** Coordinates.

The following is a partial list of coordinates computed in the campt application:

Cube

A cube is a 3-dimensional image with axis: samples, lines, and bands. The physical dimensions of a cube are called the number of samples (NS), number of lines (NL), and number of bands (NB).

Typically, the sample and line dimensions are used to represent spatial information while the band dimension represents spectral information.

Declination

Declination (Dec) is one of two angles of the north pole of a target body as a function of time.

For more information, refer to Euler Angles, Right Ascension and Prime Meridian

Detector Resolution

The size of the detector on the focal plane for each pixel.

Digital Number

The numeric value of a single pixel in an image. The value may represent almost any unit. For example: reflectance (I/F), radiance, elevation, or radius. Digital Numbers (DNs) can be discrete integers or floating point values.

ISIS Processing History

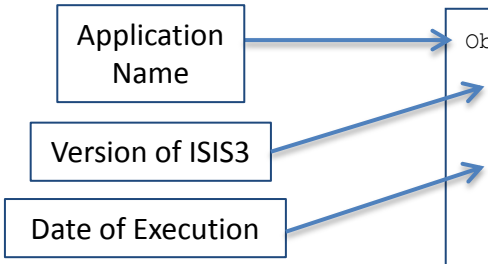
1. Print all ISIS commands applied to an ISIS cube to the screen in PVL format.
2. Write all ISIS commands applied to an ISIS cube to a text file in PVL format.
3. Print all ISIS commands applied to an ISIS cube to the screen in brief mode, (i.e. in command line format).

```
> cathist
    from=input.cub
> cathist
    from=input.cub
    to=input.pvl
> cathist
    from=input.cub
    mode=brief
```

Session Logs

- Program output
  - All executed applications append output to ***print.prt***
  - Basic accounting information on the application run is included
  - All user parameters
  - Application results

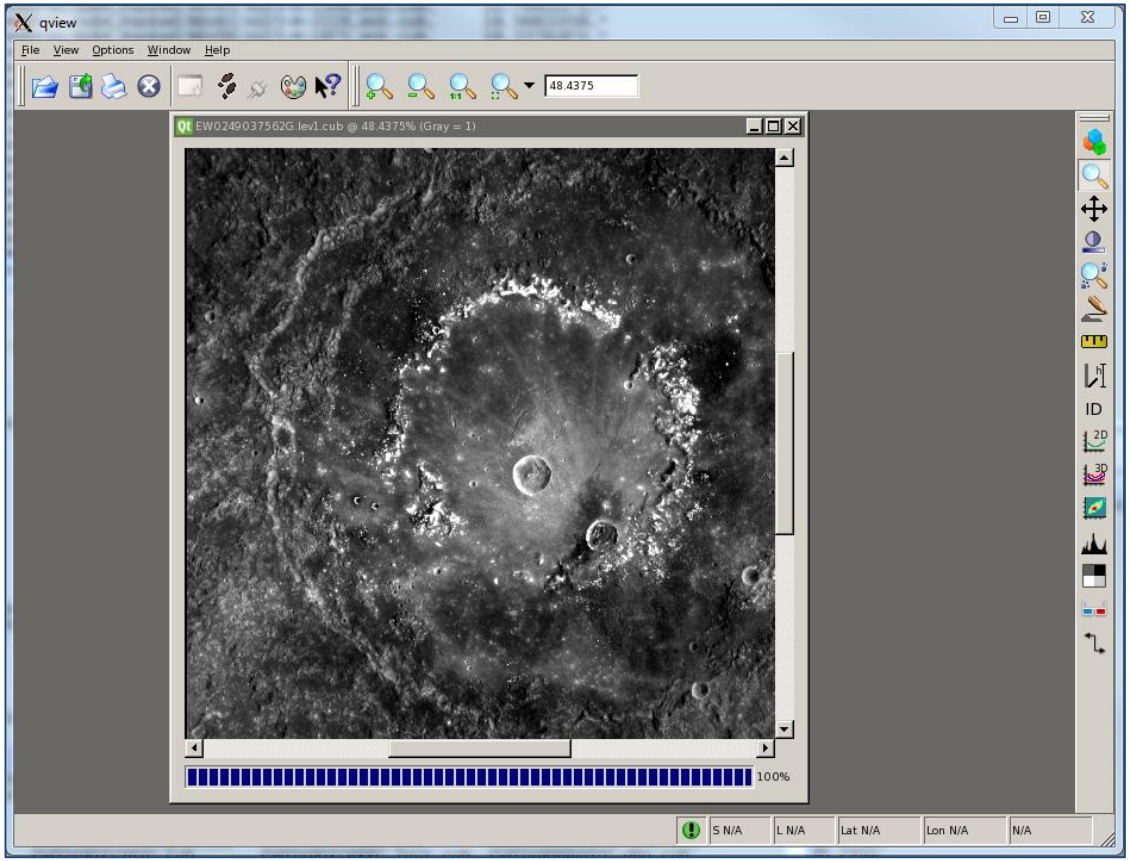
Typical Standard Output to *print.prt*



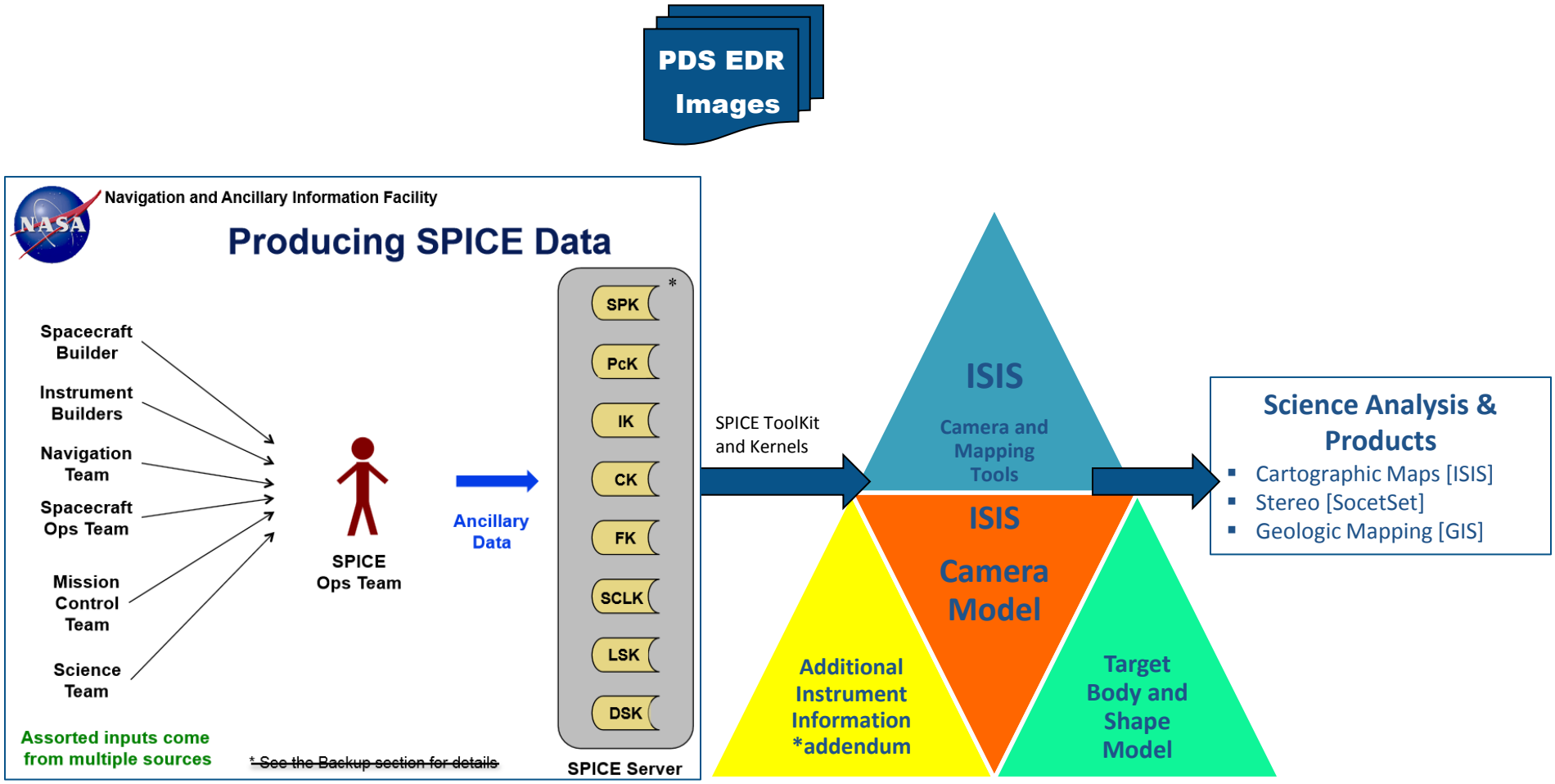

```
Object = stats
  IsisVersion       = "3.4.8.6010 stable | 2014-11-26"
  ProgramVersion    = 2013-11-12
  ProgramPath       = /usgs/pkgs/isis3.4.8/isis/bin
  ExecutionDateTime = 2015-03-06T10:52:00
  HostName          = astrovm1.wr.usgs.gov
  UserName          = tbecker
  Description       = "Generates statistics about a cube"

  Group = UserParameters
    FROM   = EW0249037420F.lev1.cub
    FORMAT = PVL
    APPEND = TRUE
  End_Group

  Group = Results
    From                     = EW0249037420F.lev1.cub
    Band                     = 1
    Average                  = 0.044823747606398
    StandardDeviation        = 0.0064236440931258
    Variance                 = 4.126320343515e-05
    Median                   = 0.043860295335343
    Mode                     = 0.041518581464479
    Skew                     = 0.44995593953587
    Minimum                  = 0.024972274899483
    Maximum                  = 0.20238755643368
    Sum                      = 46817.507899931
    TotalPixels              = 1048576
    ValidPixels              = 1044480
    OverValidMaximumPixels   = 0
    UnderValidMinimumPixels  = 0
    NullPixels               = 4096
    LisPixels                = 0
    LrsPixels                = 0
    HisPixels                = 0
    HrsPixels                = 0
  End_Group

  Group = Accounting
    ConnectTime = 00:00:00.0
    CpuTime     = 00:00:00.1
  End_Group
```

Image Visualization

- ***Qview***
  - On-line presentation regarding ***qview*** details
    - http://isis.astrogeology.usgs.gov/IsisWorkshop/uploads/8/8b/Qview.pdf



PDS → NAIF → ISIS3 → Science Analysis & Products



*__Note__: For additional information on SPICE, refer to: http://naif.jpl.nasa.gov/naif/Ancillary_Data_Production.pdf

ISIS3 Cartographic Processing Terminology/Steps

- **Level 0**
  - Decompressed spacecraft data
  - Import PDS EDR into ISIS3
  - Ingestion application names for specific cameras will include the mission/camera acronym
    - *mdis2isis* (messenger mdis data)
    - *ciss2isis* (cassini ISS data)
    - *mroctx2isis*

- **SPICE (*required*)**
  - **S**pacecraft & **P**lanetary ephemerides, **I**nstrument **C**-matrix and **E**vent kernels
  - ISIS3 uses the NAIF ToolKit for SPICE
  - '***spiceinit***'
    - Applies to all supported ISIS camera models
    - Writes pointing, spacecraft position to attached tables in input cube
    - Initialize with a DEM to orthorectify image

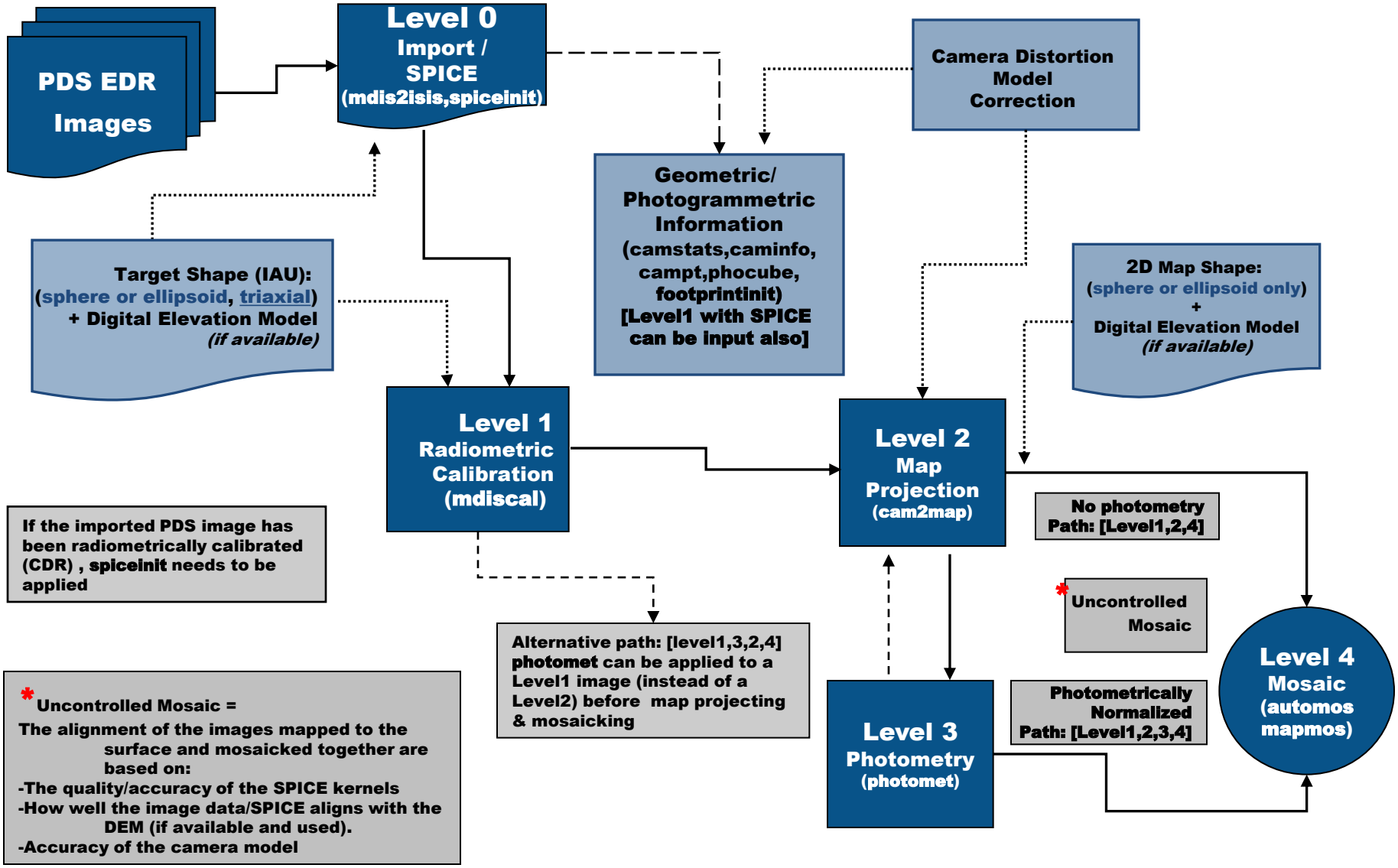
- **Level 1**
  - Radiometric calibration
  - Noise Removal (optional)
  - Application names applied to input Level0 and Level1 cubes will include a '**cam**' prefix
    - *campt, camstats, cam2map*

- **Level 2**
  - Project image to map coordinates
  - Camera distortion correction applied
  - Application names applied to input Level2 cubes will include a '**map**' prefix
    - *mappt, map2map*

- **Level 3**
  - Photometric normalization (optional)

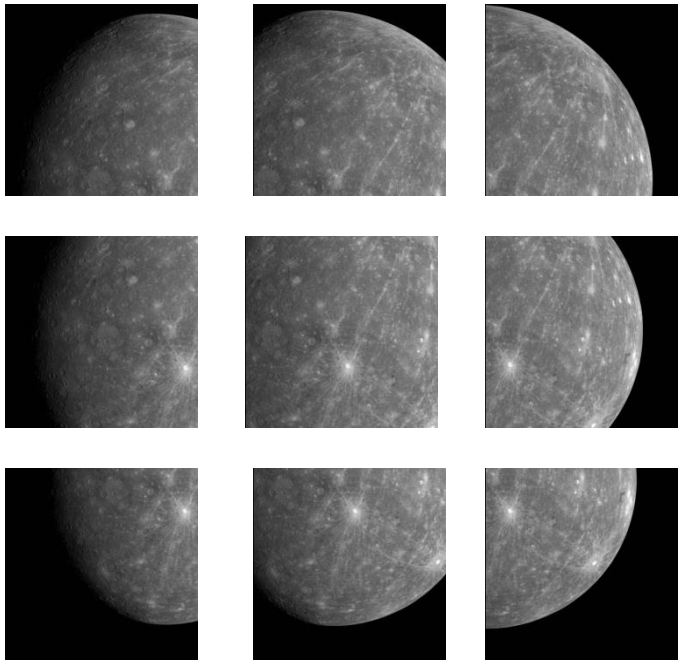
- **Level 4**
  - Mosaicking (optional)
  - Requires customized map projected input Level2 cubes

Standard MESSENGER Processing Flow within ISIS3

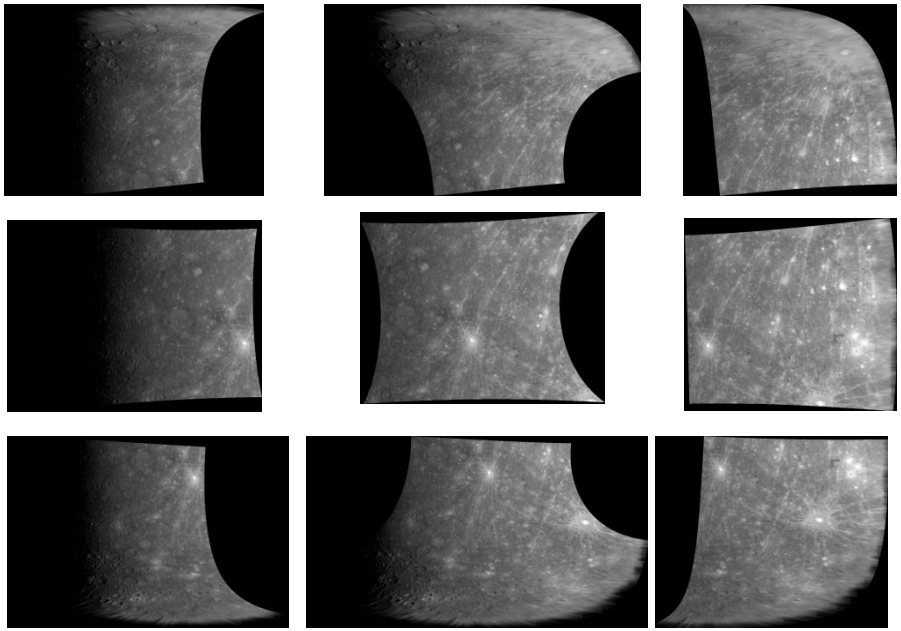


M2 Departure Color 3x3 WAC-G Filter Images

Level 1
Camera
Calibrated with SPICE



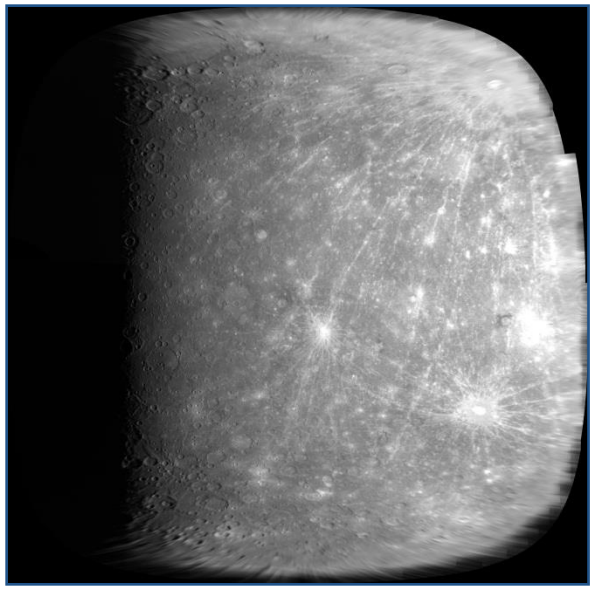
Level 2
Equirectangular
Map Projection*



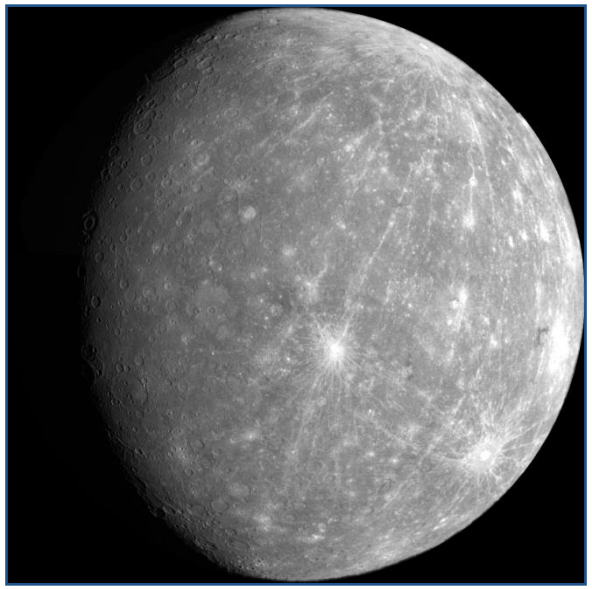
*The same map resolution and center longitude was defined for all images as required to mosaic

