Mars Science Laboratory (MSL)

Software Interface Specification

Interface Title: **Camera & LIBS Experiment Data Record (EDR) and Reduced Data Record (RDR) Data Products**

Mission: MSL   Date: July 18, 2018

Module ID: SIS-SCI006-MSL

Module Type (REFerence Only or MISsion-specific info included): MIS

Reference Module ID: N/A   Date: N/A

Signatures

GDS Generating Elements:

Ops Product Generation Subsystem (OPGS)
Amy Culver

____
Subsystem Lead   Date

Science Receiving Elements:

MSL Deputy Project Scientist
Joy Crisp

____
Deputy Project Scientist   Date

GDS Receiving Elements:

MSL GDS System Engineer
Alex Cervantes

____
System Engineer   Date

MSL GDS IPE System Engineer
Alicia Allbaugh

____
Subsystem Lead   Date

Concurrence:

MSL Imaging Investigation Scientist
Justin Maki

____
Investigation Scientist   Date

MSL ChemCam Investigation Scientist
Diana Blaney

____
Investigation Scientist   Date

ChemCam Instrument Lead
Dot DeLapp

____
Instrument Lead   Date

PDS Program Manager
Tom Morgan

____
Manager   Date

PDS Geosciences Node Manager
Ray Arvidson

____
Manager   Date

PDS Discipline Node Manager
Sue Lavoie

____
Manager   Date

Mars Science Laboratory Project

Software Interface Specification (SIS)

Camera & LIBS
Experiment Data Record (EDR) and
Reduced Data Record (RDR)
Data Products

Version 4.1

Authors:
Doug Alexander, Robert Deen

Custodian:
Eugenie Song

JPL D-38107

July 18, 2018
Jet Propulsion Laboratory
California Institute of Technology

CHANGE LOG

DATE	SECTIONS CHANGED	CHANGE TYPE	REVISION
8/14/13	<u>Section 6.1.1</u> • Added type "PRC" to list of RDR products for filename field "prodid". <u>Appendix A</u> • Added keyword SUN_VIEW_DIRECTION. <u>Appendix F</u> • Added description for keyword SUN_VIEW_DIRECTION.	Update	Version 3.2
9/14/13	<u>Section 4.2.4.2</u> • Added text updating NavMap EDR description. <u>Section 5.2.1.9</u> • Added text clarifying the meaning of DN "0" in Arm Reachability products. <u>Section 5.4</u> • Added text introducing concept of "shared" PDS label for four Mosaic files, with reference to new Appendix example. <u>Appendix B</u> • Added this Appendix as example Mosaic RDR detached PDS label.	Update	Version 3.3
3/2/14	<u>Sections 5.2.1.5 & 5.3.1 & 6.1.1</u> • Added "RNM" and as 3-char RDR product type.	Update	Version 3.4
8/13/14	<u>Section 1.3</u> • Added "MER ICER User Guide" as reference. <u>Sections 5.2.1.5 & 5.3.1 & 6.1.1</u> • Added "ZZO" and as 3-char RDR product type. <u>Section 5.2.1.4.1</u> • Added this new section to describe Rover Mask RDR. <u>Appendix A</u> • Updated columns for Mosaics to be "p" for keyword PRODUCT_VERSION_ID to reflect "only present in detached PDS labels". • Added Object "CCAM_SOH_EDR_ANCILLARY" to ChemCam EDR label. • Added STEREO_BASELINE as Ops keyword. <u>Appendix D</u> • Updated "CCAM_SOH_ANCIALLARY" FMT file. • Added "CCAM_SOH_EDR_ANCIALLARY" FMT file. <u>Appendix F</u> • Updated definition for keyword PRODUCT_VERSION_ID. • Updated definition for keywords INST_CMPRS_SEGMENT_QUALITY and INST_CMPRS_SEGMENT_STATUS. • Added keyword STEREO_BASELINE. <u>Appendix G</u> • Added keyword STEREO_BASELINE.	Update	Version 3.5

DATE	SECTIONS CHANGED	CHANGE TYPE	REVISION
5/31/17	Front Matter • Custodians Doug Alexander and Robert Deen now listed as Authors. Custodian is now Eugenie Song. Table 1.3.1 • Added "marsortho", "mslreach", "mslrough" to Mars Program Suite list. Section 5.2.1.13.11 • Added "ChemCam Finder Mosaic RDR" description ("MCE", "MCR", "MCF"). Section 5.2.1.13.12 • Added "Mosaic-to-Image Coregistration Map RDR" description. ("IDX", "ICM") Section 5.2.1.15 • Added "AEGIS RDR" description. ("AGS", "AGM") Section 5.2.1.16 • Added "Terrain Classifier RDR" description. ("TEN", "TER") Tables 5.3.1.1, 5.3.1.2, 6.1.1.1, 6.1.1.2 • Updated with details on new product types. Section 6.4.2 • Corrected translation factors for converting from RMECH to RNAV frames. Section 6.1.3 • Corrected description for "*ver*" Version Identifier for Unified Terrain Mesh RDR filenames. Appendix A • Added "Target Map (AGS)" and "Misc. RDR" categories, added new keywords. Appendix C • Updated example labels for LIBS RDR, CCS, and MOC (concatenated version) products. Appendix D • Added "CCAM_NAF_SCIDATA_COLS" FMT Appendix F • Added new keyword definitions to Group DERIVED_IMAGE_PARMS: o CLASS_DESCRIPTION o CLASSIFIER_BAND_ID o CLASSIFIER_VERSION o MAX_RANGE o PROJECTION_SCALE o REGION_COUNT • Updated definition for keywords START_AZIMUTH and STOP_AZIMUTH. • Updated keyword DERIVED_IMAGE_TYPE list of Valid Values with new product types. • Updated Input Source Notes for MMM z-stack and rangemap products for keywords SPACECRAFT_CLOCK_START_COUNT & SPACECRAFT_CLOCK_STOP_COUNT. o Also affects keywords LOCAL_TRUE_SOLAR_TIME,	Update	Version 4.0

DATE	SECTIONS CHANGED	CHANGE TYPE	REVISION
	PLANET_DAY_NUMBER, SOLAR_AZIMUTH, SOLAR_ELEVATION, SOLAR_LONGITUDE, START_TIME, STOP_TIME. Appendix G • Removed keywords that are included in PDS labels (only applicable for CCAM data products). • Added new keywords that are omitted from PDS labels. Various format/design updates and spelling/grammar corrections.		
7/18/18	Table 2.1.1 • Corrected Hazcam Field of View from 124x124 to 126x126 degrees. • Corrected Front Hazcam Baseline Stereo Separation from 16 to 16.6 cm. Section 5.2.1.2.2 • Updated description of Navcam Flatfield ground correction for MIPLRAD. Section 5.2.1.4-6, 5.2.1.11 • Corrected several references to MER RDR product names to use MSL naming convention: o "XYA" to "XYF" o "RNA" to "RNF" o "UVA" to "UVF" o "IEA" to "IEF" Section 5.2.1.16 • Updated description of Terrain Classifier RDR keywords ("TEN", "TER"). Section 5.2.1.17 • Added "ChemCam Range" and "ChemCam Exclusion Mask" RDR descriptions. ("CXR", "CXM"). Tables 5.3.1.1, 6.1.1.2 • Updated with details on new product types. Appendix A • Updated Terrain Classifier keywords under DERIVED_IMAGE_PARMS. Appendix F • Added new keywords: o ARTICULATION_DEV_POSITION o ARTICULATION_DEV_POSITION_NAME o IMAGE_BLENDING_FLAG o IMAGE_REGISTRATION_FLAG o INTERPOLATION_METHOD o INTERPOLATION_VALUE o LINEARIZATION_PRODUCT_ID • Updated keyword names and definitions for Terrain Classifier products ("TEN", "TER") in keyword Group DERIVED_IMAGE_PARMS: o CLASSIFIER_BAND_HUE o CLASSIFIER_BAND_INDEX o CLASSIFIER_BAND_INDEX_NAME o CLASSIFIER_LABEL_HUE	Update	Version 4.1

DATE	SECTIONS CHANGED	CHANGE TYPE	REVISION
	○ CLASSIFIER_LABEL_INDEX ○ CLASSIFIER_LABEL_INDEX_NAME ○ CLASSIFIER_MAX_RANGE ○ CLASSIFIER_PROJECTION_SCALE ○ CLASSIFIER_VERSION • Updated keyword definitions and/or valid values for: ○ BAYER_MODE ○ COMMUNICATION_SESSION_ID ○ DETECTOR_TO_IMAGE_ROTATION ○ FILTER_NAME ○ FILTER_NUMBER ○ FLAT_FIELD_FILE_NAME ○ FLAT_FIELD_FILE_DESC ○ EXPECTED_TRANSMISSION_PATH ○ LINEARIZATION_MODE ○ PRODUCT_COMPLETION_STATUS ○ STEREO_PRODUCT_ID • Updated Keyword Location in Label values for the following keywords to include exceptions for MAHLI Z-stack and Range Map products. ○ AUTO_EXPOSURE_DATA_CUT ○ AUTO_EXPOSURE_PERCENT ○ AUTO_EXPOSURE_PIXL_FRACTION ○ EXPOSURE_COUNT ○ EXPOSURE_DURATION_COUNT ○ EXPOSURE_TYPE ○ FILTER_NUMBER ○ FIRST_LINE ○ FIRST_LINE_SAMPLE ○ INSTRUMENT_FOCUS_MODE ○ INSTRUMENT_FOCUS_POSITION ○ INSTRUMENT_FOCUS_STEP_SIZE ○ INSTRUMENT_SERIAL_NUMBER ○ INST_CMPRS_MODE ○ INST_CMPRS_QUALITY ○ LINE_SAMPLES ○ LINES ○ MAX_AUTO_EXPOS_ITERATION_COUNT ○ SAMPLE_BIT_MODE_ID ○ START_IMAGE_ID		

OPEN ISSUE ITEMS

REVISION	OPEN ISSUE	CLOSED

TABLE OF CONTENTS

APPENDICES

LIST OF FIGURES

LIST OF TABLES

ACRONYMS AND ABBREVIATIONS

AEGIS	Autonomous Exploration for Gathering Increased Science
APID	Application Process Identifier
APSS	Activity Planning and Sequencing Subsystem
ASCII	American Standard Code for Information Interchange
ATLO	Assembly, Test, Launch and Operations
CAHV	Center, Axis, Horizontal, Vertical (camera model)
CAHVOR	Center, Axis, Horizontal, Vertical, Optical, Radial (camera model)
CAHVORE	Center, Axis, Horizontal, Vertical, Optical, Radial, Entrance (camera model)
CCD	Charged Coupled Device
CCBU	Chemistry Camera Body Unit
CCMU	Chemistry Camera Mast Unit
ChemCam	Chemistry Camera
CHIMRA	Collection and Handling for Interior Martian Rock Analysis
CNES	Centre National d'Etudes spatiales (French Space Agency)
CODMAC	Committee on Data Management and Computation
CSV	Comma-separated-value
DEA	Digital Electronics Assembly
DEM	Digital Elevation Map
DN	Digital Number
DOY	Day of Year
DP	Data Product (telemetry)
DPO	Data Product Object
DRT	Dust Removal Tool
DTE	Direct to Earth
DVT	Data Validity Time
EDL	Entry, Descent and Landing
EDR	Experiment Data Record
EHA	Engineering, Housekeeping & Accountability (EH&A)
EM	Engineering Model
EMD	Earth Metadata file (".emd")
EPDU	End-of-Product PDU
ERT	Earth Received Time
FDD	Functional Design Document
FEI	File Exchange Interface
FGICD	Flight-Ground ICD
FM	Flight Model
FOV	Field of View

FPGA	Field Programmable Gate Array
FSW	Flight Software
GSFC	Goddard Space Flight Center
GDS	Ground Data System
GSE	Ground Support Equipment
Hazcam	Hazard Avoidance Camera
HGA	High Gain Antenna
IC	Inlets Cover
ICD	Interface Control Document
ICER	Image compression algorithm (not an acronym)
ID	Identification
IDPH	Image Data Product Header
IFOV	Instantaneous Field of View
ILUT	Inverse Lookup Table
IPE	Integrated Planning and Execution (MS element)
IRAP	Institut de Recherche en Astrophysique et Planétologie
ISIS	Integrated Software for Imagers and Spectrometers
IVP	Inertial Vector Propagation
JPEG	Joint Photographic Experts Group (compression)
JPL	Jet Propulsion Laboratory
LANL	Los Alamos National Laboratory
LIBS	Laser-Induced Breakdown Spectrometer (ChemCam)
LOCO	LOw-COmplexity, LOssless COmpression
MAHLI	Mars Hand Lens Imager (MSSS)
MARDI	Mars Descent Imager (MSSS)
Mastcam	Mast Camera (MSSS)
MER	Mars Exploration Rover
MIPL	Multimission Instrument Processing Laboratory
MMM	Mastcam, MAHLI, MARDI (MSSS cameras)
MOS	Mission Operations System
MPCS	Mission data Processing and Control Subsystem
MPDU	Metadata Protocol Data Unit
MPF	Mars Pathfinder
MS	Mission System
MSL	Mars Science Laboratory
MSLICE	MSL Operations InterfaCE
MSSS	Malin Space Science Systems
NASA	National Aeronautics and Space Administration
Navcam	Navigation Camera

ODL	Object Description Language
ODS	Operations Data Store
OPGS	Operations Product Generation Subsystem
PDS	Planetary Data System
PDU	Protocol Data Unit
PPDU	Product Data Protocol Data Unit
PRT	Platinum Resistance Thermometer
PSDD	Planetary Science Data Dictionary
RA	Robotic Arm
RCE	Rover Compute Element
RDR	Reduced Data Record
RMI	Remote Micro-Imager (ChemCam)
ROI	Region of Interest
RSM	Remote Sensing Mast
RTO	Real Time Operations (MS element)
RSVP	Rover Sequencing and Visualization Program
SAPP	Surface Attitude, Positioning and Pointing
SCID	Spacecraft ID
SCLK	Spacecraft Clock
SCM	Spacecraft Configuration Manager
SFDU	Standard Format Data Unit
SIS	Software Interface Specification
SOH	State of Health (ChemCam)
SOL	Mars Solar Day
SOWG	Science Operations Working Group
SPaH	Sample Processing and Handling
SPICE	Spacecraft, Planet, Instrument, C-matrix, Events kernels
SRAM	Static Random Access Memory
SwRI	Southwest Research Institute
TBD	To Be Determined
TDS	Telemetry Delivery Subsystem
UDR	Unprocessed Data Record
USGS	United States Geological Survey
VCID	Virtual Channel Identifier
VICAR	Video Image Communication and Retrieval
WEB	Warm Electronic Box

1. INTRODUCTION

1.1 Purpose and Scope

The purpose of this Data Product Software Interface Specification (SIS) is to provide consumers of MSL instrument Experiment Data Record (EDR) and Reduced Data Record (RDR) data products with a detailed description of the products and how they are generated, including data sources and destinations. Content in this document supports EDR and RDR data products generated by the Operations Product Generation Subsystem (OPGS) for the following instruments:

- Engineering Camera instrument suite:
    a. Navigation Cameras (Navcams)
    b. Hazard Avoidance Cameras (Hazcams)

- Chemistry Camera (ChemCam) instrument suite:
    a. Remote Micro-imager (RMI) camera
    b. Laser-Induced Breakdown Spectrometer (LIBS)

  Note: The non-imaging LIBS instrument is included in this SIS because ChemCam science activities intertwine RMI imaging with LIBS spectra acquisition in the same general timeline. RMI imaging provides geomorphological context to the LIBS data. As a result, a portion of each instrument's data product metadata is common in context with the other, and it is convenient to describe all here in a single document.

  Note: The ChemCam EDR data processing also includes extraction of specific state-of-health (SOH) metadata into a separate EDR.

- Malin Space Science Systems (MSSS) instrument suite:
    a. Mast Camera (Mastcam)
    b. Mars Hand Lens Imager (MAHLI)
    c. Mars Descent Imager (MARDI)

  Note: For convenience, the above instruments are often referred to in triplet as "MMM".

The users for whom this SIS is intended include OPGS, the Activity Planning and Sequencing Subsystem (APSS), users and developers of Science Operations Analysis Software (SOAS), member scientists of the project's Science Operations Working Group (SOWG) who will analyze the data, and other scientists in the general planetary science community.

In this document, the EDR data product is the raw, uncalibrated, uncorrected image data acquired by the MSL instrument. It may include decompression if there was data product compression performed onboard the rover by the instrument. Within the group of camera instruments, the full frame image EDR data products are identical in format, except for some product label differences. The LIBS spectrum EDR is of a different format than the image EDR, with a portion of the product label identical in label items to the image product.

The RDR data products described in this document are limited to camera instruments only, and are derived directly from one or more image EDR or image RDR data product(s). They are comprised of radiometrically calibrated and/or camera model corrected and/or geometrically altered (including reprojected) versions of the raw camera data.

1.2 Contents

This Data Product SIS describes how the EDR data product is acquired by the MSL instrument and how it is processed, formatted, labeled, and uniquely identified, and how the image RDR data product is derived from image EDR or image RDR data products. The document discusses standards used in generating the product and software that may be used to access the product. The EDR and RDR data product structure and organization is described in sufficient detail to enable a user to read the product. Finally, examples of composite EDR/RDR labels are provided, along with the definitions of the keywords in the label.

1.3 Constraints and Applicable Documents

This SIS is meant to be consistent with the contract negotiated between the MSL Project and the MSL Principal Investigators (PI) for the Engineering cameras and ChemCam instrument suite in which reduced data records and documentation are explicitly defined as deliverable products. By agreement with the MMM instrument PI, products generated by OPGS from MMM data processing will be deliverable to the Project only in a backup capacity and will not be archived to the Planetary Data System (PDS). Because this SIS governs the specification of data products used during mission operations, any proposed changes to this SIS must be impacted by all affected software subsystems observing this SIS in support of operations (e.g., APSS, OPGS, SOAS).

Product label keywords may be added to future revisions of this SIS. Therefore, it is recommended that software designed to process EDRs and RDRs specified by this SIS should be robust to (new) unrecognized keywords.

This Data Product SIS is responsive to the following MSL documents:

1. Pointing, Positioning, Phasing & Coordinate Systems (3PCS), “Volume 1”, Santi Udomkesmalee, MSL-376-1297, JPL D-34642, May 29, 2007.
2. Mars Science Laboratory Surface Attitude, Positioning, and Pointing Functional Design Description (FDD), Steve Peters, MSL-376-1089, JPL D-34217, December 13, 2010.
3. Mars Science Laboratory Surface Engineering Camera Imaging Functional Design Description (FDD), Justin Maki, MSL-375-1083, JPL D-34213, December 15, 2009.
4. Mars Science Laboratory Surface ChemCam Functional Design Description (FDD), “Baseline Release, Revision B”, Noah Warner, MSL-375-1231, JPL D-34221, December 6, 2010.
5. Mars Science Laboratory Surface MARDI/Mastcam/MAHLI (MMM) Functional Design Description (FDD), “Baseline Release, Rev A”, Justin Maki, MSL-375-1744, JPL D-38155, January 19, 2010.
6. Mars Science Laboratory Flight-Ground Interface Control Document (FGICD), “Volume 1, Downlink Update Release Version 2.2.1”, Sanford Krasner, JPL D-27356, MSL 232-0219, July 21, 2010.
7. Mars Science Laboratory Surface ChemCam Interface Control Document (ICD), “Revision A”, Elisabeth Morse, MSL-336-0315, JPL D-27360, January 12, 2008.
8. MSL Archive Generation, Validation, and Transfer Plan, J. Crisp, JPL D-35281, May 28, 2010.
9. MSL Real Time Operations (RTO) Element Data Management Plan, G. Smith, JPL D-65858, March 2, 2011.
10. Mars Science Laboratory PLACES User Guide, “Release 2.0, Rev-B”, Bob Deen, MSL-586-3653, JPL D-71121, August 8, 2011.

Additionally, this SIS is also consistent with the following Planetary Data System documents:

11. Planetary Science Data Dictionary Document, Version 1.81, November 24, 2010.
12. Planetary Science Data MSL Local Data Dictionary, Version 1.0, January 15, 2013.
13. Planetary Data System Archive Preparation Guide, Version 1.4, JPL D-31224, April 1, 2010.
14. Planetary Data System Data Standards Reference, JPL D-7669, Version 3.8, Part 2, February 27 2009.
15. MSL ChemCam Science Team and PDS Geosciences Node Interface Control Document (ICD), S. Slavney and D. DeLapp, Version 2.0, May 14, 2007.
16. MSL MAHLI, MARDI, Mastcam Science Team and NASA PDS Imaging Node Science Data Archiving Interface Control Document (ICD), E. Jensen, Version 1.1, September 27, 2011.
17. MSL Experiment Data Record (EDR) and Engineering Cameras Reduced Data Record (RDR) Archive Volume Software Interface Specification (SIS), R. Alanis, JPL D-64995, Version 1.0 Draft, September 6, 2011.

Finally, this SIS makes reference to the following documents for technical background information:

18. A System for Extracting Three-Dimensional Measurements from a Stereo Pair of TV Cameras, Y. Yakimovsky and R. Cunningham, January 7, 1977.
19. Camera Calibration, D. Gennery, JPL IOM 347/86/10, February 5, 1986.
20. Sensing and Perception Research for Space Telerobotics at JPL, D. Gennery et al., *Proceedings of the IEEE Intern. Conf. on Robotics and Automation,* March 31 - April 3, 1987.
21. Camera Calibration Including Lens Distortion, D. Gennery, JPL D-8580, May 31, 1991.
22. Algorithm for Using CAHV to Determine SGI Graphics Viewpoint and Perspective, B. Bon, JPL IOM 3472-91-057, August 6, 1991.
23. Inclusion of Old Internal Camera Model in New Calibration, D. Gennery, JPL IOM 386.3-94-001, February 22, 1994.
24. Least-Squares Camera Calibration Including Lens Distortion and Automatic Editing of Calibration Points, Calibration and Orientation of Cameras in Computer Vision, D. Gennery, ISBN 3-540-65283-3, 2001.
25. Computations for Generalized Camera Model Including Entrance, Part 1 and Part 2, D. Gennery, unpublished, May 23, 2001.
26. Generalized Camera Calibration Including Fish-Eye Lenses, D. Gennery, JPL D- 03-0869, 2002.
27. Issues with Linearization, R. Deen, JPL Docushare Collection 2700, File 75670, 2003.
28. Mastcam Multispectral Imaging on the Mars Science Laboratory Rover: Wavelength Coverage and Imaging Strategies at the Gale Crater Field Site, J.F. Bell III et al., *43rd Lunar and Planetary Science Conference*, 2012.
29. Mars Science Laboratory Participating Scientists Program Proposal Information Package, Anderson, R.C., et al., December 14, 2010.
30. Seeing in Three Dimensions: Correlation and Triangulation of Mars Exploration Rover Imagery, Deen, R.G. and J.J. Lorre (2005), submitted to 2005 IEEE International Conf. on Systems, Man, and Cybernetics, Waikoloa, Hawaii.
31. Mars Exploration Rover (MER) Project ICER User's Guide, Aaron Kiely, MER 420-8-0538, JPL D-22103, January 5, 2004.
32. Rover Mast Calibration, Exact Camera Pointing, and Camera Handoff for Visual Target Tracking, W.S. Kim et al., *12th International Conference on Advanced Robotics, 2005 (ICAR '05)*, July 18-20, 2005.

1.3.1 Relationships with Other Interfaces

Changes to this EDR/RDR data product SIS document affect the following products, software, and/or documents.

Table 1.3.1 - Product and Software Interfaces to this SIS

Name	Type P = product S = software D = document	Owner
MIPL database schema	P	MIPL (JPL)
MSLEDRGEN	P	MIPL (JPL)
MSL Camera and LIBS EDRs • Navcam • Hazcam • ChemCam	P	MIPL (JPL)
Navcam RDRs	P	MIPL (JPL)
Hazcam RDRs	P	MIPL (JPL)
ChemCam RDRs • RMI	S	ChemCam Team (IRAP in France)
RSVP	S	RSVP Dev Team (JPL)
MSLICE	S	MSLICE Dev Team (JPL)
Mars Program Suite • MARSCAHV • MARSRAD • MARSJPLSTEREO • MARSCOR3 • MARSXYZ • MARSUVW • MARSRANGE • MARSREACH • MARSROUGH • MARSMAP • MARSMOS • MARSMCAULEY • MARSNAV • MARSTIE • MARSINVERTER • MARSDEBAYER • MARSBRT • MARSERROR • MSLFILTER • MARSFILTER • MARSMASK • MARSDISPCOMPARE • MARSORTHO • MSLREACH • MSLROUGH • XVD • CRUMBS	S	MIPL (JPL)

2. INSTRUMENT OVERVIEW

In this section, overviews are provided for the three MSL instrument payload suites mentioned in the previous section: a) Engineering Camera instrument suite, b) ChemCam instrument suite, and c) MMM Camera instrument suite. Largely, the instruments described in this SIS are cameras, with the exception being the LIBS spectrometers in the ChemCam instrument suite. The LIBS instruments are included in this SIS based on some shared commonality in metadata with the RMI camera, a result due to the manner in which RMI images are used to document LIBS laser events.

The MSL rover instrument payload includes 17 individual cameras. The main differences between these instruments are in the optics, mounted position, and articulation methods. The cameras are monochromatic, except for the color-capable Mastcam. The engineering cameras (Navcam, Hazcam), share the identical electronics design and spacecraft interfaces. In these cases, the detectors are 1024×1024 pixel CCDs, and the electronics provide 12-bit analog-to-digital conversion. Of the 17 cameras, there are 4 sets of stereo pairs and three single cameras, as listed in Table 2a below.

Table 2a - Tabulation of MSL Cameras

MSL Camera Instrument	Location	Number
Navigation Camera (Navcam)	Stereo pair on Remote Sensing Mast (RSM)	4
“Front” Hazard Avoidance Camera (Hazcam)	Stereo pair at front of Warm Electronics Box (WEB)	4
“Rear” Hazard Avoidance Camera (Hazcam)	Stereo pair at rear of WEB	4
Remote Micro-imaging (RMI of ChemCam)	Monoscopic on RSM	1
Mast Camera (Mastcam)	Stereo pair on RSM	2
Mars Hand Lens Imager (MAHLI)	Robotic Arm	1
Mars Descent Imager (MARDI)	WEB	1
	TOTAL	17

The MSL rover instrument payload also includes the LIBS instrument, comprised of multiple lasers and spectrometers. It shares use of a telescope, boresighted on the RSM with the Mastcam and Navcam, with the RMI camera to ensure targeting the lasers on the same spots imaged by the RMI. See Table 2b below for the LIBS components.

Table 2b - Tabulation of LIBS Components

LIBS Instrument Component	Location	Number
LIBS laser	Mast Unit (RSM)	1
Continuous Wave (CW) laser	Mast Unit (RSM)	1
LIBS telescope	Mast Unit (RSM)	1
Spectrometers (Near-Infrared, Visible, Ultra-violet)	Body Unit (WEB)	3
	TOTAL	6

Camera and LIBS mounting locations are shown in Figure 2 below.

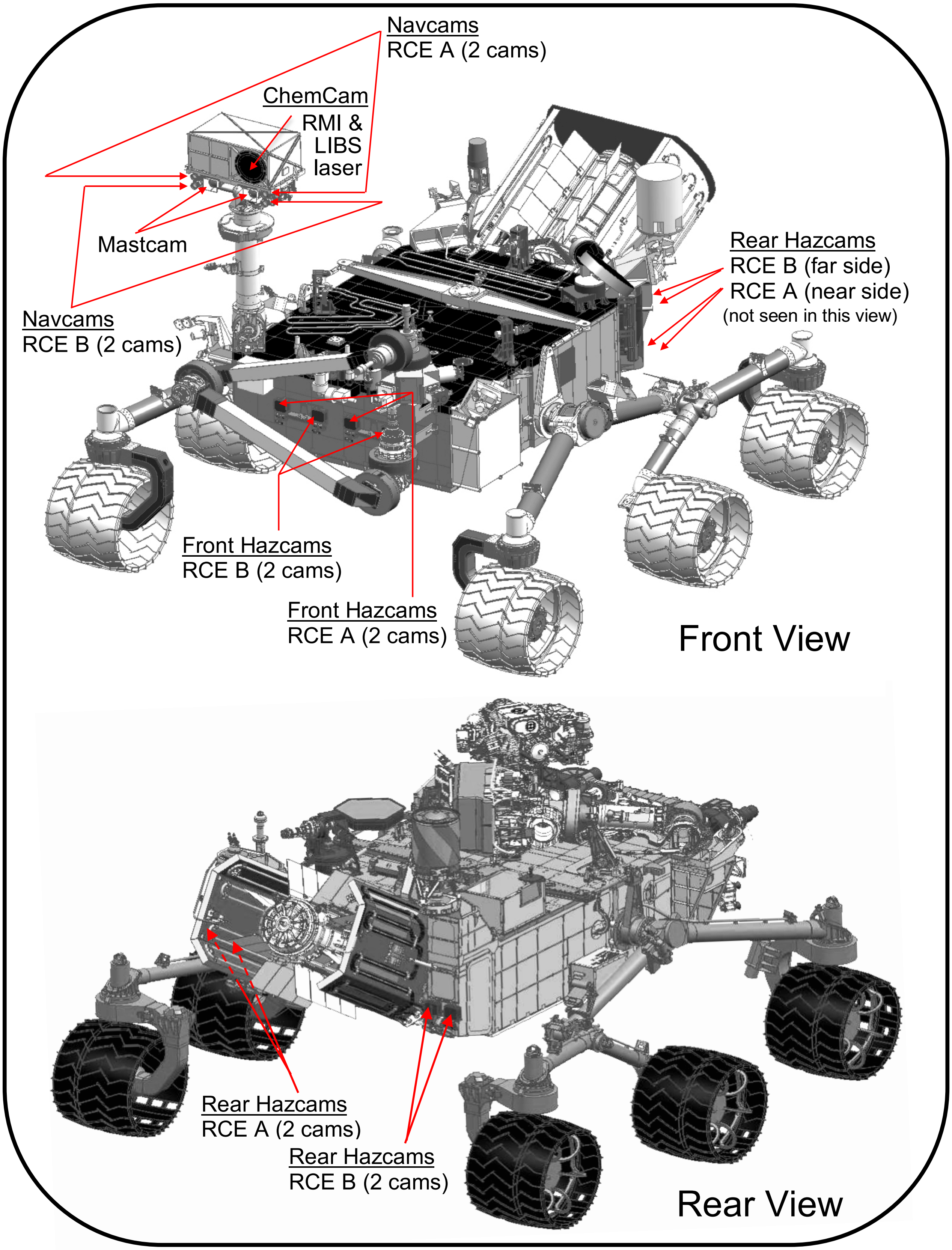


Figure 2 - MSL Camera and LIBS Locations

2.1 Engineering Camera Instrument Suite

The Engineering Camera instrument suite is comprised of the Hazcams and Navcams for the purposes of providing near-range (Hazcams) and mid-range (Navcam) fields-of-view to facilitate image acquisition important to rover traverse planning. For each camera, a total of 1024 x1024 image pixels plus 32 reference pixels per line (totaling 1024 x 1056) are each digitized to 12 bits resolution. For details about the bit processing of the scene content, refer to Section 4.4.1. The camera acquisition of the scene in hardware, beginning at origin (0,0), and subsequent onboard storage and readout of EDR image data is illustrated in Figure 2.1 below. Note that the image product label references (1,1) as the starting line and starting sample, respectively. Reference Pixels are returned separately:

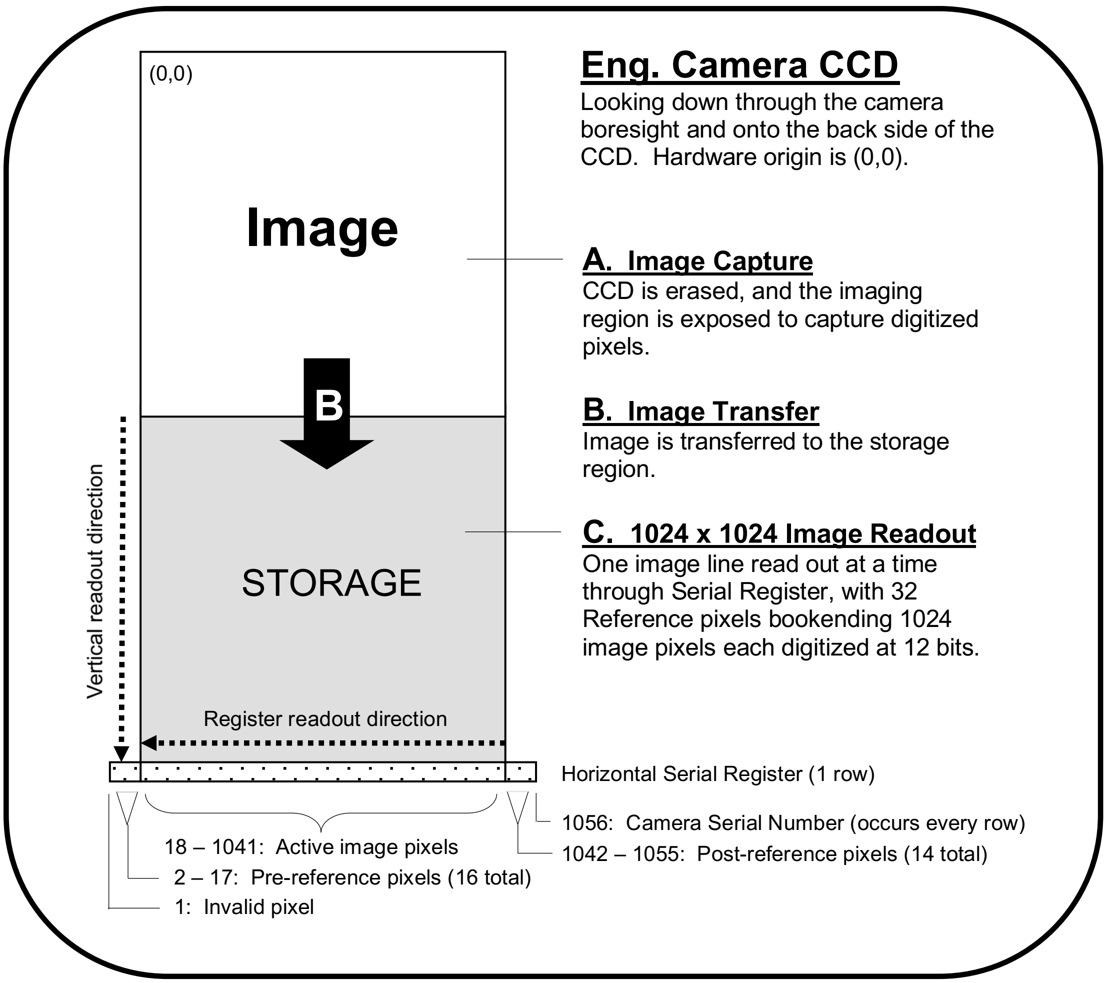


Figure 2.1 - Acquisition and Readout of Engineering Camera Image Data

2.1.1 Hazard Avoidance Camera (Hazcam)

The Hazard Avoidance Cameras (Hazcams) are two stereo pairs of engineering cameras with fish-eye lenses at 16 cm (front) and 10 cm (rear) baseline separation mounted at both the front and rear ends of the Warm Electronics Box (WEB). Hazcam assembly includes 2 cameras with a fixed Red 200 nm bandpass filter (identical to the Navcams).