M2 Departure Color 3x3 WAC-G Filter Images

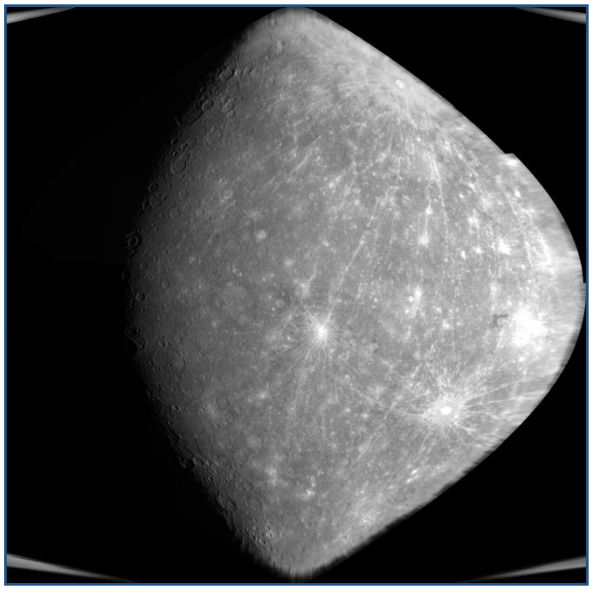
Level 4 MOSAICS
Different Map Projections



Equirectangular



Orthographic



Sinusoidal

ISIS Support for MESSENGER

NASA, MESSENGER Project, Johns Hopkins University Applied Physics Laboratory and Arizona State University have provided funding and/or support to the USGS for the development of ISIS3 software and MDIS data products

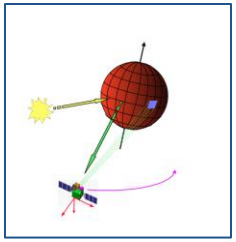
- MESSENGER Development and Support Activities
  - Develop and maintain camera models for
    - Narrow Angle (NAC)
    - Wide Angle (WAC)
  - Distribution of SPICE kernels
  - Maintenance of Radiometric calibration
  - Camera distortion correction
  - Photometric correction
    - Parameter setting are not released with ISIS. They are supplied by command line in the workshop lessons for NAC and WAC filters
  - Generating global monochrome and color maps
  - Processing of PDS MDIS EDR, CDR, BDR and MDR data
  - Participate in development of special products (uncontrolled/controlled) maps, updated kernels, stereo products and DEMs

ISIS Camera Model Coordinate System Default

In **ISIS**, it is IMPORTANT to note that unless otherwise specified, ALL raw 'camera' programs expect as input and/or result in the following geometric reference (regardless of target body)

- ISIS Universal Coordinate System
  - Longitude System defaults to the 0 - 360 range
    - Longitude Domain = 360
  - Longitude Direction default is Positive East
    - i.e. increase as you move to the east
  - Latitude System default is Planetocentric
    - Spherical bodies: Planetocentric = Planetographic

Geometric and Photometric Camera Tools



campt (Camera Point)

- **Computes camera geometric and photometric information at a given pixel location in the input image cube.**
- **The user may enter**
  - An image coordinate (sample, line), ***OR***
  - A ground coordinate (planetocentric latitude, positive east longitude)

camstats (Camera Statistics)

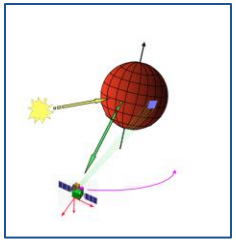
- **Generates and outputs camera statistics for an unprojected cube with SPICE information attached. (Level 0 or Level 1)**

caminfo (Camera Information)

- **Compiles and outputs spacecraft and instrument-related information**
  - Camera Statistics
  - Image Statistics
  - Geometry Information
  - Polygon Information
  - Target and Map Template Information
  - Image Footprints
  - Useful for creating geometric info for GIS database

*__Note__: For additional information on SPICE, refer to: http://naif.jpl.nasa.gov/naif/Ancillary_Data_Production.pdf

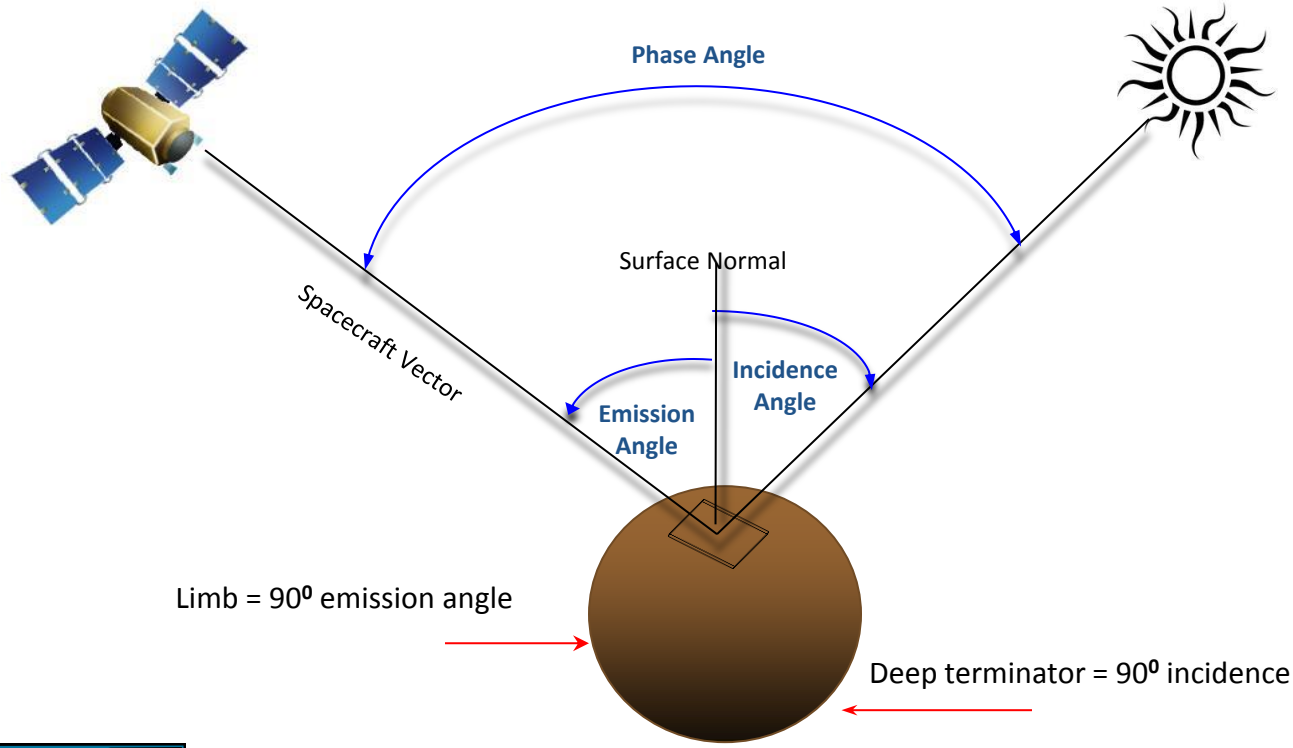
Geometric and Photometric Camera Tools

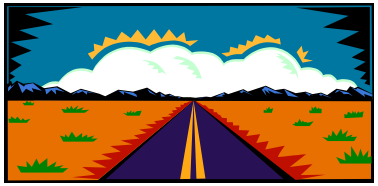


```
Group = GroundPoint
  Filename                = FC_Mars.cub
  Sample                  = 512.0
  Line                    = 512.0
  PixelValue              = 0.00260657
  RightAscension          = 19.722259374476
  Declination             = -59.58327428227
  PlanetocentricLatitude  = 48.354237551489
  PlanetographicLatitude  = 48.689922325464
  PositiveEast360Longitude = 277.86849622301
  PositiveEast180Longitude = 82.131503776986
  PositiveWest360Longitude = 82.131503776986
  PositiveWest180Longitude = 82.131503776986

...
```

```
  # Sun Information
  SunPosition             = (-177533702.4284,
                             -103045154.79627,
                             -47419881.109118) <km>
  SubSolarAzimuth         = 359.48348806842
  SolarDistance           = 1.4082928265777 <AU>
  SubSolarLatitude        = -13.007723818623
  SubSolarLongitude       = 210.13198846511
  SubSolarGroundAzimuth   = 218.08431231392

  # Illumination and Other
  Phase                   = 65.955039121282
  Incidence               = 85.578537079895
  Emission                = 20.534413411423
  NorthAzimuth            = 111.81291300246

  # Time
  EphemerisTime           = 288188598.81919 <seconds>
  UTC                     = 2009-02-18T00:22:12.634
  LocalSolarTime          = 16.51576718386 <hour>
  SolarLongitude          = 211.92252854089
End_Group
```

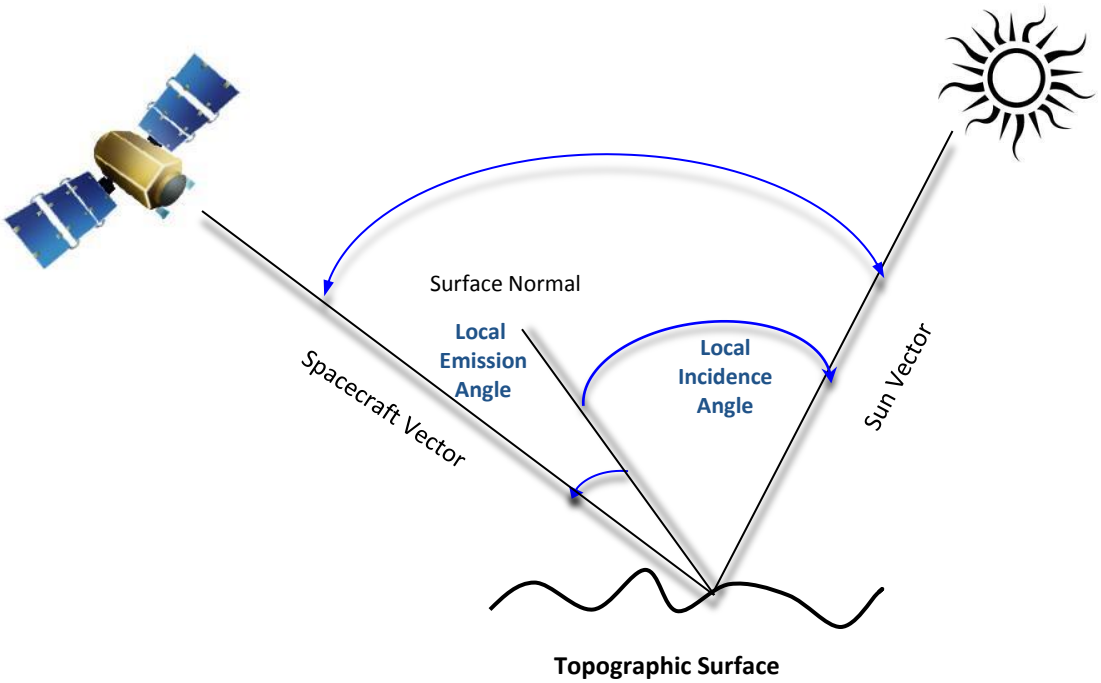
Photometric Angles (reported in degrees)





Horizon= 90⁰ emission angle

Photometric Angles (reported in degrees)

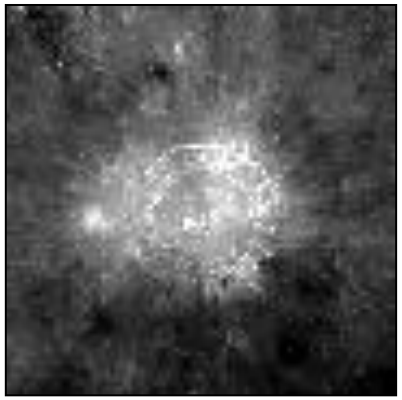


With a DEM shape model, a “local” emission and incidence angle is computed at the pixel point of interest

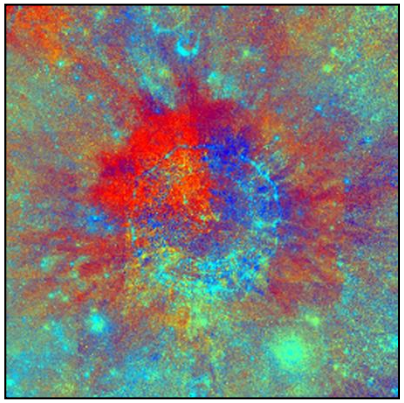
Photometric Angles (reported in degrees)

Clementine Image (750 nm)
Copernicus

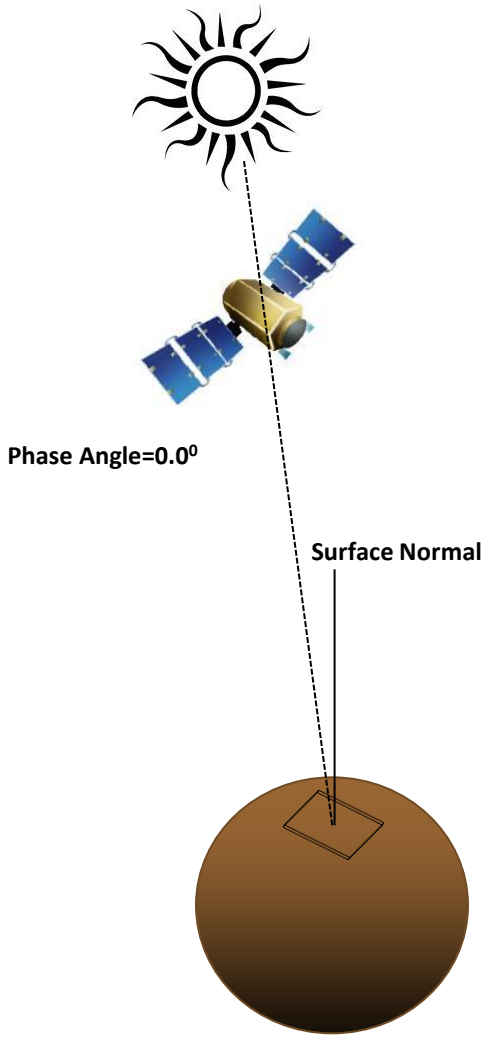
Phase Angle = ~0^0



- No shadows
- No visible topography

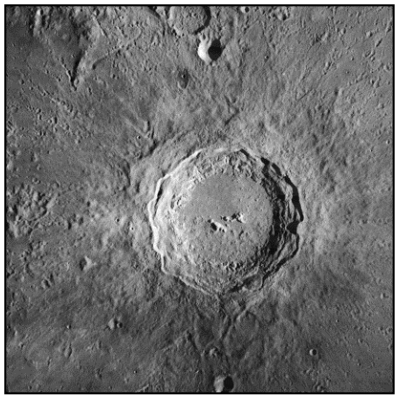


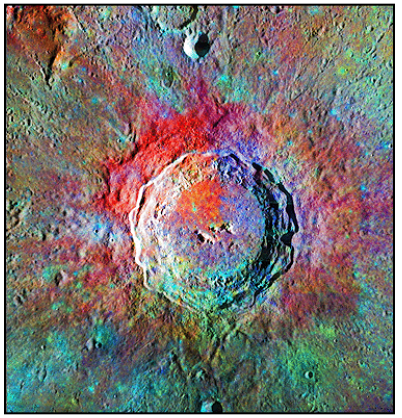
Clementine Multi-Spectral
Color Ratio of Copernicus



Lunar Orbiter IV
Copernicus

Phase Angle = ~65^0



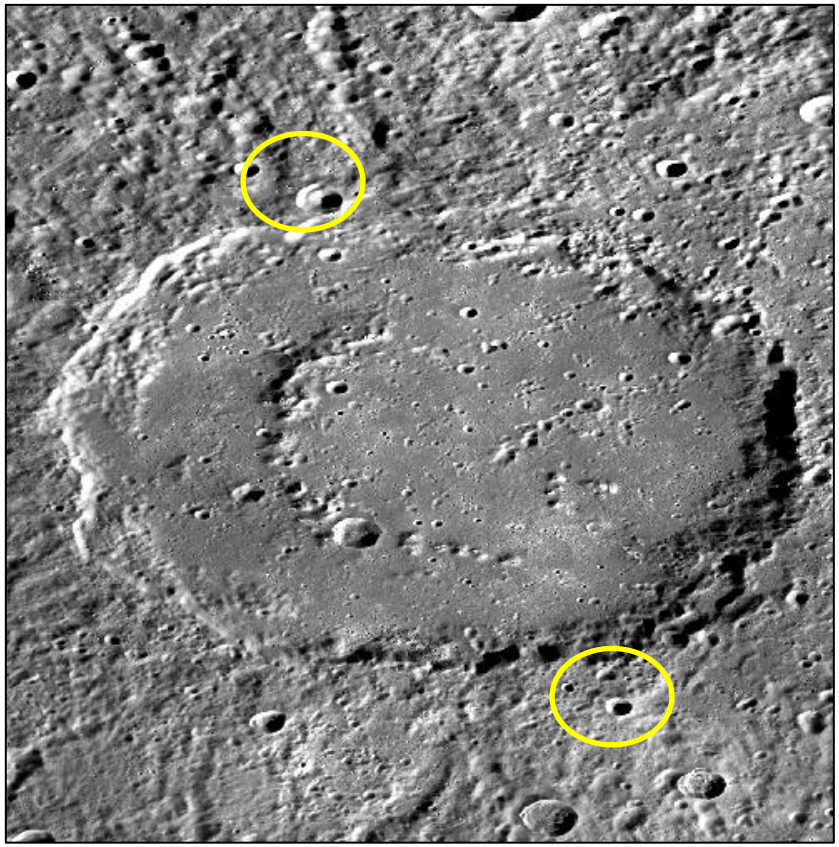


Merge of Lunar Orbiter IV
Clementine Multi-Spectral

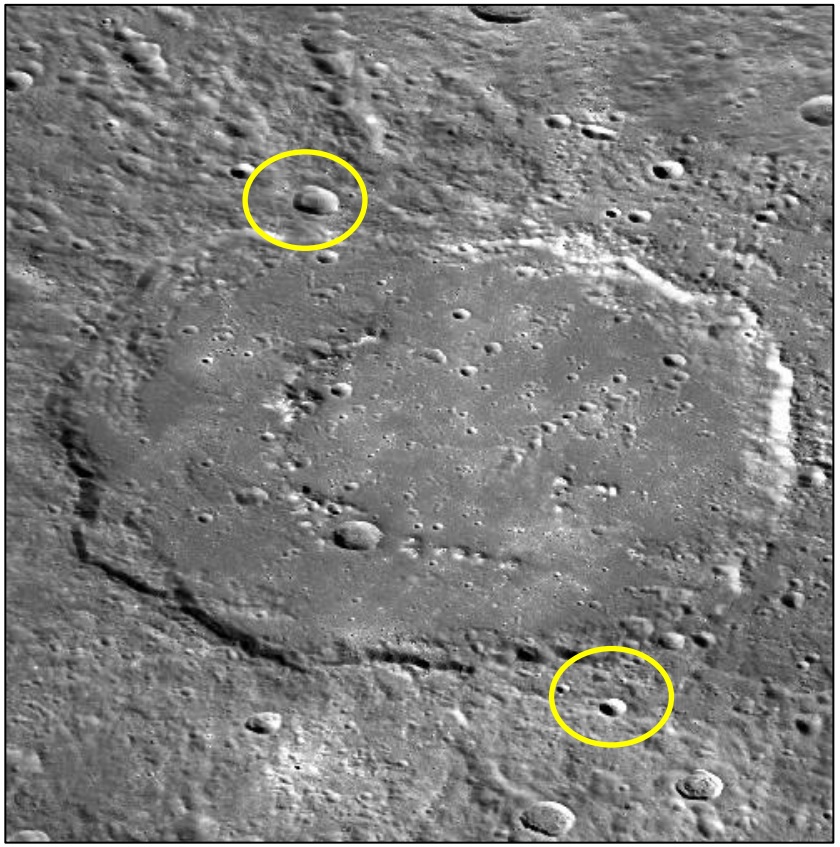
Subsolar Ground Azimuth

Messenger MDIS Mercury –

- Two separate mosaics of the same region where the sun illumination is opposite
- The image data was separated into two ranges of the solar ground azimuth



Solar Ground Azimuth Range 0-179 degrees

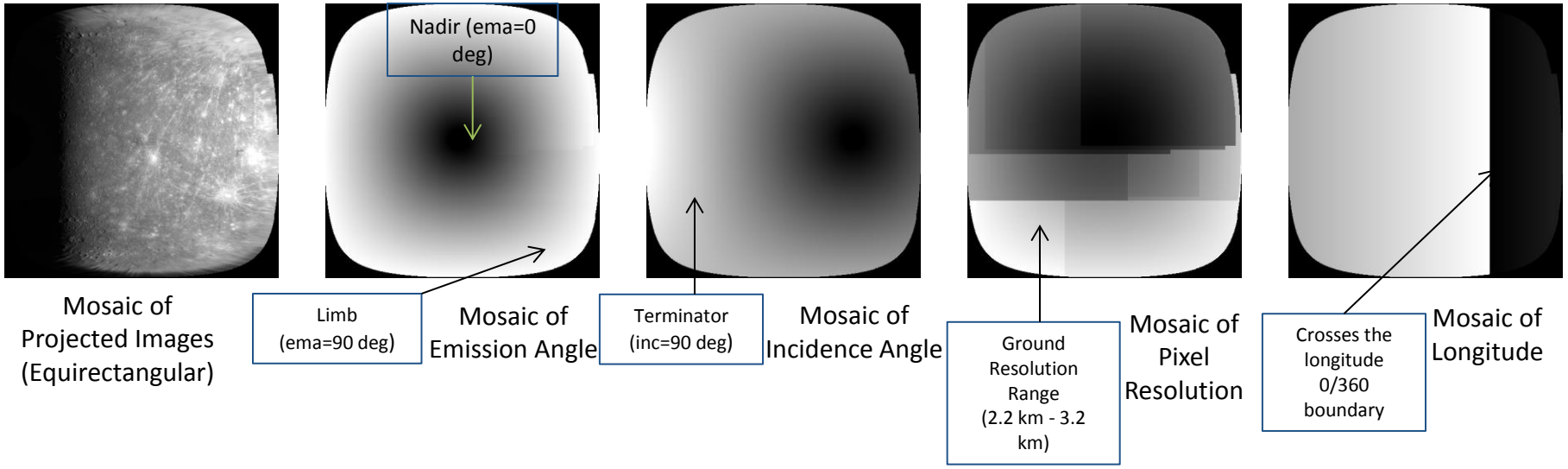


Solar Ground Azimuth 180-360 degrees

Geometric and Photometric Camera Tools

phocube

- Application generates data for every pixel and creates an output cube
- Can be applied to a raw camera image (Level1) or a map projected image (Level2)



*There are currently 20 data options available in **phocube***

Geometric and Photometric Camera Tools

- Applications applied to Level1 images
  - ***camrange*** – compute the latitude and longitude range of a level1 image
  - ***camtrim*** – trim pixels outside of a latitude and longitude range
    - *trim* means that pixels that fall outside the criteria are set to NULL
  - ***cam2cam*** – convert an input level1 camera cube to a different camera geometry

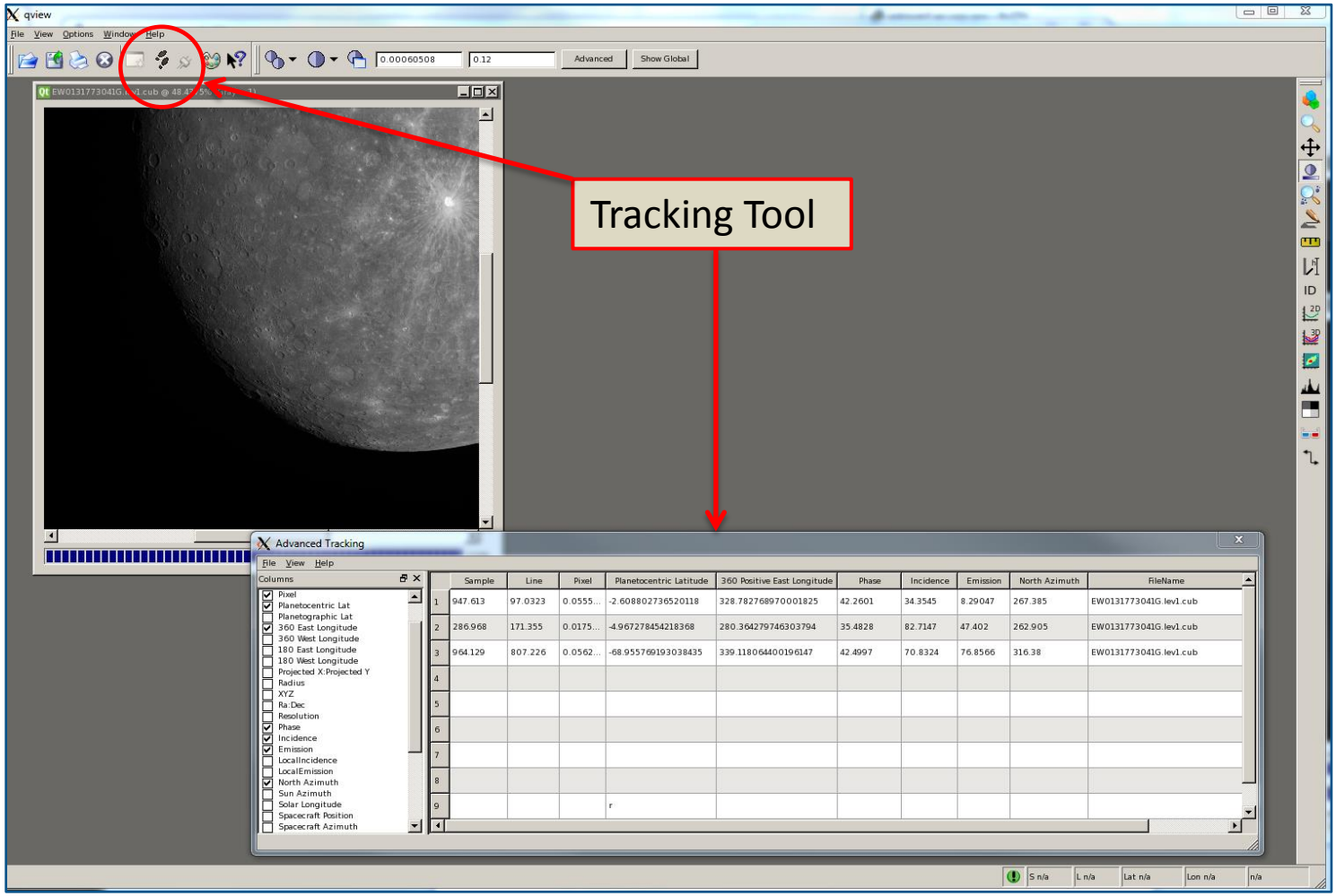
- Applications applied to Level1 or Level2 images
  - ***phocube*** – compute photometric and geometric values for every pixel
  - ***photrim*** – trim pixels outside of photometric angle ranges
    - *trim* means that pixels that fall outside the criteria are set to NULL

- ***skypt*** – compute coordinates from a given sample/line pixel location or ra/dec

http://isis.astrogeology.usgs.gov//Application/index.html#Cameras

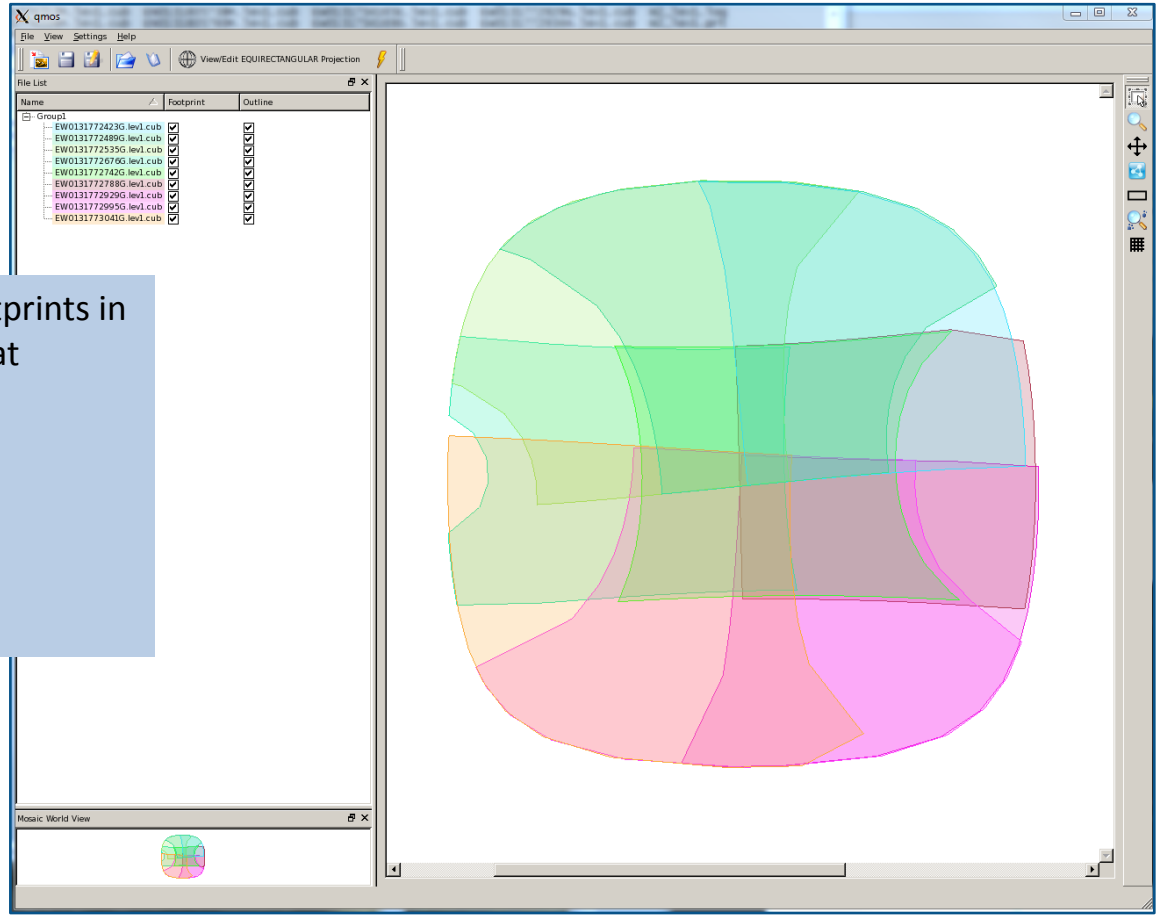
Geometric and Photometric Camera Tools

- ***qview*** – Interactively reports geometric and photometric information at every pixel



Geometric and Photometric Camera Tools

- ***qmos*** – Interactive Footprint Display
  - On-Line presentation on qmos details
    - http://isis.astrogeology.usgs.gov/IsisWorkshop/uploads/b/b1/Qmos.pdf



Displays Level1 or Level2 image footprints in any ISIS map projected format

Requirements:

-SPICE (***spiceinit***)

-Polygon (***footprintinit***)

-Geometry/Photometric Info (***camstats attach=true***)

Supported Map Projections in ISIS3

- Equirectangular
- Lambert Conformal
- Mercator
- Oblique Cylindrical
- Orthographic
- Point Perspective
- Polar Stereographic
- Simple Cylindrical
- Sinusoidal Equal Area
- Transverse Mercator

Interactive on-line map projection tutorial:
http://isis.astrogeology.usgs.gov/IsisWorkshop/index.php/Learning_About_Map_Projections

System map template files:
$ISIS3DATA/base/templates/maps

USGS Map Projections Poster:
http://egsc.usgs.gov/isb//pubs/MapProjections/projections.html

Map Projections (Level2)

- Map projection of images are based on original SPICE data
- All ISIS3 map projection programs default to
  - Sinusoidal map projection ***($ISIS3DATA/base/templates/maps/sinusoidal.map)***
  - IAU target body shape (ellipsoid or sphere)
  - Planetocentric Latitude System
  - Positive Longitude = East
  - Longitude Domain = 360 [longitude range is defined as 0 to 360 degrees]
  - Remaining required map characteristics are computed
    - Pixel resolution (meters/pixel)
    - Ground range (latitude and longitude coverage)
    - Center latitude/longitude
- Mosaic **Requirements** (user must override defaults with a map template file)
  - All images must be a map projected ISIS cube (Level2) with the following:
    - Same projection
    - Same map resolution
    - Same center latitude/longitude
    - Same Band/Wavelength

Define your own Map Projection

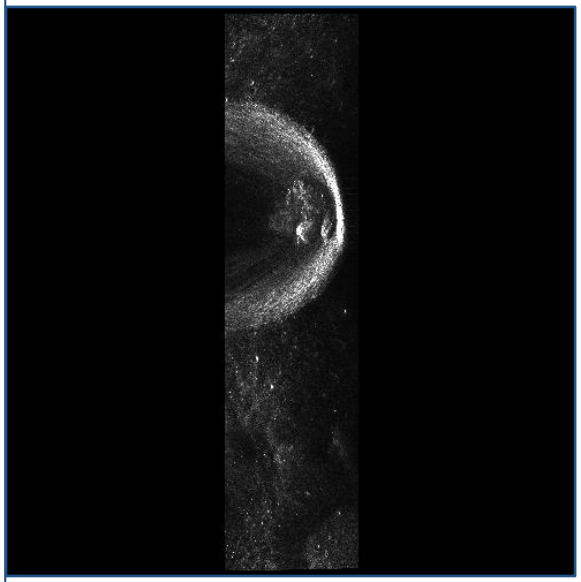
- Map Template files are used to define the output map projection
  - The map template file is used to override the cartographic defaults
- A map template file is available for every supported projection
  - $ISIS3DATA/base/templates/maps/
- Applications with a GUI interface to the map parameters
  - ***maptemplate***
  - ***mosrange***
- Other ISIS3 map projected image cubes
  - ISIS cubes in a map projection can be used as map templates
  - Provide an ISIS image cube in the ***map*** parameter to ***cam2map*** or ***map2map***

Target Body Properties for Mercury

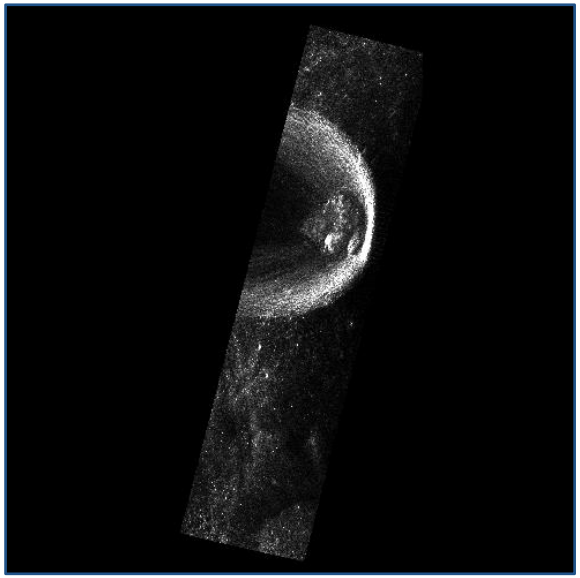
- ISIS3 normally uses the IAU target radius values from NAIF SPICE kernels for the output map projection
  - The IAU radius for Mercury is: 2439.7 km
  - **The MESSENGER Team uses: 2440.0 km**
    - The Team radius is supported as the default within the ISIS3 MESSENGER kernel subsystem
  - Be aware of this when merging or analyzing Mercury map products (i.e., Mariner 10) that are mapped to the IAU radius
    - Mariner 10, supported in ISIS3, defaults to the IAU radius value of 2439.7 km, not the MESSENGER Team value of 2440.0 km for Mercury's radius!
  - An ellipsoid or mean radius will be used for a triaxial body (e.g., Io, Vesta)

Orthorectification

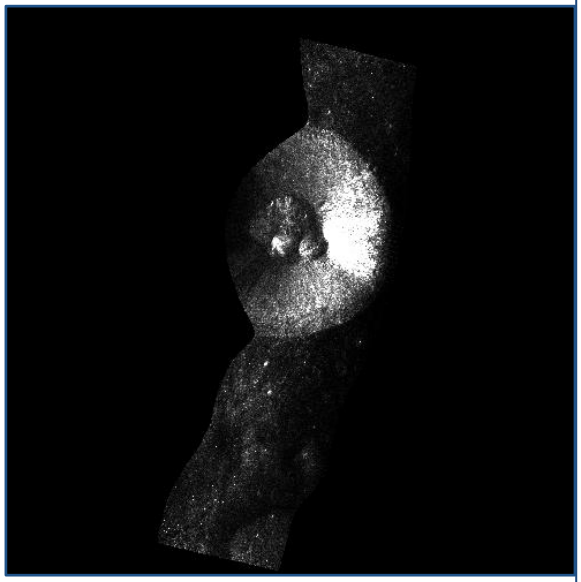
- A Digital Elevation Model (DEM) can be used when projecting images
  - ISIS3 defaults to LOLA for the Moon and MOLA for Mars
  - New Mercury shape models coming
    - September 2015
- Removes effects of topography in the output map projection
  - The DEM must be provided in the ***spiceinit*** application
  - Shackleton crater example below
    - [LRO-MiniRF LSZ_02261_1CD_XKU_89S140_V1]



LRO MiniRF Level 1



Projected onto sphere



Projected using DEM

Hands-On Lesson 1

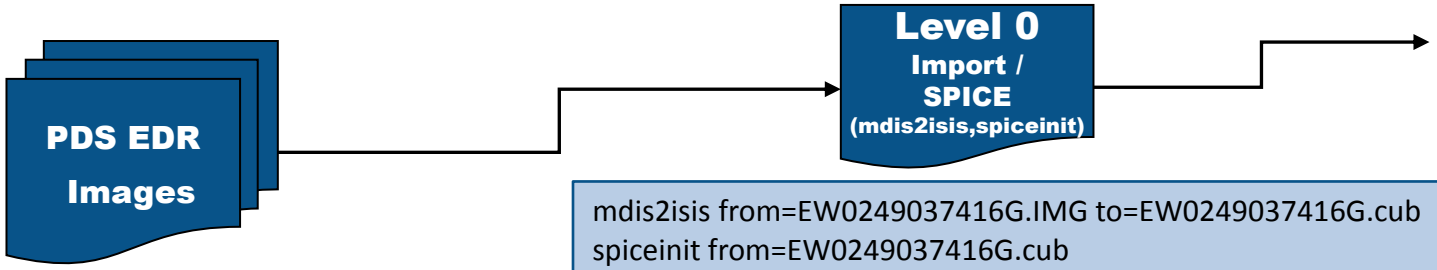
Create a Monochrome Map Mosaic (EDRs)

Create a MDIS Monochrome Mosaic

- Objective: Use ISIS3 to create an MDIS monochrome mosaic of a specific region of interest (ROI)
  - Create from PDS EDRs
  - http://pdsimage.wr.usgs.gov/Missions/MESSENGER/MSGRMDS_1001/
- Target ROI: **Raditladi Basin**
- General Processing Steps
  - Determine MDIS images to include in mosaic (PDS, PILOT, etc…)
    - http://pilot.wr.usgs.gov/
  - Download images from source
  - Process using ISIS3
    - Import, apply SPICE, radiometric calibration (CDR), map project, photometric correction and assemble mosaic
  - Refer to the Hands-On **RaditladiBasin** Lesson

Import and Apply SPICE

- Region of interest criteria: **Raditladi Basin**
  - http://www.nasa.gov/mission_pages/messenger/multimedia/messenger_orbit_image20130114_1.html
  - Latitude Range=(17N to 36.5N) Longitude Range=(111E to 128E)
  - Wavelength Filter = G (748.7 NM) InstrumentId = MDIS-WAC



Selected PDS EDR Images:

2012_176/EW0249037416G.IMG
2012_176/EW0249066407G.IMG
2012_176/EW0249037545G.IMG
2012_176/EW0249037688G.IMG
2012_176/EW0249066345G.IMG
2012_176/EW0249037671G.IMG
2012_176/EW0249037562G.IMG

```
mdis2isis from=EW0249037416G.IMG to=EW0249037416G.cub
spiceinit from=EW0249037416G.cub
mdis2isis from=EW0249066407G.IMG to=EW0249066407G.cub
spiceinit from=EW0249066407G.cub
mdis2isis from=EW0249037545G.IMG to=EW0249037545G.cub
spiceinit from=EW0249037545G.cub
mdis2isis from=EW0249037688G.IMG to=EW0249037688G.cub
spiceinit from=EW0249037688G.cub
mdis2isis from=EW0249066345G.IMG to=EW0249066345G.cub
spiceinit from=EW0249066345G.cub
mdis2isis from=EW0249037671G.IMG to=EW0249037671G.cub
spiceinit from=EW0249037671G.cub
mdis2isis from=EW0249037562G.IMG to=EW0249037562G.cub
spiceinit from=EW0249037562G.cub
```

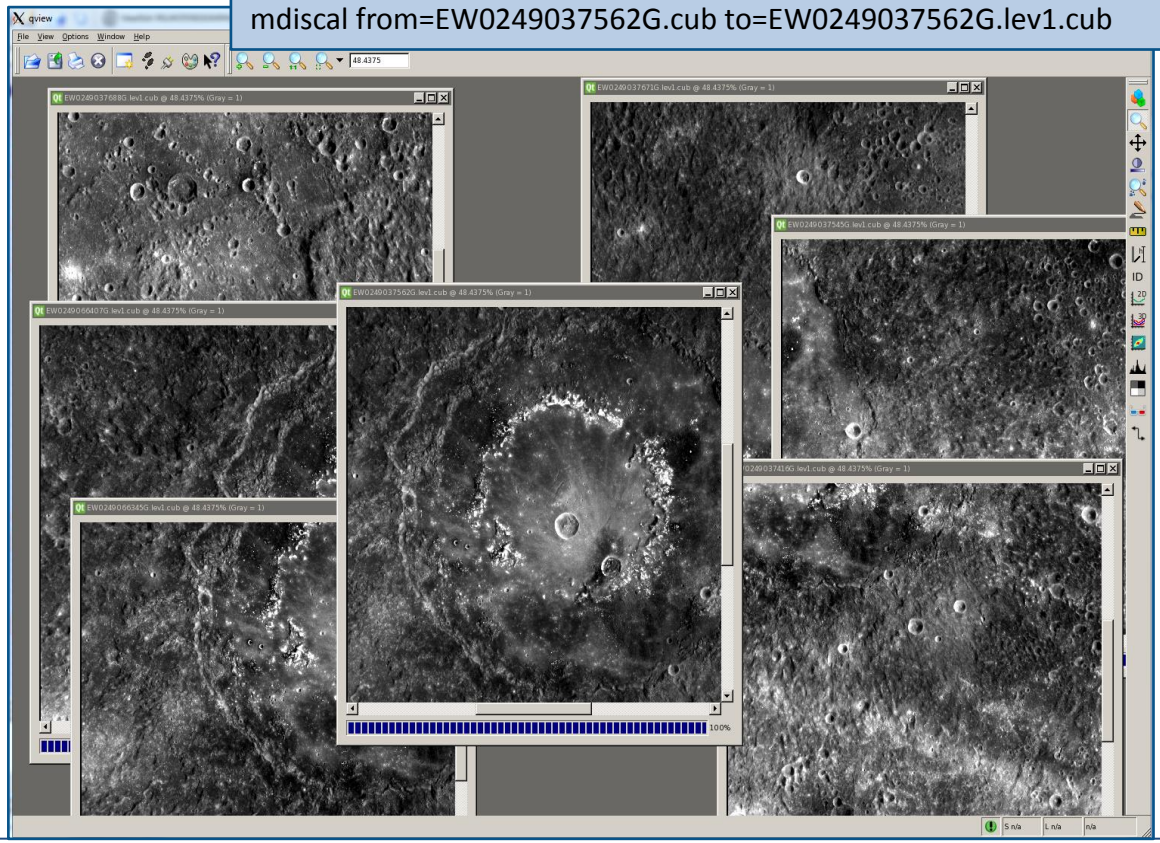
http://pdsimage.wr.usgs.gov/Missions/MESSENGER/MSGRMDS_1001/DATA/2012_176/