The Hazcams provide imaging primarily of the near field (< 5 m) both in front of and behind the rover. These cameras will be used to determine safe driving directions for the rover and provide for on-board hazard detection using stereo data to build range maps. They also support science operations for selecting near field target and robotic arm operations.

Hazcam optics characteristics useful in the analysis of EDR and RDR products are described in Table 2.1.1, with Hazcam fields of view shown in Figure 2.1.1 below:

Table 2.1.1 - Hazcam Operational Characteristics

Characteristic	Value
Field of View (FOV)	126 x 126 deg
Baseline Stereo Separation	16.6 cm for front, 10 cm for rear
Angular Resolution	2.1 mrad/pixel at center
Spectral Bandpass	600 - 800 nm
Focal Length	5.58 mm
f/number	15
Depth of Field	0.1 m - infinity
Best Focus	0.5 m

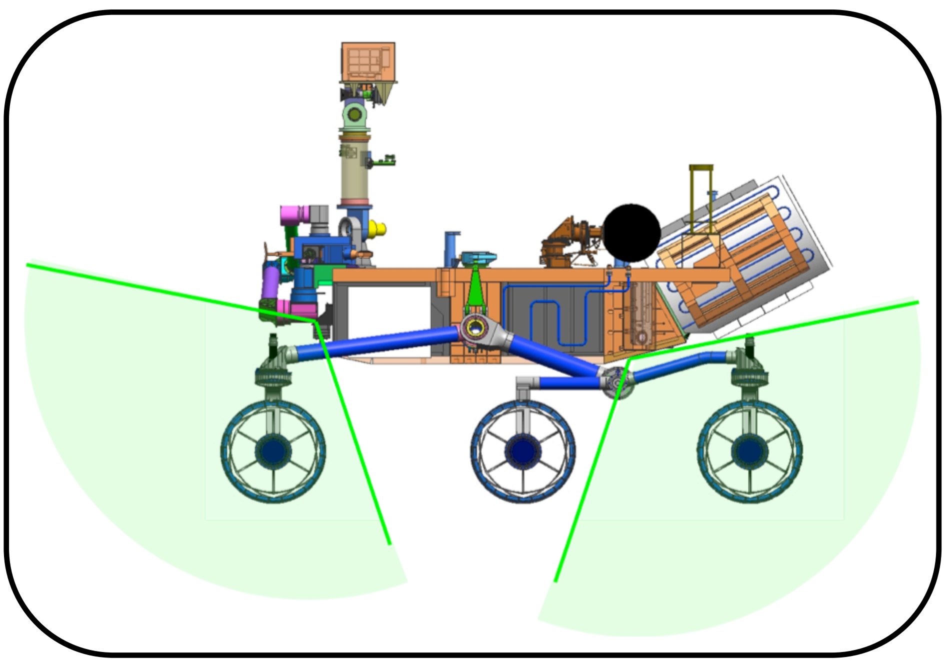


Figure 2.1.1 - Hazcam Fields of View

2.1.2 Navigation Camera (Navcam)

The Navigation Camera (Navcam) is a mast-mounted stereo pair of engineering cameras at 42.4 cm baseline separation with a spectral bandpass at approximately 200 nm. It will primarily be used for navigation purposes and general site characterization (360° panoramic images and targeted images of interest, including terrain not viewable by the Hazcams).

The cameras are boresighted with the Mastcam, and Navcam images will also be used for Science target selection and analysis.

Navcam optics characteristics useful in the analysis of EDR and RDR products are described in Table 2.1.2, with Navcam field of view shown in Figure 2.1.2 below:

Table 2.1.2 - Navcam Operational Characteristics

Characteristic	Value
Field of View (FOV)	45 x 45 deg
Baseline Stereo Separation	42.4 cm
Angular Resolution	0.82 mrad/pixel at center
Spectral Bandpass	600 - 800 nm
Focal Length	14.67 mm
f/number	12
Depth of Field	0.5 m - infinity
Best Focus	1.0 m

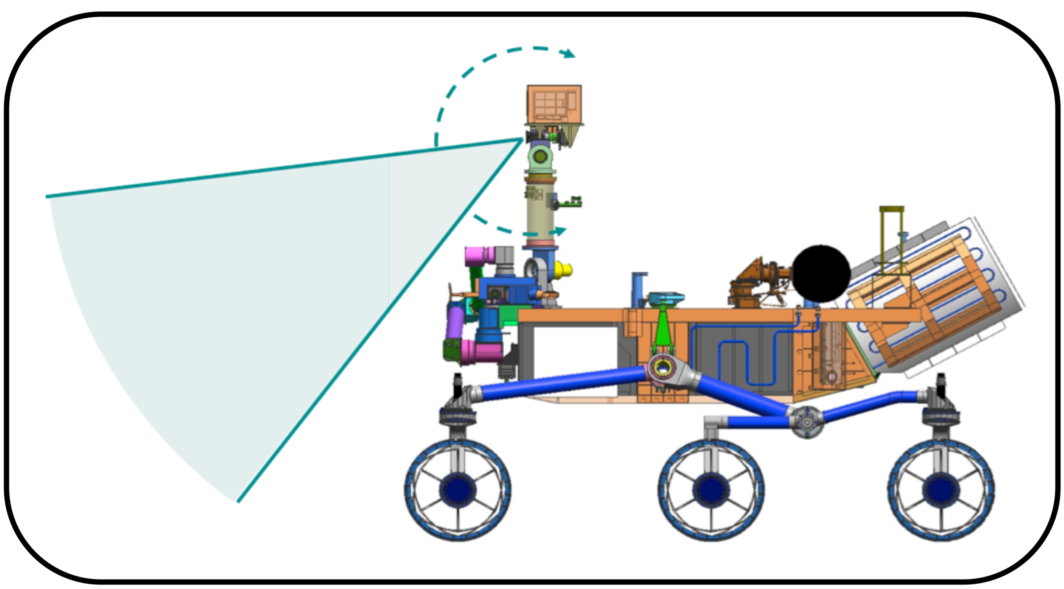


Figure 2.1.2 - Navcam Field of View

2.2 ChemCam Instrument Suite

The ChemCam instrument suite is a spectroscopy science payload comprised of the RMI camera subsystem and the LIBS laser/spectrometer subsystem. The ChemCam instrument payload is packaged onboard the rover in two instrumentation units: a) the Mast Unit (affixed to the RSM), and b), the Body Unit (housed within the rover WEB), both shown in Figure 2.2.1.

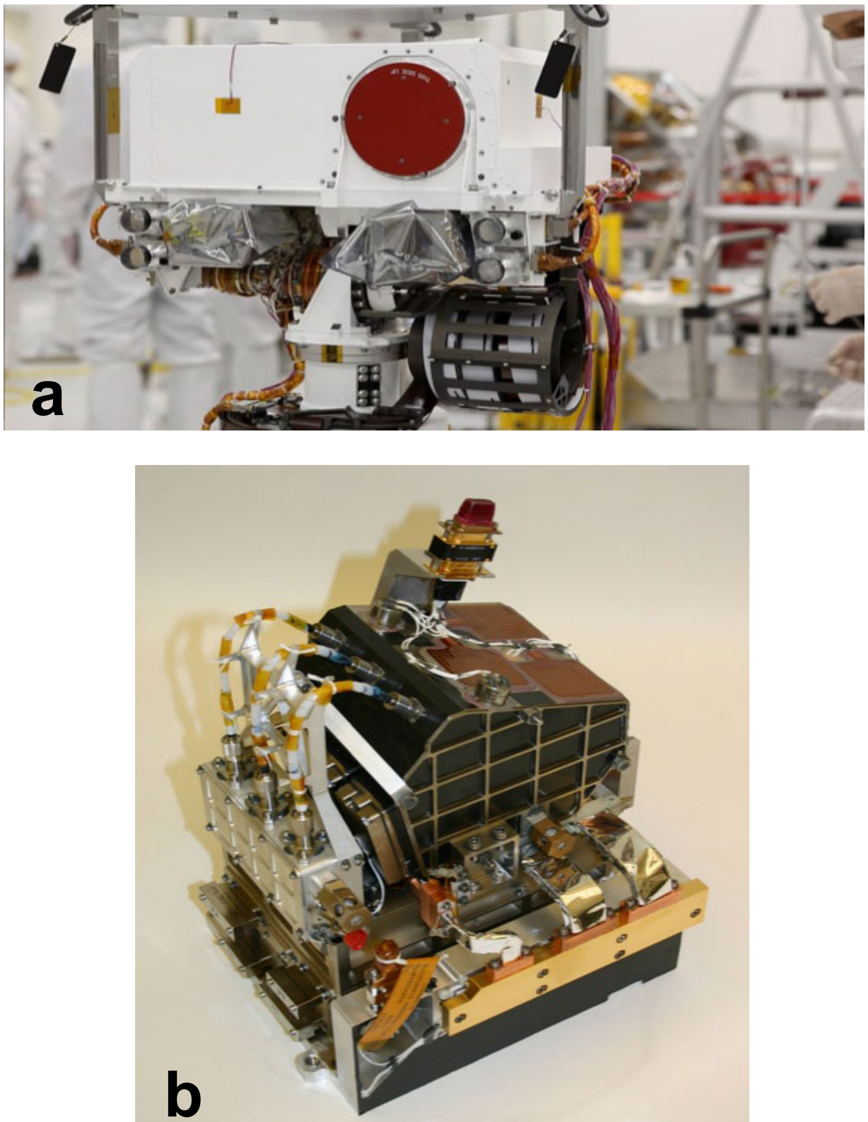


Figure 2.2.1 – Mast Unit (a) in white housing with red telescope cover, and Body Unit (b)

The RMI and lasers are contained in the Mast Unit and the spectrometers reside in the Body Unit. The LIBS will focus powerful laser pulses on targets and then determine the elemental compositions by measuring the emission lines from the ablated material in three spectral ranges. Within the Mast Unit, the design provides for the optical paths of the RMI camera and LIBS lasers to be co-aligned by using the same telescope component, enabling the focus spot for each to be registered to the same target. In this manner, the RMI data provides geomorphological context to the LIBS spectroscopy activity in image format. Refer to Figure 2.2.2 for a schematic illustration of the interfaces in a typical event.

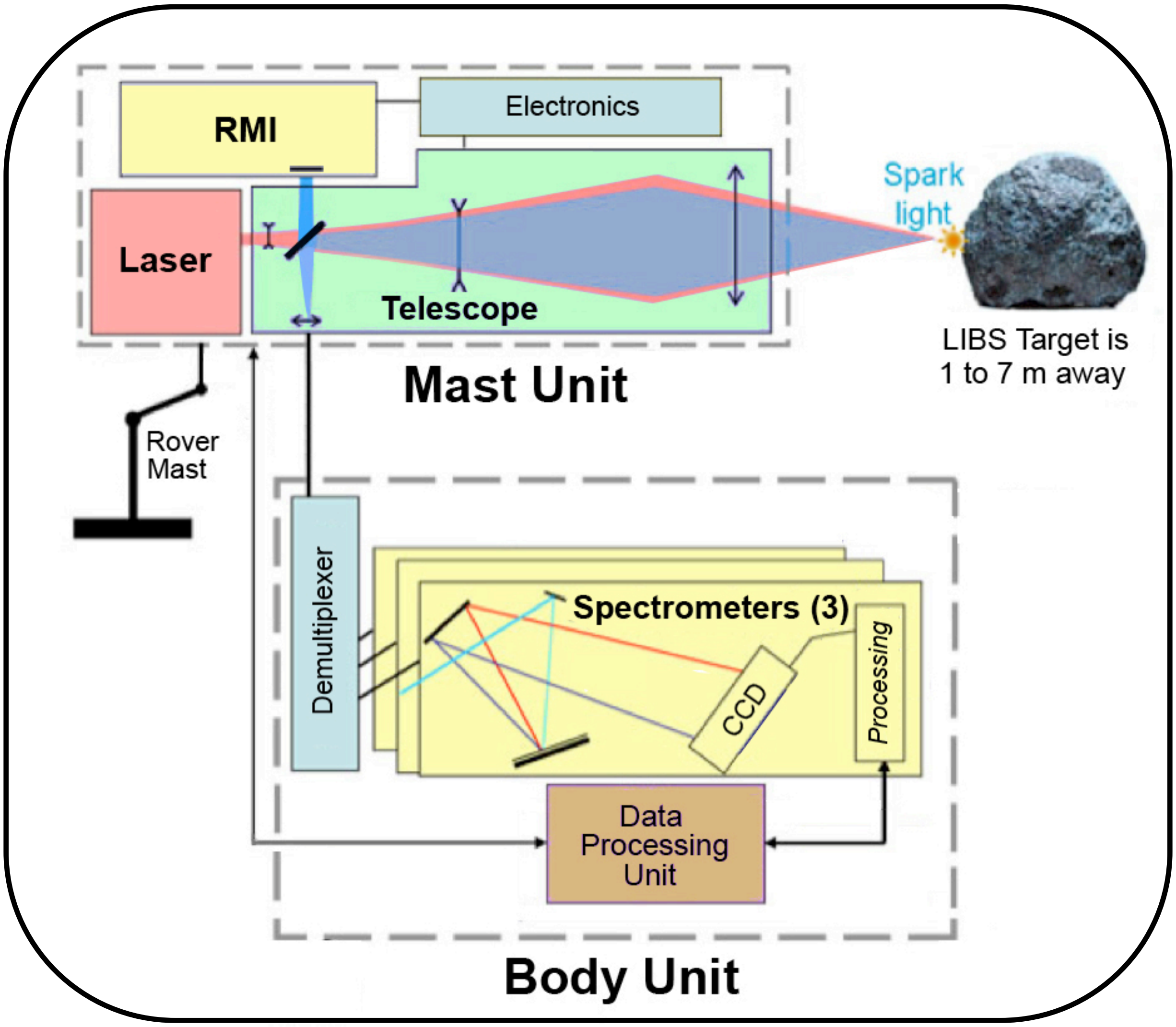


Figure 2.2.2 - Block Diagram of LIBS Spectroscopy Event

2.2.1 Remote Micro-Imager (RMI) Camera

The Remote Micro-Imager (RMI) camera provides high-resolution context images of the LIBS sampling area as well as imaging rock morphologies and distant features. It includes an adjustable focus capability. A total of 1024 x 1024 image pixels plus 16 reference pixels on the left per line and 12 reference pixels on the top and bottom (24 total) are each digitized to 10 bits/pixel resolution. Its pixel field of view is 21-22 μrad. Its effective resolution exceeds that of MER Pancam by a factor of 5 to 10. Resolution in the near-field is within a factor of 2-3 of MER MI (at closest-focus distance of 1 m for RMI versus 6 cm for MI). As with the engineering cameras, the binary data may be returned uncompressed or compressed. The Reference Pixels are returned separately. For details about the bit processing of the scene content, refer to Section 4.4.2.

RMI optics characteristics useful in the analysis of EDR and RDR products are described in Table 2.2.1.1 below:

Table 2.2.1.1 - RMI Operational Characteristics

Characteristic	Value
Field of View (FOV)	22.5 mrad
Spatial Resolution	< 1 mm @ 10 m
Angular Resolution	78-85 μrad vertical, 87-105 μrad horizontal
Spectral Bandpass	~ 250 - 900 nm
Focus Range	1.161 m to infinity
Number of Spectral Filters	1

RMI image data return compression modes are listed in Table 2.2.1.2 below:

Table 2.2.1.2 - RMI Image Compression Modes

Mode	Description	Factor
0	No compression	1
1	LOCO compression (done by RCE)	2
2	ICER compression (done by RCE)	10
3	Region-of-Interest, or ROI * (done by RCE)	64

* ROI size is variable and commandable from 8x8 pixels up to 1024x1024 pixels (128x128 is default) and the ROI can be rectangular (i.e., 256x128 pixels) if desired.

The camera acquisition of the scene and subsequent onboard storage and readout of image data is illustrated in Figure 2.2.1.1 below:

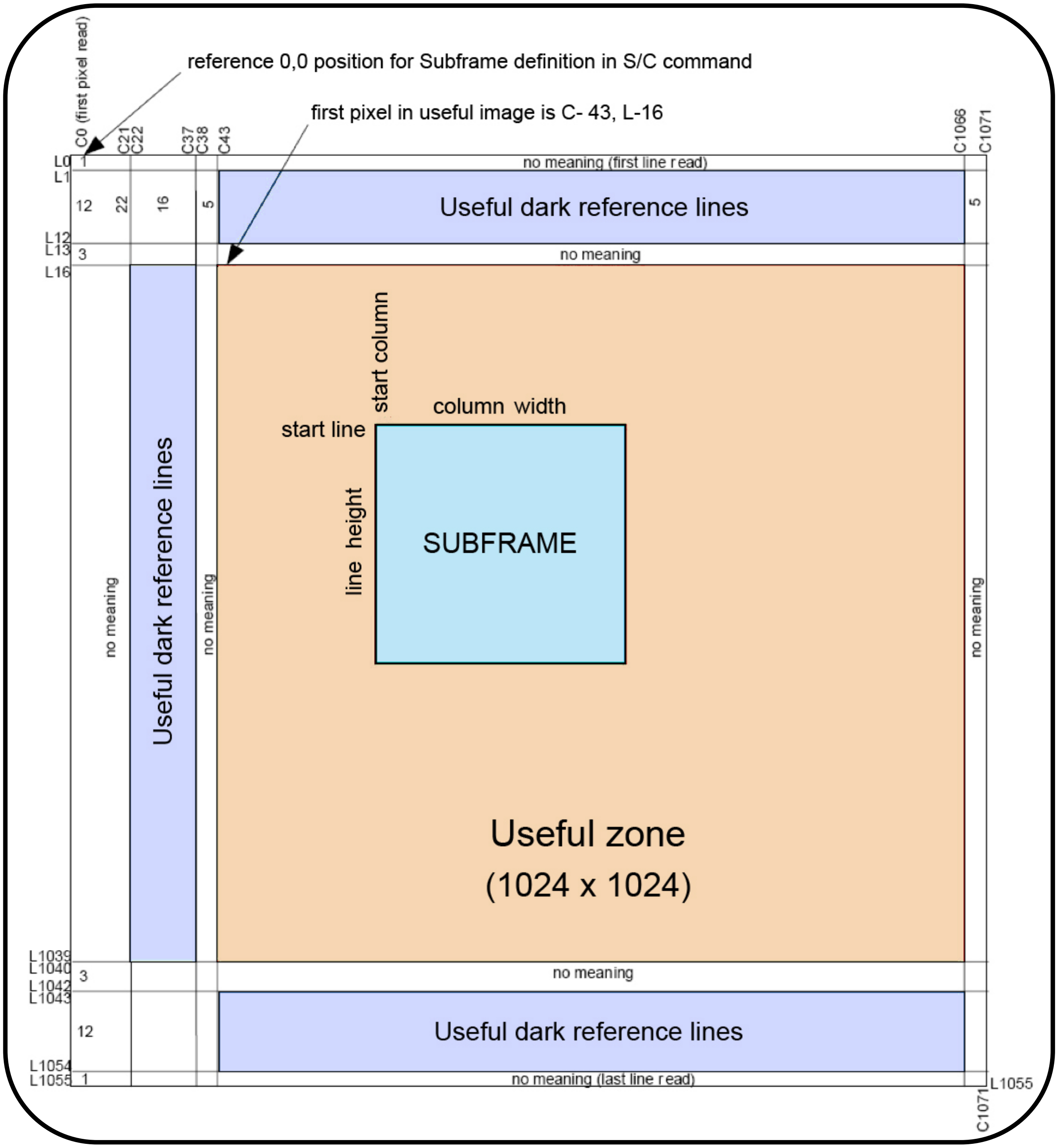


Figure 2.2.1.1 - Acquisition and Readout of RMI Camera Image Data

Generally, a subframe image in any size from 8x8 to 1024x1024 (default is 128x128, rectangular is possible on board) will be made prior to LIBS analysis, while after the laser shots a full frame image that includes the region of interest around the analyzed spot will be acquired. In addition to the 1024x1024 useful image area, the full image contains a prescan area (1024x22), a dark area (1024x16) and a reference pixel (dark collapsed to 1). Alternatively, the image may consist of a 16x16 or 32x32 thumbnail, computed on board, without the extra elements. The returned product depends on the image compression option and the size of the frame, and generally should be turned into a useful image record.

2.2.2 Laser-induced Breakdown Spectrometer (LIBS)

The ChemCam LIBS instrument payload has two lasers and three spectrometers distributed in two instrumentation units: the Mast Unit on the RSM, and the Body Unit housed in the rover WEB. The lasers reside in the Mast Unit and include a) the LIBS laser, a very high-powered beam used to ablate targeted material, and b) the Continuous Wave (CW) laser, used for the autofocus function of the LIBS instrument. The spectrometers reside in the Body Unit. Figure 2.2.2, shown previously in Section 2.2, schematically illustrates the spectroscopy of a rock target, with the active LIBS components configured in Mast and Body Units. The LIBS activity involves the lasers focusing on a small spot on target rock and soil samples within 15 m of the rover to ablate atoms and ions in electronically excited states, from which they decay, producing an illuminated plasma analyzed by the spectrometers.

Some general operational characteristics of LIBS are listed in Table 2.2.2.1 below:

Table 2.2.2.1 – LIBS Operational Characteristics

Characteristic	Value
Focus Range	≥ 1.161 m
Spectrometer Field of View (FOV)	0.65 mrad FWHM
Spectral Bandpass	240.8 – 905.5 nm
Pointing Accuracy in Mast context	± 4 mrad

The three LIBS spectrometers produce raw or averaged spectra in three different wavelength ranges. Each spectrum is coded on the center 2048 out of 2148 channels (it is also possible to return a 2D diagnostic image 2048x256). The three spectra are combined into one product. The measurement can be done after a laser shot (active spectra), without laser activation (passive spectra), or on the calibration targets. In these three cases, the same type of product or record is returned. Figure 2.2.2.1 illustrates LIBS spectra data plotted for the three spectrometer cases as a means of visualizing the spectra measurements.

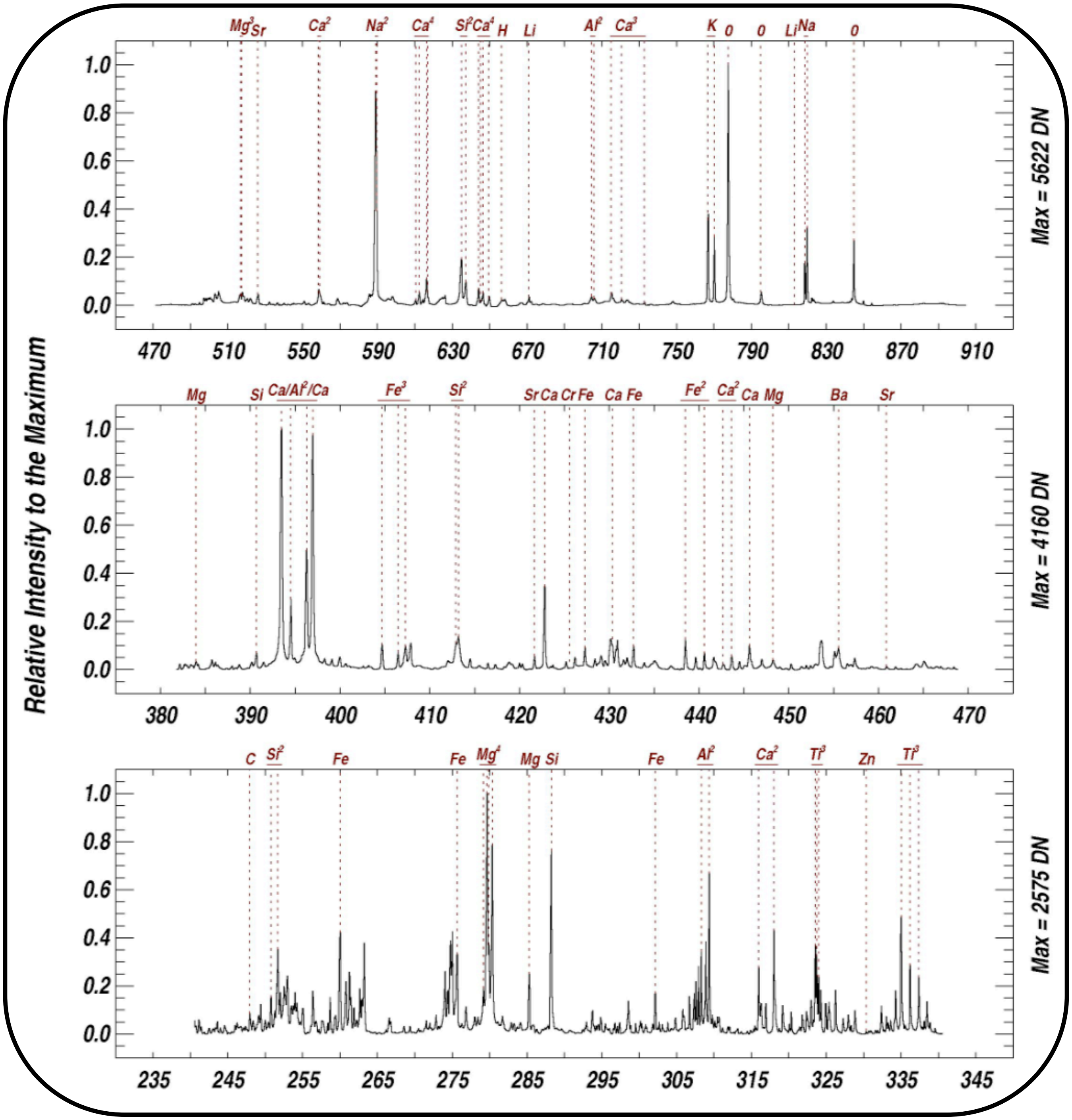


Figure 2.2.2.1 – Spectrum Data Plotted for Three LIBS Spectrometers

2.3 MMM Camera Instrument Suite

2.3.1 Mast Camera (Mastcam)

The Mast Camera (Mastcam) is comprised of a pair of color-capable focusable stereo cameras ("eyes") mounted on the rover's Remote Sensing Mast (RSM). Each camera has a different focal length and set of filters. Together, they can acquire images of up to 1600 x 1200 pixels and are capable of video. They acquire color via Bayer-pattern filters on the CCD, but also have selectable filters for science/geology applications. For more details on Bayer pattern filters, see Section 4.3.3.1.

Mastcam optics characteristics that are useful in the analysis of EDR and RDR products are described in Table 2.3.1.1 below:

Table 2.3.1.1 - Mastcam Operational Characteristics

Characteristic	Left Eye (M-34)	Right Eye (M-100)
Field of View (FOV)	15 x 15 deg	5.1 x 5.1 deg
Baseline Stereo Separation	24.5 cm	24.5 cm
Spatial Resolution	450 μrad/pixel at 2 m 22 cm/pixel at 1 km	150 μrad/pixel at 2 m 7.4 cm/pixel at 1 km
Angular Resolution	0.22 mrad/pixel	0.074 mrad/pixel
Focal Length	34 mm	100 mm
f/number	8	10
Focus Range	2.1 - infinity	2.1 - infinity
Number of Spectral Filters	8 plus Bayer pattern	8 plus Bayer pattern

Each Mastcam camera has an 8-position filter wheel. One of the positions is a broadband filter for use with the Bayer color capability of the CCD. One additional filter per eye has a neutral density coating to provide direct solar imaging capability in two colors. The remaining filters are used for science imaging.

Thirteen of the sixteen filters provide color imaging capability in eleven unique wavelengths from 400 to 1100 nm, two of the remaining filters have neutral density coatings to provide direct solar imaging capability in two colors, and one filter wheel position has been left empty to provide for maximum broadband light sensitivity. The spectral bandwidths [Ref 28] are described in Table 2.3.1.2 below:

Table 2.3.1.2 - Mastcam Spectral Filters and Bandpasses

Filter Position	Left Eye Wavelength (± Bandwidth), λ_{eff} ± HWHM (nm)	Right Eye Wavelength (± Bandwidth), λ_{eff} ± HWHM (nm)
0	590 ± 88 (Broadband)	575 ± 90 (Broadband)
	640 ± 44 (Bayer filter Red)	638 ± 44 (Bayer filter Red)
	554 ± 38 (Bayer filter Green)	551 ± 39 (Bayer filter Green)
	495 ± 37 (Bayer filter Blue)	493 ± 38 (Bayer filter Blue)
1	527 ± 7	527 ± 7
2	445 ± 10	447 ± 10
3	751 ± 10	805 ± 10

Filter Position	Left Eye Wavelength (± Bandwidth), λ_{eff} ± HWHM (nm)	Right Eye Wavelength (± Bandwidth), λ_{eff} ± HWHM (nm)
4	676 ± 10	908 ± 10
5	867 ± 10	937 ± 10
6	1012 ± 21	1013 ± 21
7	880 ± 10 ND5	440 ± 20 ND5

The Mastcam filter response profiles, including those of the RGB Bayer filters, are plotted below for the left (M-34) and right (M-100) camera eyes in two separate figures. Figure 2.3.1.2.1 [Ref 28] shows the Bayer filter profiles in red, green and blue when the filter wheel is set to Filter 0 (broadband IR cutoff shown as dashed line).

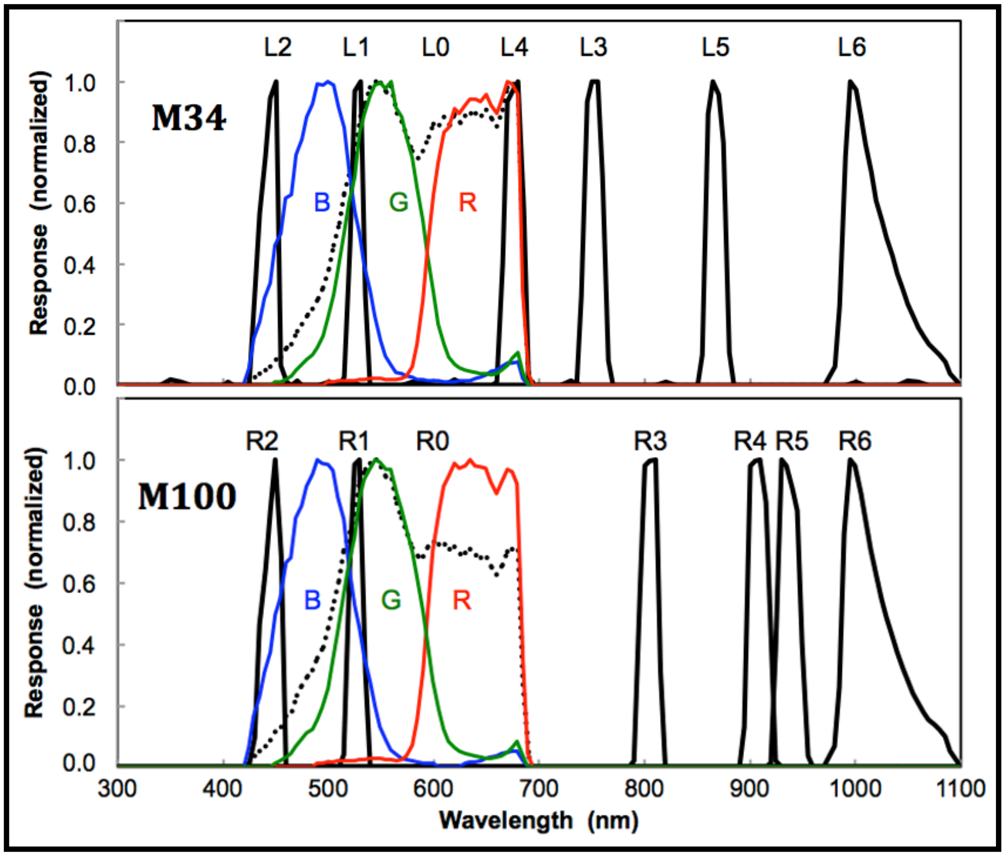


Figure 2.3.1.2.1 - Mastcam Filter Profiles, Filter Wheel Set to Filter 0

Figure 2.3.1.2.2 [Ref 29] shows the Bayer filter profiles in red, green and blue when the filter wheel is set to a nonzero number. Additionally, the quantum efficiency of the CCD detector is graphically depicted as a dashed line. The Bayer filter array and CCD quantum efficiency profiles are not shown to scale.