Radiometric Calibration

qview

Level 1
Radiometric Calibration (mdiscal)

```
mdiscal from=EW0249037416G.cub to=EW0249037416G.lev1.cub
mdiscal from=EW0249066407G.cub to=EW0249066407G.lev1.cub
mdiscal from=EW0249037545G.cub to=EW0249037545G.lev1.cub
mdiscal from=EW0249037688G.cub to=EW0249037688G.lev1.cub
mdiscal from=EW0249066345G.cub to=EW0249066345G.lev1.cub
mdiscal from=EW0249037671G.cub to=EW0249037671G.lev1cub
mdiscal from=EW0249037562G.cub to=EW0249037562G.lev1.cub
```



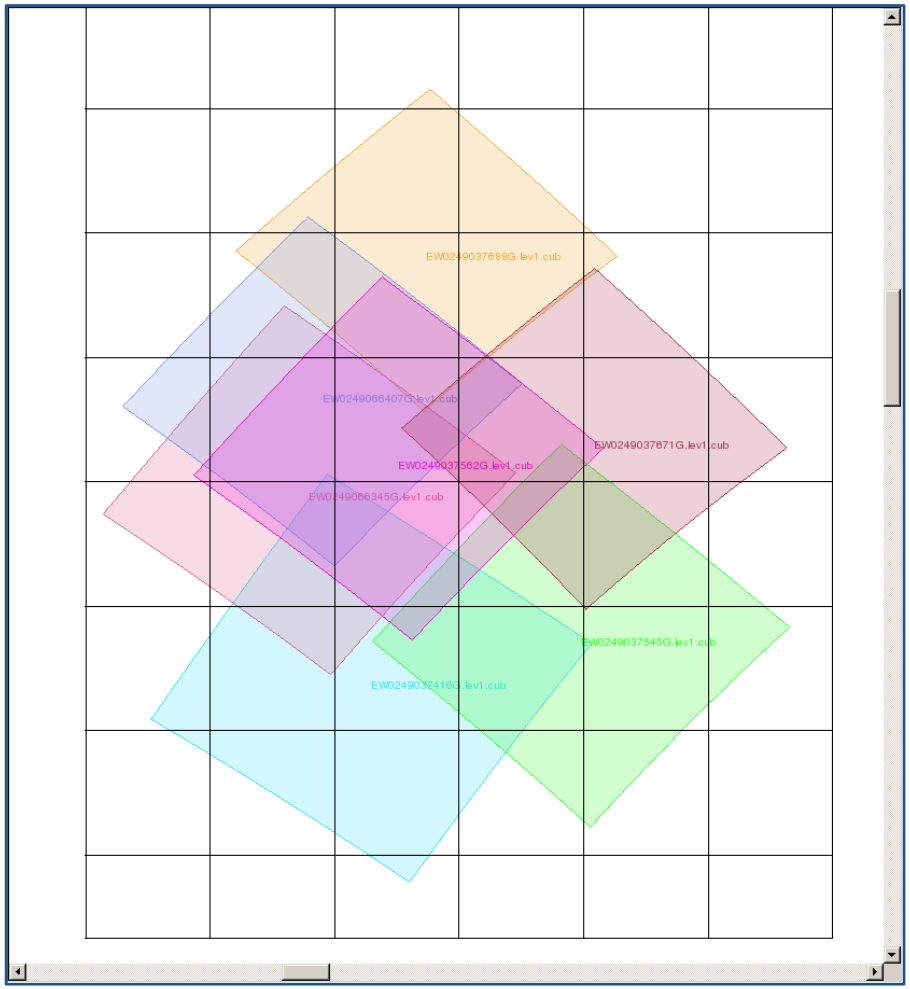
Geometric and Photometric Camera Tools

qmos

Geometric/ Photogrammetric Information (camstats,caminfo, campt,phocube, footprintinit)

```
camstats from=EW0249037416G.lev1.cub attach=true
footprintinit from=EW0249037416G.lev1.cub
camstats from=EW0249066407G.lev1.cub attach=true
footprintinit from=EW0249066407G.lev1.cub
camstats from=EW0249037545G.lev1.cub attach=true
footprintinit from=EW0249037545G.lev1.cub
camstats from=EW0249037688G.lev1.cub attach=true
footprintinit from=EW0249037688G.lev1.cub
camstats from=EW0249066345G.lev1.cub attach=true
footprintinit from=EW0249066345G.lev1.cub
camstats from=EW0249037671G.lev1.cub attach=true
footprintinit from=EW0249037671G.lev1.cub
camstats from=EW0249037562G.lev1.cub attach=true
footprintinit from=EW0249037562G.lev1.cub
```

Applications recommended before ***qmos***



Generate a Custom Output Map Template

- Create a map template file to define output map
  - For a mosaic, it is required that all input files are map projected to the exact same pixel resolution and center latitude and longitude
- Customized Map Template Options
  - ***maptemplate*** or
  - ***mosrange***
  - Manual edit
  - Existing map projected ISIS cube

Contents of *equi.map*

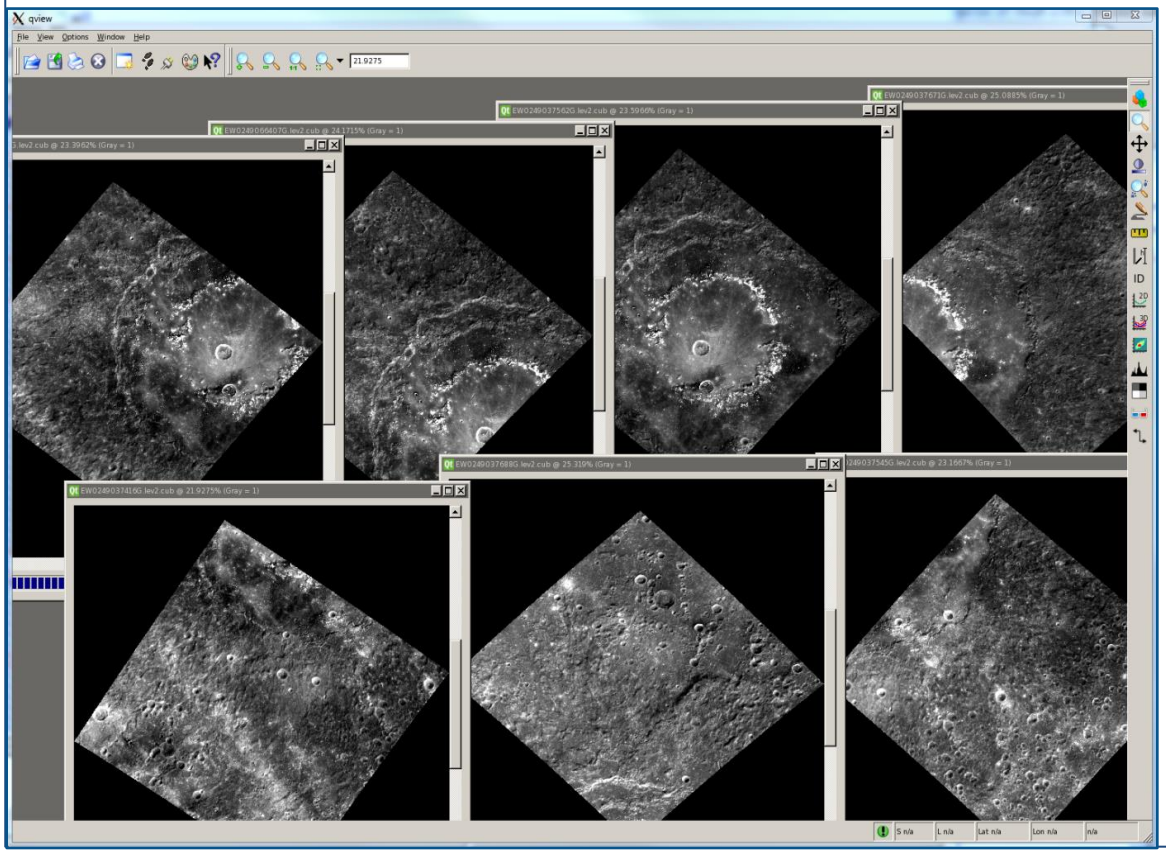
```
Group = Mapping
  ProjectionName          = EQUIRECTANGULAR
  TargetName              = Mercury
  EquatorialRadius        = 2440000.0 <meters>
  PolarRadius             = 2440000.0 <meters>
  LatitudeType            = Planetocentric
  LongitudeDirection      = PositiveEast
  LongitudeDomain         = 360
  PixelResolution         = 262.0 <meters/pixel>
  Scale                   = 163.0 <pixels/degree>
  MinPixelResolution      = 226.92996378504 <meters>
  MaxPixelResolution      = 296.53943218364 <meters>
  CenterLongitude         = 120.0
  CenterLatitude          = 27.0
  MinimumLatitude         = 17.0
  MaximumLatitude         = 37.0
  MinimumLongitude        = 111.0
  MaximumLongitude        = 128.0
End_Group
```

ls *.lev1.cub > lev1.lis
mosrange fromlist=lev1.lis to=**equi.map** projection=EQUIRECTANGULAR precision=0

Map Projection

Level2
Map Projection
(cam2map)

```
cam2map from=EW0249037416G.lev1.cub to=EW0249037416G.lev2.cub map=equi.map pixres=map
cam2map from=EW0249066407G.lev1.cub to=EW0249066407G.lev2.cub map=equi.map pixres=map
cam2map from=EW0249037545G.lev1.cub to=EW0249037545G.lev2.cub map=equi.map pixres=map
cam2map from=EW0249037688G.lev1.cub to=EW0249037688G.lev2.cub map=equi.map pixres=map
cam2map from=EW0249066345G.lev1.cub to=EW0249066345G.lev2.cub map=equi.map pixres=map
cam2map from=EW0249037671G.lev1.cub to=EW0249037671G.lev2.cub map=equi.map pixres=map
cam2map from=EW0249037562G.lev1.cub to=EW0249037562G.lev2.cub map=equi.map pixres=map
```



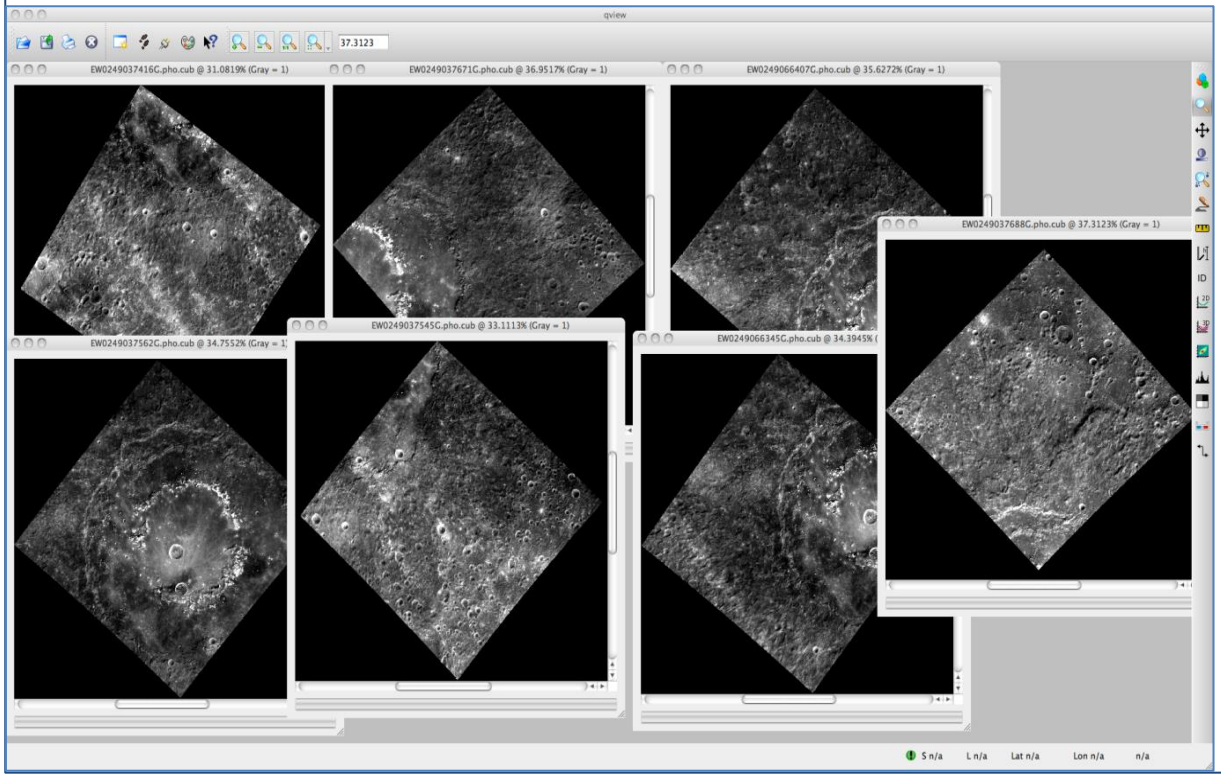
- Pixel resolution is the same for all images (*pixres=map*)
- Latitude/longitude range varies image-to-image (*defaultrange=minimize)*
- CenterLatitude and CenterLongitude must exist in map file
- ***mosrange*** result defines the map projection specifications (*map=equi.map*)

Photometric Correction

Level 3
Photometry
(photomet)

General Form (batch processing mode):

photomet –batchlist=basename.lis from=\$1.lev2.cub to=\$1.pho.cub
phtname=hapkehen theta=17.76662946 wh=0.278080114
hg1=0.227774899 hg2=0.714203968 hh=0.075 b0=2.3
zerob0standard=false **normname=albedo**
incref=30.0 incmat=0.0 thresh=10e30 albedo=1.0

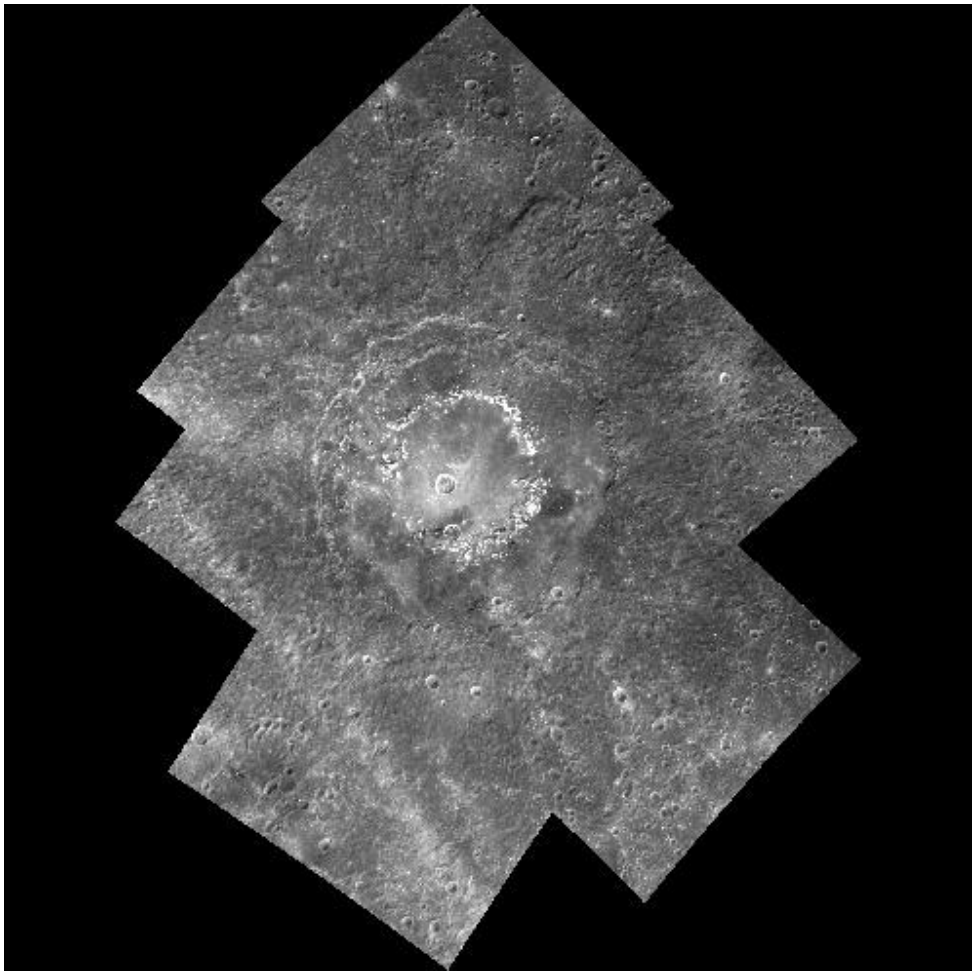


NOTES:

- Apply a photometric correction on all projected ".lev2.cub's" using values for the "G" filter
- MDIS NAC and WAC-G use the same parameters as they are very close in wavelength
- The photometry values are work in progress and expected to change

Level 4
Mosaic
(automos mapmos)

Raditladi Basin

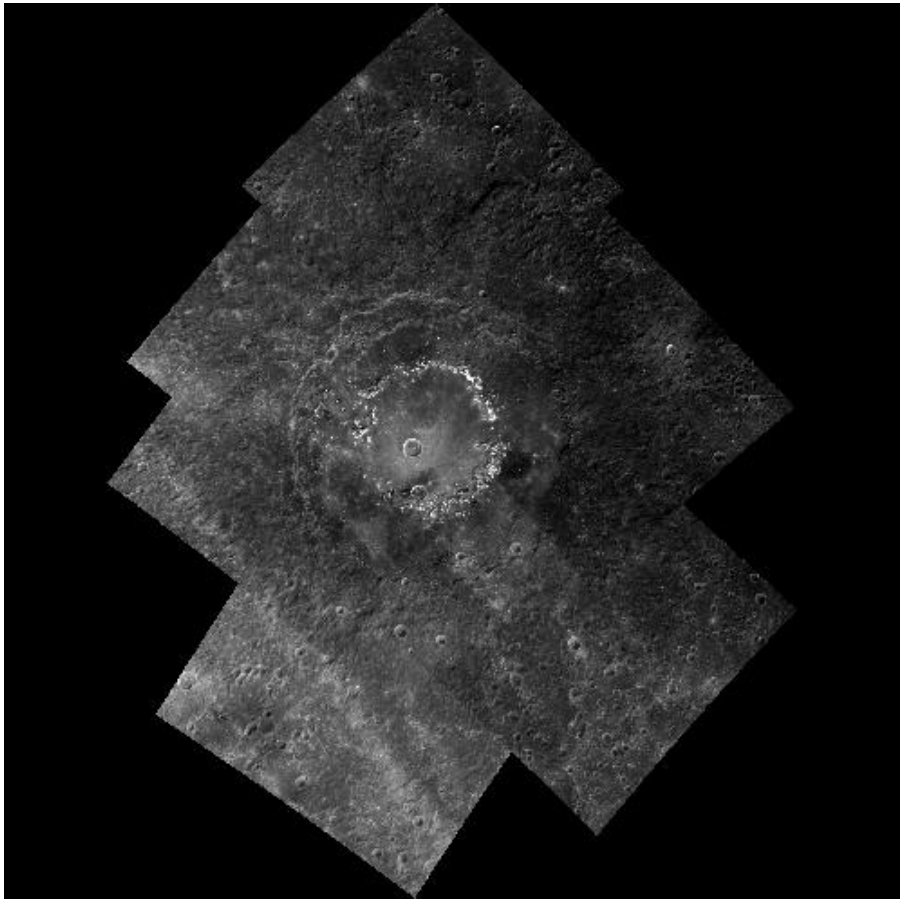


Assemble Monochrome Mosaic

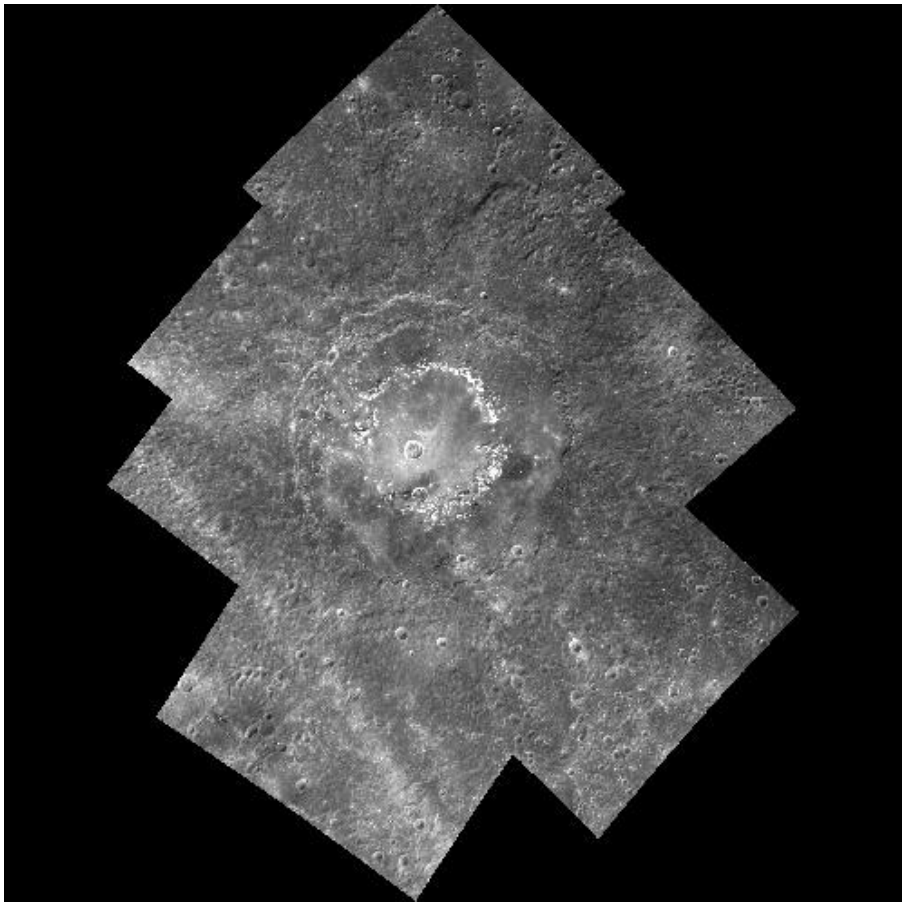
```
ls *.pho.cub > pho.lis
automos  fromlist=pho.lis mosaic=RaditladiBasin.cub
```

Photometric Correction

No Photometric Correction



After Photometric Correction



Hands-On Lesson 2

Create a 3-Color Map Mosaic (EDRs)

Create a 3-Color Regional Mosaic Map

- Objective: Use ISIS3 to create a 3-color MDIS mosaic of Raditladi Basin
  - This lesson includes some of the WAC-G images from the previous monochrome mosaic lesson
    - These images are part of the "Three Color" image campaign
    - Determined by OBSERVATION_TYPE keyword in PDS labels
    - There is a WAC-F and WAC-I accompanying image for each WAC-G
- Target ROI: Raditladi Basin
- Processing steps
  - Determine MDIS images to include in mosaic (PDS, PILOT, etc…)
    - http://pilot.wr.usgs.gov/
  - Apply the same Hands-On Lesson1 Monochrome processing steps up through Photometric Correction
  - Additional processing required for color registration within color sets
    - Coregistration of images to one another within each color set
    - Stack coregistered color set images into wavelength ordered single cube
    - Trim excess around edges within sets for seamless presentation
    - Assemble mosaic
- Refer to the Hands-On Lesson **RaditladiBasin_3Color**

Identify Color Sets

- Region of interest criteria: **Raditladi Basin**
  - http://www.nasa.gov/mission_pages/messenger/multimedia/messenger_orbit_image20130114_1.html
  - Latitude Range=(17N to 36.5N) Longitude Range=(111E to 128E)
  - Wavelength Filter = G, F, I InstrumentId = MDIS-WAC

http://pdsimage.wr.usgs.gov/Missions/MESSENGER/MSGRMDS_1001/DATA/2012_176/

Selected PDS-EDR Images [2012_176]

Color Set1:	EW0249037562**G**.IMG	EW0249037566**F**.IMG	EW0249037570**I**.IMG
Color Set2:	EW0249066345G.IMG	EW0249066349F.IMG	EW0249066353I.IMG
Color Set3:	EW0249066407G.IMG	EW0249066411F.IMG	EW0249066415I.IMG
Color Set4:	EW0249037688G.IMG	EW0249037692F.IMG	EW0249037696I.IMG
Color Set5:	EW0249037617G.IMG	EW0249037621F.IMG	EW0249037625I.IMG

Stacking Order for Color Mosaic (Blue,Green,Red)

EW0211111676F.IMG	Center = 433.2 <NM>
EW0211111682G.IMG	Center = 748.7 <NM>
EW0211111674I.IMG	Center = 996.2 <NM>

Standard Image Processing

#Ingestion

ls *.IMG | sed 's/\.IMG//' > basename.lis

mdis2isis -batchlist=basename.lis from=\$1.IMG to=\$1.cub

#SPICE

spiceinit -batchlist=basename.lis from=\$1.cub

#Radiometric Calibration

mdiscal -batchlist=basename.lis from=\$1.cub to=\$1.lev1.cub

#Prepare for qmos; evaluate the geometry characteristics of the images

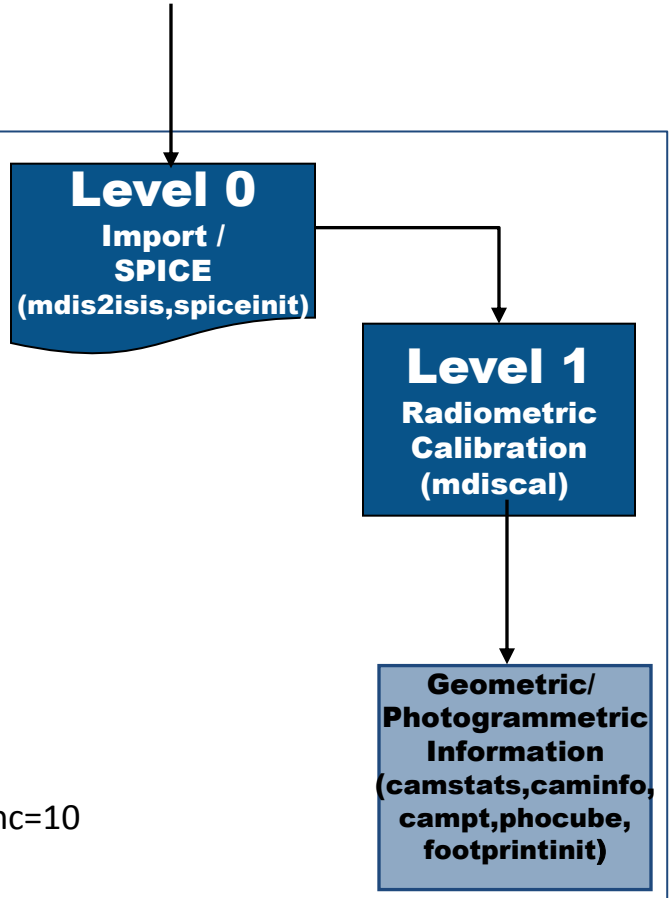
camstats -batchlist=basename.lis from=\$1.lev1.cub attach=true linc=10 sinc=10

footprintinit -batchlist=basename.lis from=\$1.lev1.cubcolorsets.lis

#Map template – Compute the average resolution, Center Latitude and Center Longitude, etc.

ls *G.lev1.cub > G_lev1.lis

mosrange fromlist=G_lev1.lis to=equi.map projection=equirectangular precision=0



Standard Image Processing

#Unix command using the 'stream' editor to remove the filename extensions to retain only the base name

ls *G.lev1.cub | sed 's/\.lev1\.cub//' > Gonly_basename.lis

#First map project the "G" filter images

cam2map -batchlist=Gonly_basename.lis from=\$1.lev1.cub to=\$1.lev2.cub map=equi.map pixres=map

Level 2
Map Projection
(mosrange, cam2map)

Contents of *colorsets.lis*

Column \$1 (match)	Column \$2	Column \$3
EW0249037562**G**	EW0249037566**F**	EW0249037570**I**
EW0249066345G	EW0249066349F	EW0249066353I
EW0249066407G	EW0249066411F	EW0249066415I
EW0249037688G	EW0249037692F	EW0249037696I
EW0249037617G	EW0249037621F	EW0249037625I

#Using a multiple column input list (space delimited), project the "F" and "I" images to match "G" Level2 output

Project the F images:

cam2map -batchlist=***colorsets.lis*** from=\$**2**.lev1.cub to=\$**2**.lev2.cub map=\$**1**.lev2.cub matchmap=true

Project the I images:

cam2map -batchlist=***colorsets.lis*** from=\$**3**.lev1.cub to=\$**3**.lev2.cub map=\$**1**.lev2.cub matchmap=true

Wavelength Dependent Photometric Correction

 # Photometry correction with coefficient values for "G"

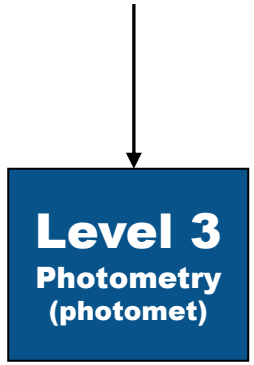
photomet -batchlist=colorsets.lis from=\$1.lev2.cub to=\$1.pho.cub phtname=hapkehen
theta=17.76662946 wh=0.278080114 hg1=0.227774899 hg2=0.714203968
hh=0.075 b0=2.3 zerob0standard=false
normname=albedo incref=30.0 incmat=0.0 thresh=10e30 albedo=1.0

 # Photometry correction with coefficient values for "F"

photomet -batchlist=colorsets.lis from=\$2.lev2.cub to=\$2.pho.cub phtname=hapkehen
theta=12.07775431 wh=0.153713769 hg1=0.221313433 hg2=0.887633784
hh=0.075 b0=2.3 zerob0standard=false
normname=albedo incref=30.0 incmat=0.0 thresh=10e30 albedo=1.0

 # Photometry correction with coefficient values for "I"

photomet -batchlist=colorsets.lis from=\$3.lev2.cub to=\$3.pho.cub phtname=hapkehen
theta=18.41686847 wh=0.35324478 hg1=0.276538744 hg2=0.613700193
hh=0.075 b0=2.3 zerob0standard=false
normname=albedo incref=30.0 incmat=0.0 thresh=10e30 albedo=1.0



Color Co-Registration

 # Pattern-match control points and "rubber sheet" the F-filter and I-filter to the G Filter for each color set

 # See important discussion on pattern matching

http://isis.astrogeology.usgs.gov/documents/PatternMatch/PatternMatch.html

Contents of *colorsets.lis*

Column \$1 (match)	Column \$2	Column \$3
EW0249037562**G**	EW0249037566**F**	EW0249037570**I**
EW0249066345G	EW0249066349F	EW0249066353I
EW0249066407G	EW0249066411F	EW0249066415I
EW0249037688G	EW0249037692F	EW0249037696I
EW0249037617G	EW0249037621F	EW0249037625I

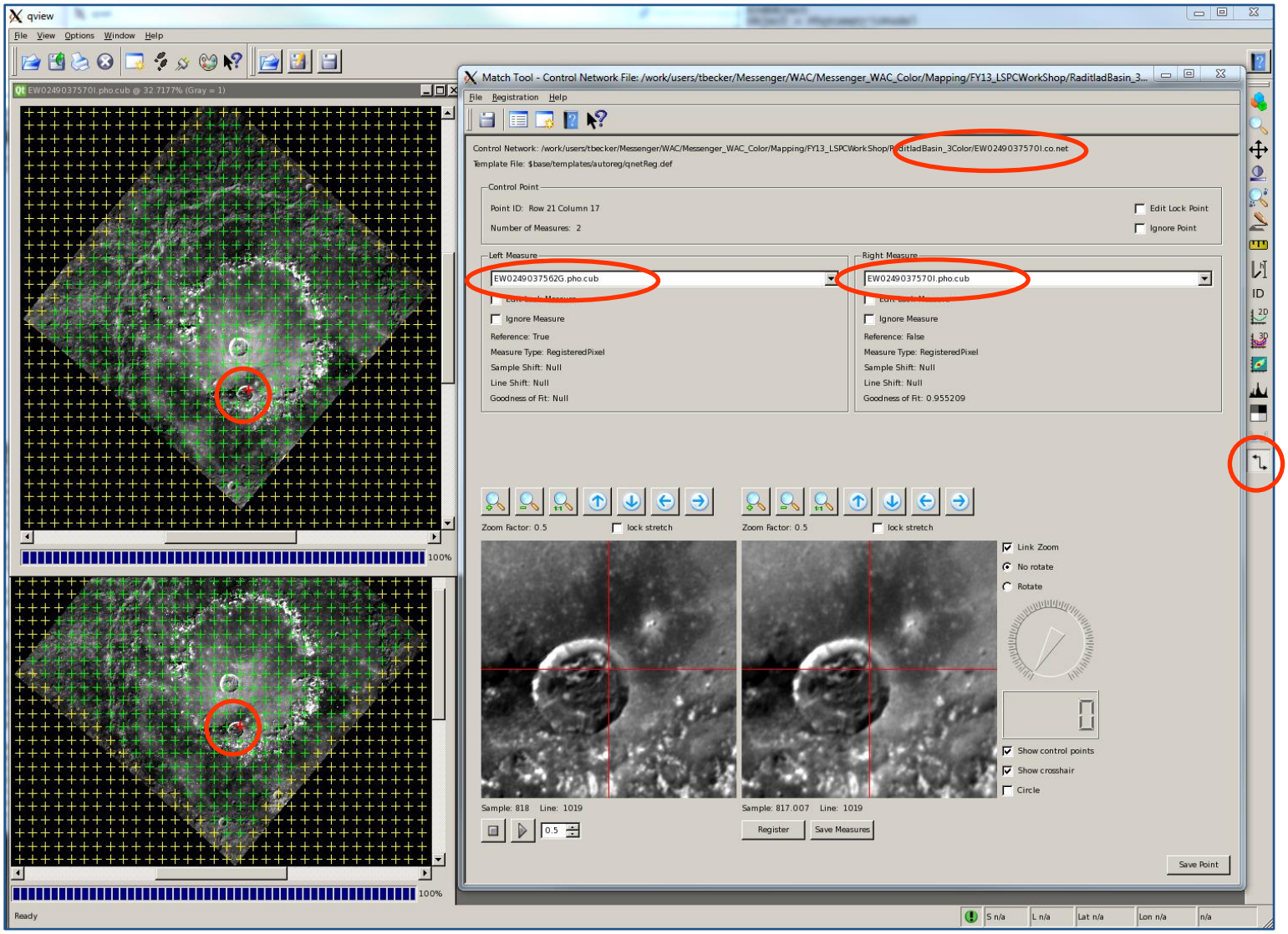
coreg -batchlist=colorsets.lis from=\$2.pho.cub match=\$1.pho.cub to=\$2.co.cub deffile=coreg.def
onet=\$2.co.net transform=warp degree=2 interp=bilinear rows=32 columns=32

coreg -batchlist=colorsets.lis from=\$3.pho.cub match=\$1.pho.cub to=\$3.co.cub deffile=coreg.def
onet=\$3.co.net transform=warp degree=2 interp=bilinear rows=32 columns=32

Color Co-Registration

Qview: *MatchTool*

Displays the **coreg** output control point network of registered pixel location measurements between “I” and “G” images.
The ***MatchTool*** supports interactive editing of the points



Multi-Band Processing

 # Stack sets into a multi-band cube (*cubeit*) and trim down to common coverage (*bandtrim*)

```
cubeit from=set1.lis to=set1.cub
bandtrim from=set1.cub to=set1_tr.cub

cubeit from=set2.lis to=set2.cub
bandtrim from=set2.cub to=set2_tr.cub

cubeit from=set3.lis to=set3.cub
bandtrim from=set3.cub to=set3_tr.cub

cubeit from=set4.lis to=set4.cub
bandtrim from=set4.cub to=set4_tr.cub

cubeit from=set5.lis to=set5.cub
bandtrim from=set5.cub to=set5_tr.cub
```

EW0249037566F.co.cub
EW0249037562G.pho.cub
EW0249037570I.co.cub

EW0249066349F.co.cub
EW0249066345G.pho.cub
EW0249066353I.co.cub

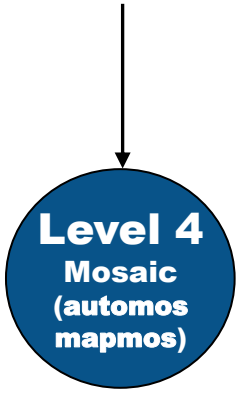
EW0249066411F.co.cub
EW0249066407G.pho.cub
EW0249066415I.co.cub

EW0249037692F.co.cub
EW0249037688G.pho.cub
EW0249037696I.co.cub

EW0249037621F.co.cub
EW0249037617G.pho.cub
EW0249037625I.co.cub

#Create a multi-band mosaic
 # *automos* computes the full lat/lon extents of all input images for the user

```
ls *set*tr.cub > color_mos.lis
automos fromlist=color_mos.lis mosaic=RaditladiBasin_rgb.cub
```

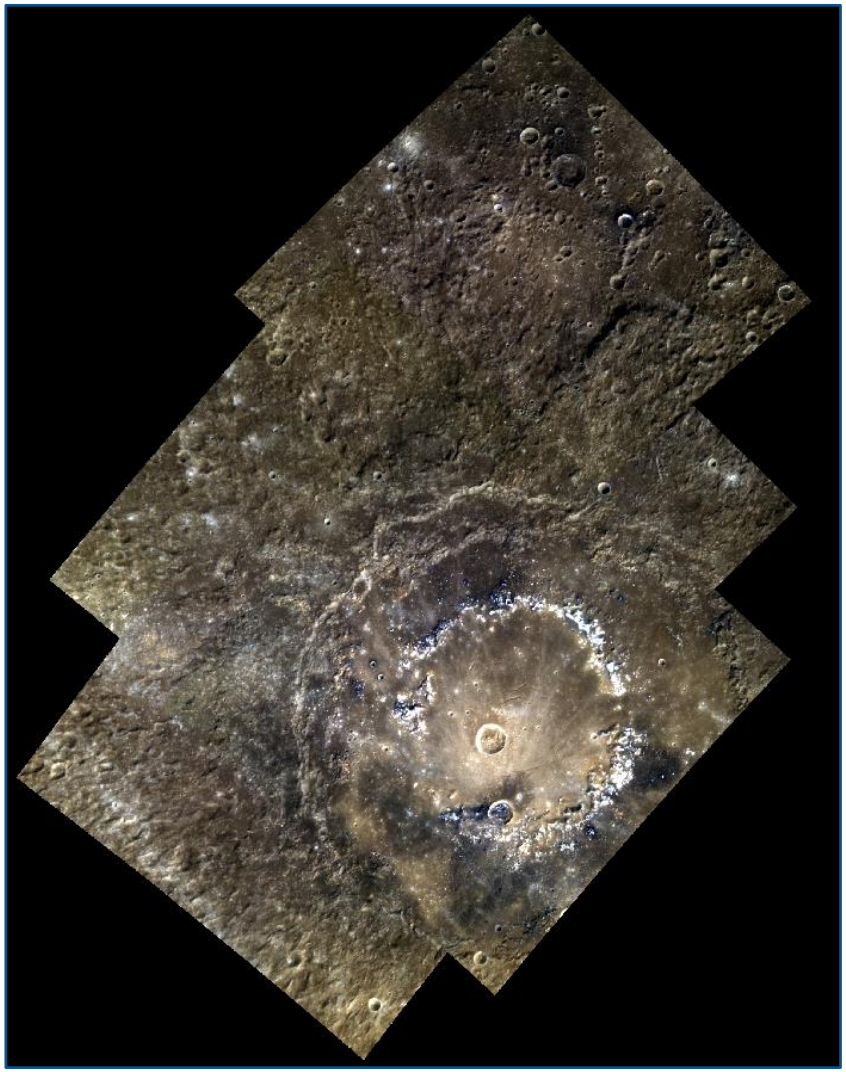


Raditladi Basin In 3-Color

Assembled 3-Color Mosaic

Three-Color Observation Filters

I (Red)	Center = 996.2 <NM>
G (Grn)	Center = 748.7 <NM>
F (Blu)	Center = 433.2 <NM>



Hands-On Lesson 3

Color Processing of an 8-Wavelength Observation

Color Processing of an 8-Wavelength Observation

- Objective: Use ISIS3 to create an 8-color MDIS set of Praxiteles
  - Images are part of the "Color" image campaign
    - Determined by OBSERVATION_TYPE keyword in PDS labels
    - Sets are comprised of one each of WAC-F, WAC-C, WAC-D, WAC-E, WAC-G, WAC-L, WAC-J and WAC-I (in wavelength order) filters
- Target ROI: **Praxiteles**
- Processing steps
  - Apply the same processing steps as in the Lesson 3-Color up through stacking
  - Eight filters requires specialized order of images to retaining increasing wavelength order in color image cube
- Refer to the Praxiteles_8Color lesson