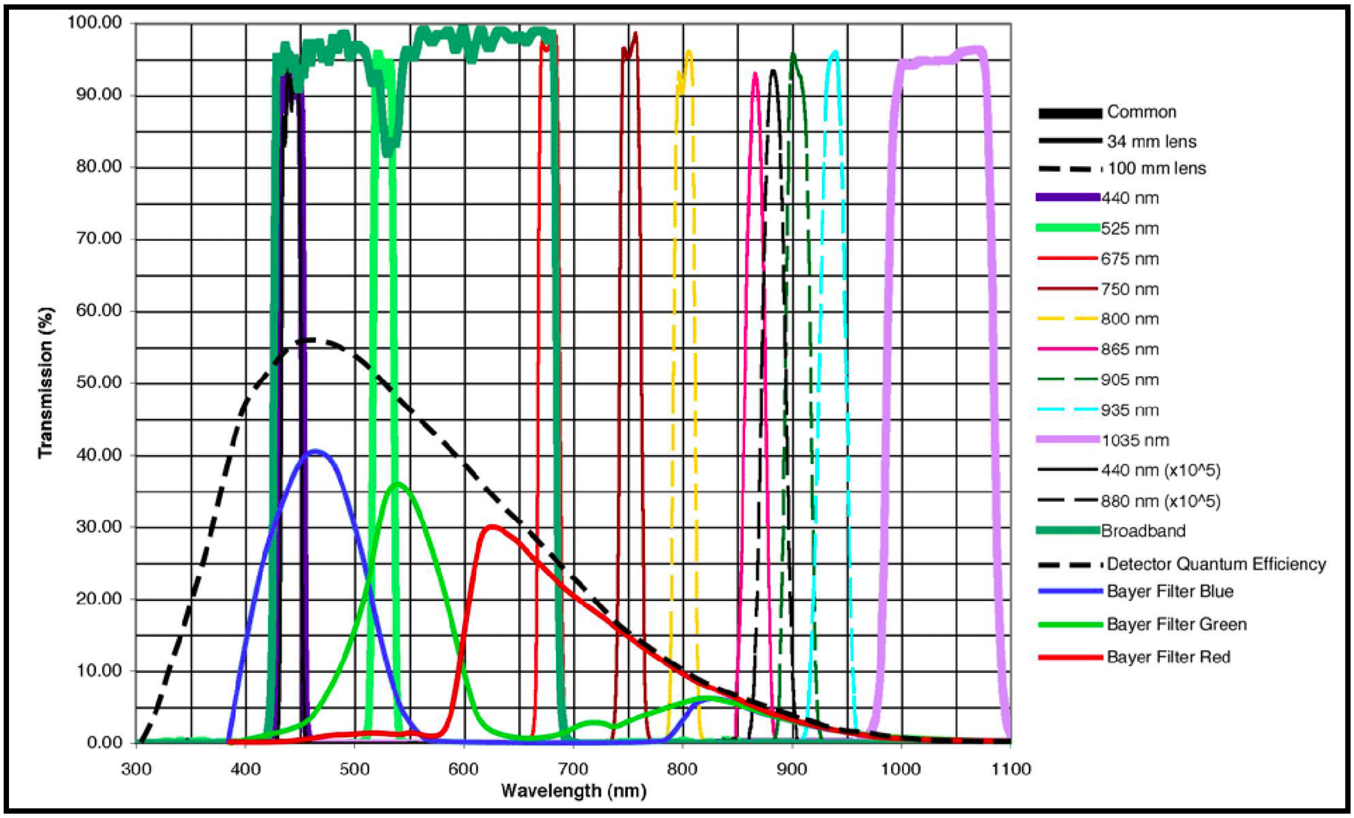


Figure 2.3.1.2.2 - Mastcam Filter Profiles, Filter Wheel Set to Nonzero

2.3.2 Mars Hand Lens Imager (MAHLI)

The Mars Hand Lens Imager (MAHLI) is a focusable color camera located on the turret at the end of the MSL robotic arm. The instrument acquires images of up to 1600 by 1200 pixels with a color quality equivalent to that of consumer digital cameras using a Bayer pattern. For details on Bayer pattern filters, see Section 4.3.3.1. It is also capable of video. MAHLI optics characteristics useful in the analysis of EDR and RDR products are described in Table 2.3.2 below.

Table 2.3.2 - MAHLI Operational Characteristics

Characteristic	Value
Field of View (FOV)	34.0 - 39.4 deg diagonal
Spatial Resolution	15 µm/pixel at 25 mm distance
Angular Resolution	0.3 - 0.34 mrad/pixel
Spectral Wavelength ± Bandwidth (λeff ± HWHM)	590 ± 88 nm (Broadband)
	640 ± 44 nm (Bayer filter Red)
	554 ± 38 nm (Bayer filter Green)
	495 ± 37 nm (Bayer filter Blue)
Focal Length	18.3 - 21.3 mm
f/number	9.8 - 8.5
Depth of Field	1.6 mm - >4800 mm
Focus Range	20.5 mm - infinity
Number of Spectral Filters	1 plus Bayer pattern on CCD

Note that the spatial resolution in Table 2.3.2 measures the working distance, which is not the same as the distance from the camera model (C) point. Spatial resolution may be calculated by:

$$\text{Pixel scale } (\mu\text{m/pixel}) = 6.9001 + [3.5201 * \text{Working Distance (cm)}]$$

Note also that the spectral bandpasses in Table 2.3.2 are approximate and use the Left Mastcam bandpasses as the representative values.

2.3.3 Descent Imager (MARDI)

The Mars Descent Imager (MARDI) is a fixed-focus color camera fixed-body-mounted to the fore-port-side of the MSL rover, even with the bottom of the rover chassis. The optical axis points in the +Z direction (toward the ground in the Rover Nav coordinate systems). The camera will take 1600 x 1200 pixel images at ~5 frames per second throughout the period of time between heatshield separation and touchdown plus a few seconds. MARDI optics characteristics useful in the analysis of EDR and RDR products are described in Table 2.3.3 below. For details on Bayer pattern filters, see Section 4.3.3.1.

Table 2.3.3 - MARDI Operational Characteristics

Characteristic	Value
Field of View (FOV)	70 - 55 deg
Spatial Resolution	1.5 m at 2 km - 1.5 mm at 2 m
Angular Resolution	0.76 mrad/pixel
Spectral Wavelength ± Bandwidth (λeff ± HWHM)	590 ± 88 nm (Broadband)
	640 ± 44 nm (Bayer filter Red)
	554 ± 38 nm (Bayer filter Green)
	495 ± 37 nm (Bayer filter Blue)
Depth of Field	2 m - infinity
Number of Spectral Filters	1 plus Bayer pattern on CCD

Note that the spectral bandpasses in Table 2.3.3 are approximate and use the Left Mastcam bandpasses as the representative values.

3. GENERAL DATA PRODUCT OVERVIEW

3.1 Data Processing Levels

This documentation uses the “Committee on Data Management and Computation” (CODMAC) data level numbering system. The MSL camera instrument EDRs referred to in this document are considered “Level 2” or “Edited Data” (equivalent to NASA Level 0). The EDRs are to be reconstructed from “Level 1” or “Raw Data”, which are the telemetry packets within the project specific Standard Formatted Data Unit (SFDU) record. They are to be assembled into complete images, but will not be radiometrically or geometrically corrected.

MSL camera instrument RDRs are considered “Level 3” (“Calibrated Data” equivalent to NASA Level 1-A), “Level 4” (“Resampled Data” equivalent to NASA Level 1-B), or “Level 5” (“Derived Data” equivalent to NASA Level 1-C, 2 or 3). The RDRs are to be reconstructed from “Level 2” edited data, and are to be assembled into complete images that may include radiometric and/or geometric correction.

Refer to Table 3.1 for a breakdown of the CODMAC and NASA data processing levels.

Table 3.1 - Processing Levels for Science Data Sets

NASA	CODMAC	Description
Packet data	Raw - Level 1	Telemetry data stream as received at the ground station, with science and engineering data embedded.
Level 0	Edited - Level 2	Instrument science data (e.g., raw voltages, counts) at full resolution, time ordered, with duplicates and transmission errors removed.
Level 1-A	Calibrated - Level 3	Level 0 data that have been located in space and may have been transformed (e.g., calibrated, rearranged) in a reversible manner and packaged with needed ancillary and auxiliary data (e.g., radiances with the calibration equations applied).
Level 1-B	Resampled - Level 4	Irreversibly transformed (e.g., resampled, remapped, calibrated) values of the instrument measurements (e.g., radiances, magnetic field strength).
Level 1-C	Derived - Level 5	Level 1A or 1B data that have been resampled and mapped onto uniform space-time grids. The data are calibrated (i.e., radiometrically corrected) and may have additional corrections applied (e.g., terrain correction).
Level 2	Derived - Level 5	Geophysical parameters, generally derived from Level 1 data, and located in space and time commensurate with instrument location, pointing, and sampling.
Level 3	Derived - Level 5	Geophysical parameters mapped onto uniform space-time grids.

3.2 Product Label and Header Descriptions

3.2.1 Overview of Labels

There are up to three different sets of product labels (metadata) that can be associated with an EDR/RDR file, to be described subsequently in this section: a) detached PDS (version 3) label, b) attached ODL (Object Description Language) [Ref 4] label, and c) attached VICAR (Video Communications and Retrieval) label. While formats and keywords may differ, all three can be reliably used to extract metadata.

It is important to note that, with one exception, the three possible labels contain the same semantic information and are in sync with each other. The exception is the PDS label, which may contain items (such as comments, keywords, group and object delimiters) that are named differently than counterparts in the ODL and VICAR labels, or may omit some items altogether. Item differences are noted in Appendices A and F and omissions are concisely listed in Appendix G. Teams that update only one or two of the labels are responsible for removing the labels they do not update, to avoid confusion. Appendix C shows example ChemCam LIBS RDR labels.

The primary label supporting operations is the attached ODL label. ODL is the format used by PDS3 labels. Therefore, from a syntactic point of view, they look identical to PDS labels (with the exception of the initial keyword and other item omissions and/or name differences) and can be processed using most PDS tools. The difference is that the keywords are not validated and are not guaranteed to be in the PDS Data Dictionary. The reason for the ODL label is because the PDS standards approval process typically is out of phase with operations delivery schedules; keywords cannot be approved and appear in the Data Dictionary within the timeline needed in operations. Every attempt is made to keep the ODL and PDS keywords the same, but there are some discrepancies. The most common difference is that many PDS keywords have a “MSL:” prefix to indicate a local data dictionary, while the ODL keywords do not.

In the case of image EDR/RDR files only, a secondary operations label exists that is the attached VICAR label. This is used by MIPL/OPGS software to operate on the images. Again, the keywords for the most part match the ODL keywords; the difference is mostly formatting and syntax.

The primary label from the archive perspective is the detached PDS3 label. This is a separate file with the same base name as the image file, with a “.LBL” extension. The detached label references the EDR/RDR filename via a keyword. This label is fully compliant with all PDS archive standards.

3.2.2 PDS and ODL Labels

As implied in the previous section, EDRs and RDRs described in this document, with the exception of the OPGS Terrain RDR, have an attached ODL label and a detached PDS label. Each institution is responsible for converting PDS-formatted products to be compatible with their own software systems (such as VICAR, IDL, ISIS, etc.).

Because the PDS and ODL formats are identical, they are described together here, with differences in content noted as such.

Per PDS standards, the detached PDS label starts with the keyword assignment:

PDS_VERSION_ID = PDS3

The attached ODL label, which contains keywords that may not be compliant to PDS standards, starts with the keyword assignment:

ODL_VERSION_ID = ODL3

This is how the two label types are distinguished. With the exception of this keyword, any PDS-format reader that does not validate against the PDS Data Dictionary should be able to read the ODL labels.

A PDS/ODL label is object-oriented and describes the objects in the data file. The PDS/ODL label contains keywords for product identification, along with the data object definition containing descriptive information needed to interpret or process the data in the file. The PDS spreadsheet object is used to describe the format of the products.

PDS/ODL label statements have the form of "keyword = value". Each label statement is terminated with a carriage return character (ASCII 13) and a line feed character (ASCII 10) sequence to allow the label to be read by many operating systems. Pointer statements with the following format are used to indicate the location of data objects in the file:

^object = location

where the carat character (^, also called a pointer) is followed by the name of the specific data object. The location is the 1-based starting record number for the data object within the file. Alternatively, it could be the 1-based byte location within the file if it includes a <bytes> unit tag. The PDS detached label includes the filename as part of the pointer:

^object = (filename, location)

Pointers are used to define the locations of the binary instrument data itself (^IMAGE for image data), the VICAR header in the case of images (^IMAGE_HEADER), and the ancillary binary data (ChemCam only, e.g. ^CCAM_SOH_DPO).

3.2.2.1 PDS Local Data Dictionary

The PDS label contains many keywords with a "MSL:" prefix. These indicate keywords that are defined in the MSL local data dictionary, rather than the primary PDS data dictionary. The ODL label does not contain these prefixes.

3.2.2.2 Keyword Length Limits

All PDS/ODL keywords are limited to 30 characters in length (Section 12.7.3 in PDS Standards Reference) [Ref 14], not including the "MSL:" prefix. Therefore, software that reads MSL PDS/ODL labels must be able to ingest keywords up to 30 characters in length.

For image RDR-producing institutions wishing to accommodate the VICAR mapping (see Section 3.2.2) of PDS/ODL keywords that use a <*unit*> tag after the value, such keywords must be limited to 24 characters in length. Otherwise, those keywords will not be transcoded from the PDS/ODL label into a VICAR label.

3.2.2.3 Data Type Restrictions

In order to accommodate VICAR dual-labeled files, 16-bit data must be stored as signed data. Unsigned 16-bit data are not supported. 12-bit unsigned data from the cameras are stored in a 16-bit signed value. 8-bit data are unsigned.

3.2.2.4 Interpretation of N/A, UNK, and NULL

During the completion of data product labels or catalog files, one or more values may not be available for some set of required data elements. In this case PDS provides the symbolic literals "N/A", "UNK", and "NULL", each of which is appropriate under different circumstances. As a note, if any one of these three symbolic literals are used in place of a keyword value that is normally followed by a Unit Tag(s) (e.g., "<*value*>"), the Unit Tag(s) is <u>removed</u> from the label.

- "N/A" ("Not Applicable") indicates that the values within the domain of this data element are not applicable in this instance. For example, a data set catalog file describing NAIF SPK kernels would contain the statement:

  INSTRUMENT_ID = "N/A"

  because this data set is not associated with a particular instrument.

  "N/A" may be used as needed for data elements of any type (e.g., text, date, numeric, etc.).

- "UNK" ("Unknown") indicates that the value for the data element is not known and never will be. For example, in a data set comprising a series of images, each taken with a different filter, one of the labels might contain the statement:

FILTER_NAME = "UNK"

if the observing log recording the filter name was lost or destroyed and the name of the filter is not otherwise recoverable.

"UNK" may be used as needed for data elements of any type.

- "NULL" is used to flag values that are *temporarily* unknown. It indicates that the data preparer recognizes that a specific value should be applied, but that the true value was not readily available. "NULL" is a placeholder. For example, the statement:

  DATA_SET_RELEASE_DATE = "NULL"

  might be used in a data set catalog file during the development and review process to indicate that the release date has not yet been determined.

  "NULL" may be used as needed for data elements of any type.

  Note that all "NULL" indicators should be replaced by their actual values prior to final archiving of the associated data.

3.2.2.5 PDS/ODL Label Constructs "Class", "Object" and "Group"

The PDS has designed a set of formal and informal constructs for labeling data products. In the PDS realm, "formal" implies a standardized design or set of rules that provides a protocol across multiple data products (e.g., multiple flight missions) for PDS validation tools, and involves a rigorous approval process. "Informal" implies a less rigorous process by which the construct meets PDS approval. For both formal and informal constructs, the PDS member keywords must be defined in the *Planetary Science Data Dictionary* (PSDD) [Ref 11] or the *Planetary Science Data MSL Local Dictionary* [Ref 12]. ODL label keywords need not be in the data dictionaries. For the EDRs and RDRs described in this document, the PDS/ODL label includes the following "formal" and "informal" constructs:

- *Class* - The Class construct is informal and resides in a PDS/ODL label as a grouping of keywords that are thematically tied together. Classes are usually preceded by a label comment, although it is not required. PDS/ODL label comments are character strings bounded by "/* */" characters.

  In the MSL Camera PDS/ODL label a Class of keywords will be preceded by a comment string as follows:

  ```
  /*  comment string  */
  keyword     =   keyword value
  keyword     =   keyword value
  ```

- *Object* - The Object construct is formal and is a set of standard keywords used for a particular data product. In the PSDD, each Object definition lists the elements required to be present each time the Object is used in a product label. The PSDD also provides a list of additional, optional keywords that are frequently used in the Object. Any element defined in the PSDD may be included as an optional element in any Object definition, at the discretion of the data preparer.

  In the MSL Camera PDS/ODL label an Object's set of keywords is specified as follows:

  ```
  OBJECT          =   Object identifier
  ```

keyword	=	*keyword value*
keyword	=	*keyword value*
END_OBJECT	=	*Object identifier*

- *Group* - The Group construct can be either a formal or informal grouping of keywords that are not components of a larger Object. Group keywords may reside in more than one Group within the label.

  The Group construct is further described in section 12.4.5 of the PDS Standards Reference, "Object Description Language Specification and Usage: GROUP Statement".

  In the MSL Camera PDS/ODL label, a Group's set of keywords is specified as follows:

GROUP	=	*Group identifier*
keyword	=	*keyword value*
keyword	=	*keyword value*
END_GROUP	=	*Group identifier*

3.2.2.6 PDS/ODL Image Object

An IMAGE object is a two-dimensional array of values, all of the same type, each of which is referred to as a *sample*. IMAGE objects are normally processed with special display tools to produce a visual representation of the samples by assigning brightness levels or display colors to the values. An IMAGE consists of a series of lines, each containing the same number of samples.

The required IMAGE keywords define the parameters for simple IMAGE objects:

- LINES is the number of lines in the image.
- LINE_SAMPLES is the number of samples in each line.
- SAMPLE_BITS is the number of bits in each individual sample.
- SAMPLE_TYPE defines the sample data type.

The IMAGE object has a number of keywords relating to image statistics. These keywords will be present in all EDRs. In RDRs, they are optional, and if they are present, they must be updated to reflect the current statistics of the image (often they will be omitted for the sake of computational efficiency). Note that the VICAR label never contains these keywords; see section 3.2.2. The statistics keywords are:

- MEAN
- MEDIAN
- MAXIMUM
- MINIMUM
- STANDARD_DEVIATION
- CHECKSUM

Many variations on the basic IMAGE object are possible with the addition of optional keywords and/or objects. The "^IMAGE" keyword identifies the start of the image. Recommended image formats are described and illustrated in Reference 4 (Appendix A.19).

3.2.2.7 PDS/ODL Table Objects and ".FMT" Files

In lieu of the binary Image Data Product Header (IDPH) Data Product Object (DPO) contained in the Engineering Camera and MMM EDR products, ChemCam EDRs include binary DPOs that contain ancillary information about the EDR. Of these, the primary carrier of such information is the Ancillary DPO. Refer to Figure 4.4 (Diagrams B and C) for a visual schematic of the ChemCam EDR structures, wherein the DPO information is noted as "auxiliary" data. Some of the data from these DPOs are pulled out into standard PDS/ODL label items. Section 4.4.2 discusses the ChemCam EDR structure and how the DPOs reside within the EDR.

The ^pointer at the beginning of the label points to the data itself. There is also a PDS/ODL Object in the label, which defines the contents and structure of the data. How the structure is described is defined by the PDS standards documents [Ref 14], so it is not fully defined here.

Of particular note is that most of the structure is offloaded to a separate format (.FMT) file via a ^STRUCTURE keyword within the object. The various ".FMT" files supporting the ChemCam EDR labels are described explicitly in Appendix D.

An example of the label object is shown below:

```
OBJECT                         =  TABLE
   NAME                        =  "CCAM_SOH_DPO_TABLE"
   COLUMNS                     =  20
   ROWS                        =  80
   ROW_BYTES                   =  46
   INTERCHANGE_FORMAT          =  "BINARY"
   DESCRIPTION                 =  "ChemCam State of Health Structure"

   /*   This format file defines columns 1-43, the first part of the SCIDATA object.     */

   ^STRUCTURE                  =  "CCAM_SOH_DPO_TABLE.FMT"

   /*   This container includes columns 44, 45, and 46. This set of columns may          */
   /*   be repeated up to 20 occurrences.                                                */

   OBJECT                      =  CONTAINER
      NAME                     =  "CCAM_SOH_TO_RCE_CONTAINER"
      START_BYTE               =  129
      BYTES                    =  100
      REPETITIONS              =  20    /*  or any number between 1 and 20               */
      DESCRIPTION              =  "ChemCam SOH to RCE structure.
         Includes arrays ccam_DPU_SOH_struct and ccam_MU_SOH_struct."

      /*   This format file defines columns 44, 45, and 46, the last part                */
      /*   of the SCIDATA object.                                                        */

      ^STRUCTURE               =  "CCAM_SOH_TO_RCE_CONTAINER.FMT"

   END_OBJECT                  =  CONTAINER
END_OBJECT                     =  TABLE
```

3.2.3 VICAR Label

For all image EDR data products and MIPL produced image RDR data products, an embedded VICAR label follows the ODL label and is pointed to by the PDS/ODL pointer "^IMAGE_HEADER". The VICAR label is also organized in an ASCII, "keyword = value" format, although there are only spaces between keywords (no carriage return/line feeds as in PDS). The information in the VICAR label is an exact copy of the information in the PDS label as defined in the next section. The reader is referred to the VICAR File Format document for details of the format, which is available at the URL "http://www-mipl.jpl.nasa.gov/external/VICAR_file_fmt.pdf". The following text is an excerpt that describes the basic structure:

> A VICAR file consists of two major parts: the labels, which describe what the file is, and the image area, which contains the actual image. The labels are potentially split into two parts, one at the beginning of the file, and one at the end. Normally, only the labels at the front of the file will be present. However, if the EOL keyword in the system label (described below) is equal to 1, then the EOL labels (End Of file Labels) are present. This happens if the labels expand beyond the space allocated for them. The VICAR file is treated as a series of fixed-length records, of size RECSIZE (see below). The image area always starts at a record boundary, so there may be unused space at the end of the label, before the actual image data starts.
> The label consists of a sequence of "keyword=value" pairs that describe the image, and is made up entirely of ASCII characters. Each keyword-value pair is separated by spaces. Keywords are strings, up to 32 characters in length, and consist of uppercase characters, underscores ("_"), and numbers (but should start with a letter). Values may be integer, real, or strings, and may be multiple enclosed in parentheses (e.g. an array of 5 integers, but types cannot be mixed in a single value). Spaces may appear on either side of the equals character (=), but are not normally present. The first keyword is always LBLSIZE, which specifies the size of the label area in bytes. LBLSIZE is always a multiple of RECSIZE, even if the labels don't fill up the record. If the labels end before LBLSIZE is reached (the normal case), then a 0 byte terminates the label string. If the labels are exactly LBLSIZE bytes long, a null terminator is not necessarily present. The size of the label string is determined by the occurrence of the first 0 byte, or LBLSIZE bytes, whichever is smaller. If the system keyword EOL has the value 1, then End-Of-file Labels exist at the end of the image area (see above). The EOL labels, if present, start with another LBLSIZE keyword, which is treated exactly the same as the main LBLSIZE keyword. The length of the EOL labels is the smaller of the length to the first 0 byte or the EOL's LBLSIZE. Note that the main LBLSIZE does not include the size of the EOL labels. In order to read in the full label string, simply read in the EOL labels, strip off the LBLSIZE keyword, and append the rest to the end of the main label string.

Note that the EOL labels will not appear in archive products.

A binary header may appear in between the VICAR label and the image, containing arbitrary binary data that are not interpreted by VICAR. The number of records in this header is defined by the VICAR system keyword NLB. This binary header area is where the ChemCam DPOs are stored.

3.2.4 Mapping of PDS/ODL and VICAR Labels

In the cases of the attached ODL and VICAR labels, information content is identical, by definition. ODL and VICAR labels may be used interchangeably, for any purpose in the mission. Any software that modifies one label must also modify the other, or remove them entirely. This is often most easily accomplished by stripping off one of the headers, processing the remaining label as desired locally, and then running a conversion tool to re-create the missing header. Such tools will be provided by MIPL.

In the case of the detached PDS label, information content is not necessarily identical to the ODL label, and therefore not necessarily identical to the VICAR label when applicable. It is intended that a PDS label can have reduced content, but not increased content, in comparison to the ODL and VICAR labels. Appendix G tracks the label keywords that are not present in the PDS label.

It is important to note that only product labels containing the label keyword assignment "PDS_VERSION_ID = PDS3" are valid PDS products.

The mapping between PDS and ODL keywords is straightforward, and keyword names are usually the same. However, there are some keyword name differences. Appendix A shows a label in ODL format, and identifies differences in ODL and PDS keyword names. For space reasons in this document, the corresponding VICAR label is omitted from this document, but it is required. The mapping rules are as follows:

- For mapped PDS, ODL and VICAR (if applicable) keywords, values are identical in all cases with the exception of differences mandated by the file format itself, such as quoting rules. See the respective PDS and VICAR documents for details, but in general, PDS/ODL uses double quotes (") while VICAR uses single quotes (').

- For ODL and VICAR label keywords, with the exception of those defining the file format itself (described below), names are identical in both cases.

- For PDS labels, resident keywords map 1-to-1 with ODL keywords, but not all ODL keywords have a mapping to PDS keywords. ODL keywords not represented in PDS labels are listed in Appendix G.

- For PDS labels, keyword names are usually identical to keywords in the ODL label, but not necessarily so. Name differences between keywords are identified in Appendix A and F.

- Any ODL label group maps 1-to-1 to a VICAR property set with the same name (Group) name == property set name). All contained keywords are identical in both cases. The GROUP and END-GROUP keywords are omitted from the VICAR label; PROPERTY keywords are used instead (as per the VICAR file format definition).

- For PDS labels, groups map 1-to-1 with the ODL label, but group names can differ. Name differences between groups are identified in Appendix A.

- Any set of PDS/ODL keywords not in a group (in PDS terms, a class) is identified by an introductory comment (e.g. /* IDENTIFICATION DATA ELEMENTS */). Such classes map 1-to-1 to a VICAR property set. The name of the VICAR property set and the name of the PDS introductory comment map as follows:

Table 3.2.4.1 – PDS/ODL Class to VICAR Property Set Mappings

PDS/ODL Class Comment	VICAR Property Set Name
/* FILE DATA ELEMENTS */	Special case, see below
/* POINTERS TO DATA OBJECTS */	Special case, see below
/* IDENTIFICATION DATA ELEMENTS */	IDENTIFICATION
/* TELEMETRY DATA ELEMENTS */	TELEMETRY
/* HISTORY DATA ELEMENTS */	PDS_HISTORY_PARMS
/* COMPRESSION RESULTS */	COMPRESSION_PARMS

- For VICAR labels, PDS/ODL comments (i.e., /* *string* */) are stored in a keyword named "PDS_COMMENT". This keyword appears in the VICAR property containing the elements immediately following the comment. When converting from VICAR to PDS/ODL, the comment is placed immediately before the group or class. Blank lines should surround the comment. Note that with OPGS-generated EDR and RDR data products, multiple comment lines in a Group are not supported.

- The PDS/ODL objects IMAGE_HEADER and IMAGE, as well as the keywords in /* FILE DATA ELEMENTS */ and the ^IMAGE and ^IMAGE_HEADER pointers (in /* POINTERS TO DATA OBJECTS */) in the table above, do not map directly to VICAR. They all describe the layout of the file and the image data. The VICAR equivalent for all of these items is the VICAR System label. Information maps between these in a straightforward way. It should be trivial to construct a VICAR system label and the above-referenced PDS entities after referring to the respective file-format-definition documents. Note that the /* FILE DATA ELEMENTS */ and /* POINTERS TO DATA OBJECTS */ comments are constant and so are not mapped to PDS_COMMENT keywords in the VICAR label. They are inserted automatically as part of the system label conversion process.

- The statistics-related keywords in the PDS/ODL IMAGE object are MEAN, MEDIAN, MAXIMUM, MINIMUM, STANDARD_DEVIATION, and CHECKSUM. These keywords are never transferred to the VICAR label. For VICAR -> PDS/ODL conversion, they can be computed from the image, or simply omitted from the PDS/ODL image (for RDRs only - EDRs require them).

- The few remaining items in the PDS/ODL_IMAGE object are treated specially. The FIRST_LINE, FIRST_LINE_SAMPLE, INVALID_CONSTANT, and MISSING_CONSTANT keywords are transferred to the VICAR IMAGE_DATA property set.

- Any PDS/ODL keyword with a <*unit*> tag after the value is transferred to the VICAR label without the unit tag. A VICAR keyword with the same name, but with "__UNIT" (two underscores) appended to the end, is added with the value of the unit. So, for example, the PDS/ODL keyword "EXPOSURE_TIME = 1.5 <s>" would translate to two VICAR keywords: "EXPOSURE_TIME = 1.5" and "EXPOSURE_TIME__UNIT = s". Note that because of this, any PDS/ODL keyword that can support a unit is limited to 24 characters. If there is more than one value (an array), a unit is associated with each. In this case, the "__UNIT" VICAR keyword becomes multi-valued also, with each unit copied in sequence. If one of the elements does not have a unit (but others do), the corresponding entry is "N/A" (which is not copied to the PDS/ODL label). For example, PDS/ODL "CONTRIVED_ANGLE = (1.2 <rad>, 22.0, 54.1 <deg>)" would map to VICAR "CONTRIVED_ANGLE = (1.2, 22.0, 54.1)" and "CONTRIVED_ANGLE__UNIT = (rad, N/A, deg)".

- The VICAR history label is omitted from the PDS header

- PDS/ODL TABLE objects (used to describe the ChemCam DPOs) map to VICAR property sets with each table, or container within the table (subobject), being a separate property set. The name of the property set matches the name of the table object (OBJECT = name). Certain keywords are handled specially, as described in the following table:

Table 3.2.4.2 – Referencing PDS/ODL Objects in VICAR Label

VICAR Label	PDS / ODL Meaning
PDS_OBJECT__TYPE=TABLE	Type of PDS/ODL object.
PDS_OBJECT__PTR=recnum	^pointer = (recnum + pointer to header area)
PDS_OBJECT__OFFSET=offset	Byte offset within record for pointer, if needed.
PDS_OBJECT__LOC=BINARY HEADER	Tells where the object is for proper PTR adjustment.
PDS_OBJECT__CHILD=(xxx,yyy)	Names of subobjects (PDS/ODL containers).
PDS_OBJECT__PARENT=aaa	Name of parent object.

Note: PDS_OBJECT__PTR is a 1-based record number within the binary header, so it must be adjusted to be relative to the start of the file for the ODS/ODL pointer.

- PDS/ODL ^pointer objects other than ^IMAGE and ^IMAGE_HEADER that point to tables or containers map to PDS_OBJECT__* keywords as described in the table above. Any remaining pointers, such as ^STRUCTURE, map to VICAR by removing the leading caret and appending "__PTR". Thus "^STRUCTURE" becomes "STRUCTURE__PTR" in VICAR.

3.3 Binary Data Storage Conventions

EDR and RDR data products for MSL camera image and LIBS spectra data are stored as binary data. For the image EDRs, the data formats include rescaled 8-bit integers stored in an unsigned byte, as well as 10-bit or 12-bit integers stored in signed 16-bit integers. The spectrum EDR cases vary between unsigned 16-bit and unsigned 32-bit integers. The PDS and VICAR labels are stored as ASCII text.

3.3.1 Bit and Byte Ordering

The ordering of bits and bytes is only significant for instrument (image and spectra) and binary header data; all other labeling information is in ASCII.

For non-byte instrument data, which includes 8-bit unsigned shorts, 16-bit signed shorts, 32-bit signed ints, and 32- and 64-bit IEEE floating-point numbers, the data may be stored in either Most Significant Byte first ("big-endian", as used by e.g. Sun computers and Java), or Least Significant Byte first ("little-endian", as used by e.g. Linux and Windows computers). In an EDR/RDR product, the instrument data can have only one ordering, but it is dependent on the host platform where the data was processed. Binary header data, applicable to ChemCam, can have a different ordering than the instrument data. This follows both the PDS/ODL and VICAR file format conventions.

For Engineering Camera and MMM image data, the PDS/ODL label carries keyword SAMPLE_TYPE in the IMAGE Object to define which ordering is used in the file. The VICAR label carries keywords

INTFMT and REALFMT in the System portion of the label to define the ordering. Both of these file formats specify that bit 0 is the least significant bit of a byte. See the respective PDS and VICAR file format definition documents.

For ChemCam instrument data, RMI image data ordering is defined by keyword SAMPLE_TYPE in the IMAGE Object of the PDS/ODL label and by keywords INTFMT and REALFMT in the System portion of the VICAR label. LIBS spectra data ordering is defined by pointer STRUCTURE in the CCAM_LIBS_TABLE Object of the PDS/ODL label. ChemCam binary headers are of varying data type, as described in their PDS/ODL label Object definitions and the external ".FMT" files. However, they are always in MSB ("big-endian") format, even if the instrument data are LSB.

Table 3.3.1 - MSL Image EDR/RDR and Spectrum EDR Bit Ordering

Address	MSB-first	LSB-first
n	most significant byte	least significant byte
n+1	next	next
n+2	next	next
n+3	least significant byte	most significant byte

4. EDR PRODUCT SPECIFICATION

MSL instrument EDRs and RDRs described in this document will be generated by JPL's Multimission Instrument Processing Laboratory (MIPL) under the OPGS subsystem of the MSL GDS Realtime Operations (RTO) element. Other RDRs described in this document will be generated by the ChemCam Science Team.

The EDR consists of unprocessed experiment data stored in binary format. The EDR structures defined in this document vary depending on instrument, with all containing attached metadata labels.

For MSL, with the exception noted in Section 1.3, the camera image EDR and LIBS spectrum EDR are the fundamental instrument data archive product. They will be generated as "raw" uncalibrated data within an automated pipeline process managed by MIPL under OPGS at JPL. The size of an image EDR data product is approximately 2 MB, while the size of a spectrum EDR is approximately 260 KB. The total estimated volume of image EDRs and spectrum EDRs over the course of the nominal 2-year MSL mission is approximately 376 Gigabytes and 6 Gigabytes, respectively.

4.1 EDR General Processing

The EDR processing begins with the reconstruction of packetized telemetry data resident on the TDS by the Mission data Processing and Control Subsystem (MPCS) into a binary ".dat" data product and associated ".emd" Earth meta-data file. The data product and meta-data are written by MPCS to the Operations Data Store (ODS) and messages are generated on a Java Message Server (JMS) bus, where they are ingested by MIPL's EDR generator "msledrgen" and processed with SPICE kernels provided by NAIF. The EDR will be generated within 60 seconds after the JMS message describing the ODS location of the respective the binary data product and associated Earth meta-data file has been received by the OPGS pipeline system. This data flow is illustrated in Figure 4.1, and is elaborated subsequently in this section:

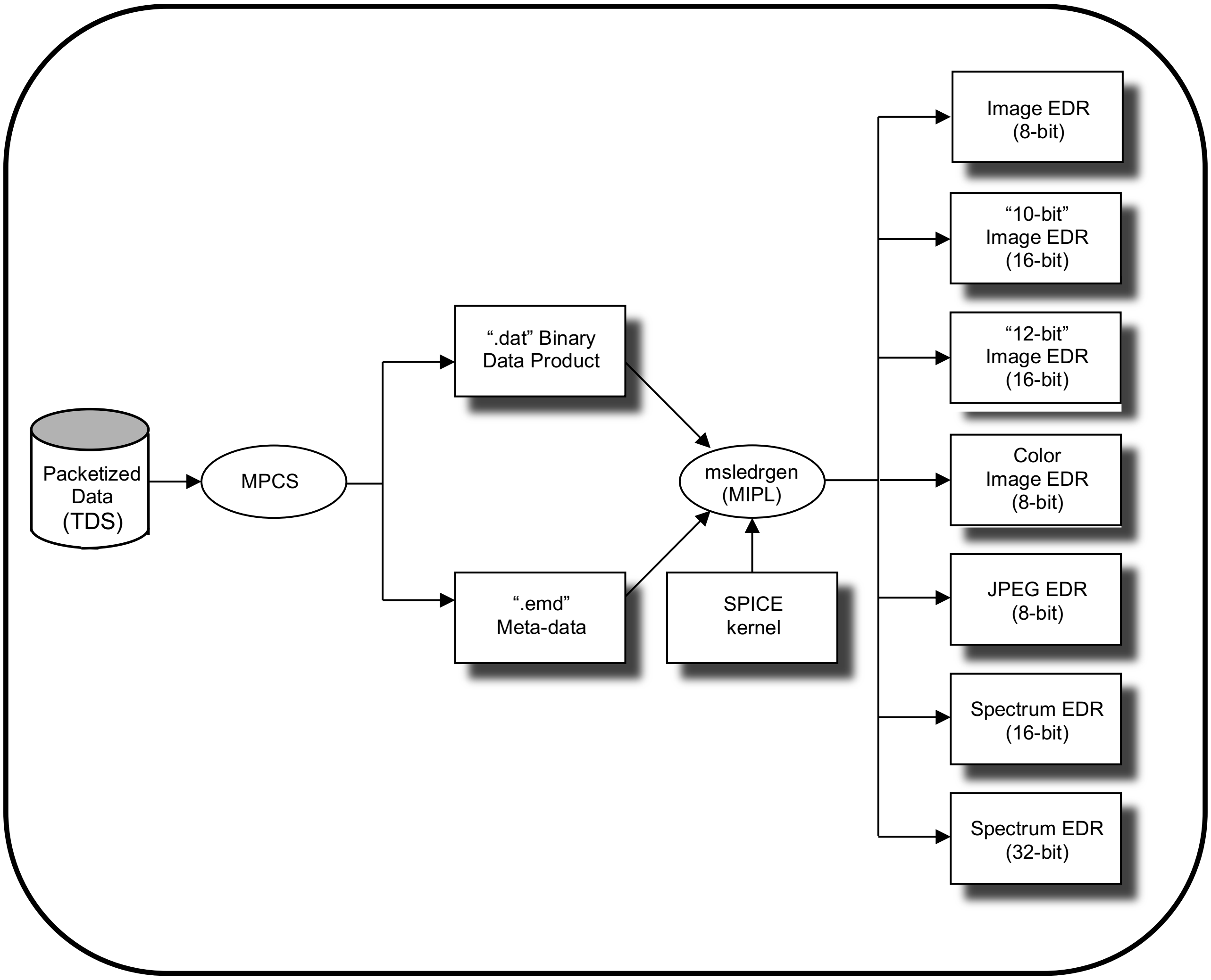


Figure 4.1 - EDR Processing Flow

In all EDR cases, missing packets will be identified and reported for retransmission to the ground as "partial datasets". Prior to retransmission, the missing EDR data will be filled with zeros. The EDR data will be reprocessed only after all "partial datasets" are retransmitted and received on the ground. In these cases, the EDR version will be incremented so as not to overwrite any previous EDR versions. The EDR data product will be placed into FEI for distribution and to facilitate the archiving process.

4.2 EDR Product Types

Descriptions for the various EDR product types are provided in this section. They are broken down into four groupings: a) Image, b) Image Support, c) Spectroscopy, and d) Health & Safety. Refer also to Table 4.3 for a mapping between the source MSL instrument and the EDR product type.

4.2.1 Image EDRs

4.2.1.1 Standard Image EDR

All of the imaging instruments create standard image EDRs (for engineering camera and RMI, this is the only image EDR type). These images are uncompressed, raster-format data products. They are nominally 12-bit products, stored as 16-bit signed integers. If 12-to-8-bit scaling (called "companding" by the MMM instruments) is performed onboard, then pixels are stored as 8-bit unsigned bytes. Engineering camera and RMI images are always a single band. MMM EDRs may consist of one or three bands, depending on whether color processing was done onboard.

4.2.1.2 MMM-specific Image EDR

For MMM, there is a Bayer pattern of R,G,B filters superimposed on the CCD. This may be processed onboard to create a color image. If this is done, the data will always be sent as 3-band, 8-bit JPEG color, and the EDR will be 3-band byte format. The MMM cameras can also send data without color processing, in which case it will either be a single-band 8-bit JPEG, or an 8- or 12-bit uncompressed or losslessly-compressed image, stored as described above. In any case, if a JPEG is sent, a JPEG EDR will also be created.

4.2.1.2.1 MMM "JPEG" EDR

The MMM instruments often compress their images using JPEG for downlink. The JPEG EDR contains this JPEG as originally downlinked; it has not been uncompressed or recompressed. This type is not produced unless JPEG compression used onboard (i.e. no ground compression). It may be a color or grayscale (single band) JPEG.

4.2.1.2.2 MMM "Z-stack" EDR

The MAHLI (most typically) and Mastcam instruments may create a Z-stack image onboard, by combining images taken at different focus settings. The result is a best-focus image with a much greater depth of field. Each pixel contains data from the source image(s) that are in best focus (potentially interpolated between neighboring images). This image may be color or a single band.

NOTE: The metadata regarding vehicle state for Z-stack images reflects the state of the vehicle at the time the Z-stack product itself was created. This is generally not the same as the vehicle state at the time the imagery was acquired. Most importantly, this means the arm state and mast state, and thus camera model, for these images do not properly describe the image. The metadata must be obtained from the thumbnail or full frame of one of the images that went into making the Z-stack. Determining the proper image to use is beyond the scope of this document, but can often be inferred by inspection of the available data.

4.2.1.2.3 MMM "Depth Map" EDR

As part of MMM Z-stack processing, a Depth Map (also referred to as a Range Map) image may also be produced. This 8-bit, single-band EDR indicates which image the best-focus data came from. Pixels with a digital number (DN) of 0 indicate the first image in the Z-stack, while a DN of 255 indicates the last, with linear scaling between. For example, with a 5-image stack and expected DNs of 0, 64, 128, 192, and 255, a DN of 96 would indicate the depth is halfway between images 2 and 3.

NOTE: The metadata regarding vehicle state for Depth Map images reflects the state of the vehicle at the time the Depth Map product itself was created. This is generally not the same as the vehicle state at the time the imagery was acquired. Most importantly, this means the arm state and mast state, and thus camera model, for these images do not properly describe the image. The metadata must be obtained from the thumbnail or full frame of one of the images that went into making the Depth Map. Determining the proper image to use is beyond the scope of this document, but can often be inferred by inspection of the available data.

4.2.1.2.4 MMM "Video" EDR

The MMM Video EDR is identical in format to the standard image EDR. It represents a single frame of a video sequence. The EDR type is separate in order to better distinguish video frames from still frames.

4.2.1.2.5 MMM "Recovered" EDR

An MMM Recovered EDR is an EDR for which the metadata is unavailable. It existed in camera memory but for whatever reason the onboard data product describing it was lost or deleted. Such products have extremely limited metadata, with lots of "UNK" (unknown) values. This can occur to any MMM EDR type.

4.2.1.3 Image Sampling Types

Except where otherwise noted, in general, the image types described above can be sent in one of three pixel sampling types: 1) Full-frame, 2) Subframe, or 3) Downsampled. They may also be sent as a Thumbnail, in addition to (or instead of) the three aforementioned formats.

4.2.1.3.1 "Full Frame" EDR

Full-frame EDRs contain the entire contents of the CCD. For the engineering cameras and RMI, this means a 1024x1024 image. For MMM instruments, this means 1648 (columns) x 1200 (rows).

4.2.1.3.2 "Subframe" EDR

Sub-frame EDRs are a subset of rows and columns of the full-frame image. They can be thought of as a window on the CCD, with the same resolution but smaller coverage area.

4.2.1.3.3 "Downsampled" EDR

Downsampled EDRs contain a smaller version of the image, resulting in reduced resolution of the same coverage area. They apply to engineering camera and RMI camera images, but not to MMM camera images. Downsampling can be done via one of three methods: 1) nearest neighbor pixel averaging, 2) pixel averaging with outlier rejection, or 3) computing the median pixel value. Note that downsampling can be applied to subframes, although this is not normally done.

4.2.1.3.4 "Thumbnail" EDR

Thumbnails are a reduced-resolution version of the original image, sent in addition to, or instead of, the original image. They apply to all image EDRs. The main purpose of a Thumbnail EDR is to provide an image summary using a very low data volume compared to the original image. Decisions about downlinking the original image can be made using the Thumbnail. For the engineering cameras and RMI, Thumbnails are produced relative to the full-frame image, even if the product is downsampled or subframed. For MMM camera images, Thumbnails are of the subframe region only, and not the full-frame.

4.2.2 Image Support EDRs

4.2.2.1 "Row Summation" EDR

A row summation EDR is the summing of the rows of a full-frame or subframed image and returning the results. The EDR is an nx1 array of 32-bit integers (whose length is equal to the image height) where the DN value of the ith element is the value of the sum of all the pixels in the ith row. Applicable to the Engineering Camera instrument suite.

4.2.2.2 "Column Summation" EDR

A column summation EDR is the summing of the columns of a full-frame or sub-framed image and returning the results. The EDR is a 1xn array of 32-bit integers (whose length is equal to the image width) where the DN value of the ith element is the value of the sum of all the pixels in the ith column. Applicable to the Engineering Camera instrument suite.

4.2.2.3 "Reference Pixel" EDR

The onboard CCD array has "Reference" dark pixels (12-bits) located before and after the image data. For the Eng. Cameras, there are 16 "pre-Reference" and 15 "post-Reference" pixels followed by the camera hardware serial number (left-shifted by 4 bits if 12-bit data) in each row. For the ChemCam RMI, there are 12 "pre-Reference" lines before the image lines and 12 "post-Reference" lines after the image lines. Also, there are 16 "pre-Reference" pixels before the image data in each row. Applicable to the Engineering Camera instrument suite and ChemCam RMI.

4.2.2.4 "Histogram" EDR

The histogram EDR is a 32-bit integer array storing the histogram of the image. A 1x256 or 1x4096 array will be returned. Applicable to the Engineering Camera instrument suite.

4.2.3 Spectroscopy EDRs

4.2.3.1 "Spectrum" EDR

The spectrum EDR is comprised of one or more ChemCam LIBS spectroscopy intensity (spectra) values, or digital numbers (DNs), stored as an array of 16-bit or 32-bit unsigned integers. The data range is dependent on which of the three possible spectrometers (UV, Visible, Near-infrared) were commanded. Applicable to the ChemCam LIBS instrument.

4.2.4 Engineering EDRs

4.2.4.1 "State-of-Health" (SOH) EDR

The SOH EDR is comprised of binary metadata describing the health and safety of ChemCam instrumentation. There are five variants: a) State-of-Health, b) Power On, c) Power Off, d) Sun Safe and e) Initialize. Stored in 16-bit unsigned integer format, they are a vector of 9 entities for the ChemCam Body Unit and a vector of 39 entities for the ChemCam Mast Unit. Applicable to the ChemCam instrument suite as a standalone product.

4.2.4.2 "NavMap" EDR

The NavMap products are engineering products used to assess the performance of the rover's on-board navigation software. All 3-char product identifiers starting with "N" (form: Nxx) are NavMap products.

The list of NavMap products and their short descriptions are provided in Table 4.3. Most, but not all, are images of some sort in the standard VICAR/ODL/PDS format with varying data types, sizes, and numbers of bands. Because these are detailed engineering products, no further description of them is provided in this document. They will not be archived.

4.3 EDR Product Format

Description of EDR product formats in this section will be by instrument suite: a) Engineering Cameras, b) ChemCam Instruments, c) MMM Instruments.

The EDR will be formatted according to this SIS, following the general terms of labeling and bit ordering previously discussed in Sections 3.2 and 3.3, respectively. This section details the specifics of a variety of formats across all image EDRs and spectrum EDRs. The various EDR formats and their data sizes, across all instrument suites, are listed in Table 4.3 and discussed subsequently in this section:

Table 4.3 – List of EDR Types and Formats

Type	Product Identifier	Size	Format (bits)	Instruments	Description
Standard Image	EDR	Full-frame • 1024 lines x 1024 Samples for Eng. Cams • 1024 lines x 1024 samples without Ref Pixels (1056 lines x 1072 samples with Ref Pixels) for RMI • Up to 1200 lines x 1648 samples for MMM Cams	8-bit unsigned or 16-bit signed integer	Eng. Cams, RMI, MMM Cams	Nominal full sized, full resolution data product. Nominally, data are acquired at 10-bit (RMI) or 12-bit (Eng. Cams) resolution, stored as the last 12 bits of a 16-bit integer. If "12 to 8-bit" scaling was commanded (Eng. Cams), the 12-bit data has been scaled onboard as 8-bit pixels, stored on the ground in a single unsigned byte. Note that as an RDR, these scaled 8-bit data can be unscaled back to 12 bits, using an Inverse Lookup Table (ILUT), stored as the last 12 bits of a 16-bit integer.
		Subframe Variable size	8-bit unsigned or 16-bit signed integer	Eng. Cams, RMI, MMM Cams	Same format as Full Frame, but only a selected row (line) and/or column sub-frame is read back. ICER is also capable of subframing. The bit scaling rules described for the Full Frame case above also apply here.
		Downsampled Variable size (usually 1/4 size of Full Frame)	8-bit unsigned or 16-bit signed integer	Eng. Cams, MMM Cams	Images are converted to smaller images via a) nearest neighbor pixel averaging, b) pixel averaging with outlier rejection, or c) computing the median pixel value. The bit scaling rules described for the Full Frame case above also apply here.
		Thumbnail • 64 lines x 64 samples for Eng. Cams, nominally • 64 lines x 64 samples for RMI, nominally • Variable size for MMM Cams	8-bit unsigned or 16-bit signed integer	Eng. Cams, RMI, MMM Cams	This data product is a spatially sized down version of an existing Full Frame, so is less than full size and less than full resolution. The bit scaling rules described for the Full Frame case above also apply here.
MMM-specific Image	EJP	JPEG Variable size	8-bit unsigned	MMM Cams	Contains JPEG-compressed data as originally downlinked. It may be a color or grayscale (single band) JPEG.

Type	Product Identifier	Size	Format (bits)	Instruments	Description
	EZS	Z-stack Variable size	8-bit unsigned	MMM Cams	A best-focus Z-stack image, from a combination of images at different focus settings, with a much greater depth of field. Typically MAHLI data. It may be color or a single band.
	EDM	Depth Map Variable size	8-bit unsigned	MMM Cams	Part of MMM Z-stack processing. This single-band image indicates which image the best-focus data came from. DN of 0 indicates first image in the Z-stack, 255 is the last, with linear scaling between.
	EVD	Video Variable size	8-bit unsigned	MMM Cams	Identical in format to the Standard image EDR. It represents a single frame of a video sequence.
Image Support	ERS	Row Summed N lines x 1 sample	32-bit unsigned	Eng. Cams	Array of 32-bit integers whose length is equal to image height, wherein the DN value for the Jth element equals the sum of all pixels in the Jth row.
	ECS	Column Summed 1 line x N samples	32-bit unsigned	Eng. Cams	Array of 32-bit integers whose length is equal to image width, wherein the DN value for the Jth element equals the sum of all pixels in the Jth column.
	ERP	Reference Pixel • 1024 lines x 32 (16 “pre”, 16 “post”) samples for Eng.Cams • 1024 lines x 16 “pre” samples, plus 12 “pre” lines and “post” lines, for RMI	16-bit unsigned	Eng. Cams, RMI	Dark pixels bookending (pre- and post-) image pixels during serial register readout. There are “pre-” Reference and “post-” Reference pixels, and in the case of Eng. Cameras, an additional 1 for the camera hardware serial number (left-shifted by 4 bits if 12-bit data).
	EHG	Histogram • 1 line x 4096 samples • 1 line x 256 samples	32-bit unsigned	Eng. Cams	DN histogram computed from image can have either 256 or 4096 bins, each capable of holding count values of up to 4,194,304.
	EID	IDPH-only 1 line x 1 sample	8-bit unsigned or 16-bit signed integer	Eng. Cams	IDPH (Image Data Product Header) only, with no image data. The data is formatted as a 1x1 image with a 0 pixel value. This product is generated when the cameras are commanded to acquire a picture (for example, to pre-point the RSM), but no image data are requested from the camera. These types of products are intended to serve as metadata only and are typically used to identify RSM pre-point activities.
Spectroscopy	EDR	variable	16-bit or 32-bit	LIBS	Array of spectrum intensity values stored in 16-bit or 32-bit data ranges, depending on commanded

Type	Product Identifier	Size	Format (bits)	Instruments	Description
			unsigned		mode of ChemCam LIBS spectrometers.
State-of-Health	EIN EPW EPO EWU ESS	variable	16-bit unsigned	RMI, LIBS	Binary metadata fields stored in 16-bit integer. For ChemCam Body Unit, vector of 9 entities. For ChemCam Mast Unit, vector of 39 entities.
Navigation Map	NGD	n/a	n/a	Eng. Cams	Nav Map Goodness – measure of how safe rover would be, centered at each cell
	NCE	n/a	n/a	Eng. Cams	Nav Map Certainty – measure of how certain the Goodness map values are
	NID	n/a	n/a	Eng. Cams	Nav Map Idles – number of steps since a cell was updated
	NMN	n/a	n/a	Eng. Cams	Nav Map Minimum Count – measure of how often this cell was Minimum elevation
	NMX	n/a	n/a	Eng. Cams	Nav Map Maximum Count – measure of how often this cell was Maximum elevation
	NEL	n/a	n/a	Eng. Cams	Nav Map Elevation – cell elevation
	NNX	n/a	n/a	Eng. Cams	Nav Map Normal X – X component of surface normal at cell
	NNY	n/a	n/a	Eng. Cams	Nav Map Normal Y – Y component of surface normal at cell
	NNZ	n/a	n/a	Eng. Cams	Nav Map Normal Z – Z component of surface normal at cell
	NTL	n/a	n/a	Eng. Cams	Nav Map Tilt – tilt at this cell
	NRS	n/a	n/a	Eng. Cams	Nav Map Residual – plane fit residual centered at this cell
	NOF	n/a	n/a	Eng. Cams	Nav Map Offset – plane equation term
	NMO	n/a	n/a	Eng. Cams	Nav Map Moments – First, Second order moments of stereo points in this cell
	NFP	n/a	n/a	Eng. Cams	Nav Map Footprint – moment footprint
	NRK	n/a	n/a	Eng. Cams	Nav Map Rock – find a Rock map
	NMC	n/a	n/a	Eng. Cams	Nav Map Minimum Cells – *to be determined*
	NFV	n/a	n/a	Eng. Cams	Nav Map FOV – *to be determined*
	NFE	n/a	n/a	Eng. Cams	Nav Map FOV Edge – *to be determined*
	NPI	n/a	n/a	Eng. Cams	Nav Map Path Information – binary path evaluations
	NLY	n/a	n/a	Eng. Cams	Nav Map Layer Certainty – certainty of Layer data
	NED	n/a	n/a	Eng. Cams	Nav Map Elevation Difference – elevation difference
	NL0	n/a	n/a	Eng. Cams	Nav Map Layer 0 – Layer 0 goodness map

Type	Product Identifier	Size	Format (bits)	Instruments	Description
	NL1	n/a	n/a	Eng. Cams	Nav Map Layer 1 – Layer 1 goodness map
	NL2	n/a	n/a	Eng. Cams	Nav Map Layer 2 – Layer 2 goodness map
	NL3	n/a	n/a	Eng. Cams	Nav Map Layer 3 – Layer 3 goodness map
	NL4	n/a	n/a	Eng. Cams	Nav Map Layer 4 – Layer 4 goodness map
	NL5	n/a	n/a	Eng. Cams	Nav Map Layer 5 – Layer 5 goodness map
	NL6	n/a	n/a	Eng. Cams	Nav Map Layer 6 – Layer 6 goodness map
	NL7	n/a	n/a	Eng. Cams	Nav Map Layer 7 – Layer 7 goodness map
	NL8	n/a	n/a	Eng. Camas	Nav Map Layer 8 – Layer 8 goodness map
	NL9	n/a	n/a	Eng. Cams	Nav Map Layer 9 – Layer 9 goodness map
	NLA	n/a	n/a	Eng. Cams	Nav Map Layer 10 – Layer 10 goodness map
	NLB	n/a	n/a	Eng. Cams	Nav Map Layer 11 – Layer 11 goodness map
	NLC	n/a	n/a	Eng. Cams	Nav Map Layer 12 – Layer 12 goodness map
	NLD	n/a	n/a	Eng. Cams	Nav Map Layer 13 – Layer 13 goodness map
	NLE	n/a	n/a	Eng. Cams	Nav Map Layer 14 – Layer 14 goodness map
	NLF	n/a	n/a	Eng. Cams	Nav Map Layer 15 – Layer 15 goodness map
	NT0	n/a	n/a	Eng. Cams	Nav Map Temporary Storage Layer 0 – temporary storage for Layer 0 goodness map
	NT1	n/a	n/a	Eng. Cams	Nav Map Temporary Storage Layer 1 – temporary storage for Layer 1 goodness map
	NT2	n/a	n/a	Eng. Cams	Nav Map Temporary Storage Layer 2 – temporary storage for Layer 2 goodness map
	NT3	n/a	n/a	Eng. Cams	Nav Map Temporary Storage Layer 3 – temporary storage for Layer 3 goodness map
	NT4	n/a	n/a	Eng. Cams	Nav Map Temporary Storage Layer 4 – temporary storage for Layer 4 goodness map
	NT5	n/a	n/a	Eng. Cams	Nav Map Temporary Storage Layer 5 – temporary storage for Layer 5 goodness map
	NT6	n/a	n/a	Eng. Cams	Nav Map Temporary Storage Layer 6 – temporary storage for Layer 6 goodness map
	NT7	n/a	n/a	Eng. Cams	Nav Map Temporary Storage Layer 7 – temporary storage for Layer 7 goodness map
	NT8	n/a	n/a	Eng. Cams	Nav Map Temporary Storage Layer 8 – temporary storage for Layer 8 goodness map

Type	Product Identifier	Size	Format (bits)	Instruments	Description
	NT9	n/a	n/a	Eng. Cams	Nav Map Temporary Storage Layer 9 – temporary storage for Layer 9 goodness map
	NTA	n/a	n/a	Eng. Cams	Nav Map Temporary Storage Layer 10 – temporary storage for Layer 10 goodness map
	NTB	n/a	n/a	Eng. Cams	Nav Map Temporary Storage Layer 11 – temporary storage for Layer 11 goodness map
	NTC	n/a	n/a	Eng. Cams	Nav Map Temporary Storage Layer 12 – temporary storage for Layer 12 goodness map
	NTD	n/a	n/a	Eng. Cams	Nav Map Temporary Storage Layer 13 – temporary storage for Layer 13 goodness map
	NTE	n/a	n/a	Eng. Cams	Nav Map Temporary Storage Layer 14 – temporary storage for Layer 14 goodness map
	NTF	n/a	n/a	Eng. Cams	Nav Map Temporary Storage Layer 15 – temporary storage for Layer 15 goodness map
	NWR	n/a	n/a	Eng. Cams	Nav Map Wraparound – *to be determined*
	NSF	n/a	n/a	Eng. Cams	Nav Map Stereo Filters – pixel-by-pixel labeling of stereo filter that rejected pixel value
	NSD	n/a	n/a	Eng. Cams	Nav Map Stereo Disparity – left disparity image
	NLR	n/a	n/a	Eng. Cams	Nav Map Left Rectified – left rectified image
	NRR	n/a	n/a	Eng. Cams	Nav Map Right Rectified – right rectified image
	N2F	n/a	n/a	Eng. Cams	Nav Map Second Stereo Filters – *to be determined*
	N2D	n/a	n/a	Eng. Cams	Nav Map Second Stereo Disparity – *to be determined*
	N2L	n/a	n/a	Eng. Cams	Nav Map Second Left Rectified – *to be determined*
	N2R	n/a	n/a	Eng. Cams	Nav Map Second Right Rectified – *to be determined*
	NVF	n/a	n/a	Eng. Cams	Nav Map VO Features – visual odometry feature list
	NMS	n/a	n/a	Eng. Cams	Nav Map Memory Manager Status – ASCII dump of NAV memory allocation
	NDC	n/a	n/a	Eng. Cams	Nav Map D-star Cost – D-star cost map
	NPC	n/a	n/a	Eng. Cams	Nav Map D-star Planning Cost – *to be determined*
	NDR	n/a	n/a	Eng. Cams	Nav Map D-star Layer Cost – *to be determined*
	NDF	n/a	n/a	Eng. Cams	Nav Map D-star Field – D-star field map
	NDL	n/a	n/a	Eng. Cams	Nav Map D-star Look Ahead – D-star Lookahead buffer
	NKO	n/a	n/a	Eng. Cams	Nav Map Keepout – keepout zones

Type	Product Identifier	Size	Format (bits)	Instruments	Description
	NKS	n/a	n/a	Eng. Cams	Nav Map Keepout Site – *to be determined*
	NKP	n/a	n/a	Eng. Cams	Nav Map Keepout Path Site – *to be determined*
	NMI	n/a	n/a	Eng. Cams	Nav Map IDPH – *to be determined*
	NSG	n/a	n/a	Eng. Cams	Nav Map Step Goodness – *to be determined*
	NTG	n/a	n/a	Eng. Cams	Nav Map Tilt Goodness – *to be determined*
	NRG	n/a	n/a	Eng. Cams	Nav Map Roughness Goodness – *to be determined*
	NGK	n/a	n/a	Eng. Cams	Nav Map Good Keep – *to be determined*
	NSP	n/a	n/a	Eng. Cams	Nav Map Stereo Points – *to be determined*
	NNE	n/a	n/a	Eng. Cams	Nav Map Num Entries – *to be determined*

4.3.1 Engineering Camera Instrument Suite

The binary content of the Engineering Camera image EDR data product is a copy of the scene that had been projected onto the camera instrument's charge-coupled device (CCD) and shifted into the CCD memory buffer, as shown in Figure 2.1 previously. The image data will be decoded and decompressed in single frame form as the image EDR. The Full Frame form of a standard image EDR has the maximum dimensions of 1024 lines by 1024 samples.

For the Engineering Camera image EDR, there are two possible radiometric formats of telemetered image data. In the first case, 12-bit data scaled onboard to 8-bit via a "12 to 8-bit" Lookup Table (LUT) or, by bit shifting, will be downlinked as 8-bit data and stored "as is" on the ground in a single unsigned byte. In the second case, 12-bit data without onboard LUT scaling or bit shifting will be downlinked as 12-bit data and stored "as is" in the 12 lowest bits of the signed 16-bit integer. The binary data may be returned in uncompressed or compressed form. Data returned as compressed are ICER [Ref 31] or LOCO encoded and will be decompressed as part of the EDR processing.

4.3.2 ChemCam Instrument Suite

4.3.2.1 RMI Image EDR

The binary content of the RMI image EDR product is a copy of the scene that had been projected onto the camera instrument's CCD onboard and read out from the CCD memory buffer, as shown in Figure 2.2.1.1 previously. The Full Frame form of a standard image EDR has the maximum dimensions of 1024 lines by 1024 samples. The image data, telemetered down at 10 bits/pixel, will be decoded in single frame form and stored in the 10 lowest bits of a signed 16-bit integer as the image EDR. Like the Engineering Camera instruments, data returned as compressed are ICER or LOCO encoded and will be decompressed as part of the EDR processing.

4.3.2.2 LIBS Spectrum EDR

The binary content of the LIBS spectrum EDR is comprised of intensity values, or data numbers (DNs). Depending on which of the three variants of spectroscopy activity were commanded, the values fall somewhere within the stored 16-bit or 32-bit data range. Unlike the RMI image data, LIBS spectrum data are only returned as uncompressed.

4.3.2.3 State-of-Health (SOH) EDR

The binary content of the SOH EDR is comprised of metadata DPOs only. For the ChemCam Body Unit, the metadata are stored as a vector of 9 entities in 16-bit unsigned integer format. For the ChemCam Mast Unit, the metadata are stored as a vector of 39 entities in 16-bit unsigned integer format.

4.3.3 MMM Camera Instrument Suite

The data from the MMM camera suite can come down in one of four different ways: color or grayscale JPEG, losslessly compressed, or uncompressed. These methods apply across all of the image EDR types (standard image, Z-stack, depth map, video frame) and geometry types (full-frame, subframe, thumbnail).

4.3.3.1 Bayer Pattern

The MMM cameras create images of 1648x1200 pixels. Each pixel on the CCD has an individual red (R), green (G), or blue (B) filter arranged in a Bayer pattern, which consists of repeated squares of pixels:

(**R**,**G**)
(**G**,**B**)

This allows an RGB color picture to be taken in a single exposure. Additional geology filters are available, which interact with the Bayer filters in complex ways. See Section 2.3 for an overview or the MMM camera documentation for details.