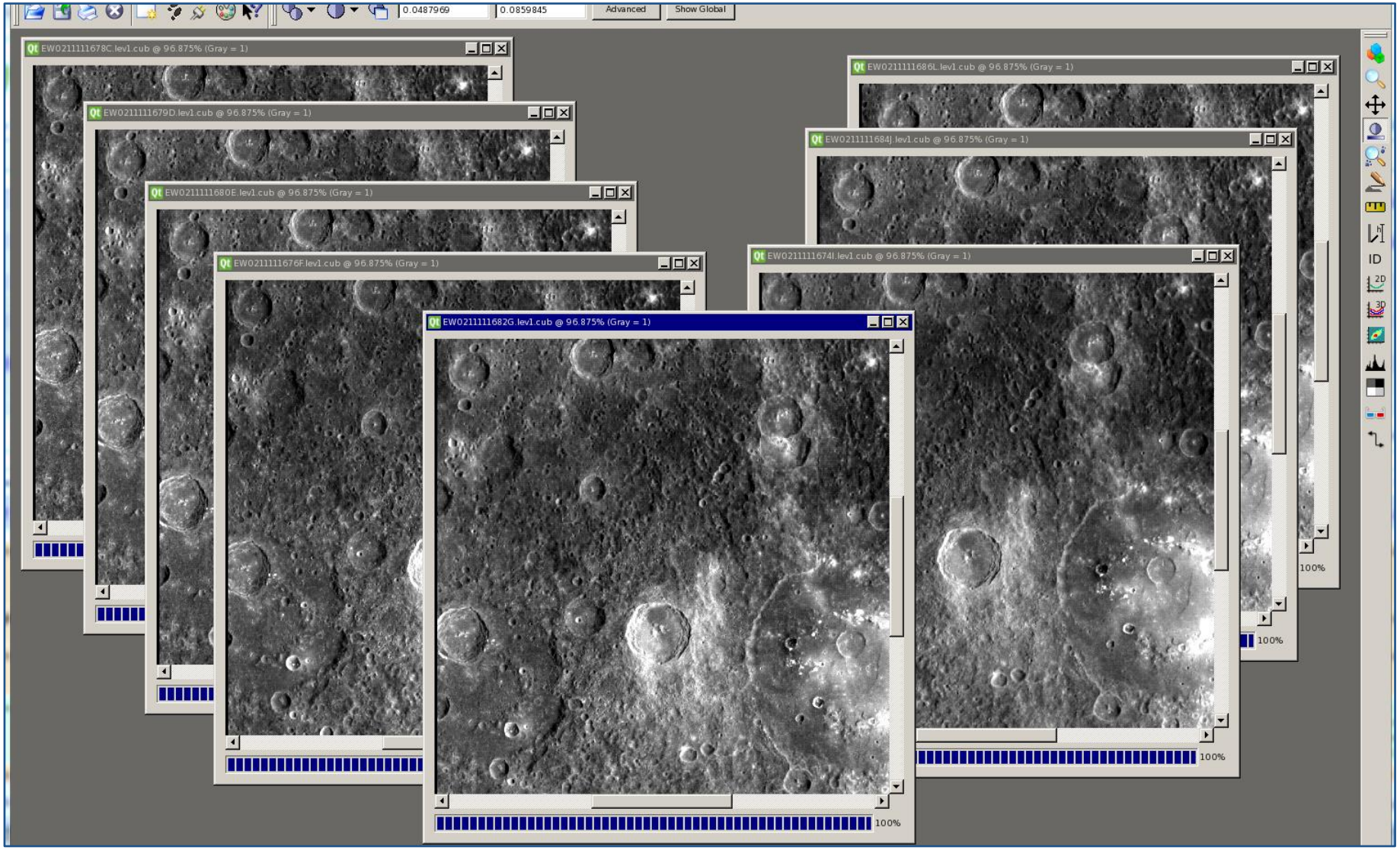
Create a Multi-Spectral Map

- Region of interest criteria: **Praxiteles**
  - http://www.nasa.gov/mission_pages/messenger/multimedia/messenger_orbit_image20130114_1.html
  - Latitude Range=(24N to 39N) Longitude Range=(290E to 307E)
  - Wavelength Filter = All available InstrumentId = MDIS-WAC

Selected PDS-EDR Images [2011_102]

EW0211111676F.IMG	Center = 433.2 <NM>
EW0211111678C.IMG	Center = 479.9 <NM>
EW0211111679D.IMG	Center = 558.9 <NM>
EW0211111680E.IMG	Center = 628.8 <NM>
EW0211111682G.IMG	Center = 748.7 <NM>
EW0211111686L.IMG	Center = 828.4 <NM>
EW0211111684J.IMG	Center = 898.8 <NM>
EW0211111674I.IMG	Center = 996.2 <NM>

Create a Multi-Spectral Map



Standard Image Processing

#Ingestion

ls *IMG | sed 's/\.IMG//' > basename.lis

mdis2isis -batchlist=basename.lis from=\$1.IMG to=\$1.cub

#Spice

spiceinit -batchlist=basename.lis from=\$1.cub

#Radiometric Calibration

mdiscal -batchlist=basename.lis from=\$1.cub to=\$1.lev1.cub

#Map template – Compute the average resolution, Center Latitude and Center Longitude, etc.

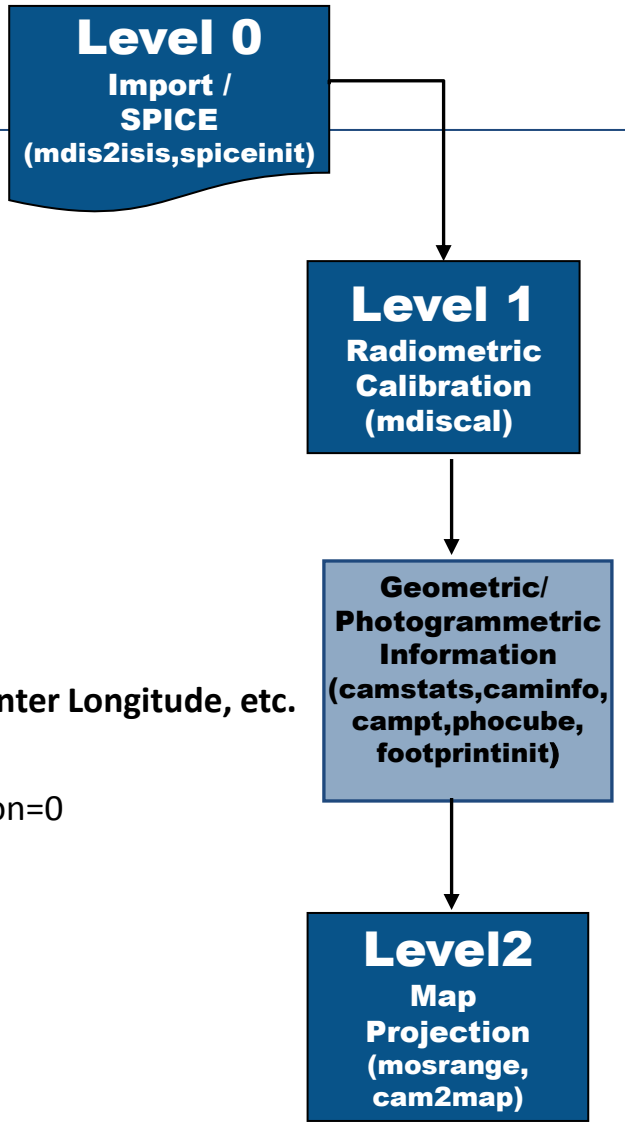
ls *.lev1.cub > lev1.lis

mosrange fromlist=lev1.lis to=equi.map projection=equirectangular precision=0

#Map project all eight wavelength images

cam2map -batchlist=basename.lis from=\$1.lev1.cub to=\$1.lev2.cub

map=equi.map pixres=map defaultrange=map



Wavelength Dependent Photometric Correction

 # All photometric parameters for 8 WAC filters

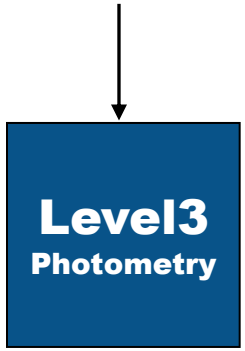
photomet from=EW0211111678**C**.lev2.cub to=EW0211111678C.pho.cub phtname=hapkehen
theta=13.82780392 wh=0.182212955 hg1=0.212533357 hg2=0.856934992
hh=0.075 b0=2.3 zerob0standard=false
normname=albedo incref=30.0 thresh=10e30 albedo=1.0

photomet from=EW0211111679**D**.lev2.cub to=EW0211111679D.pho.cub phtname=hapkehen
theta=15.78892162 wh=0.215984749 hg1=0.206649235 hg2=0.811417942
hh=0.075 b0=2.3 zerob0standard=false
normname=albedo incref=30.0 thresh=10e30 albedo=1.0

photomet from=EW0211111680**E**.lev2.cub to=EW0211111680E.pho.cub phtname=hapkehen
theta=16.87688941 wh=0.239885991 hg1=0.209211245 hg2=0.773783766
hh=0.075 b0=2.3 zerob0standard=false
normname=albedo incref=30.0 thresh=10e30 albedo=1.0

photomet from=EW0211111676**F**.lev2.cub to=EW0211111676F.pho.cub phtname=hapkehen
theta=12.07775431 wh=0.153713769 hg1=0.221313433 hg2=0.887633784
hh=0.075 b0=2.3 zerob0standard=false
normname=albedo incref=30.0 thresh=10e30 albedo=1.0

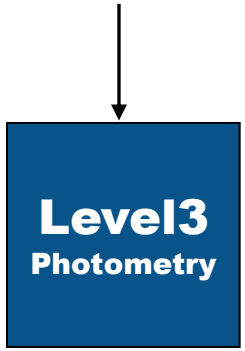
photomet from=EW0211111682**G**.lev2.cub to=EW0211111682G.pho.cub phtname=hapkehen
theta=17.76662946 wh=0.278080114 hg1=0.227774899 hg2=0.714203968
hh=0.075 b0=2.3 zerob0standard=false
normname=albedo incref=30.0 thresh=10e30 albedo=1.0



8-Color Set Registration

 # All photometric parameters for 8 WAC filters, continued…

```
photomet  from=EW0211111674I.lev2.cub to=EW0211111674I.pho.cub phtname=hapkehen
        theta=18.41686847 wh=0.35324478 hg1=0.276538744 hg2=0.613700193
        hh=0.075 b0=2.3 zerob0standard=false
        normname=albedo incref=30.0 thresh=10e30 albedo=1.0

photomet  from=EW0211111684J.lev2.cub to=EW0211111684J.pho.cub phtname=hapkehen
        theta=18.07191127 wh=0.32654443 hg1=0.261680383 hg2=0.650146228
        hh=0.075 b0=2.3 zerob0standard=false
        normname=albedo incref=30.0 thresh=10e30 albedo=1.0

photomet  from=EW0211111686L.lev2.cub to=EW0211111686L.pho.cub phtname=hapkehen
        theta=17.96224797 wh=0.304047732 hg1=0.245886415 hg2=0.678657724
        hh=0.075 b0=2.3 zerob0standard=false
        normname=albedo incref=30.0 thresh=10e30 albedo=1.0
```



#Unix command to create a list of the photomet output, strip off the filename extension and remove the G filter from the output list

```
ls  *.pho.cub  |  sed 's/\.pho\.cub//'  | grep -v G  >  no_g.lis
```

 # Pattern match control points and rubber sheet all the filters to the G Filter for each color set

```
coreg  -batchlist=no_g.lis  from=\$1.pho.cub deffile=coreg.def
    to=\$1.co.cub  onet=\$1.co.cub  match=EW0211111682G.pho.cub  transform=warp degree=2 interp=bilinear rows=32 columns=32
```

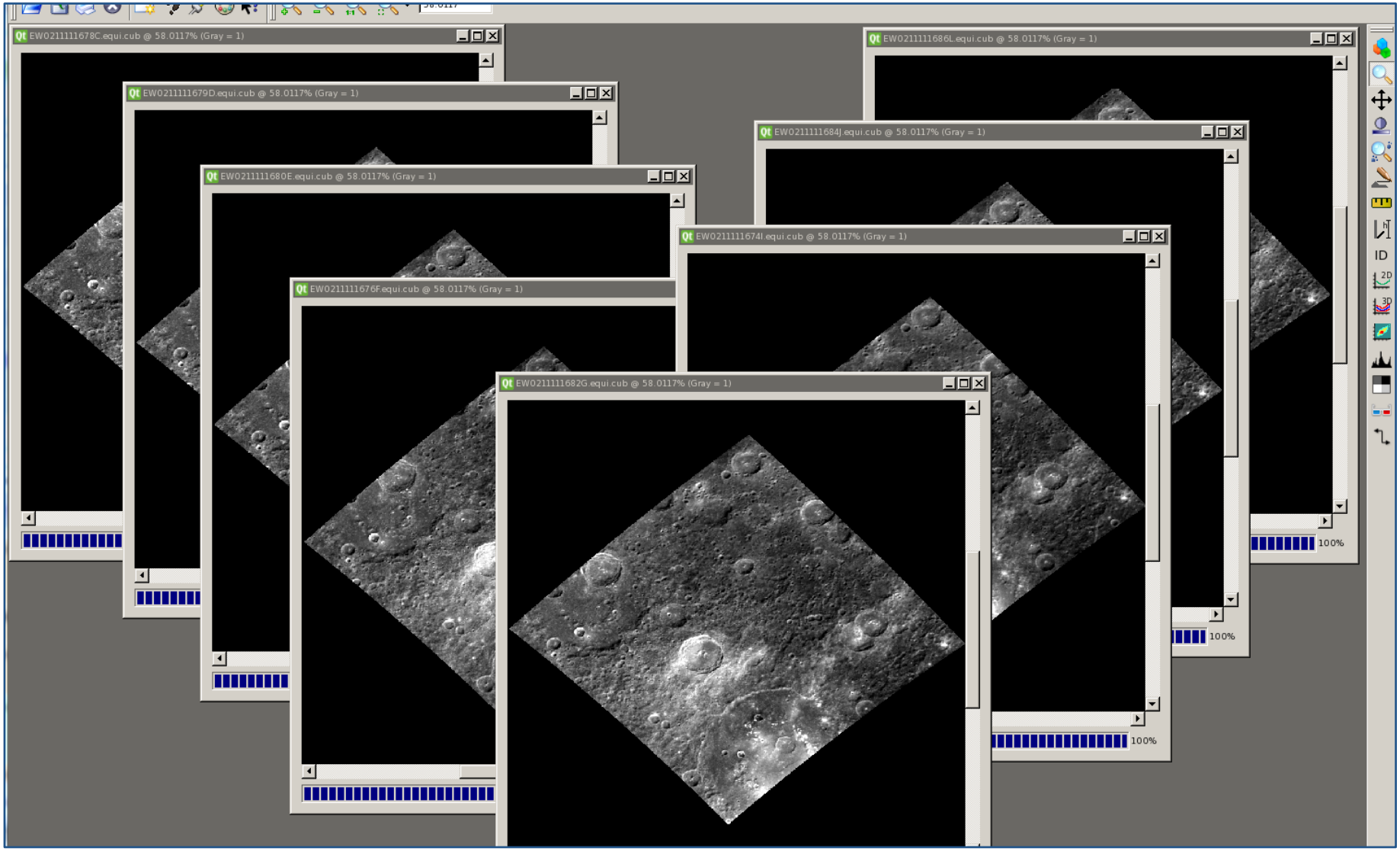
#Stack the results

```
cubeit fromlist=color_order.lis to=Praxiteles_stack.cub
```

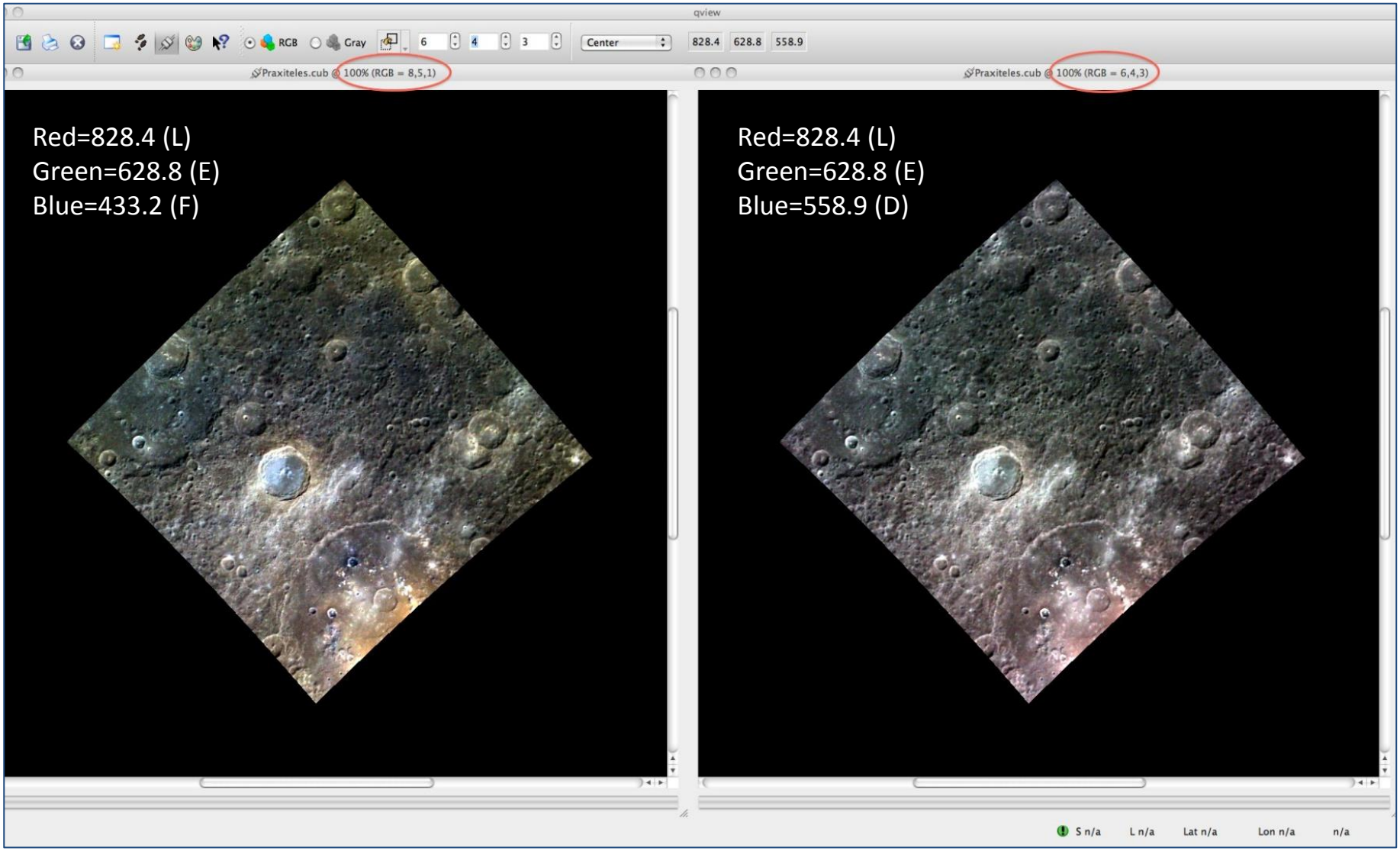
#Trim down to common coverage through the stack

```
bandtrim from=Praxiteles_stack.cub to=Praxiteles.cub
```

8-Color Set Registration



Praxiteles: RGB-Color Combinations



Processing MESSENGER PDS BDR and MDR Data Map Products

Processing MESSENGER PDS BDR and MDR Data Products

- High level global data products available through PDS
  - Intended to free users from having to derive their own products from raw (EDR) data
- Two MESSENGER PDS data products
  - Map Projected Basemap RDR (BDR)
    - http://pdsimage.wr.usgs.gov/Missions/MESSENGER/MSGRMDS_4001/
  - Map Projected Multispectral Basemap RDR (MDR)
    - http://pdsimage.wr.usgs.gov/Missions/MESSENGER/MSGRMDS_5001/
- ISIS3 provides a PDS ingestion application called ***pds2isis***
  - Imports the data and label keywords into ISIS3 format
  - Translates the PDS map projection keywords and properties
  - After ingestion, ISIS3 can be used to produce maps of different projection types and resolutions (***map2map***)

Processing MESSENGER PDS BDR and MDR Data Products

- MESSENGER Monochrome Basemap Mosaic (BDR)
  - Product Highlights
    - Coverage is very nearly global
    - Comprised of MDIS NAC and WAC-G images
    - Map projected to Equirectangular projection
      - Fifty-four non-overlapping *tiles* in fifteen quadrangles
    - North and South poles are available in Polar Stereographic
    - Map Resolution = ~166 meters/pixel (265 pixels/degree)
    - Contains five additional reference data *backplanes*
      - ObservationId, BDR stacking metric (mosaic ordering), Solar Incidence angle, Emission angle and Phase angle
- RDR SIS contains complete description of BDR products
  - http://pdsimage.wr.usgs.gov/Missions/MESSENGER/MSGRMDS_4001/DOCUMENT/MDIS_CDR_RDRSIS.PDF
- Important reference *prior* to usage and analysis with this data!
  - http://pdsimage.wr.usgs.gov/Missions/MESSENGER/MSGRMDS_4001/CATALOG/MDIS_BDR_DS.CAT