Of the 1648 pixels per line, only 1608 are photoactive pixels. The line structure is shown in Figure 4.3.3.1 below, and is broken down into the following:

- 2 dark pixels from the end of the **p**revious line (“P” in Figure 4.3.3.1)
- 1 invalid **A**DC pipeline pixel from the interline time (“A” in Figure 4.3.3.1)
- 4 **i**solation pixels (“I” in Figure 4.3.3.1)
- 16 **d**ark pixels (“D” in Figure 4.3.3.1)
- 1608 photoactive RGB pixels (“R”, “G”, “B” in Figure 4.3.3.1)
- 17 dark, isolation, and overscan pixels (“X” in Figure 4.3.3.1)

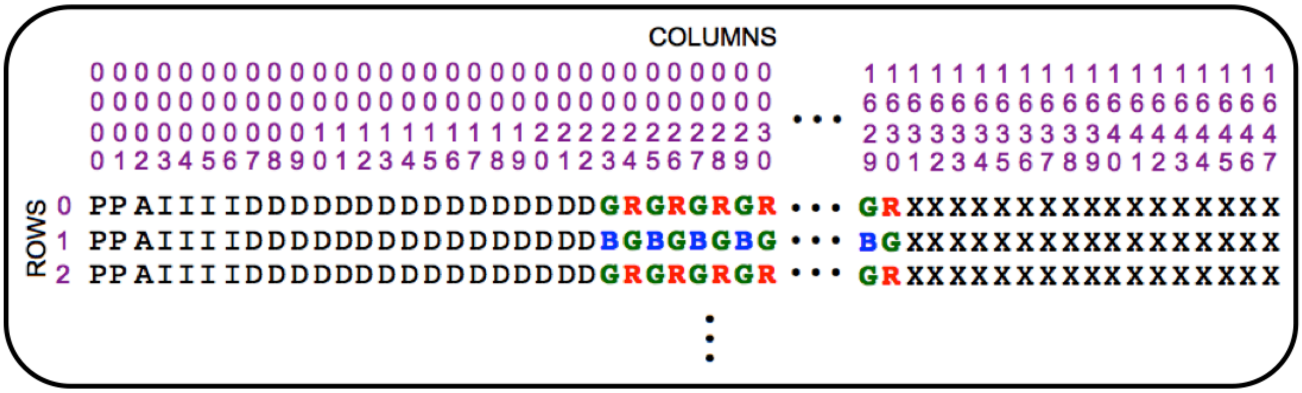


Figure 4.3.3.1 - MMM Camera RGB in Bayer Pattern Layout on CCD

Because there are an odd number of starting dark pixels, the first valid pixel (23) on even lines will be G (GRGRGR...) while on odd lines it will be B (BGBGBG...).

4.4 EDR Product Structure

As in the previous section “EDR Product Format”, description of EDR product structures in this section will be by instrument suite: a) Engineering Cameras, b) ChemCam Instruments, c) MMM Instruments. This section details the specifics of a variety of EDR structures, taking into account the concept of product labels and the product’s binary content described previously in Sections 3.2 and 3.3, respectively.

Figure 4.4 shows EDR products in this SIS with four possible structures: 1) an image EDR that has a VICAR label wrapped by an ODL label, Diagram A, 2) an image EDR identical in structure to #1 but with additional “auxiliary” data (discussed in Section 4.4.2), shown in Diagram B, 3) a spectrum EDR that has an ODL label followed by “auxiliary” data, shown in Diagram C, and 4) a state-of-health EDR that has only an ODL label, shown in Diagram D. All are shown with detached ASCII PDS-compliant labels for archive purposes.

For a description of the PDS and ODL labels, see Section 3.2.2, and for a description of the VICAR Label, see Section 3.2.3, and for a mapping between PDS/ODL and VICAR, see Section 3.2.4.

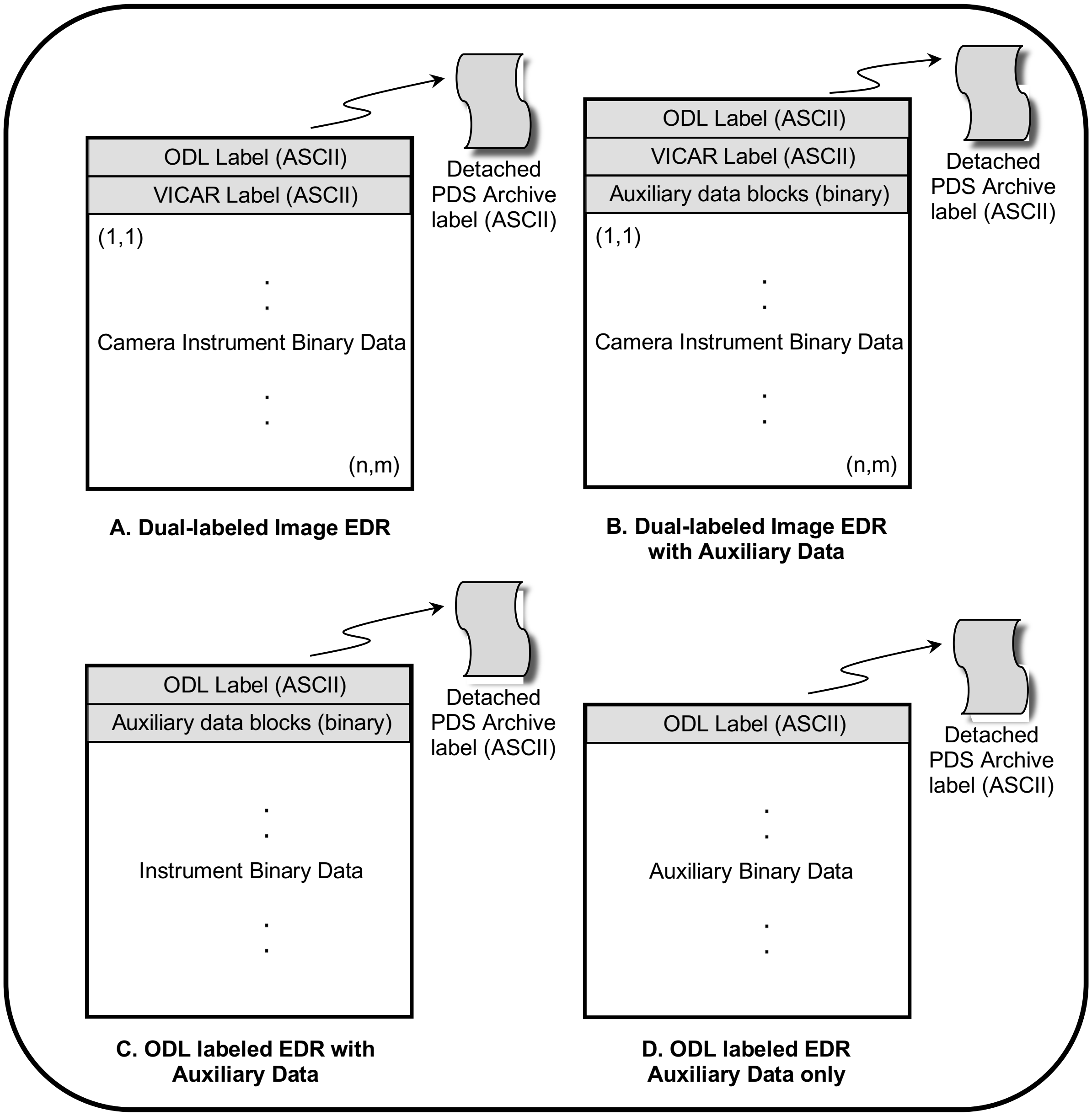


Figure 4.4 - EDR Product Structures

4.4.1 Engineering Camera Instrument Suite

All image EDRs for the Engineering Camera instrument suite have the same structure, containing the four attributes shown in Diagram A of Figure 4.4. They are listed below. The first three attributes comprise the image file and consist of an integral number of "records", where the record size is the size needed to contain one line of image data. The fourth attribute is an external (detached) label file for archive purposes:

1) An operations label in ODL (ASCII) format
2) An operations label in VICAR (ASCII) format
3) An n x m block of binary image data with the origin at the upper left pixel in line (row) 1, sample (column) 1. Note that some camera EDR products will be rotated so that the origin (1,1) is not the same as the CCD origin.

4) A detached PDS-compliant label, in a separate file. Its availability to users will not coincide with the delivery of the EDR to the ODS file system; it will be delivered later with the release of the Archive Volume by PDS.

4.4.2 ChemCam Instrument Suite

The RMI image EDR, LIBS spectrum EDR and state-of-health (SOH) EDR are common in one aspect: they each include "auxiliary" data that augments the binary instrument data and other metadata parts of the EDR. Here, the term "auxiliary" refers to a set of one or more binary Data Product Objects (DPOs) that are the components of the binary ".dat" file generated by MPCS (refer to Section 4.1). They contain instrument data (image or spectrum information) and/or metadata (commands, temperature readouts, etc.). Within the ChemCam EDR, these DPOs are stored in the "binary header" section of the file in the format provided by the spacecraft, following all labels but preceding the binary instrument content. The size and number of DPOs in any ChemCam EDR can vary, depending on the commanding of onboard data acquisition for that EDR. Each DPO starts on a record boundary and is padded to the record length if necessary. At minimum, all ChemCam EDRs contain the DPO type "Ancillary" and multiple instances of the DPO type "State-of-Health" (SOH), which also provide sources for some of the product label metadata items. Refer to Section 3.2.2.7 for a description of the label pointers that provide the necessary offsets to parse the DPOs in the EDR.

Because RMI image EDRs are VICAR formatted files, they differ in the way they get constructed in comparison to LIBS and SOH EDRs. So it follows that their respective labels, necessary in describing the embedded DPOs, are constructed differently. In building the RMI image EDR, program MSLEDRGEN reads the RMI ".emd" metadata file to identify each DPO that will be extracted from the RMI ".dat" file. Each DPO is padded to the width of the VICAR image during its insertion into the EDR as a separate binary line before the image. To accommodate the VICAR file format, the RMI label is minimal in content descriptive of the DPOs. In lieu of containing the detailed DPO describing objects, the RMI EDR label contains pointers that reference external ".FMT" files containing such objects. This differs from the LIBS and SOH EDR labels, which generally contain the detailed DPO describing objects in the body of the label. As an example, the DPO type "SOH" embedded in the RMI EDR requires three descriptive objects to appropriately define it. In the LIBS EDR, those three objects are contained in the EDR label. However, the RMI EDR label has a pointer to an external ".FMT" file that contains the three SOH DPO describing objects.

4.4.2.1 RMI Image EDR

The image EDR structure for the RMI camera consists of five parts. The first four of these are part of the image file itself; the fifth is a detached PDS label. Each of the four parts of the image file consists of an integral number of "records", where the record size is the size needed to contain one line of image data.

So structurally, the EDR matches Diagram B in Figure 4.4 with the following design:

1) An operations label in ODL (ASCII) format
2) An operations label in VICAR (ASCII) format
3) Zero or more records of binary header data, which contain auxiliary information in the form of DPOs previously described in Section 4.4.2.
4) An n x m block of binary image data with the origin at the upper left pixel in line (row) 1, sample (column) 1. Note that some camera EDR products will be rotated so that the origin (1,1) is not the same as the CCD origin.
5) A detached PDS-compliant label, in a separate file. Its availability to users will not coincide with the delivery of the EDR to the ODS file system; it will be delivered later with the release of the Archive Volume by PDS.

4.4.2.2 LIBS Spectrum EDR

The spectrum EDR structure for the LIBS instrument is similar to the RMI image EDR structure, minus the VICAR label. So, the EDR consists of four parts. The first three of these are part of the spectrum (one or more spectra) file itself; the fourth is a detached PDS label. Each of the three parts of the spectrum file consists of an integral number of "records", where the record size is the size needed to contain one spectrum.

Therefore, the EDR matches Diagram C in Figure 4.4 with the following design:

1) An operations label in ODL (ASCII) format
2) Zero or more records of binary header data, which contain auxiliary information in the form of DPOs previously described in Section 4.4.2.
3) A bit stream of binary spectra data.
4) A detached PDS-compliant label, in a separate file. Its availability to users will not coincide with the delivery of the EDR to the ODS file system; it will be delivered later with the release of the Archive Volume by PDS.

4.4.2.3 State-of-Health (SOH) EDR

The SOH EDR structure for the ChemCam instrument suite contains only binary auxiliary data, containing no instrument or binary header data.

So, the EDR consists of three parts and matches Diagram D in Figure 4.4 with the following design:

1) An operations label in ODL (ASCII) format
2) Zero or more records of binary auxiliary information in the form of DPOs previously described in Section 4.4.2.
3) A detached PDS-compliant label, in a separate file. Its availability to users will not coincide with the delivery of the EDR to the ODS file system; it will be delivered later with the release of the Archive Volume by PDS.

4.4.3 MMM Camera Suite

The EDR structure for the suite of MMM Cameras is the same as that described for the Engineering Cameras in Section 4.4.1.

Specifically, the MMM Camera image EDR matches Diagram A in Figure 4.4 with the following design:

1) An operations label in ODL (ASCII) format
2) An operations label in VICAR (ASCII) format
3) An n x m block of binary image data with the origin at the upper left pixel in line (row) 1, sample (column) 1. Note that some camera EDR products will be rotated so that the origin (1,1) is not the same as the CCD origin.

4.5 EDR Product Validation

Validation of the MSL EDRs will fall into two primary categories: automated and manual. Automated validation will be performed on every EDR product produced for the mission. Manual validation will only be performed on a subset.

Automated validation will be performed as a part of the archiving process and will be done simultaneously with the archive volume validation. Validation operations performed will include such things as verification that the checksum in the label matches a calculated checksum for the data product (i.e., that the data product included in the archive is identical to that produced by the real-time process), a validation of the PDS syntax of the label, a check of the label values against the database and against the index tables included on the archive volume, and checks for internal consistency of the label items. The latter include such things as verifying that the product creation date is later than the earth received time, and comparing the geometry pointing information with the specified target. As problems are discovered and/or new possibilities identified for automated verification, they will be added to the validation procedure.

Manual validation of the images will be performed both as spot-checking of data through-out the life of the mission, and comprehensive validation of a sub-set of the data (for example, a couple of days' worth of data). These products will be viewed by a human being. Validation in this case will include inspection of the image or other data object for errors (like missing lines) not specified in the label parameters, verification that the target shown / apparent geometry matches that specified in the labels, verification that the product is viewable using the specified software tools, and a general check for any problems that might not have been anticipated in the automated validation procedure.

5. RDR PRODUCT SPECIFICATION

RDR data products described in this document will be generated by MIPL personnel using the Mars Suite of VICAR image processing software within OPGS at JPL and the ChemCam Science Team using team-developed software at Los Alamos National Laboratory (LANL) in New Mexico, Institut de Recherche en Astrophysique et Planétologie (IRAP) and Centre National d'Etudes spatiales (CNES) in France. The RDRs produced will be “processed” data. The input will be one or more image or spectrum EDR or RDR data products and the output will be formatted according to this SIS. Additional metadata may be added by the software to the product label.

There may be multiple versions of image and spectrum RDRs. The RDR data product will be placed into FEI for distribution.

5.1 RDR General Processing

Processing is different for each image RDR type, as described in this section. In general, each RDR process inherits the metadata from its source EDR/RDR, modifying a portion of the metadata as necessary to reflect the subsequent output RDR.

5.1.1 Image RDRs

Image RDR data products described in this section will be generated by the EDR/RDR data product pipeline operating under the OPGS at JPL.

For Engineering Camera and MMM camera data, and to a lesser degree ChemCam RMI data, RDRs will be processed using the Mars Suite subset of the VICAR image processing software developed at MIPL. The various image RDR interfaces in the VICAR processing flow are shown in Figure 5.1.1, and their descriptions are described subsequently in Section 5.2.

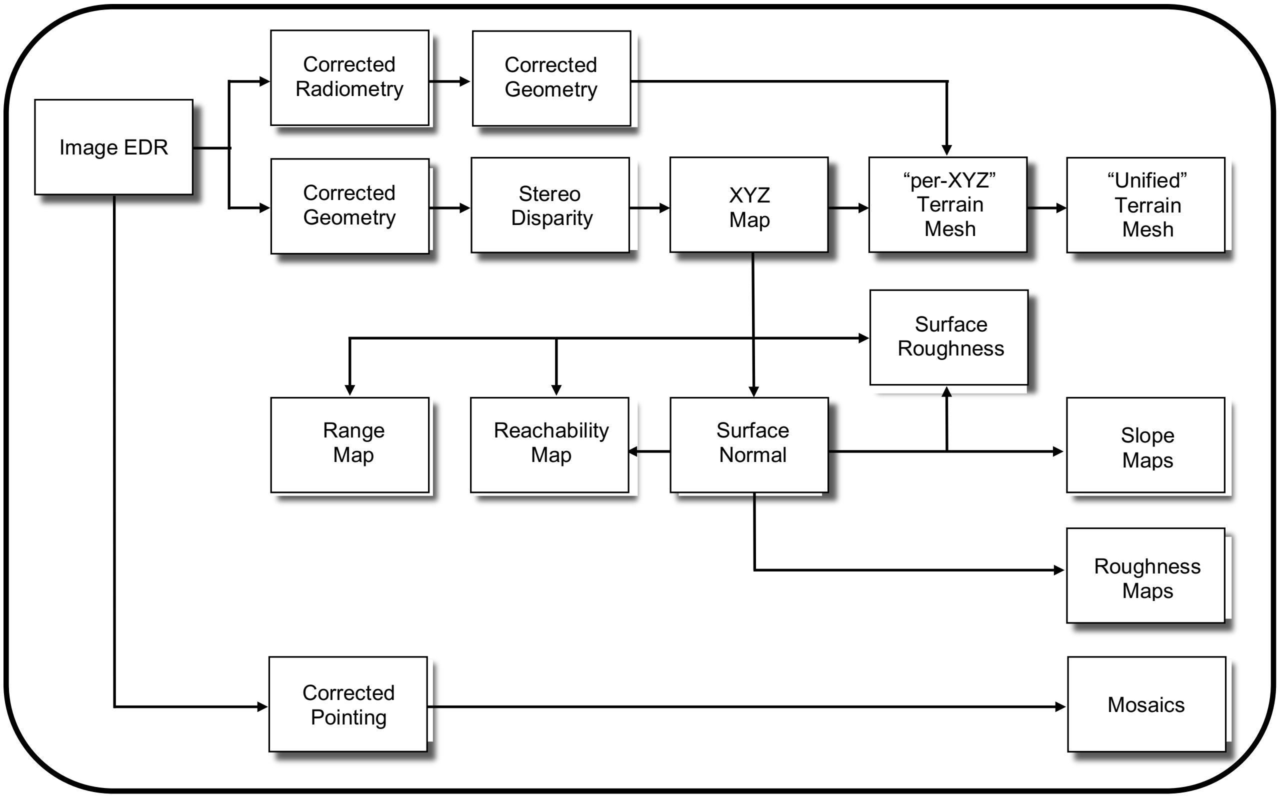


Figure 5.1.1 - Image RDR Processing Flow

5.1.1.1 Common Processing

Although the various Engineering Camera and MMM camera RDR product types are specific in nature, certain types of processing are often common to different RDR base files. These are described below.

5.1.1.1.1 Geometrically Corrected Images (Linearization)

EDRs and single-frame RDRs are described by a camera model. This model, represented by a set of vectors and numbers, permit a point in space to be traced into the image plane, and vice-versa. Linearization mode is indicated by the "geom" flag in the filename.

EDR camera models are derived by acquiring images of calibration targets with known geometry at a fixed azimuth/elevation. The vectors representing the model are derived from analysis of this imagery. These vectors are then translated and rotated based on the actual pointing of the camera to represent the conditions of each specific image. The resulting vectors make up the "camera model" for the EDR.

The Navcam, ChemCam, and MMM cameras use a CAHVOR model, while the Hazcams use a more general CAHVORE model. Both model types are nonlinear and involve some complex calculations to transform line/sample points in the image plane to XYZ positions in the scene. To simplify this, the images are "warped", or reprojected, in a process often called "linearization", such that they can be described by a linear CAHV model. See Figure 5.1.1.1.1 for a visual comparison between a Front Hazcam image EDR (left) and the Front Hazcam "linearized" image RDR (right).

Figure 5.1.1.1.1 - Image EDR (left) vs. "Linearized" Image RDR (right)

This linearization process has several benefits:

1) It removes geometric distortions inherent in the camera instruments, with the result that straight lines in the scene are straight in the image.
2) It aligns the images for stereo viewing, known as epipolar alignment. Matching points are on the same image line in both left and right images, and both left and right models point in the same direction.
3) It facilitates correlation, allowing the use of 1-D correlators in some cases.
4) It simplifies the math involved in using the camera model.

However, it also introduces some artifacts in terms of scale change and/or omitted data (see the references).

The linearized CAHV camera model is derived from the EDR's camera model by considering both the left and right eye models and constructing a pair of matched linear CAHV models that conform to the above criteria. For details on this algorithm see the references.

There are two types of linearization, indicated in the filename. For the nominal case, each image is linearized using a virtual camera model constructed to indicate what the other eye's model would look like at the same pointing. This allows each image to be processed independently, without the need to find the stereo partner (or even acquire the image), yet provides the same results as if the match had been performed.

However, this virtual partner is not always appropriate. In some cases, the stereo pair is composed of images acquired at different pointings (often needed for very close-range mast work in order to achieve sufficient overlap), different positions (for an arm-based camera such as MAHLI), different rover locations (long-baseline stereo, used to resolve distances far greater than a normal stereo pair can do), different conditions (such as different focus or zoom on the Mastcam), or even different instruments. These cases require linearizing with the actual stereo partner.

Regardless of linearization type, the image is then projected, or warped, from the CAHVOR/CAHVORE model to the CAHV model. This involves projecting each pixel through the EDR camera model into space, intersecting it with a surface (which matters only for the CAHVORE-based Hazcams and is a 1m radius sphere centered on the camera), and projecting the pixel back through the CAHV model into the output image.

C - The 3D position of the entrance pupil.
A - A unit vector normal to the image plane pointing outward (towards C).
H - A vector pointing roughly rightward in the image; it is a composite of the orientation of the CCD rows, the horizontal scale, the horizontal center.
V - A vector pointing roughly downward in the image; it is a composite of the orientation of the CCD columns, the vertical scale, the vertical center, and A.

If P is a point in the scene then the corresponding image locations x and y can be computed from:

$$x = \frac{(P - C)\,H}{(P - C)\,A} \qquad y = \frac{(P - C)\,V}{(P - C)\,A}$$

For details on the camera model math and calibration and more description of the CAHV-model family, see references [Ref 18] through [Ref 27]. Note that the X and Y positions above are 0-based coordinates; i.e., the coordinate (0,0) is the center of the upper-left pixel. This is different than the common PDS convention of 1-based coordinates, where (1,1) is the center of the upper-left pixel, used elsewhere in this document.

5.1.1.1.2 Image Overlays

Many image RDR types represent some quantity other than intensity of light, such as XYZ or slope. The value at each pixel indicates the measurement of the quantity at the corresponding point in the original image. These types can be overlaid on a background of the EDR or other intensity image, using color-coding to represent the RDR value. See Figures 5.2.1.4 (XYZ overlay), 5.2.1.5 (Range overlay) and 5.2.1.9.1 (Arm Reachability overlay) for examples. This gives a visualization of the RDR in context of the scene. Overlays are generally indicated by a product type ending with "O" in the filename.

5.1.1.1.3 Filled Images

Many RDR types do not achieve full coverage, e.g. the correlator is unable to find a solution at every point or the XYZ point failed the various filters. These "holes" are preserved in the RDRs using some value to indicate no solution (see the MISSING_CONSTANT labels). For ops work, it is critical to know where the holes are, so they are preserved in all nominal RDRs. Some RDRs created for purposes other than operations have these holes filled in using an interpolation mechanism. These are referred to as "filled" RDRs. Filled RDRs are generally indicated by a product type ending with "F" in the filename (but not all trailing "F"s mean Filled).

5.1.1.1.4 Color Processed Images

The MMM cameras routinely produce color images via their Bayer-pattern CCD's. The de-Bayering is typically done onboard but can also be done on the ground if not. In any case, RDRs that contain image data, such as ILT, RAD, RAS, etc., are produced by pulling the three bands apart into separate files, processing each independently, and then combining them into color again. Both the color image and one or more of the individual bands are kept as end products, as specified in the Config flag (3rd character) in the filename.

RDRs related to image geometry (e.g. disparity, XYZ, all downstream products) are created using only the Green band extracted from the color image. The Green band is used because there are two green pixels per Bayer cell, as opposed to one for red and blue, leading to a higher resolution image.

Note that not all MMM images are color; in particular some of the geology filters for the Mastcam produce single-band images, which are processed like any other single-band image.

5.1.1.1.5 Rover Volume Exclusion Image Masks

For the purposes of Terrain Mesh RDR generation, OPGS will create “Rover Volume Exclusion Mask” files that can be applied to several types of RDR (primarily XYZ). They are used to filter out rover features from generated terrain products, as well as the horizon and other undesirable features. They are single-band, byte files corresponding to the source image, where 255 indicates the corresponding pixel should be removed, or 0 indicates the pixel should remain in the output. These mask files typically have an “M” as the first character of the product type.

5.1.1.1.6 Masked Images

The exclusion masks, or other masks, can be combined with the RDR to create a Masked image. The contents are identical to the source RDR (most often, XYZ) except that where the mask is 255, the value is set to the value specified by MISSING_CONSTANT. Masked files typically have an “M” as the third character of the product type (exception: masked ARM is called ARK).

5.1.1.1.7 EDR-like RDRs

The LIN and BAY RDRs are exactly like EDRs except they have had linearization or de-Bayer processing done to them. This processing can be done to any image and does not change the product type code; the config, geom, and/or samp fields change instead. The LIN and BAY product type codes exist simply to ensure that the “EDR” product type code is not used for any RDRs.

5.1.2 Spectroscopy RDRs

Spectroscopy data products derived from the ChemCam LIBS spectrum EDR will be RDRs generated by software tools used and developed by ChemCam instrument team members at LANL. To generate a LIBS spectroscopy RDR product, the spectral data are extracted from the binary EDR file and converted to an ASCII format. Multivariate analyses are performed on the spectral data, and elemental abundances are reported using partial least squares regression (PLS).

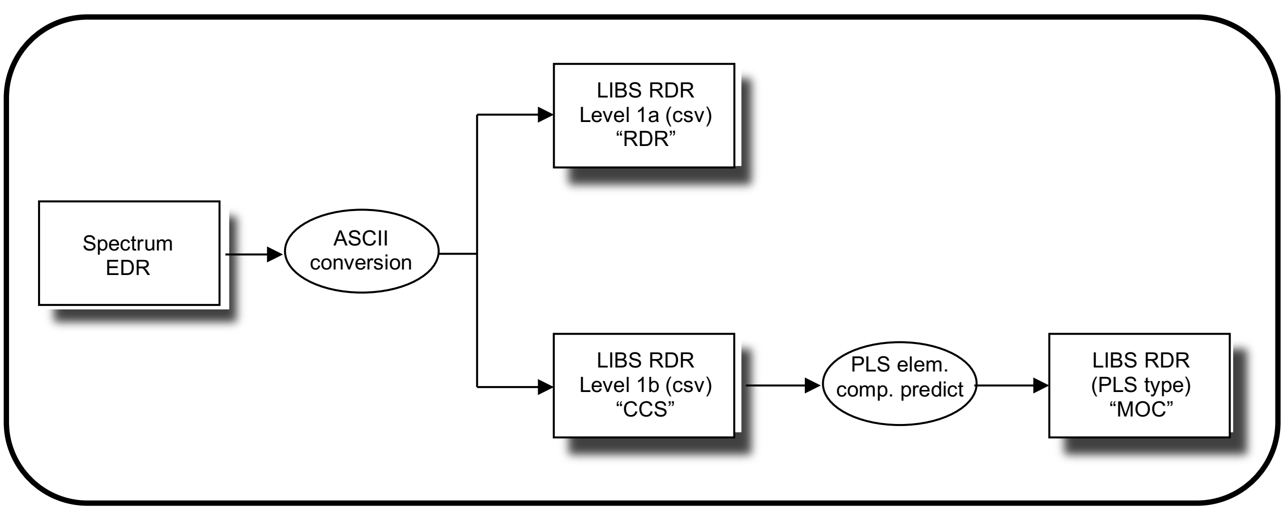


Figure 5.1.2 - Spectroscopy (LIBS) RDR Processing Flow

5.2 RDR Product Types

Descriptions for the various RDR product types are provided in this section. Refer also to Tables 5.4.1 and 5.4.2 for a concise list of the RDR product types.

5.2.1 Image RDRs

5.2.1.1 "Inverse LUT" RDR

The ILT RDR is produced by OPGS to reconstruct the original 12-bit pixels generated by the camera sensors. If the EDR is already in 12-bit format, ILT is simply a copy. However, if the EDR is in "8-bit" format (see Section 4.4) as a result of onboard "12 to 8-bit" scaling using a Lookup Table (LUT) or bit shifting, then an Inverse LUT (ILUT) is to be used to rescale the 8 lowest bits to the 12 lowest bits in the 16-bit signed integer.

The ILC type has had additional processing to remove certain instrument artifacts. At launch for MSL, this type is not used. However, operational experience with MER shows that over time, cameras degrade, with noise appearing due to radiation exposure and other effects. ILC is thus a placeholder for processing such as despiking, which removes this kind of noise, or interpolation of dead pixels.

The ILP type represents an ILT or ILC that has been pointing-corrected (see the Mosaic section) or re-localized based on ground analysis. It is expected that a special processing flag will be set for any products employing this type.

5.2.1.2 "Radiometrically Corrected" RDR

There are 3 different kinds of radiometrically corrected products. "RA*" have been corrected to absolute radiance units of W/m^2/nm/steradian. "RI*" products have been corrected for instrument effects only, and are in units of DN. "IO*" products are radiance factor (I/F) and are dimensionless. Within each kind, the values may be represented as floating-point values, or scaled to integers for ease of manipulation.

Independent of type, there are multiple methods of performing radiometric correction, distinguished by the RADIOMETRIC_CORRECTION_TYPE keyword. The two described here are CHEMRAD and MIPLRAD.

5.2.1.2.1 CHEMRAD Method

This refers to partial radiometric correction of ChemCam RMI image data performed using an image processing toolkit developed in IDL by ChemCam Science Team members at IRAP and CNES in France. It is deemed "partial" in the sense that the raw image data are not calibrated to physical radiance units. Although the calibration portion of radiometric correction is not performed for the input EDR, other corrections are applied. These corrections are a function of the image geometry, exposure time (milliseconds), sensor temperature (deg Celsius), and target distance (meters). The latter three are extractable from the SOH portion of the EDR file.

The main IDL procedure is RMI_PROCESSING.pro, which can apply successive corrections to RMI EDR image defects by incorporating, in a specific order, several lookup tables (LUTs) constructed during ground calibrations. The processing flow is shown in Figure 5.2.1.2.1.

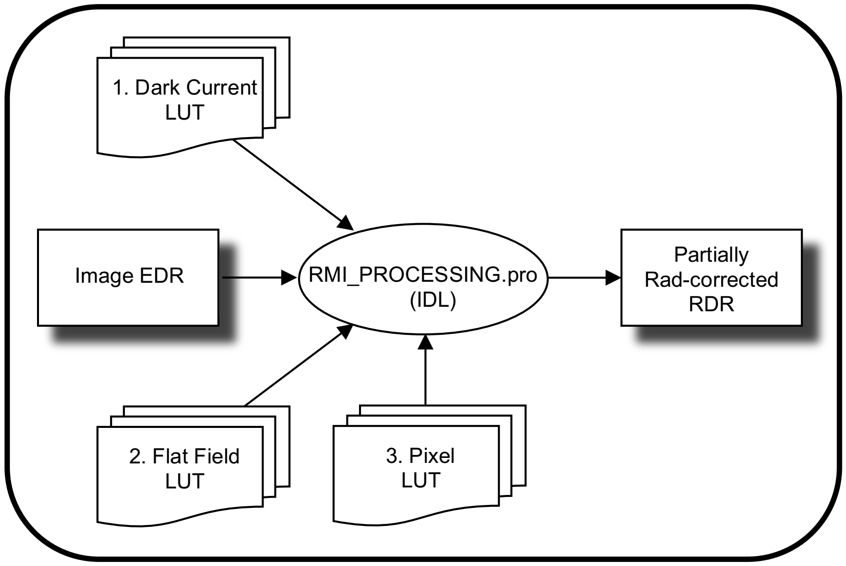


Figure 5.2.1.2.1 - Partial Radiometric Correction for ChemCam RMI

During the ground calibrations, several lookup tables were built with the constant values necessary to compute the corrections; they are included with the RMI_PROCESSING.pro code. These lookup tables will need to be revised with in-flight calibrations at a frequency to be defined.