Processing MESSENGER PDS BDR and MDR Data Products

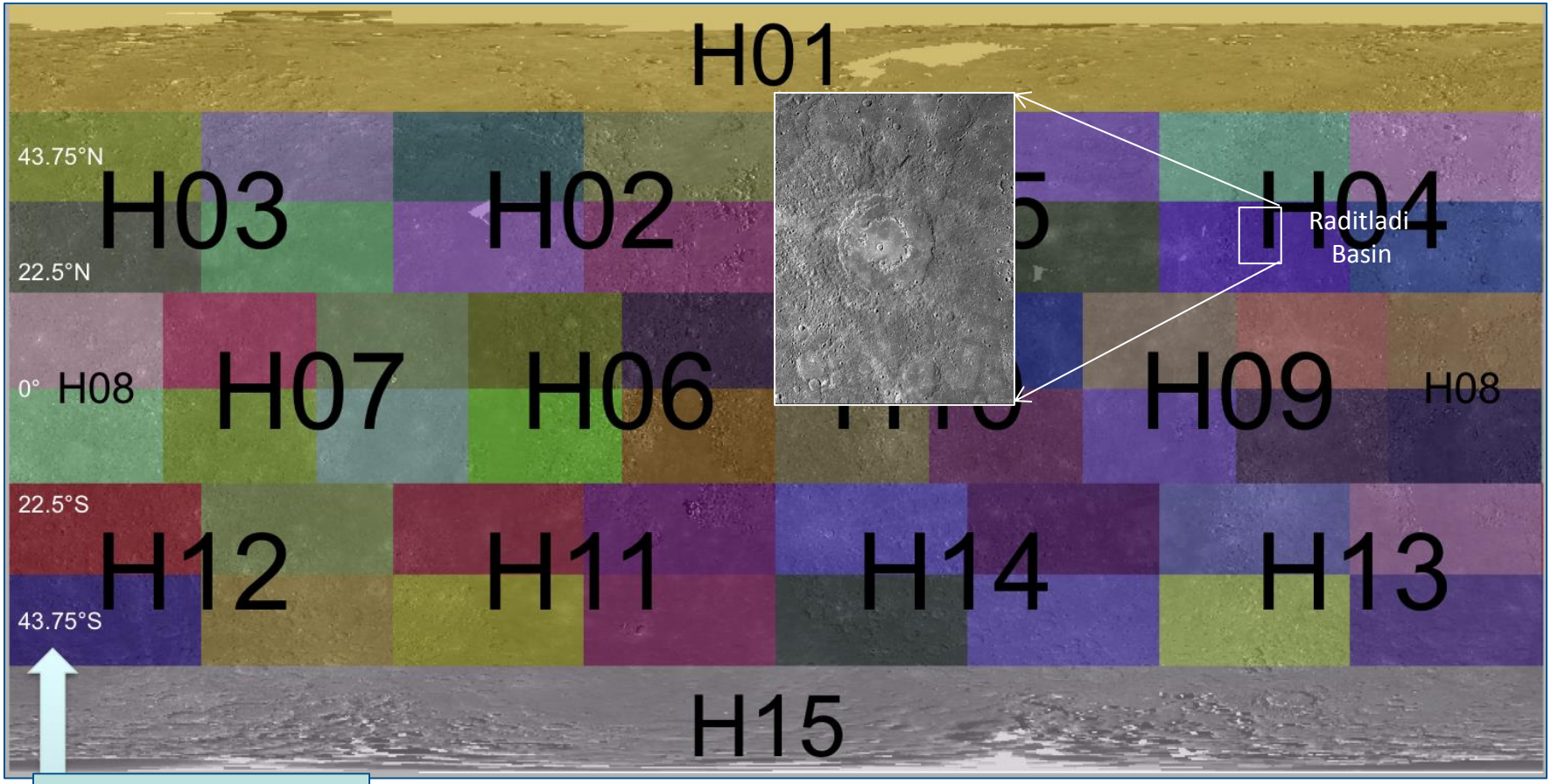
- **Objective:** Extract the Raditladi region from the PDS BDR and MDR map products
  - Remap the PDS BDR monochrome and a MDR color map products for direct comparison with the ISIS3 EDR derived monochrome and 3-color mosaics
  - This requires a re-projection of all BDR/MDR tiles that fall within the latitude and longitude extents of the ISIS3 EDR Raditladi Basin mosaics
  - The ISIS3 EDR Raditladi Basin regional mosaic coverage is Latitude range of 17°N to 36°N and Longitude range of 111° to 128° Positive East
- Details to consider:
  - Resolution Differences
    - EDR monochrome product has a map resolution of 262 meters/pixel
    - The PDS BDR product has a map resolution of 166 meters/pixel
  - Map Projection Differences
    - Each product differs in center latitude and longitude
    - Longitude Domain
      - PDS BDR and MDR products are projected in the 180° longitude domain
      - The ISIS3 EDR Lesson products are projected in the 360° longitude domain

Processing MESSENGER PDS BDR and MDR Data Products

- Table of tile boundaries for all Mercury Quadrangles
- Each tile contains four quadrants (NW, NE, SW and SE) excluding North and South poles (54 in all)
- Longitude Positive East 180° domain
- All BDR quad tiles have a different center latitude and center longitude

Quadrangle	Subdirectory name	Latitude (degrees)	Longitude (deg. east)
H-1 Borealis	H01	65 to 90	0 to 360
H-2 Victoria	H02	22.5 to 65	270 to 360
H-3 Shakespeare	H03	22.5 to 65	180 to 270
H-4 Liguria	H04	22.5 to 65	90 to 180
H-5 Apollonia	H05	22.5 to 65	0 to 90
H-6 Kuiper	H06	-22.5 to 22.5	288 to 360
H-7 Beethoven	H07	-22.5 to 22.5	216 to 288
H-8 Tolstoj	H08	-22.5 to 22.5	144 to 216
H-9 Solitudo Criophori	H09	-22.5 to 22.5	72 to 144
H-10 Pieria	H10	-22.5 to 22.5	0 to 72
H-11 Discovery	H11	-65 to -22.5	270 to 360
H-12 Michelangelo	H12	-65 to -22.5	180 to 270
H-13 Solitudo Persephones	H13	-65 to -22.5	90 to 180
H-14 Cyllene	H14	-65 to -22.5	0 to 90
H-15 Bach	H15	-90 to -65	0 to 360

Processing MESSENGER PDS BDR and MDR Data Products



Hands-On Lesson 4

MDIS BDR Monochrome Mosaic

MDIS BDR Monochrome Mosaic

- Objective: Use ISIS3 to create an MDIS monochrome mosaic from PDS BDRs of Raditladi Basin
  - http://pdsimage.wr.usgs.gov/Missions/MESSENGER/MSGRMDS_4001/BDR/
  - Because each PDS quadrant has a different CenterLatitude/CenterLongitude, reprojection is required to a common center coordinate in order to combine into map mosaic
- General Processing Steps
  - Determine quadrants in quads with ROI coverage (H04SW0, H09NE0)
  - Download images from PDS
  - Process using ISIS3
    - Import (***pds2isis***)
    - map project (***map2map***)
    - mosaic (***automos***)
- Refer to the **MDIS_BDR** Lesson

MDIS BDR Processing Sequence

- A Perl script, ***mdis_pds_proc***, provides a one line command to process a list of PDS BDR or MDR quadrant files or ***pds2isis*** converted ISIS cubes
- Parameters
  - **--list**=Raditladi.lis (A generated list of the input PDS Quads)

ls -1 MDIS_BDR_256PPD_*.LBL > Raditladi.lis

  - ***-map****=Raditladi.map* (Use the map file created in RaditladiBasin lesson)
  - ***--matchmap*** = flag that sets the matchmap parameter to true for cam2map that ensures all output map projections cover same area and resolution
  - **--mosaic** = Output filename of reconstructed BDR product

./mdis_pds_proc -v --bdr --list=Raditladi.lis –map=Raditladi.map --matchmap --mosaic=Raditladi_bdr.cub

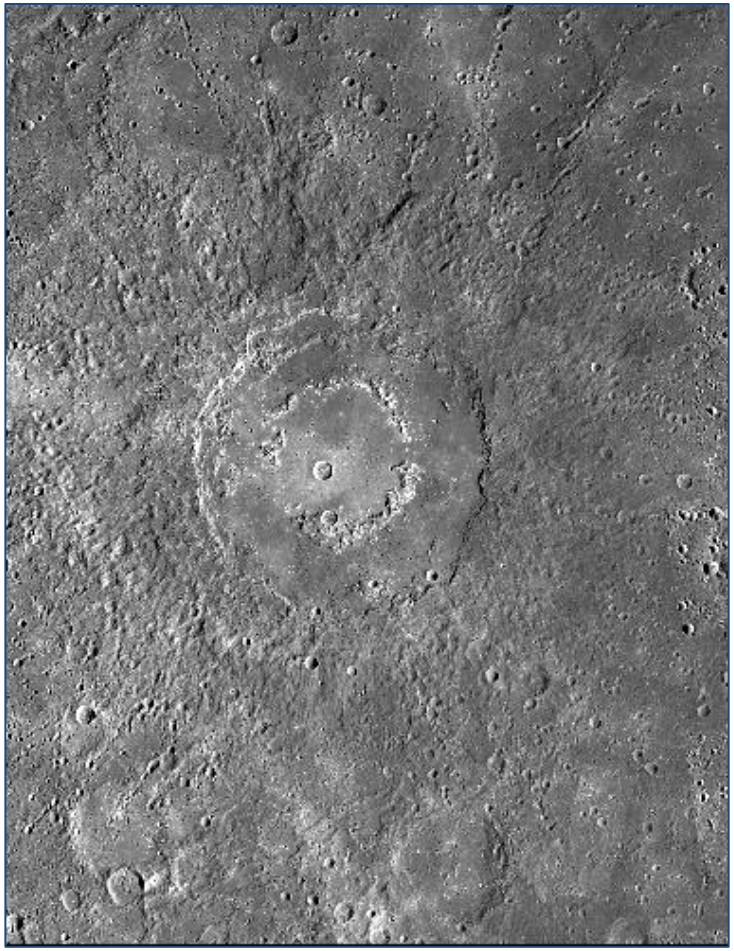
  - Use **--bdr** for monochrome option (extracts and projects first band only)

MDIS BDR Processing Sequence

#Execution sequence of the Perl script ***mdis_pds_proc*** for ***–bdr*** option

```
pds2isis from=MDIS_BDR_256PPD_H04SW0.LBL to=mdis_bdr_256ppd_h04sw0.pds.cub
map2map from=mdis_bdr_256ppd_h04sw0.pds.cub+1
            to=mdis_bdr_256ppd_h04sw0.proj.cub map=Raditladi.map matchmap=true

pds2isis from=MDIS_BDR_256PPD_H09NE0.LBL to=mdis_bdr_256ppd_h09ne0.pds.cub
map2map from=mdis_bdr_256ppd_h09ne0.pds.cub+1
             to=mdis_bdr_256ppd_h09ne0.proj.cub  map=Raditladi.map matchmap=true

ls -1 *.proj.cub > mosfiles.lis
automos fromlist=mosfiles.lis mosaic=Raditladi_bdr.cub matchbandbin=false priority=beneath
```

MDIS BDR Monochrome Mosaic



Raditladi Basin
Extracted from PDS BDR

Hands-On Lesson 5

MDIS MDR Color Mosaic

MDIS MDR 3-Color Mosaic

- Objective: Use ISIS3 to create an MDIS 3-Color (or 8-Color) mosaic from PDS MDRs of Raditladi Basin
  - http://pdsimage.wr.usgs.gov/Missions/MESSENGER/MSGRMDS_5001/MDR/
  - Like BDRs, each MDR quadrant has a different CenterLatitude/CenterLongitude
    - Reprojection is required to a common center coordinate to combine into map mosaic
- General Processing Steps
  - Same quadrants as BDR mosaic (H04SW0, H09NE0)
  - Download images from PDS
  - Process using ISIS3
    - Import (***pds2isis***)
    - map project (***map2map***)
    - mosaic (***automos***)
- Refer to the MDIS_MDR lesson

MDIS BDR Processing Sequence

- A Perl script, ***mdis_pds_proc***, provides a one line command to process a list of PDS BDR or MDR quadrant files or ***pds2isis*** converted ISIS cubes
- Parameters
  - **--list**=Raditladi.lis (A generated list of the input PDS Quads)

ls -1 MDIS_BDR_256PPD_*.LBL > Raditladi.lis

  - ***-map****=Raditladi.map* (Use the map file created in RaditladiBasin lesson)
  - ***--matchmap*** = flag that sets the matchmap parameter to true for cam2map that ensures all output map projections cover same area and resolution
  - **--mosaic** = Output filename of reconstructed BDR product

./mdis_pds_proc –v –rgb --list=Raditladi.lis –map=Raditladi.map --matchmap --mosaic=Raditladi_bdr.cub

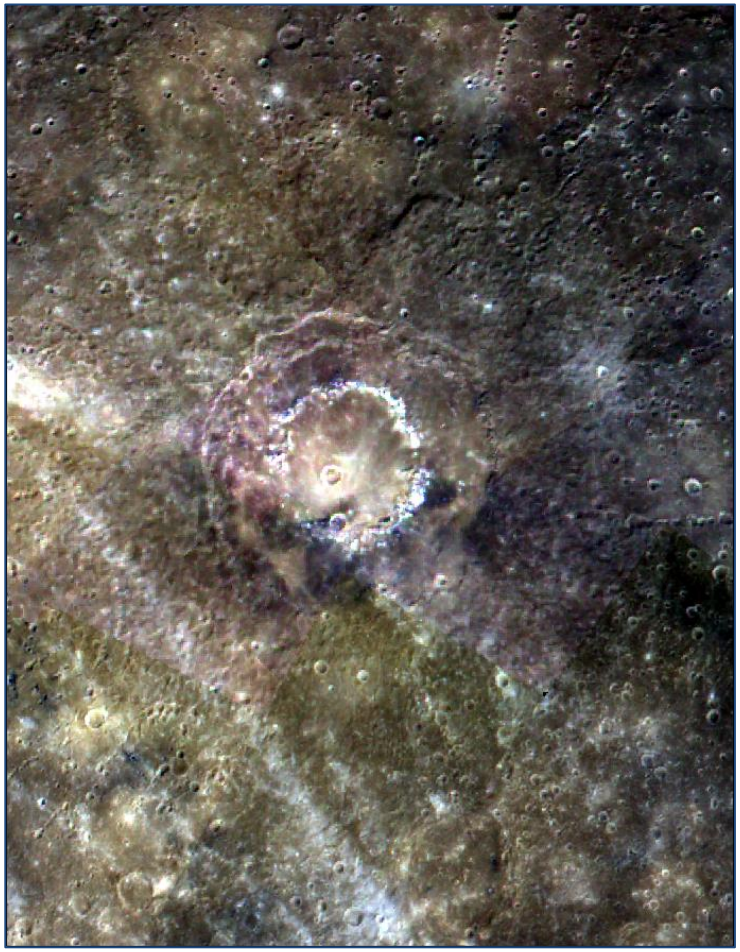
  - Use **--rgb** for 3-color option (extracts and projects bands 1, 5, 8 [I,G,F])
  - Use **--mdr** for 8-color option (extracts and projects first 1-8 bands)

MDIS MDR 3-Color Processing Sequence

 # Execution sequence of the Perl script *mdis_pds_proc* for *–rgb* option

```
pds2isis from=MDIS_MDR_064PPD_H04SW0.LBL to=mdis_mdr_064ppd_h04sw0.pds.cub
map2map from=mdis_mdr_064ppd_h04sw0.pds.cub+1,5,8
        to=mdis_mdr_064ppd_h04sw0.proj.cub map=Raditladi.map matchmap=true

pds2isis from=MDIS_MDR_064PPD_H09NE0.LBL to=mdis_mdr_064ppd_h09ne0.pds.cub
map2map from=mdis_mdr_064ppd_h09ne0.pds.cub+1,5,8
        to=mdis_mdr_064ppd_h09ne0.proj.cub map=Raditladi.map matchmap=true


ls -1 mdis_mdr_*.proj.cub > mosfiles.lis

automos fromlist=mosfiles.lis mosaic=Raditladi_rgb.cub matchbandbin=false priority=beneath
```

MDIS MDR 3-Color Processing Sequence



Raditladi Basin in 3-Color extracted from PDS-MDR

Three-Color Observation Filters

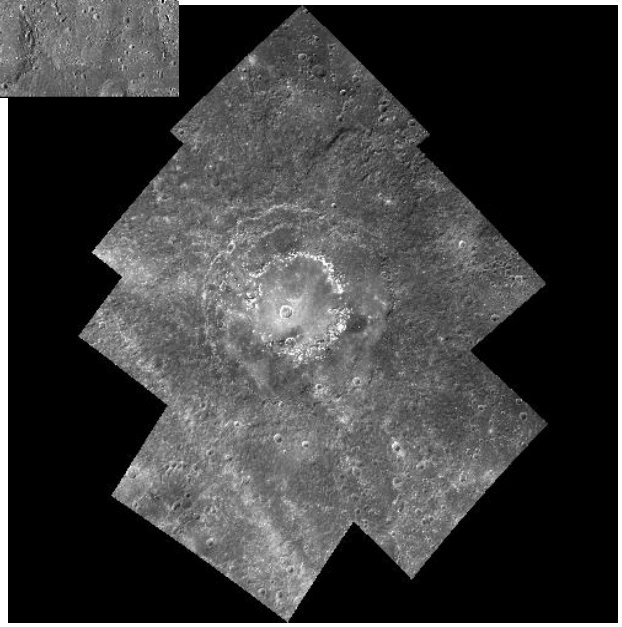
I (Red)	Center = 996.2 <NM>
G (Grn)	Center = 748.7 <NM>
F (Blu)	Center = 433.2 <NM>

Final Raditladi Basin Products

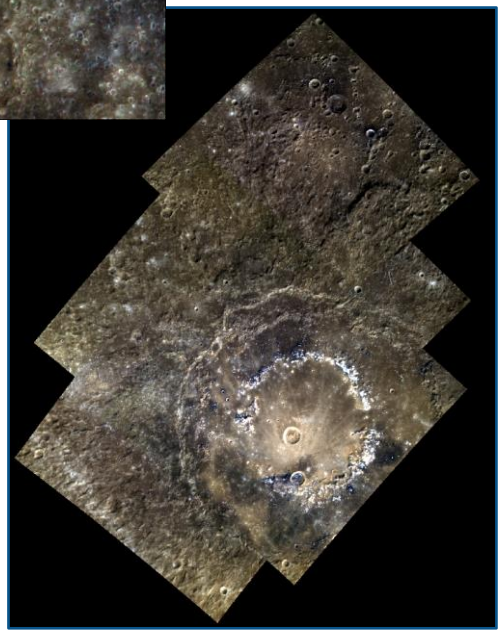
Lesson 4 - BDR

Lesson 5 – MDR (rgb)

Lesson 1 - EDR



Lesson 2- EDR



Considerations When Processing PDS BDRs and MDRs

- Use ***pds2isis*** to import all bands into an ISIS cube
- All quadrants must be reprojected to same center coordinate in order to mosaic into combined map product
- Special Perl script, **mdis_pds_proc**, is provided to simplify PDS BDR and MDR processing
  - Script documentation contains additional information/help
- PDS BDR and MDR products include additional information available for every pixel in the backplanes
- Don't project "OBSERVATION ID" band with anything other than nearest neighbor (i.e., no interpolation)
  - Otherwise computes an average of surrounding OBSERVATION IDs – nonsense!
  - Project alone or exclude entirely
    - Refer to **--bands** option in **mdis_pds_proc**
- The PDS BDR and MDR volumes are huge!
  - BDR is 69GB
  - MDR is 9GB

Hands-On Lesson 6

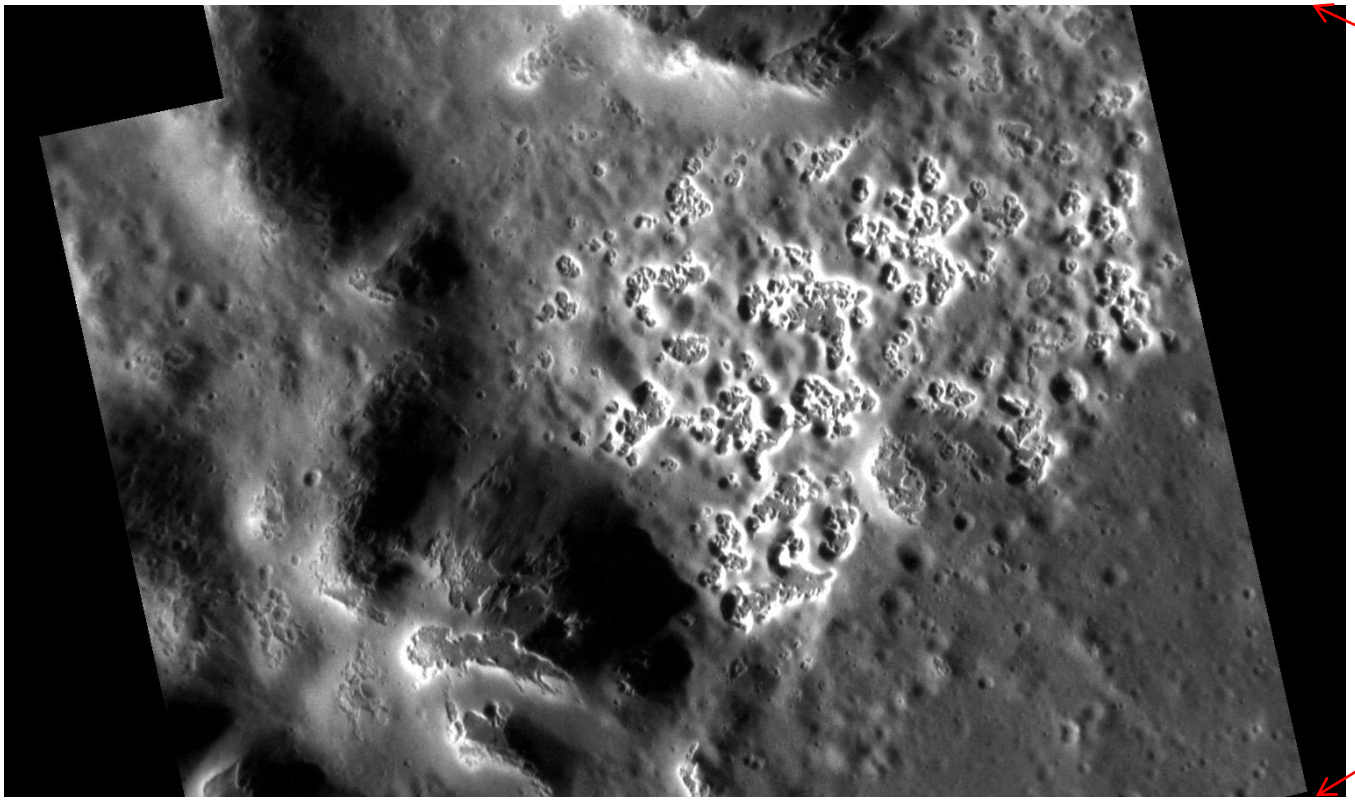
Very High Resolution NAC Mosaics

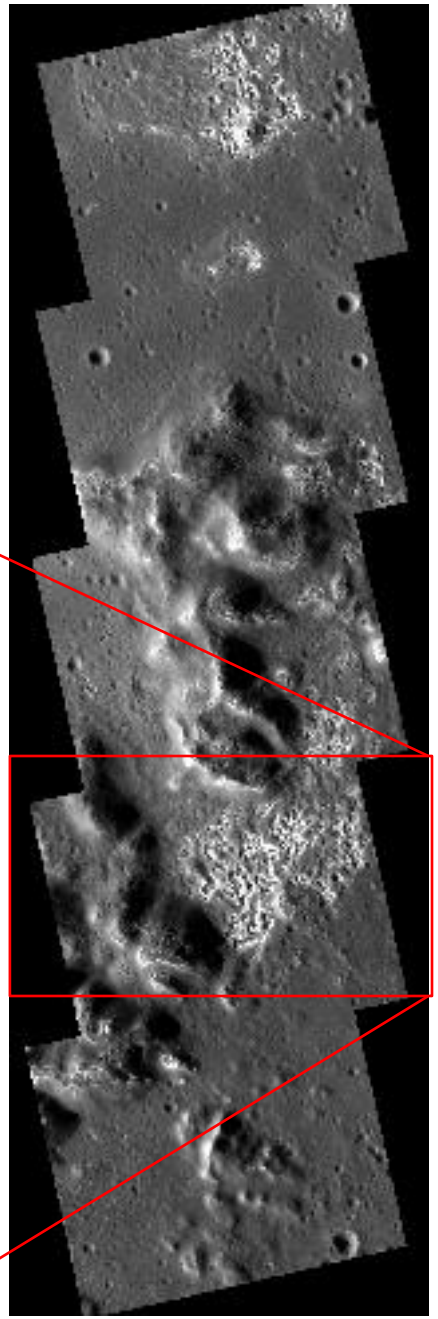
Very High Resolution NAC Mosaic

Raditladi Hollows

This MDIS NAC mosaic is ~17 meters/pixel and was acquired on 2011-08-04.