The code applies corrective image processing steps in the reverse order of the apparition of the defects in the image, since one effect can influence the subsequent ones. In order, the corrections are as follows:

1. Subtract the estimated dark current image (temperature, time). This step incorporates a dark current LUT.
2. Iteratively (row by row) subtract the smearing drag (time) due to the CCD.
3. Iteratively (row by row) subtract the ghost image that is linked to the LIBS telescope and optics.
4. Divide by the estimated flat field that is linked to the LIBS telescope and optics. This step incorporates a flat field LUT.
5. Correct for known bad pixels by replacing them with median of neighboring pixels. This step incorporates a pixel LUT.
6. Apply a mask corresponding to the footprint of the telescope on the useful part of the CCD detector.
7. Enhance the contrast of the corrected image using histogram equalization.

The resulting RDR product is only partially radiometrically corrected and is not calibrated into physical radiance units. Note that the default flat field LUT is derived from a sky observation on Sol 32. Data acquired previous to Sol 32 were reprocessed using the Sol 32 flat field LUT.

Effects not accounted for in the correction process include:

a) Non-linearity - Each pixel of the CCD is a potential well, in which electrons are trapped. The number of trapped electrons is an affine function of the lighting, but above a certain threshold the linearity is broken (saturation). At room temperatures, at gain 12, the non-linearity starts around 700 counts, and saturation is reached around 850 counts, i.e. an integration of about 10 ms. This response function of the detector is a complex function of the temperature and can vary from pixel to pixel. It has not been calibrated since the primary objective of the RMI is to give a qualitative context of the LIBS experiment.

b) Direct Light - Some light from the far field can reach the detector without passing by the telescope mirrors, despite the baffle on the secondary mirror. It can be seen as extra light in the bottom part of some images, and sometimes as the focused picture of the distant scene. The diffuse light is taken into account in the determination of the reference flat field, and thus rather well corrected, but the focused distant scene cannot be corrected. However, this rare and faint, except in the corners that are removed when the telescope mask is applied.

5.2.1.2.2 MIPLRAD Method

This refers to radiometric correction of any camera instrument data systematically performed by MIPL (OPGS at JPL) to meet tactical time constraints imposed by rover planners. The exception is ChemCam RMI data, which were processed only using the CHEMRAD method described in the previous section. The resulting rad-corrected RDRs are integrated into terrain mesh products used for traverse planning. This method is typically less precise than the methods used by the science teams.

MIPLRAD is a first-order correction only and should be considered approximate. MIPLRAD first backs out any onboard flat field that was performed. It then applies the following corrections: flat field, exposure time, and temperature-compensated responsivity. The result is calibrated to physical units of W/m^2/nm/sr. The actual algorithm and equations used for MIPLRAD are shown below. Not every step applies to each camera type. Each correction is applied in sequence, to every pixel.

- Note for Navcam B-side flat field correction:
  In early 2018, MSL OPGS began using a new flat field image to perform this correction, replacing those obtained prior to launch (“V01” and “V02”). This new version, dubbed “V03”, is derived from a series of in-flight Navcam sky observations between sols 1171-1261 [courtesy of M. Lemmon, Texas A&M], and better represents the current condition of the Navcam optics. Navcam RDRs that use the V03 flat show improved removal of lens artifacts in single-frame images, reduced or eliminated brightness seams in mosaics, and increased coverage in stereo XYZ products (compared to those flattened with V01 or V02).

  The team is working with the PDS Archive to coordinate the re-release of all B-side Navcam single-frame RDRs that have been re-processed using the in-flight sky flats (date TBD). The flat field used for a given Navcam RDR can be found in the label keyword “FLAT_FIELD_FILE_NAME”, under Derived Image Parms. See Errata documentation in the Navcam volume of the PDS Archive for more details.

1. For the Engineering Camera instrument suite only, if on-board flat-fielding has been applied, it is backed out according to the parameters in FLAT_FIELD_CORRECTION_PARM, which is described in Appendix F and defines ff(x,y).

$$\mathrm{output}(x, y) = \mathrm{input}(x, y) * \mathrm{ff}(x, y)$$

2. To apply the flat-field correction on the ground, the x and y coordinates are adjusted based on downsampling and subframing to find the corresponding pixel in the flat field, and then the DN is divided by the flat field value (see note above for Navcam B-side):

$$\mathrm{output}(x, y) = \mathrm{input}(x, y)\ /\ \mathrm{flat\_field}(x', y')$$

3. Exposure time is then removed. Exposure time comes from EXPOSURE_DURATION, converted to seconds:

$$\mathrm{output(x,y)} = \mathrm{input(x,y)}\ /\ \mathrm{exposure\_time}$$

4. The temperature responsivity is removed next. The temperature of the CCD is determined from INSTRUMENT_TEMPERATURE using the following rules, where the first valid temperature found (0.0 is ignored as no-reading, and >50C is considered broken) is the one used:

   a) Use the CCD temp of said camera
   b) Use the CCD temp of left/right partner
   c) Use the CCD temp of alternate A/B side same-eye camera
   d) Use the CCD temp of alternate A/B side left/right partner
   e) Use the CCD temp of "similar" camera (other Hazcam, other mast-mount) in a-d order
   f) Use the CCD temp of any camera
   g) Use electronics temp instead of CCD in a-f order

   The temperature is combined with parameters R0, R1, and R2, which were derived from ground calibration and come from the flat field parameter file (see Appendix F) according to the following formula:

$$\mathrm{output(x,y)} = \mathrm{input(x,y)} * \left(\mathrm{R0} + \mathrm{R1*temp} + \mathrm{R2*temp^2}\right)$$

5. Finally, the result is (optionally) converted to integers using the RADIANCE_OFFSET and RADIANCE_SCALING_FACTOR keywords:

$$\mathrm{output(x,y)} = \left(\mathrm{input(x,y)}\ \text{-}\ \mathrm{RADIANCE\_OFFSET}\right)/\mathrm{RADIANCE\_SCALING\_FACTOR}\ + 0.5$$

Note that the engineering cameras were not well-calibrated radiometrically. Specifically, only flat fields were obtained, not temperature coefficients. Since they are build-to-print copies of the MER engineering cameras, the MER temperature responsivity parameters are used. For MMM, no temperature compensation is applied by the MIPLRAD method.

5.2.1.3 "Disparity" RDR

A Disparity file contains 2 bands of 32-bit floating point numbers in the Band Sequential order (line, sample). Alternatively, line and sample may be stored in separate single-band files.

The parallax, or difference measured in pixels, between an object location in two individual images (typically the left and right images of a stereo pair) is called the "disparity". Disparity files contain these disparity values in both the line and sample dimension for each pixel in the reference image. This reference image is traditionally the left image of a stereo pair, but could also be the right image. The geometry of the Disparity image is the same as the geometry of the reference image. This means that for any pixel in the reference image the disparity of the viewed point can be obtained from the same pixel location in the Disparity image.

There are three types of disparity. In the primary type (DS*), the values in the Disparity image are the 1-based coordinates of the corresponding point in the non-reference image. Thus, the coordinates in the reference image are the same as the *coordinates* in the Disparity image, and the matching coordinates in the stereo partner image are the *values* in the Disparity image. Disparity values of 0.0 indicate no valid disparity exists, for example due to lack of overlap, correlation failure, or parallax occlusion. This value is reflected in the MISSING_CONSTANT keyword.

Disparity images can also be "delta" disparity (DD*), which measures the relative offset between coordinates in the two images. This is what most imaging scientists mean by disparity. These products are not produced in the nominal pipeline, but can be produced as special products. Missing values are flagged by the value specified in MISSING_CONSTANT, typically 0.0.

The third type is a "first-stage" disparity (DF*). This is a by-product of the two-stage MIPL correlation procedure [Ref 30] and represents the intermediate step between stages. It contains coordinate values, as in the primary disparity.

For each of the three primary types, several kinds of file can be produced: normal, line, sample, raw, error metric, and grid. Not all kinds apply to all types.

Normal disparity files contain 2 bands of floating-point numbers in (line, sample) order using the Band Sequential format. The line and sample components may be stored in separate single-band files.

Raw files contain the results before doing left->right and right->left reconciliation. They should be considered an intermediate result.

Disparity error metric files contain information about the quality of the correlation match. The tools to create them have not yet been developed; they are listed here as a placeholder for future expansion. Thus, the format has not yet been determined as of this writing.

Grid overlays are an aid to visualization that may be created on occasion. These files are single-band byte images showing how a regular grid is distorted by the disparity matches (which is itself an indication of the terrain).

5.2.1.3.1 Stereo Pair Matching Method

Inherent in the designed operation of the stereo cameras is time-synchronization in the acquisition of left and right images intended for stereo processing. So, the SCLK timestamps of the respective left and right image acquired as a stereo pair onboard will be used to automatically identify them as a stereo image pair during nominal ground data processing.

Occasionally, stereo pairs will need to be processed that were not acquired simultaneously. This could be due to a sequence error, or for special operations such as re-pointing the mast between frames or driving the rover (long-baseline stereo). These off-nominal stereo pairs will be identified manually during ops, and processed as necessary. The special processing flag will be used to identify these. Additionally, most will use the "Actual" linearization method rather than "Nominal".

5.2.1.4 "XYZ" RDR

An XYZ file contains 3 bands of 32-bit floating point numbers in the Band Sequential order. Alternatively, X, Y and Z may be stored in separate single-band files as an X Component RDR, Y Component RDR and Z Component RDR, respectively. The single component RDRs are implicitly the same as the XYZ file, which is described below. XYZ locations in all coordinate frames for MSL are expressed in meters unless otherwise noted.

The pixels in an XYZ image are coordinates in 3-D space of the corresponding pixel in the reference image. This reference image is traditionally the left image of a stereo pair, but could be the right image as well. The geometry of the XYZ image is the same as the geometry of the reference image. This means that for any pixel in the reference image the 3-D position of the viewed point can be obtained from the same pixel location in the XYZ image. The 3-D points can be referenced to any of the MSL coordinate systems (specified by DERIVED_IMAGE_PARMS Group in the PDS label).

Most XYZ images will contain "holes", or pixels for which no XYZ value exists. These are caused by many factors such as differences in overlap, correlation failures, and the failure of a result to meet quality checks in the XYZ program. Realize this list of factors is only representative, and not exhaustive. Holes are indicated by X, Y, and Z all having the same specific value. This value is defined by the MISSING_CONSTANT keyword in the IMAGE object. For the XYZ RDR, this value is (0.0,0.0,0.0), meaning that all three bands must be zero (if only one or two bands are zero, that does not indicate missing data). Note that “0.0,0.0,0.0” is technically a legal value, but could occur at most once in an image and will rarely occur at all. The value is based on legacy software from previous missions and is compatible with current mission software. Additionally, it is extremely unlikely that the value will conflict with actual data since it is between the rover's wheels in Rover Frame (which cannot be imaged in stereo except for heroic MAHLI efforts); even when the Site origin is visible (e.g. the rover moves away), the possibility of sampling an exact value of “0.0,0.0,0.0” is considered extremely low. Also, if the value were to be sampled as actual data, losing a single pixel in the image is not problematic. The file format does not support nulls, so some other sentinel value would have to be chosen instead.

An XYZ Error metric (XYE) is available, which gives the estimated error for each pixel. It is a 3-band float product, with the three bands indicating the estimated range error (in meters) along each of the X, Y and Z axes. These values together define the error ellipsoid. Note that these values are axis-aligned, while the error is naturally range-aligned. Therefore, the Range Error (RNE) product will generally be more accurate. XYE is provided as a convenience and its ellipsoid will always completely contain the RNE ellipsoid.

XYZ files can be filled (XYF). Individual X,Y,Z files can be filled as well. They can have associated rover mask files (MXY) as well as becoming masked (XYM) and saved as an overlay (XYO for XYZ, ZZO for Z-band), as shown in Figure 5.2.1.4 below. The rover mask files are discussed in more detail in Section 5.2.1.4.1.

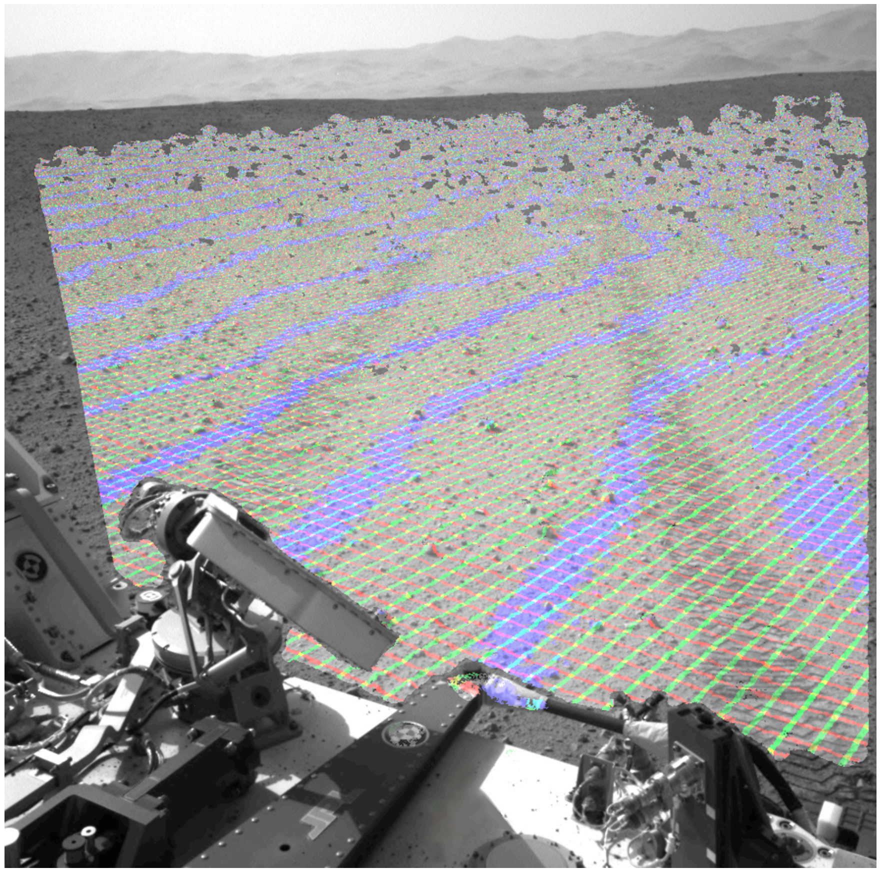


Figure 5.2.1.4 - XYZ Data Masked and Overlaid onto Image EDR

5.2.1.4.1 Rover Mask RDR

The MXY (mask for XYZ) file is a special kind of mask file, called a rover mask. This file is intended to mask off the rover, so that XYZ points correlated on the rover do not show up in terrains. It is created using a low-fidelity volumetric model of the rover (with some margin around the rover components), which is articulated based on the telemetered joint positions of the arm, wheels, and suspension. This model is then projected into the image to create a mask. The articulation means the mask will reflect the arm and wheel positions, minimizing the amount of good terrain that is masked off. If joint positions are not available, a "swept volume" is used, meaning the mask covers all possible positions of the articulating device. The HGA is always modeled as a swept volume.

Rover masks can also be generated using predicted arm joint positions; this is the basis of the "ChemCam finder" mosaics produced during operations. Such predicted-state masks are marked using the special processing flag field in the filename.

Associated with the MXY mask image is an XML file with the same base name. This file contains a polygonal representation of the articulated low-resolution volumetric model that is used to make the rover mask (before it is projected into the image). The format of this file is not described completely here, but it is straightforward XML, consisting of a large number of polygons in XYZ space.

5.2.1.5 "Range" RDR

A Range (distance) file contains 1 band of 32-bit floating-point numbers. The pixels in a Range image represent Cartesian distances from a reference point (defined by the RANGE_ORIGIN_VECTOR keyword in the PDS label) to the XYZ position of each pixel (see XYZ RDR). This reference point is normally the camera position as defined by the C point of the camera model. A Range image is derived from an XYZ image and shares the same pixel geometry and XYZ coordinate system. As with XYZ images, range images can contain holes, defined by MISSING_CONSTANT. For MSL, this value is 0.0.

The Range Error metric (RNE) gives the estimated error of the XYZ point in meters for each pixel. Like XYE, it is a 3-band float product, whose three bands define the error ellipsoid (in meters). However, for RNE the bands are interpreted differently, in a way that more naturally represents the underlying error mechanism. The first band is the error in the downrange direction - radially away from the camera. This is the primary error for any stereo ranging system. The other two bands contain the cross-range error, orthogonal to each other and to the downrange vector direction. Band 2 (first crossrange) is coplanar with the camera stereo baseline and as such can generally be thought of as the "horizontal" crossrange direction, with band 3 the vertical, for common stereo geometries. Range products can become masked (RNM), as well as filled (RNF) and overlaid (RNO), as shown in Figure 5.2.1.5 below.

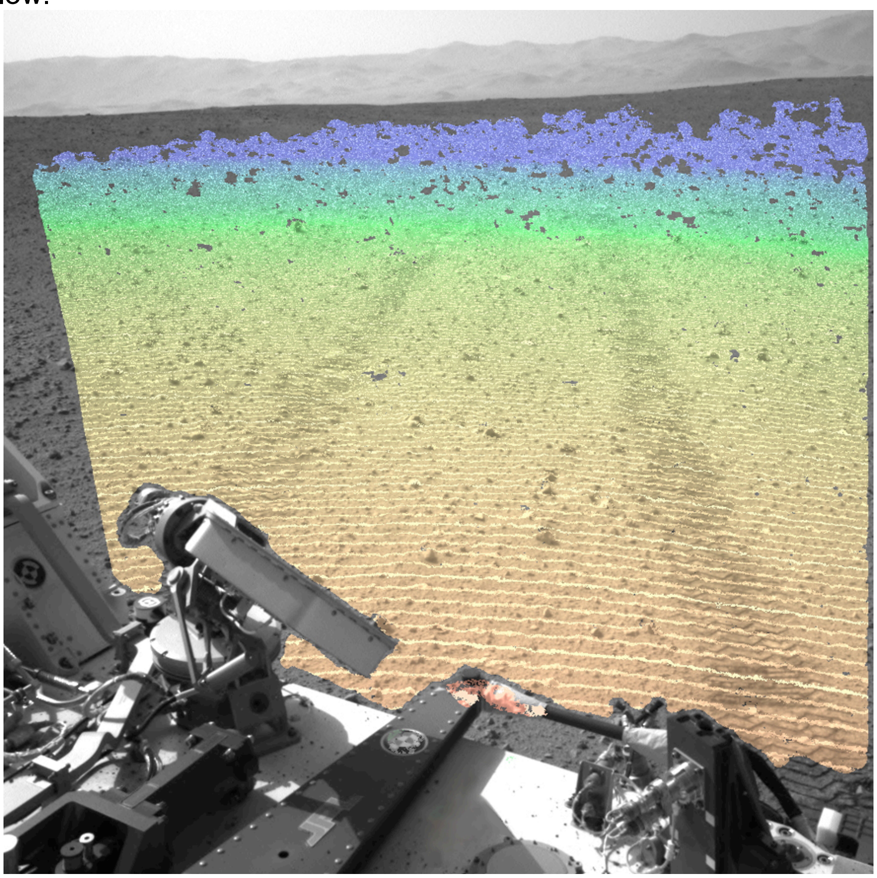


Figure 5.2.1.5 - Range (Distance) Data Overlaid onto Image EDR

5.2.1.6 "Surface Normal" (UVW) RDR

A Surface Normal (UVW) file contains 3 bands of 32-bit floating point numbers in the Band Sequential order. Alternatively, U, V and W may be stored in separate single-band files as a U Component RDR, V Component RDR and W Component RDR, respectively. The single component RDRs are implicitly the same as the UVW file, which is described below.

The pixels in a UVW image correspond to the pixels in an XYZ file, with the same image geometry. However, the pixels are interpreted as a unit vector representing the normal to the surface at the point represented by the pixel. U contains the X component of the vector, V the Y component, and W the Z component. The vector is defined to point out of the surface (e.g. upwards for a flat ground). The unit vector can be referenced to any of the MSL coordinate systems (specified by the DERIVED_IMAGE_PARAMS Group in the PDS label).

Most UVW images will contain "holes", or pixels for which no UVW value exists. These are caused by many factors such as differences in overlap, correlation failures, and insufficient neighbors to compute a surface normal. Holes are indicated by U, V, and W all having the same specific value, defined by MISSING_CONSTANT as (0.0,0.0,0.0).

Two special kinds of surface normal products are defined based on MER experience. These are not expected to be used during nominal MSL ops, but are included in case they become needed. The UVP type projects the surface normal onto an arbitrary plane in space, so the unit vectors always lie parallel to the plane. The UVT type provides the angle between the surface normal and the same arbitrary plane in a single band.

The UVW product contains the surface normal resulting from analyzing a small patch of pixels, which is appropriate for arm work. In contrast, the UVS product contains the surface normal resulting from analyzing a much larger patch of pixels (comparable to the size of the rover), which is appropriate for driving slope determination. Specific patch sizes are operational tuning parameters that have not yet been set, but will be provided in the history portion of the embedded VICAR label.

Surface normals can be filled (UVF) and overlaid (UVO). Separate U,V,W files can also be filled.

5.2.1.7 “Surface Roughness” RDR

The roughness maps, RUD (for the Drill) and RUT (for the Dust Removal Tool (DRT)) contain surface roughness estimates at each pixel in the image, along with a "goodness" flag indicating whether the roughness meets certain criteria.

For each pixel, the surface normal defines a reference plane. XYZ pixels in the area of interest are gathered, and the distance to the plane is computed. Outliers are thrown out. For the remainders, the minimum and maximum distances from the plane are found. Roughness is defined as the distance between this min and max (thus, is peak-to-peak variation within the area along the normal vector).

Two roughnesses are potentially computed. The first is an overall measurement containing all points within a radius of the central pixel. This is used for the DRT, and for the drill body. The second, used only for the drill, contains points within a ring between two radii. This is used for the drill stabilizer bars, and is not used for the DRT.

In each case, the computed roughnesses are compared to thresholds, which determine whether the point is "good" or not. The potential values of this goodness state are:

- 0.0 = No solution
- 1.0 = Both ring and overall roughnesses exceed thresholds
- 2.0 = Overall roughness (only) exceeds its threshold
- 3.0 = Ring roughness (only) exceeds its threshold
- 4.0 = Roughnesses within threshold (i.e. point is good)

Values of 1.0 and 3.0 appear only in the RUD product, since there is no ring for RUT.

The files are thus 2-band (RUT) or 3-band (RUD) float images, with the first band being the state, the second band being the overall roughness, and the third band being the ring roughness. For the second and third bands, 0.0 does not indicate a missing value (unlike most other products). Rather, 1.0 is used, as specified in MISSING_CONSTANT.

The default parameters and thresholds for these products are shown in Table 5.2.1.7 (all values in meters):

Table 5.2.1.7 - Surface Roughness Params and Thresholds

Param / Threshold	Product Type	
	RUD	RUT
Overall radius	0.075 m	0.035 m
Overall threshold	0.04 m	0.005 m
Ring inner radius	0.06 m	n/a
Ring outer radius	0.075 m	n/a
Ring threshold	0.015 m	n/a

5.2.1.8 Slope RDRs

The Slope-related RDRs represent aspects of the slope of the terrain as determined by stereo imaging. The Slope Map is derived from the UVS product, which contains the rover-sized surface normal at every point. There are several slope types, each of which can additionally be overlaid on a background.
In the equations below, *u*, *v*, and *w* are values from the UVS file, while *x*, *y*, and *z* are values from the XYZ file.

5.2.1.8.1 "Slope" (nominal) RDR

The SLP (SLO) type contains the slope in degrees for each pixel.

$$\text{slope} = \frac{180}{\pi}\left(\frac{\pi}{2} + \tan^{-1}\left(\frac{w}{\sqrt{u^2 + v^2}}\right)\right)$$

5.2.1.8.2 "Slope Rover Direction" RDR

The SRD (SRO) type contains the component of the slope (in degrees) that was facing the rover, i.e. if the rover went radially outward from its current position, this indicates the climb or descent. In the formula below, R is the rover's position.

$$\mathbf{V} = \frac{\left[x - \boldsymbol{R}_x,\ y - \boldsymbol{R}_y\right]}{\sqrt{(x - \boldsymbol{R}_x)^2 + \left(y - \boldsymbol{R}_y\right)^2}}$$

$$\mathbf{SRD} = -\frac{180}{\pi}\tan^{-1}\left(\frac{\mathbf{V}_x u + \mathbf{V}_y v}{-w}\right)$$

5.2.1.8.3 "Slope Heading" RDR

The SHD (SHO) type contains the direction of the slope as a clockwise angle from north, in degrees. Use the 4-quadrant form of arctangent to get a full 360-degree range.

$$\text{slope\_heading} = \frac{180}{\pi}\tan^{-1}\left(\frac{v}{u}\right)$$

5.2.1.8.4 "Slope Magnitude" RDR

The SMG (SMO) type contains the magnitude of the normal unit vector projected onto the horizontal plane. It is directly related to sin(slope).

$$\text{slope\_mag} = \sqrt{u^2 + v^2}$$

5.2.1.8.5 "Slope Northerly Tilt" RDR

The SNT (SNO) type contains the component of the slope in degrees that points north.

$$\text{northerly\_tilt} = \frac{180}{\pi} \sin^{-1}(u)$$

5.2.1.8.6 "Solar Energy" RDR

The SEP (SEO) type is included as a placeholder. It is not used for MSL.

5.2.1.9 Arm Reachability RDRs

The Arm Reachability maps contain information about whether or not the instruments on the Arm can "reach" (contact or image) the object or location represented by each pixel in the scene, and how hard they can push ("preload"). They are derived from the XYZ and Surface Normal (UVW) products.

The geometry of the reachability maps matches the reference XYZ, and Surface Normal (UVW) images, in that each pixel in the file directly corresponds to the pixel at the same location in the other products.

For the arm reachability map products, pixels with a DN of 0 denote an area where an arm instrument is unable to make contact (see Tables 5.2.1.9.1.1 and 5.2.1.9.1.2). To avoid ambiguity in the operational data store and archive, all arm reachability products are generated and archived, regardless of whether the reachability values are 0 or not. Consequently, there are arm reachability products comprised entirely of pixels with DN 0, that is, with no pixels indicating reachability. These products help document the decision process employed by the MSL project when identifying contact science targets.

5.2.1.9.1 "Arm Reachability" RDR

The reachability map (ARM) encodes information for each of the 5 arm instruments in each of the 8 possible arm configurations, for a total 40 values per pixel. It is stored as a 5-band image of 16-bit integers in standard Band Sequential order. Each band represents one of the 5 arm instruments in the order defined by INSTRUMENT_BAND_ID. Within each band, the 16-bit integer contains 2 bits of information for each of the 8 configurations (in the order defined by CONFIGURATION_BIT_ID) packed into the 16-bit value. The first mode in CONFIGURATION_BIT_ID is in the high-order 2 bits of the integer. The two bits represent 4 states: not reachable (0), and three levels of reachability, with 3 being the most easily reachable. Reachability is determined by checking if the contact, approach, and overdrive positions are reachable using the computed surface normal and 3 additional normals differing by a "tweak" angle from the computed one. Reachability level 3 indicates all checks pass, while 1 and 2 indicate that some did not and may require extra attention from the arm operator in order for the instrument to safely reach the point.

Table 5.2.1.9.1.1 - Arm Reachability Bit Assignments Per Configuration

16-bit Integer Bit Order (15=MSB, 0=LSB)							
15 - 14	**13 - 12**	**11 - 10**	**9 - 8**	**7 - 6**	**5 - 4**	**3 - 2**	**1 - 0**
Shoulder Out Elbow Up Wrist Up	Shoulder Out Elbow Up Wrist Down	Shoulder Out Elbow Down Wrist Up	Shoulder Out Elbow Down Wrist Down	Shoulder In Elbow Up Wrist Up	Shoulder In Elbow Up Wrist Down	Shoulder In Elbow Down Wrist Up	Shoulder In Elbow Down Wrist Down

Table 5.2.1.9.1.2 - Arm Reachability Values

Value		Description
Decimal	**Binary**	
0	00	Not reachable
1	01	Reachability quality 1
2	10	Reachability quality 2
3	11	Reachability quality 3 (best)

Note that reachability maps may include masks (MAR), be masked (ARK), and be overlaid (ARO) as shown in Figure 5.2.1.9.1 below.

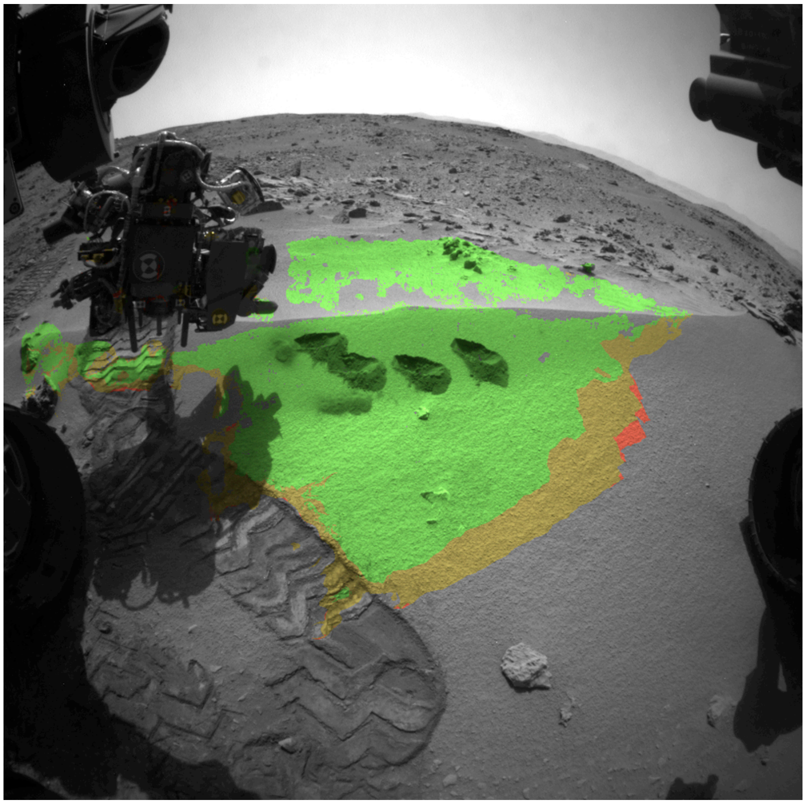


Figure 5.2.1.9.1 - Arm Reachability Qualities Color-coded, Overlaid onto Image EDR

5.2.1.9.2 "Arm Preload" RDR

The ARP type indicates the minimum and maximum preload values (in Newtons) that can be applied by the Drill instrument at the point represented by the pixel. This is a 2-band 16-bit signed integer product, where the bands represent (minimum, maximum) preloads (also defined by INSTRUMENT_BAND_ID).

5.2.1.10 Color RDRs

The MMM cameras contain a Bayer pattern of color filters on the CCD. This is a repeating pattern of 4 pixels where each “cell” contains one red, two green, and one blue pixel (see Figure 4.3.3.1). This allows acquisition of color without using the filter wheel. In order to be generally useful, the image must normally be “de-Bayered” in one of several ways. The method of de-Bayering is indicated by the “config” field in the product filename, with additional support from the “samp” field. Any time de-Bayering results in an image size change, the camera model must be adjusted to match.

For color images, the Bayer cells are typically extracted to separate color bands. This is indicated by values of “R”, “G”, or “B” for the “config” field. If the image is half-size in both dimensions as compared to the original (e.g. the RGB cells are simply extracted), then the “samp” field displays “B” to indicate Bayer subsampling. If the pixels are interpolated back to full size, the “samp” field is unchanged. Green pixels, being twice as numerous, present special challenges. For a full-size image, the “config” code is “G”. For Bayer-subsampled images, several cases are possible. If only the upper or lower green pixels are used from each cell, then the “config” values are “U” or “L”. If both are combined, the value is “G”. If both are present, so the image is twice as big in one dimension than the other, the value is “D”. Regardless of the extraction mode, if all three colors are merged back into a single 3-band image, the value is “F”, with the “samp” field indicating the size.

For non-color images taken using geology filters, there are two additional options. Individual cells can be extracted as described above. They can also be averaged in a 2x2 pattern (potentially taking into account responsivity of the filter) to create one pixel per Bayer cell. This is indicated with a “config” value of “A”. Or, each pixel can be corrected for the combined responsivity of the Bayer and geology filters; this is indicated with a value of “C”.

Note that most de-Bayering is done onboard. This is not reflected using the “config” code. 2x2 averaging or subsampling simply results in a downsampled or subsampled image, while JPEG creation results in a single 3-band EDR image. Thus, for all EDRs, the “config” field represents the filter number, not the Bayer state.

5.2.1.11 “Photometry” RDR

The IEP type contains incidence, emission, and phase angles for each pixel for use in photometry work. It is a 3-band float product derived from UVW in the order (incidence, emission, phase). It can also be filled (IEF).

5.2.1.12 Terrain Map RDRs

Terrain models are high level products that are derived using XYZ files and the corresponding image files. The XYZ files contain point clouds: sets of vertices in a specific coordinate system. The corresponding image files are used to obtain intensity or color information for each vertex in the point cloud. The terrain models are generated by triangulating point clouds using volume based surface extraction. The original image is used as a texture map to add detail and color to the polygonal surface representation. Terrain models are stored in Open Inventor binary format. Image textures are stored in SGI RGB format. Height maps (i.e., digital elevation maps, or DEMs) are also produced and used by the Rover Sequencing and Visualization Program (RSVP) for tasks which require simple and fast lookup such as rover settling. Height Maps are stored in VICAR format.

5.2.1.12.1 “Per-XYZ” Terrain RDRs

For every XYZ RDR created, the following terrain products are generated and follow the Single-frame RDR filename convention (see Section 6.1.1):

a) (*.tar) - A collection of tiles representing spatial subdivision of a point cloud. Each tile is a separate file within the tar-file. Each tile contains vertices that define terrain in multiple Level of Details (LOD). From the vertices, triangles are striped for rendering efficiency. Note that tar-files are not used for unified mesh creation and though self-contained, serve only as intermediate products.

b) (*.iv) - Open Inventor terrain representation of a point cloud defined in the XYZ RDR. It's a single file generated by combining all tiles contained in the tar-file described above and storing it as an Open Inventor binary file. It constitutes the per-XYZ mesh product.

c) (*.rgb) - Image file in SGI RGB format that is used as the Texture Map for the per-XYZ mesh product.

d) (*.ht) - Height Map (DEM) derived from the XYZ RDR, stored as an image file. It's in VICAR format but is not fully compliant to VICAR label specifications. It has 3 bands. Band 1 contains actual height data. Band 2 fills areas for which there is no actual data using interpolation. Band 3 provides metric of how close a pixel value is to the actual data.

Example:

Given the point cloud XYZ RDR named NLA_412403715XYZLF0060000NCAM15000M1.IMG and the corresponding image NLA_412403715RASLF0060000NCAM15000M1.IMG, the following files are created:

- NLA_412403715RASLF0060000NCAM15000M1.tar - Collection of vertices tiles in Open Inventor ASCII format representing spatial subdivision of the point cloud.
- NLA_412403715RASLF0060000NCAM15000M1.iv - Concatenation of all tiles into one Open Inventor file in binary format.
- NLA_412403715RASLF0060000NCAM15000M1.rgb - Texture Map image in SGI image format.
- NLA_412403715RASLF0060000NCAM15000M1.ht - Height Map image, with the following label items defining the spatial extent of the Height Map:

NL = 512	*(number of lines)*
NS = 442	*(number of samples)*
X_AXIS_MINIMUM = -39.906654	
Y_AXIS_MINIMUM = -8.1579	
MAP_SCALE = 0.087151	*(Resolution at which Height Map has been generated)*

5.2.1.12.2 "Unified" Terrain RDRs

Just as individual images can be combined into image mosaics, per-XYZ meshes can be combined into unified terrain meshes. These are the ultimate terrain products used by rover planners during tactical operations. Per-XYZ polygonal surfaces are generated using XYZ RDRs defined in a Site frame. The tool RSVP extracts Site information from the unified mesh product filename to render terrains into the proper locations for rover traverse and arm placement applications, as shown in Figures 5.2.1.12.2.1 and 5.2.1.12.2.2.

The terrain products listed below follow the unified mesh product filename convention (see Section 6.1.3), which differs from the Single-frame filenames carried by the per-XYZ terrain products:

a) (*.iv) - Open Inventor file in ASCII format that contains references to all individual binary per-XYZ Open Inventor files

b) (*.mod) - ASCII file that contains references to all corresponding individual per-XYZ ".ht" Height Maps files.

Example:

Assuming the generation of multiple per-XYZ mesh products described and exemplified in Section 5.2.1.12.1, the following unified mesh files are created:

- N_L0168_RASLF_006_0000_AUTOGENM1.iv - Collection of references to Open Inventor files including NLA_412403715RASLF0060000NCAM15000M1.iv
- N_L0168_RASLF_006_0000_AUTOGENM1.mod - Collection of references to Height Maps including N_L0168_RASLF_006_0000_AUTOGENM1.ht

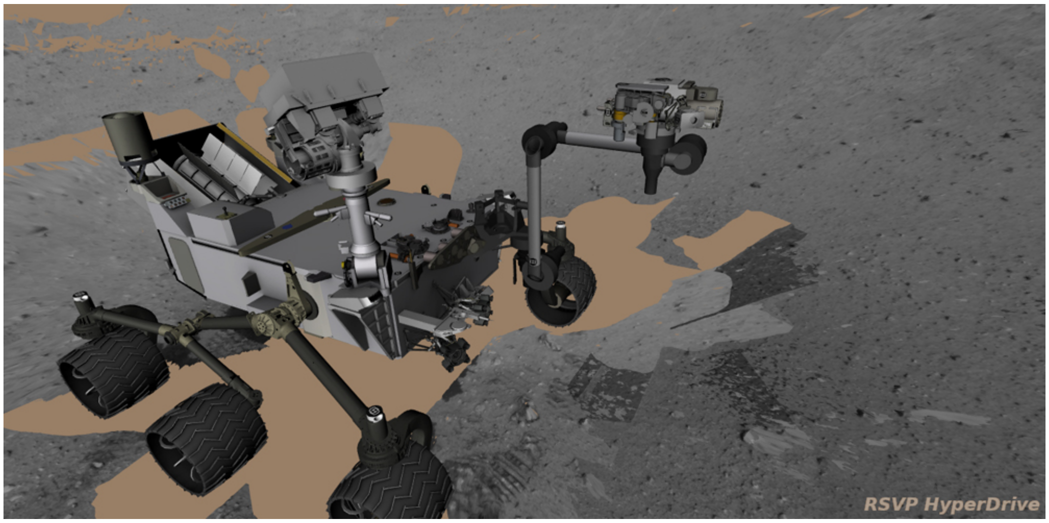


Figure 5.2.1.12.2.1 - Rover Location Rendered in Unified Terrain Mesh

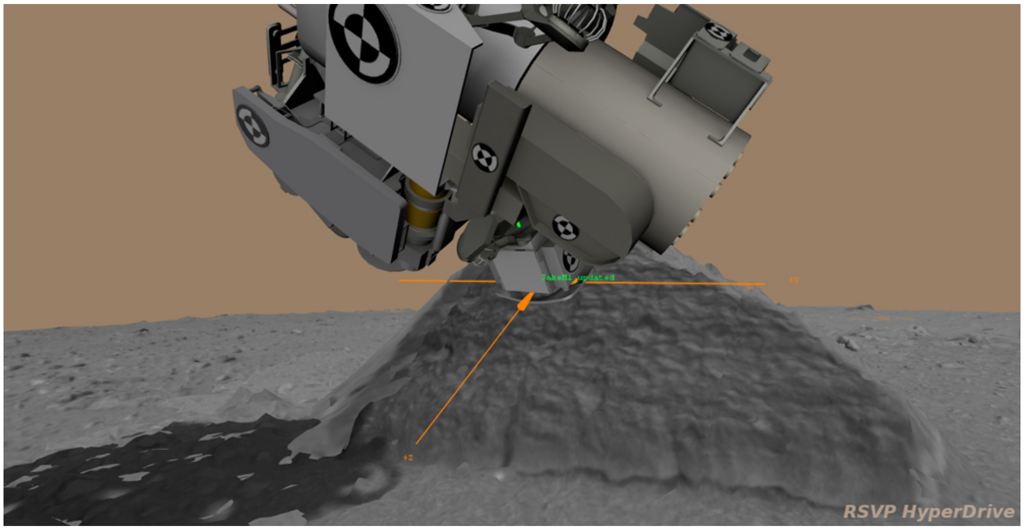


Figure 5.2.1.12.2.2 - Arm Payload Location Rendered in Unified Terrain Mesh

5.2.1.13 Mosaic RDRs

This section discusses the process of mosaicking multiple frames into a single RDR product using some projection. The text largely reflects the methods applied by MIPL under OPGS. It should be noted that governing methods and software can differ between OPGS and other operations subsystems or science instrument teams; the algorithms followed by other teams may not be the same as described here.

5.2.1.13.1 Overview of Mosaics in General

Mosaics can embody several important properties, making them very useful products. They assemble small pieces into a larger field of view. Certain projections create a level horizon, removing rover tilt. Mosaics can be calibrated so directions such as north and east can be determined - or they can be made relative to the rover to visualize forward and right. They can provide overhead views (Vertical, Polar, or Orthorectified projections) to help understand the local environment. They can be made from different types of data (such as slope). They can combine different filters to create color. Finally, they are the signature products for public outreach.

Mosaics can be assembled autonomously by tracing a view ray from each mosaic location or pixel into the scene, determining its intersection with a surface model (typically a ground plane), and then querying each input image to determine if that point lies within its field of view. In this fashion mosaics containing perhaps hundreds of images can be assembled for each spectral band. It may be necessary to refine the camera pointing in order to produce accurate mosaics. This requires the determination of the actual azimuth and elevation of each image in order to correct for errors such as gear backlash. One way to do this is to acquire tiepoints between all pairs of overlapping images. Camera pointing parameters are then estimated which cause the camera model to map the tiepoints to their correct locations. In many cases this can be accomplished automatically, but it often requires human intervention to select tiepoints because of small overlap, parallax, or changing lighting.

5.2.1.13.2 Mosaicking Method

The process used by the MIPL software to create mosaics is described below. It consists of several sub-steps. Conceptually, one can think of the process as adjusting the pointing of the inputs, projecting them down to a surface, and looking at the result from a different point of view (the output projection). In reality, the process is run in reverse for ease of interpolation (this is described below).

A. ***Pointing Correction*** - An optional (but important) first step in mosaic production is pointing correction. This is used to minimize geometric seams (discontinuities) between frames. The results of pointing correction are used in mosaics, but they can also be fed back into the RDRs (often at the ILUT stage), resulting in adjusted XYZ and other derived values and corrected meshes.

   There are several methods by which improved pointing of the cameras can be determined. The most common method is to pick tiepoints between image pairs, either automatically or with manual assistance, and use those in a global cost function minimization to determine the corrected pointing parameters. Another possibility is to analyze the shape of XYZ data in the overlap region, again using an error minimization process to derive updated pointing parameters. Pointing parameters can also be determined manually.

   Regardless of method, the result is encapsulated in a pointing correction or "nav" file. A more detailed description of this file is provided in Section 5.2.1.13.3, but fundamentally, this file contains, for each image being corrected, the original pointing parameters and the revised pointing parameters.

   Pointing parameters are simply those numbers which represent how the camera is pointing in the rover frame, reduced to available degrees of freedom. These are used as inputs to the kinematics procedures that derive the camera model. The set of pointing parameters, together with the kinematics algorithm, is referred to as a pointing model. The job of a pointing model is to take a calibration camera model and transform it using the pointing parameters to create a transformed camera model, which represents the specific image in question.

A given camera may have multiple available types of pointing models. The mast-mounted cameras have a standard model with two parameters: azimuth and elevation actuator angles. However, another model is available with three: azimuth, elevation, and "twist", which is a rotation around the camera's A axis (A being one of the CAHV camera model parameters). For MAHLI, there are two models: one with six parameters (the XYZ position plus the three Euler angles describing the orientation) and one with seven (XYZ plus the four components of a quaternion). The Hazcam and MARDI models have zero parameters, since they are rigidly attached to the rover body. Other pointing models may exist as well.

The set of available pointing model types and their full descriptions are outside the scope of this SIS; they are defined in a PDS documentation file.

For the mosaic process, the "nav" file is sufficient to describe the pointing parameters. However, if other corrected RDRs need to be produced, such as XYZs or meshes, then the pointing parameters must be stored in the label. This is accomplished via the POINTING_MODEL_NAME and POINTING_MODEL_PARAMS labels. When recomputing a camera model, if these labels are present they should be used in preference to the normal method of pointing via labels in the ARTICULATION_DEVICE_STATE groups. For most users, however, the GEOMETRIC_CAMERA_MODEL should be used directly; this will be updated properly with respect to the corrected pointing.

B. ***Output Projection Determination*** - The output projection is then determined. The parameters describing the projection are listed in Appendix A, and described in detail in Appendix F. The output projection parameters are determined by analysis of the inputs to give the "best" resulting mosaic, but can be overridden by the user. The determination process is outside the scope of this document; the results are what is important and they are in the label.

C. ***Surface Determination*** - A surface model is critical for mosaics. This is a mathematical surface, which approximates the actual scene. To the extent that the scene differs from the surface model, distortion and uncorrectable seams due to parallax can result.

Usually the surface model is a flat plane, with normal pointing upwards. This can be adjusted, however, to better match the scene. Regardless, the results are documented in the SURFACE_MODEL_PARMS group.

There are five potential surface models in the MIPL software: PLANE, INFINITY, SPHERE, SPHERE1, and SPHERE2. See SURFACE_MODEL_TYPE in Appendix F for description. Note that an appropriate surface model is often determined automatically as part of the pointing correction process, and the surface model can be stored in the "nav" file. Almost all mosaics are created using the PLANE model.

Note that the parameters (surface normal and ground point, for PLANE) for an appropriate surface model are often determined automatically as part of the pointing correction process, and the surface model can be stored in the "nav" file.

D. ***Computation of Output View Ray*** - For each pixel in the output mosaic, a view ray in 3-D space is constructed. How this view ray is constructed depends on the projection type. In this section, the pixel is at location (i,j) in 0-based coordinates, with i corresponding to sample and j to line. (0,0) is in the upper-left-hand corner. Capitalized values represent PDS label items from the SURFACE_PROJECTION_PARMS group. Unit and coordinate system conversions are applied as necessary but are not specified here. The coordinate system used is defined by REFERENCE_COORD_SYSTEM_* in SURFACE_PROJECTION_PARMS.

E. ***Projection from Output to Surface*** - Once the view ray is determined, it is projected out until it intersects with the surface model. The resulting point in XYZ space is used in the next step. If the ray does not intersect the surface, the point is assumed to be at infinity in the direction the view ray is pointing. Exception: as mentioned below, the Vertical projection will reverse the direction of its view ray; infinity is assumed only if they both miss.

   Note that the INFINITY surface model guarantees the ray will miss the surface at all times.

   The difference between the SPHERE1 and SPHERE2 models is that, if the ray intersects the spherical surface more than once, SPHERE1 will take the first intersection, while SPHERE2 will take the second. For normal rover situations, SPHERE1 thus roughly models a convex hill, while SPHERE2 roughly models a concave crater when the rover is outside the sphere.

F. ***Projection from Surface to Input*** - The XYZ location (or direction for the infinity case) is then back-projected into each input camera model in turn, using the corresponding input camera model. The first input for which the resulting pixel coordinate is inside the image (excluding border pixels which are thrown away) and non-0 stops the process; that is the image from which the output pixel value is taken. Values of 0 in the input image are ignored, with the effect that they are transparent.

   This process results in stacking the images such that the first one in the input list of images "wins". There is no feathering of overlaps; the first image is "on top" of all the others, and an image completely covered by preceding images will not be used at all.

G. ***Interpolation and Storage of the Result*** - Finally, a bilinear interpolation is performed on the input image, based on the 4 pixels surrounding the back-projected location. The result of this interpolation is the value of the output pixel.

   Bilinear interpolation is optional, but is normally done for image mosaics. Mosaics of other data types such as XYZ or Surface Normal (UVW) generally have interpolation turned off to avoid aliasing from interpolation with invalid pixels.

5.2.1.13.3 Mosaic Ancillary Files

A number of ancillary files are used to support mosaicking, and contain parameters and information describing how the mosaic was produced. With these, it is possible to maintain traceability and provenance for each pixel in a standard mosaic back to the source image. On PDS-released archive volumes, the ancillary files have the same basename as the mosaic to preserve one-to-one matching with the mosaic, albeit with a different extension. On the operational data store (ODS) for operations, they might not necessarily have the same mosaic basename, as several mosaics might share an ancillary file. In such cases, the ancillary file's name may indicate a different product type, projection, eye, filter/color, or geometric or brightness correction than the target mosaic's filename. The other filename fields should always match.

It should be noted that many if not most mosaics are produced at least partially by hand, which explains most of the inconsistencies noted below. The general case is described, but as with any hand work not all conforms exactly.

These files are not described completely here, but we hope the descriptions are sufficient to be able to decipher them:

a) **List files** - With a ".LIS" extension, list files are simple text files containing the names of the images making up the mosaic, one per line. The first name in the list references the image frame that is "on top" in the mosaic product, covering the image frames that are referenced in the list below it. The list files often contain full pathnames to disks on the operational data store (ODS) - directory paths that are not part of a PDS-released archive volume. However, the filenames themselves, minus directory paths, usually will be part of an archive volume. Occasional mosaics may have list files comprised of names for private copies of images (e.g. with "/home" in the pathname) that are the result of different scenarios of special processing: specified and unspecified. In the former case, such image files will be in an archive volume with filenames that carry a character flagging the type of special processing. In the latter case, no special character is present in the filename, though the file's name and/or metadata label will identify the ultimate source image of the unspecified processing. Note that a mosaic's list file content of filenames (minus directory paths) is also referenced in the mosaic's label using keyword INPUT_PRODUCT_ID.

b) **Nav files** - With a ".NAV" extension, nav files are XML files describing the pointing corrections that have been applied to images in a mosaic, as well as the surface model. The prologue contains identifying information. Note that the "static_parameters" file is listed as "MSL:MSL_pma.point" in some nav files; this should be corrected to "MSL:MSL_mast.point" and refers to the calibration file used to find mast kinematics parameters.

   Pointing correction works by applying a set of pointing parameters (e.g. mast azimuth and elevation) to a kinematics algorithm and using that to re-point the camera model. See definition of keyword POINTING_MODEL_NAME in Appendix F for more information.

   For each image (<solution> element), the Site and Drive components of the RMC are listed, followed by image identifier information. This identifier information contains the original pointing parameters, which allows the same correction to be applied to e.g. the left and right eyes (regardless of image ID information).

   Following that are the updated pointing parameters, and then the revised (re-pointed) camera model. In rare cases, nav files are edited by hand, which puts the accuracy of the camera model update at risk.

   At the end is usually a "surface_model" element describing the surface model determined by the MIPL software program MARSNAV (which creates the nav files). This information is repeated in the SURFACE_MODEL_PARMS group of the mosaic product's label.

c) **Tiepoint files** - With a ".TIE" or ".TPT" extension, tiepoint files are XML files containing image tiepoints used as input to program MARSNAV. These can be automatically or manually selected. The prologue relates each image ID to a key, which is used throughout the remainder of the file.

   Each tiepoint has a left and right key, and then 1-based coordinates in the corresponding files of the tiepoint. Just <left> and <right> should be used; <projected> has little value. In the flags, "quality" represents the quality of correlation match when the tiepoint was correlated (scale 0-1). The "interactive" flag has little archival value. It does not (as the name suggests) indicate whether the tiepoint was automatically selected or manually tweaked.

   Tiepoint type "0" is by far the most common and is a standard image tiepoint. Other types are rarely used and their descriptions are beyond the scope of this document. Full descriptions are in the help documentation for program MARSNAV.

d) **Brightness Correction files** - With a ".BRT" extension, brightness correction files are XML files containing information used to correct the brightness and contrast of images in a mosaic relative to one another. They are similar in concept and structure to nav files. After the prologue, each image has one <brt_solution>. The most common correction type, LINEAR, specifies an overall additive and multiplicative factor to be applied to each image (MULT is applied first, then ADD). These factors are echoed in the IMAGE_RADIANCE_FACTOR and IMAGE_RADIANCE_OFFSET keywords in the mosaic label. The HSI_LIN type is similar, except the correction (for color images) is done in Hue-Saturation-Intensity (HSI) space, with the correction applied to Intensity only. See also BRIGHTNESS_CORRECTION_TYPE.

e) **Brightness Overlap files** - With an ".OVR" extension, brightness overlap files are XML files containing information about image statistics in overlap areas, used to create the brightness correction files. They are similar in concept to tiepoint files, except the "tiepoints" are the mean and standard deviation of small areas of overlapping pixels for the mosaic.

   They start with a prologue defining the image ID-to-key mapping, as with nav files. Each overlap then has a number of images involved in the overlap, the number of pixels, and a "radius" which is a general description of the maximum size of the overlap. This is followed by the key, mean and standard deviation of the overlapping area in each image. The line and sample coordinates are provided for an arbitrary point in the overlap, just to help locate where the overlap is. The actual shape of the overlap is not specified.

   Overlap type "0" is a standard overlap. Type "1" gives the mean and standard deviation not of an overlap, but of the image as a whole (thus there is only one image). Type "2" is like type "0" but has mean and standard deviation in HSI space (intensity only). Type "3" is like type "1" but using HSI space. See the help documentation for MIPL software program MARSBRT for full details.

5.2.1.13.4 "Cylindrical Projection Mosaic" RDR

Cylindrical projections are the most common method for viewing non-stereo panoramas. The MIPL method for creating a Cylindrical projection involves computing the azimuth and elevation of the view ray, as follows:

$$\text{azimuth} = i/\text{MAP\_RESOLUTION} + \text{START\_AZIMUTH}$$
$$\text{elevation} = (\text{ZERO\_ELEVATION\_LINE} - j)/\text{MAP\_RESOLUTION}$$

The view ray emanates from the point PROJECTION_ORIGIN_VECTOR.

Figure 5.2.1.13.4 shows such a mosaic overlaid onto azimuth and elevation grid lines, with individual frame boundaries superimposed and annotated by number. In this case each pixel represents a fixed angle in azimuth and elevation. Rows are of constant elevation in the selected coordinate frame. In this case, a Site frame was used, so the horizon is level, and columns begin clockwise from Mars north.

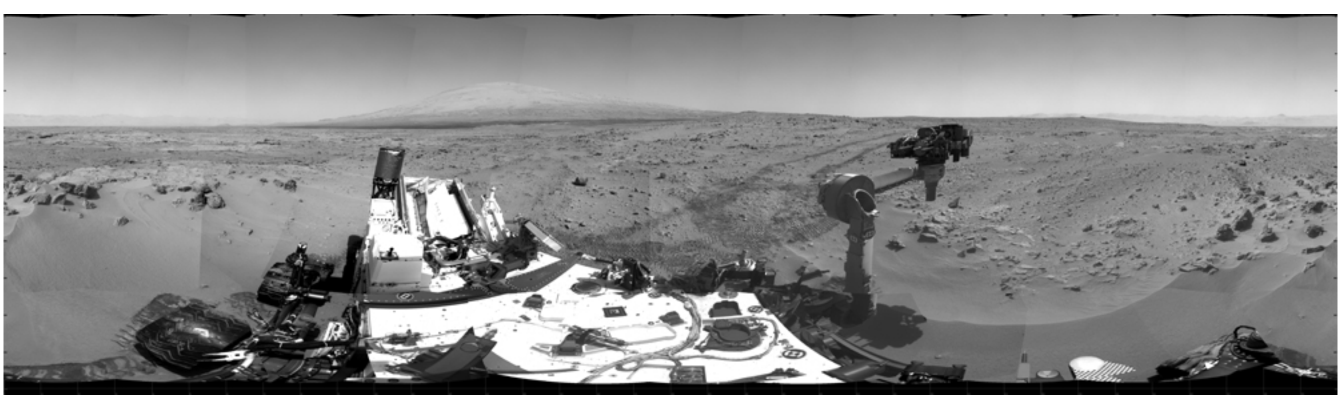


Figure 5.2.1.13.4 - Cylindrical Projection Mosaic

5.2.1.13.5 "Camera Point Perspective Mosaic" RDR

Figure 5.2.1.13.5 shows a Camera Point Perspective mosaic. It is a perspective projection with horizontal epipolar lines. The mosaic behaves as though the "camera" which acquired the image frames was an instrument with a much larger field of view. For MSL, this type of mosaic is typically in the Rover Frame and thus may have a tilted horizon if the rover is not level.

Point-perspective mosaics give the most natural view of small areas and are suitable for stereo viewing, but cannot be used for wide fields of view.

MIPL creates the Camera Point Perspective by using the output camera model (described by the GEOMETRIC_CAMERA_MODEL group in the output mosaic) to project the pixel into space. The origin of the view ray is thus the C point of the camera model, with the ray's direction being determined by the camera model. See Section 5.1.1.1.1 and references [Ref 18] through [Ref 27] for the mathematics.

Figure 5.2.1.13.5 - Camera Point Perspective Mosaic

5.2.1.13.6 "Cylindrical-Perspective Projection Mosaic" RDR

Cylindrical-Perspective mosaics are used for large stereo panoramas, and work across a full 360 degrees of azimuth. Stereo is preserved because a baseline separation is maintained between the camera eyes at different azimuths.

This projection is the most complicated projection to create. Each column i (counting from 0) in the output mosaic is assigned its own camera model. This is done in several steps:

1) Compute initial camera model. This model is a CAHV linearized model derived from the first input to the mosaic, re-pointed to azimuth 0 and elevation PROJECTION_ELEVATION. This model is stored in the GEOMETRIC_CAMERA_MODEL label group.

2) The instantaneous field of view of the "central" pixel (at the point where the A vector intersects the image plane) is computed using the formula:

$$\text{ifov} = \tan^{-1}\left(\frac{1.0}{\left|\left(\vec{\text{H}} - \vec{\text{A}} * \left(\vec{\text{H}} \bullet \vec{\text{A}}\right)\right)\right|}\right)$$

where the "•" indicates the scalar dot product of the two vectors A and H.

Alternatively, this can be derived from the image size and azimuthal extent (where the azimuths are adjusted by 360 degrees such that the result is minimally positive):

$$\text{ifov} = \frac{(\text{STOP\_AZIMUTH - START\_AZIMUTH})}{\text{LINE\_SAMPLES}}$$

3) The azimuth of the column is computed:

$$\text{azimuth} = \text{START\_AZIMUTH} + \text{i} * \text{ifov}$$

4) The initial camera model is re-pointed using kinematics as described above under the pointing correction section, using the above azimuth and PROJECTION_ELEVATION. This results in the final camera model for the column.

Step 4 is difficult to duplicate for reconstructing the set of camera models. For that reason, an alternate method is described in this paragraph. The resulting models are exact for mast-mounted cameras with no backlash correction; they are a close approximation for other cases. In general, for mast-mounted cameras, the C points of the column camera models describe a ring in space, whose diameter is approximately the baseline between the cameras. This ring is described by PROJECTION_ORIGIN_VECTOR (center), PROJECTION_AXIS_OFFSET (radius), and CAMERA_ROTATION_AXIS_VECTOR (orientation of the ring axis). These together simulate the kinematics motion of a mast-mounted camera in Rover frame. To compute the camera model for the azimuth defined in Step 3, take the camera model from the label, and rotate the entire camera model around the camera rotation axis by the azimuth amount, using the ring center as the pivot point. The C point will remain on the ring, while the camera pointing (close to but not identical to the A vector) will remain approximately tangent to the ring at that point. After this, compute the rotation required to transform CAMERA_ROTATION_AXIS_VECTOR into PROJECTION_Z_AXIS_VECTOR (which can be done by taking the cross product to get the rotation axis and the dot product to get the rotation amount). Then rotate the camera model by this amount, again using the ring center as the pivot point. This has the effect of tilting the entire ring so it is perpendicular to the PROJECTION_Z_AXIS_VECTOR. This last rotation is often used to remove the effect of rover tilt, resulting in a flat horizon with the camera model baselines (vector between the left and right eyes) aligned with the horizon (technically, perpendicular to the Z axis in Local Level frame). For this "untilt" case, the PROJECTION_Z_AXIS_VECTOR is the Local Level frame's Z axis expressed in Rover Nav frame. Note that PROJECTION_ELEVATION and PROJECTION_LINE are measured before this "untilt" rotation takes place, so they end up describing a sinusoid in the final mosaic when untilt is used.

Once the camera models have been defined, the mosaic proceeds through each pixel as with the other projections. The view ray is computed as described below (A, H, and V come from the column's camera model):

$$\text{x\_center} = \vec{\text{A}} \bullet \vec{\text{H}}$$
$$\text{y\_center} = \vec{\text{A}} \bullet \vec{\text{V}}$$
$$\text{samp} = \text{x\_center}$$

$$\text{line} = \text{y\_center} + \text{j} - \text{PROJECTION\_ELEVATION\_LINE}$$

where the "•" indicates the scalar dot product of two vectors. This (samp,line) coordinate is then projected into space using the column's camera model, and this projection becomes the view ray. The origin of the view ray is the column's C point. See Section 5.1.1.1.1 and references [Ref 18] through [Ref 27] for the mathematics of camera models.

Figure 5.2.1.13.6 shows a Cylindrical-Perspective projection in which a 360-degree view can be viewed in stereo. This is a perspective projection similar to Figure 5.2.1.13.5 except that the mosaic acts like a pinhole camera that follows the mosaic in azimuth while maintaining camera baseline separation. If the mosaic is generated with no tilt correction (i.e., CAMERA_ROTATION_AXIS_VECTOR and PROJECTION_Z_AXIS_VECTOR are the same) and the rover is tilted, the horizon will not be level, instead being sinusoidal. This preserves epipolar alignment and allows for better stereo viewing of the panorama. However, for aesthetic reasons, Cylindrical-Perspective mosaics are often created by "untilting" the rover as described above. In these cases, the horizon will be level, but stereo alignment may be compromised due to parallax effects in areas where the surface model does not closely match the actual surface. Additionally, the overall baseline between the cameras may be adjusted via the ring radius (PROJECTION_AXIS_OFFSET). This has the effect of enhancing or reducing the overall disparity, which can result in better stereo viewing in some cases. This baseline adjustment may create similar parallax effects in areas where the surface model does not match the actual surface.

Figure 5.2.1.13.6 - Cylindrical-Perspective Projection Mosaic

5.2.1.13.7 "Polar Projection Mosaic" RDR

Polar mosaics create a quasi-overhead view that still allows viewing all the way to the horizon.

MIPL creates the Polar projection by computing the azimuth and elevation of the view ray as follows:

$$x = i - \text{SAMPLE\_PROJECTION\_OFFSET}$$
$$y = \text{LINE\_PROJECTION\_OFFSET} - j$$
$$\text{range} = \sqrt{x^2 + y^2}$$
$$\text{elevation} = \text{range}/\text{MAP\_RESOLUTION} - 90 \text{ degrees}$$
$$\text{azimuth} = \text{REFERENCE\_AZIMUTH} + (90 \text{ degrees} - \tan^{-1}(y, x))$$

The view ray emanates from the point PROJECTION_ORIGIN_VECTOR.

Figure 5.2.1.13.7 shows a Polar projection. Concentric circles represent constant projected elevation. Mars nadir is at the convergent center and the horizon is corrected for lander tilt. North is up.

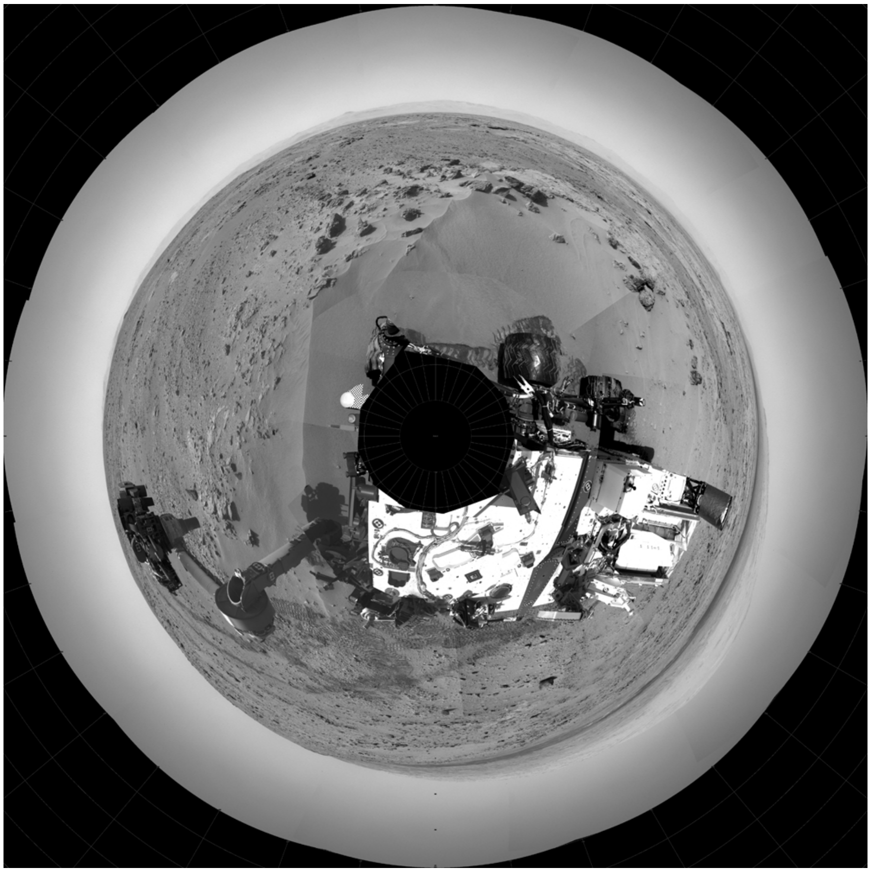


Figure 5.2.1.13.7 - Polar Projection Mosaic

5.2.1.13.8 "Vertical Projection Mosaic" RDR

Vertical mosaics provide a view of the surroundings as if you were looking straight down. They are thus quite useful for establishing the environmental context or comparing with orbital imagery, but suffer from severe distortion with any variance of the scene from the surface model. In particular, rocks are severely elongated, and the terrain is not taken into account.