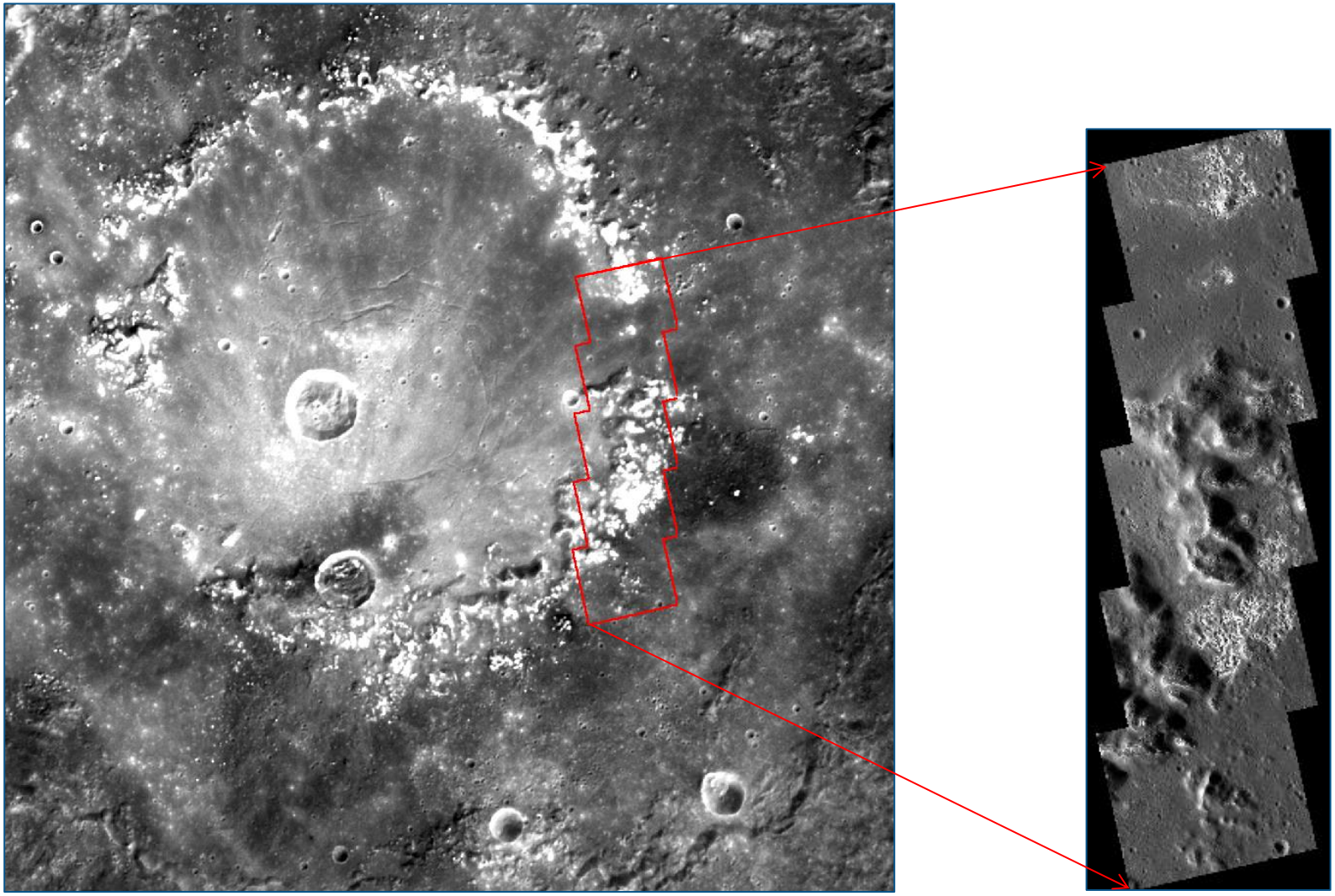
Refer to the RaditladiHollows Lesson





Very High Resolution NAC Mosaic

Raditladi Hollows



Export and Application Support for ISIS3 Products

Trent Hare
Cartographer thare@usgs.gov

Brief introduction to using data outside of ISIS.

- For more visit Astrogeology's **MRCTR GIS Lab**

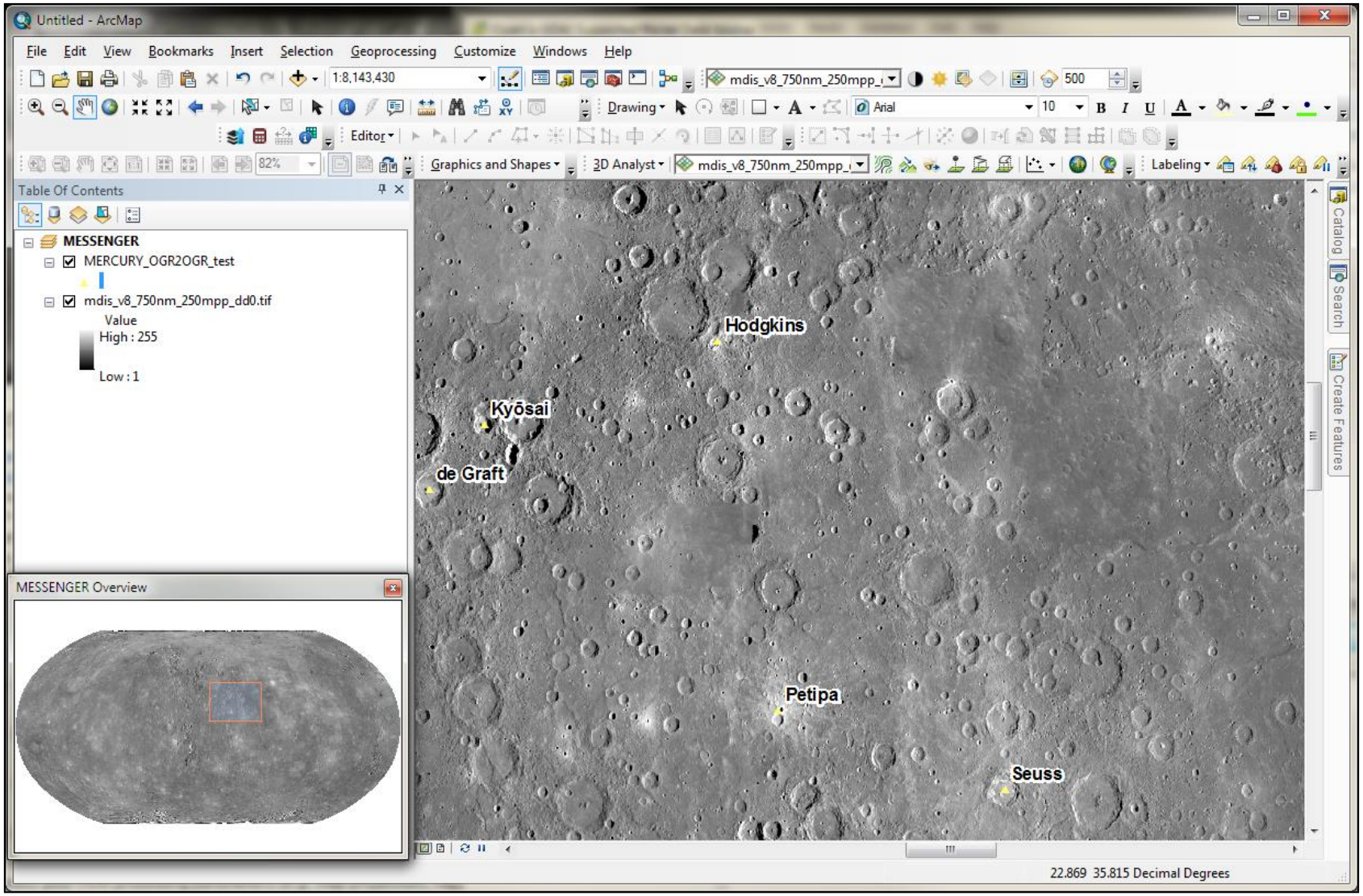
Exporting ISIS3 Data for Further Analysis

- For higher bit types and “mapping” formats users can:
  - Use the ISIS3 format **directly** (apps that use GDAL)
    - ArcMap GIS
    - QGIS, GRASS, TuiView
    - Mirone (GMT, Matlab)
    - Opticks
  - Convert to a standardized format (using GDAL tools)
    - GeoTiff (recommended, 8, 16, 32bit, 2TB)
    - GeoJpeg2000 (8, 16bit, 2TB limit?)
    - ENVI (header w/ raw image, 8, 16, 32bit, limit?)
    - 100+ others

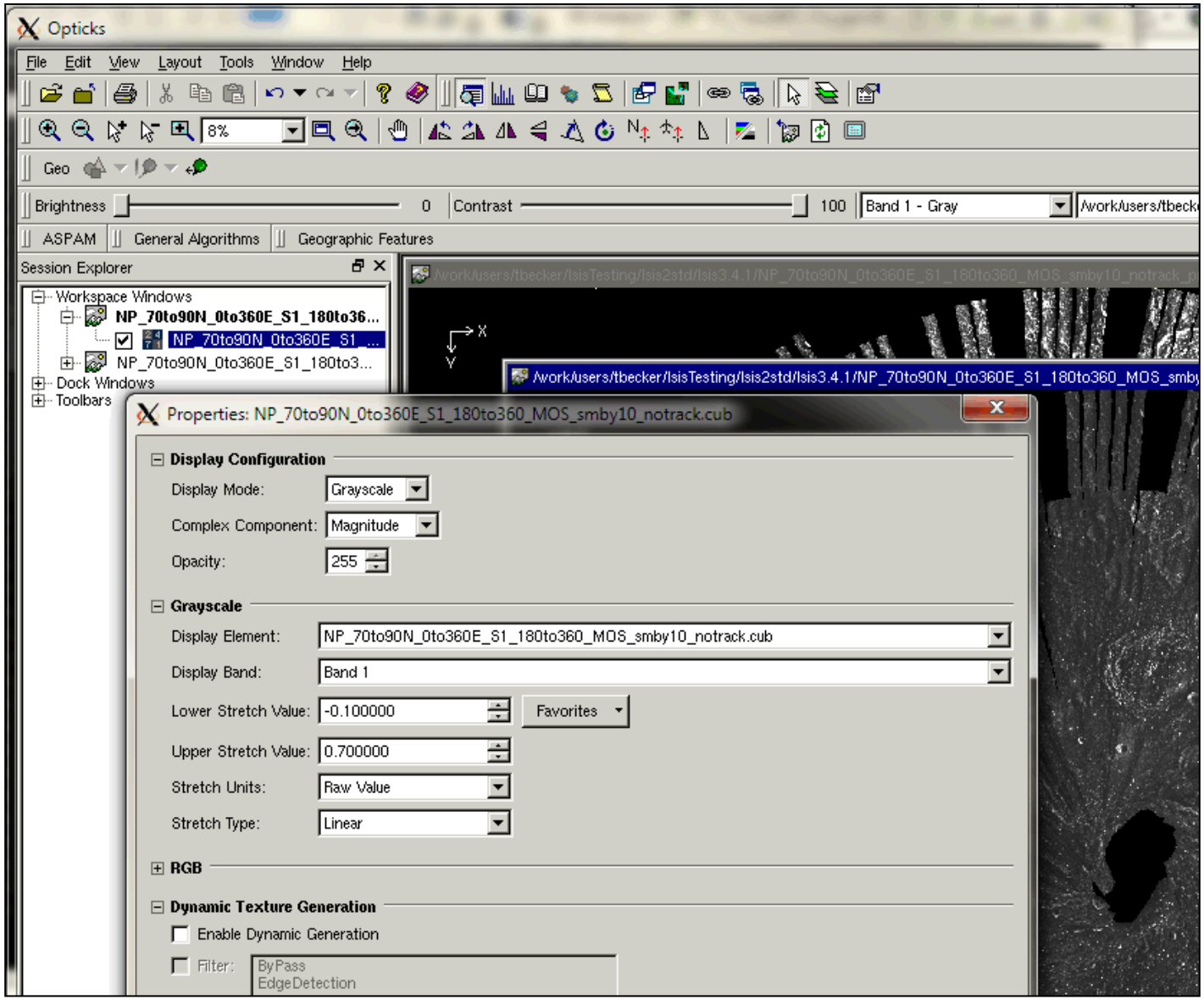
RULES of the GIS ROAD

- For ISIS processing
  - Best to set **same** projection and parameters for all
    - Note: optional to set same resolution
  - For visual (thematic) images, best to convert to 8bit
  - For "data", set all ISIS Special Pixel Values to NULL
    - e.g. DEM, Temperature -- 16,32 bit
      - Use ISIS3 applications such as: ***specpix***, ***stretch***, ***bit2bit***
  - For global
    - If LongitudeDomain=360, then set clon=180
    - If LongitudeDomain=180, then set clon=0 *(better supported)*
  - Don't use *funky* projections (e.g. Oblique Cylindrical-Cassini RADAR)

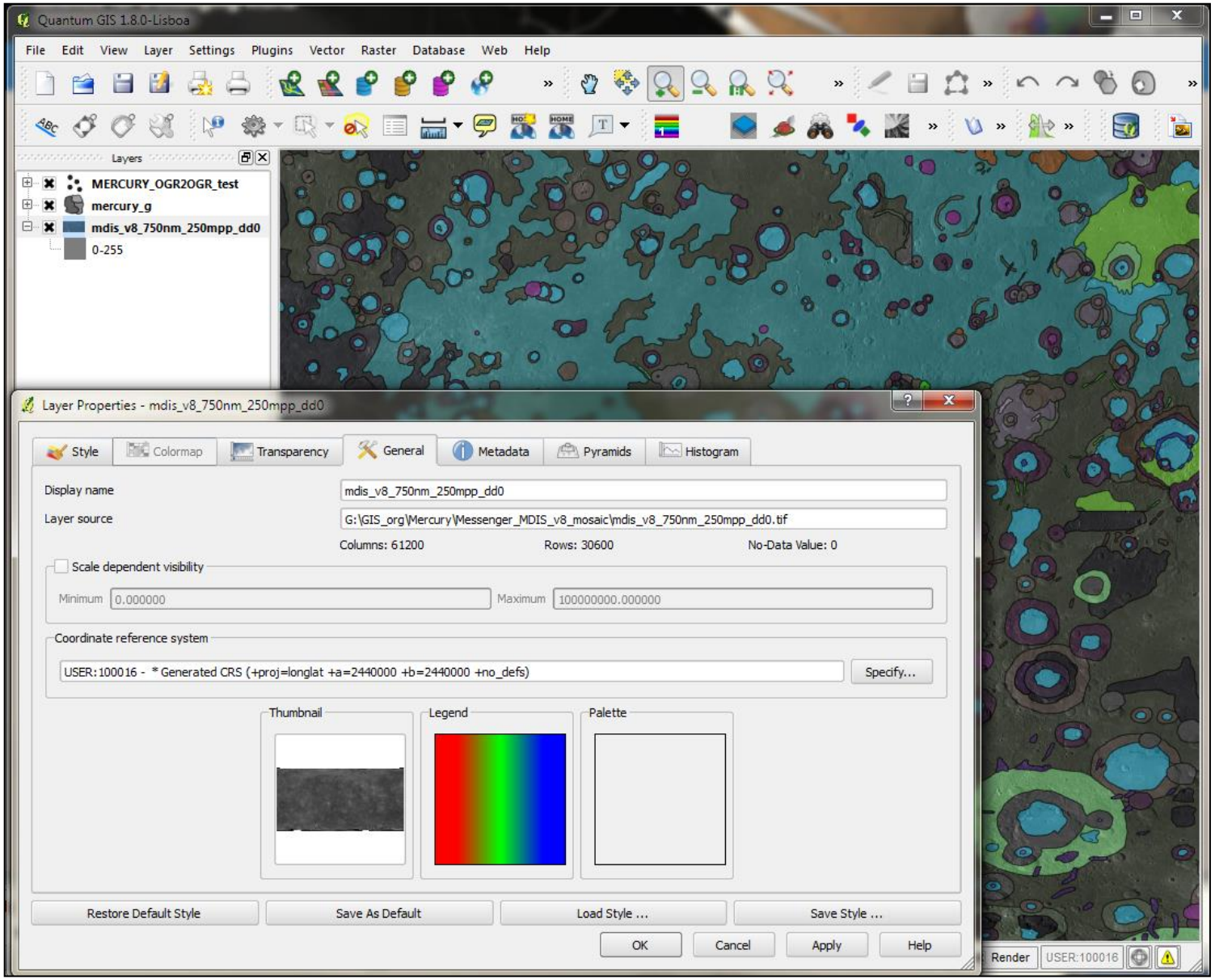
Direct Support Examples - ArcMap



Direct Support Examples - Opticks



Direct Support Examples - QGIS



GDAL for 32bit Map Projected ISIS3 (and PDS)

GDAL (binaries available using Fwtools, OSGeo4W, Mac - Kyng Chaos):

> gdal_translate –of GTIFF isis_ver3.cub isis_ver3.tif

GDAL install and conversion tips:

http://bit.ly/pprlMK

No Need for 16, 32 bit – Convert to 8bit

- ISIS3 – The "***stretch***" application seems to be the best
  - After run, import ISIS cub into ArcMap

Stretch method 1

>stretch from=input.cub to=output_8bit.cub+8bit+1:254 **USEPERCENTAGES=true**
pairs="0:1 100:254" null=0 lis=0 lrs=0 his=255 hrs=255

This allows you to specify input percentages for the mapping pairs.

Thus when USEPERCENTAGES=true is set pairs="0:1 100:254" means:
"map 0% to 1 (or the file's min value to 1) and 100% to 254 (file's max value)."

Stretch method 2

This also means you can apply a recommended 0.5% clip to remove the potential extraneous lows and highs like:

> stretch from=input.cub to=output_8bit.cub+8bit+1:254 USEPERCENTAGES=true
pairs="0:1 0.5:1 99.5:254 100:254" null=0 lis=0 lrs=0 his=255 hrs=255

No Need for 16, 32 bit – Convert to 8bit

- GDAL method

> gdalinfo –mm in.cub (returns ***min/max***, now convert)

> gdal_translate –ot byte –scale ***min max*** 1 255 –a_nodata 0 in.cub out.tif

> gdal_translate –of PNG –ot byte –scale ***min max*** 1 255 –a_nodata 0 in.cub out.png

> gdal_translate –of JP2KAK –co quality=100 –ot byte –scale ***min max***
1 255 –a_nodata 0 in.cub out.jp2

- Cshell Script helper: http://bit.ly/oxIsQ7

NumPy (Numerical Python)

For custom raster processing you can use GDAL's read support for ISIS 2, and ISIS3 formats and Python's numpy package.

NumPy:

- An array/matrix package for Python
- Well suited for image processing – i.e. one function can operate on the entire array
- Slicing by dimensions and applying functions to these slices is concise and straightforward
- Nearly 400 methods defined for use with NumPy arrays (e.g. type conversions, mathematical, logical, etc.)

From: Using Python, GDAL and NumPy for spatial analysis and modeling Outline