MIPL creates the Vertical projection as follows:

$$\mathrm{nl} = \text{Number of lines in the mosaic (IMAGE object, LINES)}$$
$$\mathrm{ns} = \text{Number of samples in the mosaic (IMAGE object, LINE\_SAMPLES)}$$
$$\mathrm{x} = (\mathrm{nl}/2 - j) * \mathrm{MAP\_SCALE}$$
$$\mathrm{y} = (i - \mathrm{ns}/2\,) * \mathrm{MAP\_SCALE}$$

The view ray emanates from (x, y, 0) and points straight down (0,0,1). If the ray misses the surface in step E of Section 5.3.13.3 above, it is changed to point straight up (0,0,-1).

Figure 5.2.1.13.8 shows a vertical view. It assumes that the field is a plane tangent to the Martian surface with up pointing north. This is not an Orthorectified rendering but is still useful in many situations.

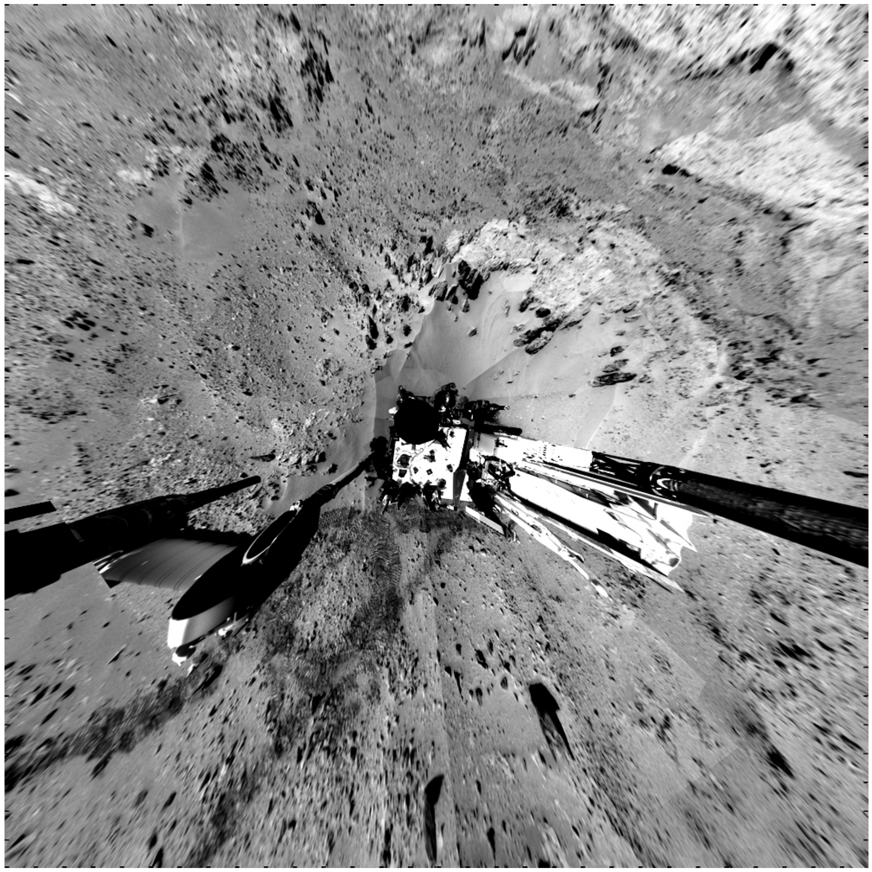


Figure 5.2.1.13.8 - Vertical Projection Mosaic

5.2.1.13.9 "Orthographic Projection Mosaic" RDR

The Orthographic projection is a generalization of the Vertical projection intended primarily for use with MAHLI data. It differs from Vertical in that an arbitrary projection plane can be specified.

If O is the point specified by the PROJECTION_ORIGIN_VECTOR and $\hat{X}$ and $\hat{Y}$ are the unit vectors given by PROJECTION_X_AXIS_VECTOR and PROJECTION_Y_AXIS_VECTOR respectively, then an arbitrary point P will have projection coordinates (X,Y) as follows:

$$\vec{X} = \left(\vec{P} - \vec{O}\right) \bullet \hat{X}$$
$$\vec{Y} = \left(\vec{P} - \vec{O}\right) \bullet \hat{Y}$$

where the "•" indicates the scalar dot product of two vectors. PROJECTION_Z_AXIS_VECTOR is the direction of projection; the three vectors form a right-handed orthonormal basis.

All of these quantities must be specified with respect to a single frame defined by the REFERENCE_COORD_SYSTEM_NAME and REFERENCE_COORD_SYSTEM_INDEX. Additional relevant parameters for the projection are MAP_SCALE, X_AXIS_MINIMUM, X_AXIS_MAXIMUM, Y_AXIS_MINIMUM, and Y_AXIS_MAXIMUM.

A Vertical projection is the same as Orthographic with PROJECTION_X_AXIS_VECTOR = (1,0,0), PROJECTION_Y_AXIS_VECTOR = (0,1,0), and PROJECTION_Z_AXIS_VECTOR = (0,0,1).

5.2.1.13.10 "Orthorectified Projection Mosaic" RDR

Orthorectified mosaics are used to show a "true" view of the scene from a different point of view, without distortion due to parallax. The point of view is usually overhead, resulting in an image suitable for comparison to satellite imagery. The removal of parallax necessarily leads to holes or gaps in the mosaic, which do not occur with the other projections.

The Orthorectified mosaic is projected to a plane in a similar manner to the Orthographic or Vertical projections. However, unlike any of the other projections, the XYZ locations of the pixels are taken into account. This is what allows parallax to be removed.

A simple way to think of this, for the case of an Orthorectified-Vertical projection, is to attach the XYZ coordinate (derived from stereo analysis) to each input image pixel, chop off the Z coordinate, and use the XY coordinates as the position in the output image. The more general (non-Vertical) case is similar in concept, just rotate the XYZ values to the frame defined by the projection plane first.

The most common projection plane is to look straight down, which corresponds to the same point of view as the Vertical projection. For this case, PROJECTION_X_AXIS_VECTOR = (1,0,0), PROJECTION_Y_AXIS_VECTOR = (0,1,0), and PROJECTION_Z_AXIS_VECTOR = (0,0,1).

The specific algorithm must deal with filling holes in the output mosaic and is still to be determined, as the software for this projection remains under development as of this writing.

Figure 5.2.1.13.10 shows an orthorectified rendering.

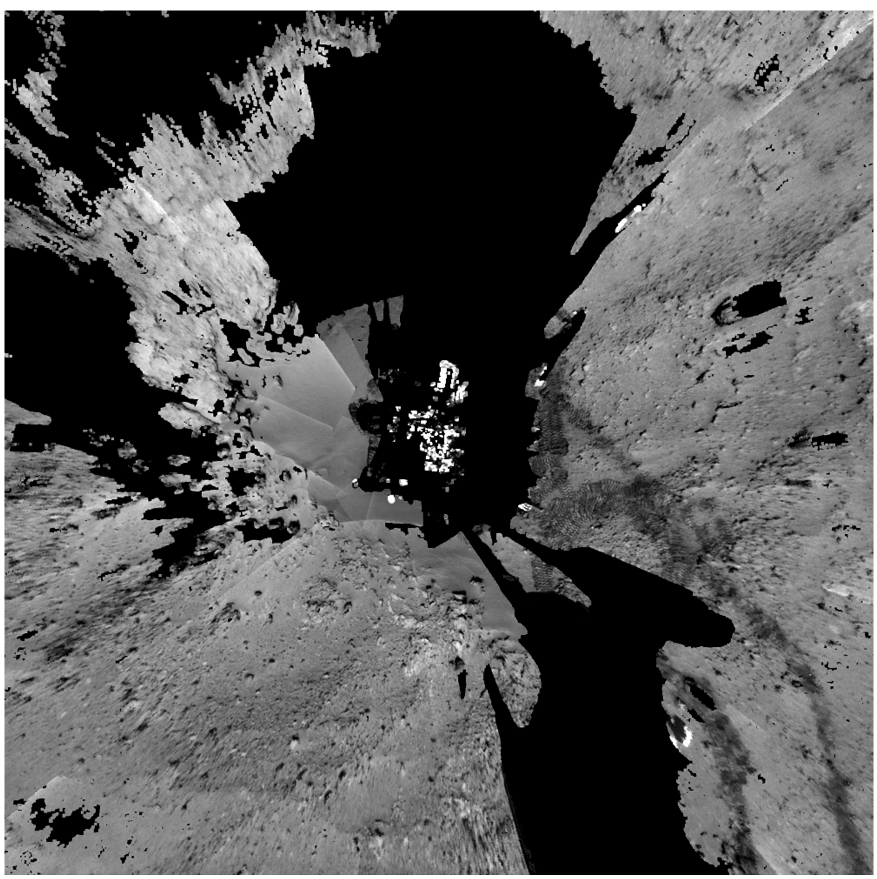


Figure 5.2.1.13.10 - Orthorectified Projection Mosaic

5.2.1.13.11 ChemCam Finder Mosaic RDR

ChemCam Finder Mosaic products are a use case for mosaics of image masks of the projected location of rover hardware from the point of view of the ChemCam instrument (on RSM). These products can be overlaid on Navcam mosaics of the same projection. They are used by the ChemCam operations team to determine if a desired target is clear of rover hardware at a given time, based on either Actual or Predicted knowledge of the rover's arm orientation. The 5-character "mosid" field (see 6.1.2 for details on mosaic file naming convention) begins with "A" for Actual position, and "P" for Predicted position, followed by a time stamp in LMST. For example, a file ending in "_A1520" is based on the Actual arm position on that sol at 15:20 LMST. These products have a non-PDS compliant PRODUCT_ID (44 chars vs. 40). This is acceptable because the product is not necessarily archived with the PDS.

The ChemCam Finder product types are "MCR" (Risk mask), "MCE" (Error mask), and "MCF" (Combo Risk/Error mask – Figure 5.2.1.13.11 below). The difference between the MCE and MCR products is in the amount of margin applied to the mask – the MCE "Error" product applies a small, constant margin to all of the rover hardware, while the MCR "Risk" product applies greater margin to the Arm than the rover body to account for variations in the projection of the Arm due to parallax, depending on its position and distance from the RSM. The MCF "Combo" product includes both Risk and Error masks, where a DN of 2 = MCR-only (shown in Red below), DN of 3 = MCE (shown in Yellow below), and DN of 0 = Clear of any risk of hardware collision.

Note that some versions of these mosaics have target locations plotted on them. These targets are derived from the operations target database.

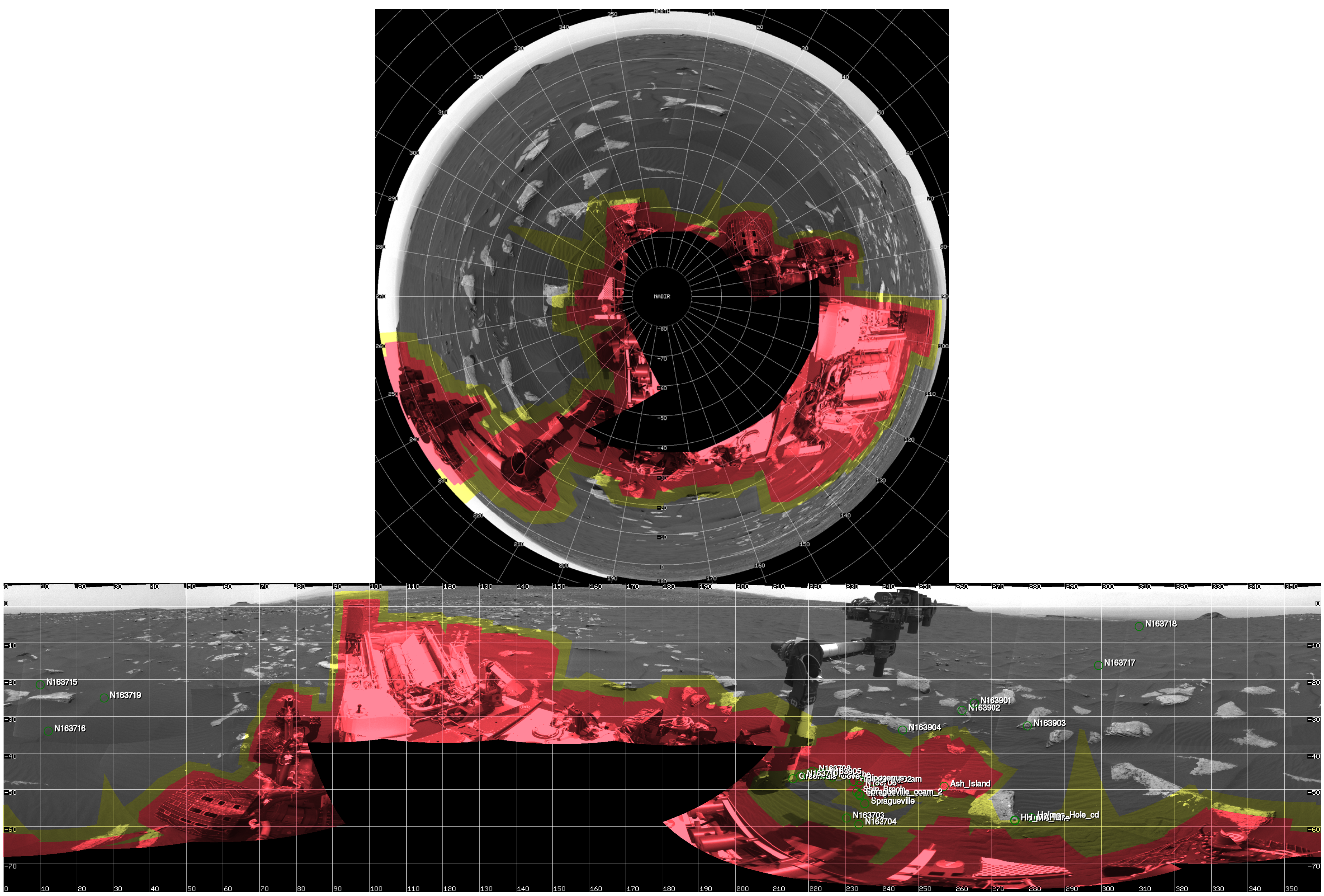


Figure 5.2.1.13.11 – ChemCam Finder "MCF" masks for 2 different arm configurations, applied to the same Navcam mosaic basemap in polar (top) and cylindrical (bottom) projections.

5.2.1.13.12 "Mosaic-to-Image Coregistration Map" RDR

The "IDX" and "ICM" products collectively map each pixel in a given mosaic back to its original single-frame image source. The "IDX" Index image product identifies which original input image that the pixel came from by assigning a 1-based integer value to each image, which corresponds to its position within the mosaic's LIST file. The "ICM" product is a 2-band floating-point file containing 1-based line and sample coordinates of each pixel in its original single-frame image. The "ICM" product is very similar in format to the Disparity RDR products (Section 5.2.1.3).

Figure 5.2.1.13.12 shows an example Navcam mosaic with associated "IDX" and "ICM" products. These products are generated by request and are not necessarily available for all mosaics.

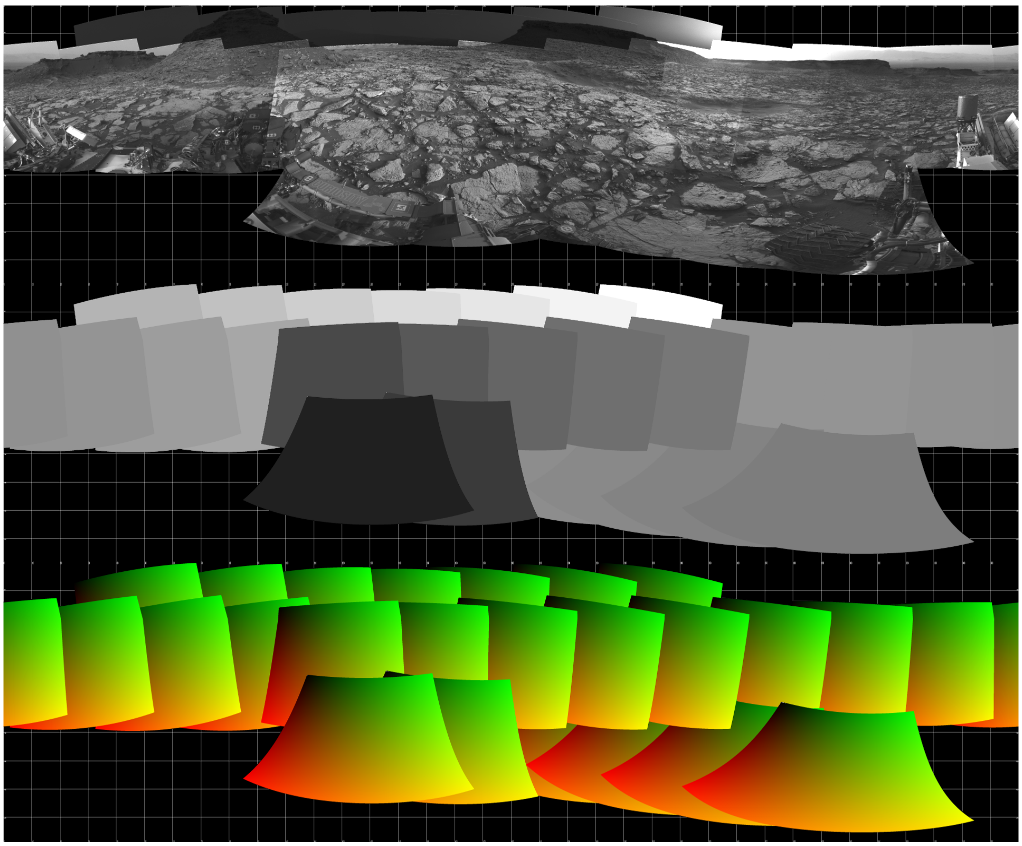


Figure 5.2.1.13.12 – A 26-frame Navcam mosaic (top), with associated IDX (middle) and ICM (bottom) products. The ICM image has the original LINE index (band 1) mapped to Red, and original SAMPLE index (band 2) mapped to Green.

5.2.1.13.13 Non-image Mosaics

Normally mosaics are created using imagery, where each pixel is either a raw or radiometrically corrected intensity value. However, mosaics can also be created using other types of pixels. In fact, any of the RDR types using an image format (e.g. not meshes) can be mosaicked. The most useful of these are mosaics of XYZ, surface normal, and the various Slope types.

For example, an XYZ mosaic contains XYZ values for each pixel in the mosaic rather than intensity values. The inputs to the mosaic program are XYZ files (or individual X, Y, or Z components), and the pixels are interpreted in the same way - as the coordinate of the corresponding pixel in Cartesian space. Like XYZ images, the mosaics may consist of a single 3-band file with X, Y, and Z components, or separate 1 band files for each component. A Z-only mosaic of a Vertical or

Orthorectified-Vertical projection creates a digital elevation model (DEM) - approximate in the Vertical case, correct for Orthorectified-Vertical.

As another example, Slope mosaics are often created and then overlaid on an image mosaic using the same projection parameters to help with rover navigation.

For MSL, a common product is the "ChemCam Finder" mosaic. This consists of a set of three Polar projection mosaics: the imagery, the range, and the rover exclusion mask. The mask is adjusted based on actual or predicted arm locations. The overlay of the mask mosaic over the image aids the ChemCam team in targeting (to avoid the rover and the arm), while the overlay of range helps determine focus distances for commanding.

Care must be taken while producing these mosaics to ensure that a consistent coordinate system and data type are used for all the input images. No transform is done on the data; the output mosaic may have only one coordinate system in which the values are defined, and one DERIVED_IMAGE_TYPE.

Non-image mosaics are often created without interpolation; the nearest pixel is used instead. This avoids aliasing effects when pixels are interpolated with neighboring invalid pixels.

5.2.1.14 “Anaglyph” RDR

A stereo anaglyph is a method of displaying stereo imagery quickly and conveniently using conventional display technology (no special hardware) and red/blue glasses. This is done by displaying the left eye of the stereo pair in the red channel, and displaying the right eye in the green and blue channels. An anaglyph data product simply captures that into a single 3-band color image, which can be displayed using any standard image display program with no knowledge that it is a stereo image. The red (first) band contains the left eye image, while the green and blue (second and third) bands each contain the right eye image (so the right image is duplicated in the file).

The Anaglyph method can also apply to multi-frame mosaic products. OPGS-generated mosaic Anaglyphs occasionally required some subtle pixel-shifting of the right eye mosaic data to improve the stereo effects. Mosaic Anaglyph products are distinguishable in the Mosaic RDR filename convention (see Section 6.1.2).

5.2.1.15 AEGIS RDRs

The AGS Target Map product (Figure 5.2.1.15) is a mask that shows each of the areas of interest for a ChemCam LIBS observation as determined by the onboard AEGIS software. Each area has a different DN between 1-255, which maps to the AGM file described below, and a DN of 0 indicates an area that is not of interest.

The corresponding non-image AGM product is a JSON formatted text file with an array of targets, each of which includes metadata about that target - AEGIS match score, distance from rover and the RMI camera, its XYZ position, and other metadata details. The index location within the target array corresponds to the DN given to its mask in the AGS product (1-based array).

This product is intended for operational use by AEGIS Flight Software team, and is not necessarily included in the PDS archive.

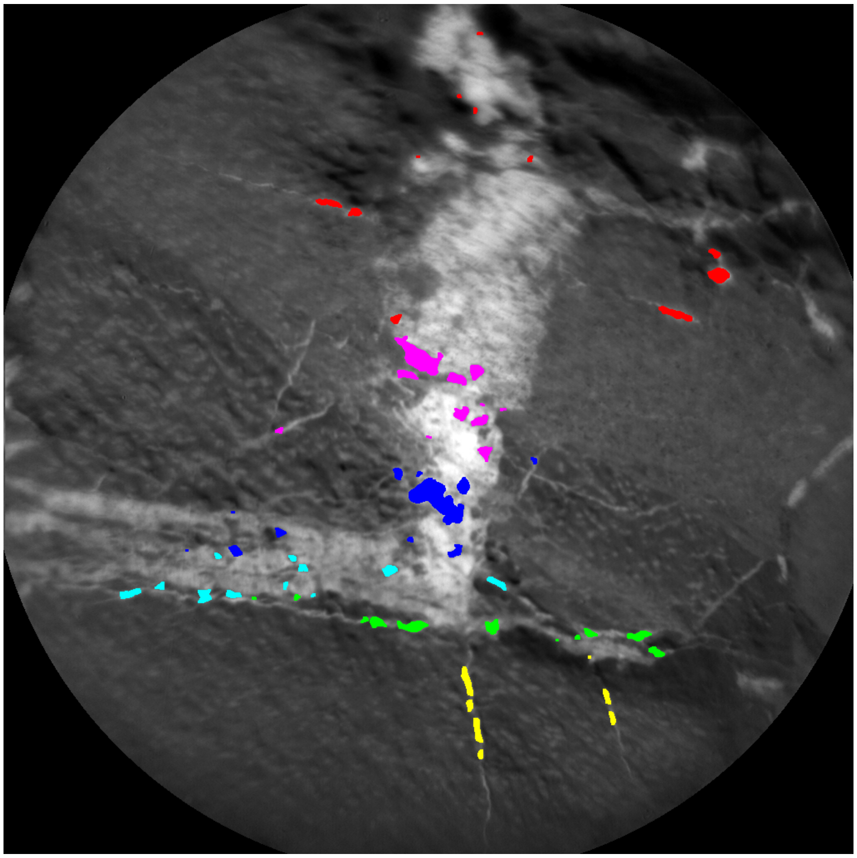


Figure 5.2.1.15 – AEGIS AGS Target Map (colorized) over a ChemCam RMI image.

5.2.1.16 Terrain Classifier RDRs

Terrain Classifier RDR products are derived from Navcam images, and are used operationally to improve predictions of rover slip for planning drives by attributing a different set of slip parameters to each terrain type. The RDR products are generated by OPGS using a machine learning image classification algorithm. These products are integrated into various planning tools used by the operations team, but are not necessarily included in the PDS archive.

The "TEN" Terrain Probability map is a multiband product with 1 band per terrain class, where the DN value of a given band corresponds to the classification confidence score (Figure 5.2.1.16a). It is registered geometrically to the source EDR. The total number of bands is not fixed in order to allow flexibility for new terrain classes to be added. Updates or additions to existing class-to-band definitions will result in an increment of the "CLASSIFIER_VERSION" keyword.

Terrain classes are defined by the "CLASSIFIER_BAND_INDEX_NAME" keyword, and may represent geological features such as sand ripples, caprock, etc.; or rover traversability characteristics like high slip, medium slip, etc. The band number associated with each class is listed in the "CLASSIFIER_BAND_INDEX" keyword as unsigned integers starting from 0.

There is a range label keyword ("CLASSIFIER_MAX_RANGE"), which describes the maximum distance in meters at which terrain classes were applied, as the classification performance varies with distance from the camera. The max range is applied on a line-by-line basis.

The “TER” Terrain Classification map is a 2-band product that is derived from the “TEN” map, where the first band is a map of the most probable terrain class for each pixel (Figure 5.2.1.16b), and the second band is a confidence score for the assigned class.

The list of possible classes in the TER product is included in the label keyword “CLASSIFIER_LABEL_INDEX_NAME”. These will match the classes listed in the TEN product, but will always have an additional “NONE” class appended to the end of the list. Pixels assigned to the “NONE” class may contain rover hardware or have range values greater than the max range constraint, and thus were masked out of the final products. The “CLASSIFIER_LABEL_HUE” keyword assigns a hex color code hue to each class in band 1 – this keyword is used by Mars Viewer and RSVP operations tools to display the classifications. See Appendix F for full details on TEN and TER keywords (“CLASSIFIER_[...]”).

Two additional products are specifically generated for use with the RSVP Hyperdrive planning tool: “.ht_tc”, which presents TER RDR data in orthographic projection to geometrically register it to the corresponding DEM RDR (“.ht”). “.mod_tc” is a non-image product that contains a list of “.ht_tc” files and other metadata that enables RSVP Hyperdrive to load all relevant “.ht_tc” files and select the optimal product to use at a given location.

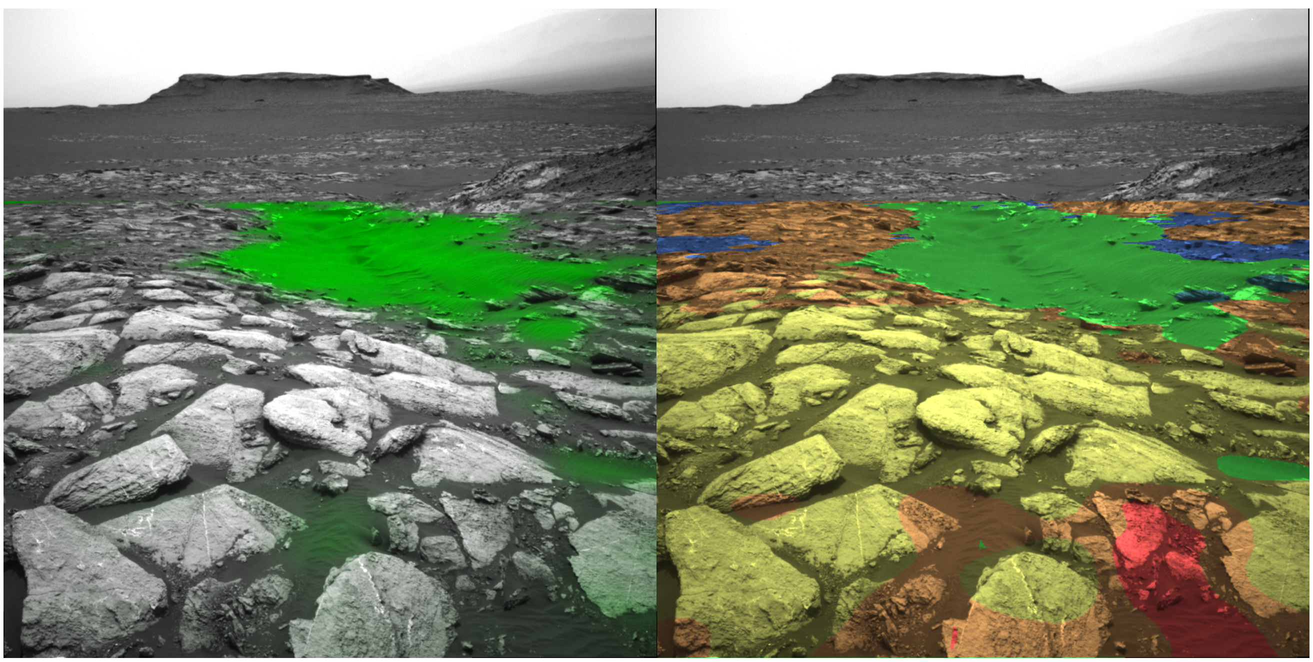


Figure 5.2.1.16a (Left) Band 1 (“Sand” class confidence score) of TEN probability map. Figure 5.2.1.16b (Right): Band 1 of TER classification map. “Sand” class is in green.

5.2.1.17 ChemCam Range and Exclusion Mask RDRs

The pixels in a ChemCam Range RDR (“CXR”) represent discrete Cartesian distances in meters between the center of the ChemCam Camera optical frame and the XYZ position of each pixel (derived from Navcam stereo). Distances are expressed in the Rover Navigation Frame (see “XYR” RDR). A ChemCam Range RDR contains 1 band of 32-bit floating point numbers.

For each pixel in the Navcam image with an XYZ value, a virtual pointing is computed using an inverse kinematic solution for RSM azimuth and elevation derived from [Ref 32]. Virtual pointing and position values of the ChemCam Camera change for each pixel; consequently, the RANGE_ORIGIN_VECTOR keyword in the PDS label is set to a value of “(0, 0, 0)”, unlike Navcam

Range RDR products. ChemCam Range images depend on Navcam observations for visual representation, therefore the products share the same pixel geometry as the XYZ images from which they are derived. As with XYZ images, ChemCam range images can contain holes, defined by MISSING_CONSTANT. For MSL, this value is 0.0.

A ChemCam Exclusion Mask (“CXM”) product is derived from ChemCam Range “CXR” RDR. The mask describes the region about the rover where use of LIBS is precluded, which is a function of distance from the ChemCam Camera to a ground target.

This 1-band product is represented in byte format. It has a similar format to the Rover Mask Image products (“MXY”), where unmasked points have a value defined by the MISSING_CONSTANT, 0. Mask points have a value of 255, and represent targets which lie in the Exclusion Zone.

5.2.2 Spectroscopy RDRs

ChemCam EDR LIBS spectra will be used to generate three standard RDR products.

5.2.2.1 “Initial LIBS Spectrum” RDR

This product will contain the spectrum data for each spectrum available in the EDR file, the processed median of the spectrum data and the processed average of all spectra. In a later release the Standard Deviation, 1st and 3rd quartile will be added. Processing includes the following steps: (1) dark subtraction, (2) correction for drift, (3) denoising, (4) continuum removal, (5) calibration in wavelength and (6) resampling. The identifier in the filename for this product is “RDR”.

The format of this RDR is described in Table 5.2.2.1:

Table 5.2.2.1 - LIBS RDR

Column Number	Name	Data Type	Description
1	Wavelength	Double	Calibrated wavelength
2 … N+1 (where N = # spectra received)	Spectra Digital Number (DN)	Double	Intensity in DN for each spectrum
N+2	Median Spectrum Digital Number (DN)	Double	Intensity in DN of Median of all shots
N+3	Average Spectrum Digital Number (DN)	Double	Intensity in DN of Average of all shots

The data will be preceded by several rows of comments, and key parameters used in the data processing. (total rows: 6144 + comments)

5.2.2.2 “Intermediate Clean Calibrated Spectra” RDR

This product will contain the spectrum for each shot available in the EDR file and an averaged spectrum that have been: (1) dark subtracted, (2) corrected for drift, (3) denoised, (4) continuum removed, (5) calibrated in wavelength, (6) resampled and (7) corrected for instrument response. The identifier in the filename for this product is “CCS”.

The format of this RDR is described in Table 5.2.2.2, where term “radiance” equates to “photon/shot/mm^2/sr/nm”:

Table 5.2.2.2 – Clean Calibrated Spectra

Column Number	Name	Data Type	Description

Column Number	Name	Data Type	Description
1	Wavelength	Double	Calibrated wavelength
2 … N+1 (where N = # spectra received)	Spectra Radiance	Double	Intensity in radiance for each spectrum processed
N+2	Median Spectrum Radiance	Double	Intensity in radiance of Median of all spectra processed
N+3	Average Spectrum Radiance	Double	Intensity in radiance of Average of all spectra processed

The data will be preceded by several rows of comments, and key parameters used in the data processing. (total rows: 6144 + comments)

5.2.2.3 "Multivariate Prediction of Oxide Composition" RDR

Through a partial least-square model based on laboratory spectra of geostandard reference samples, derive the target composition in oxide weight percent, including uncertainties. The analysis may produce results for individual shots or an average of the individual shots (excluding the first five shots which typically represent dust). The identifier in the filename for this product is "MOC".

The format of this RDR is described in Table 5.2.2.3:

Table 5.2.2.3 - Elemental Composition Predictions (Oxide Weight)

Row Number	Name	Data Type	Description
1	Oxides	Text	Oxides and elements are labeled across the top of each section
2	RMSEP	Double	Root mean square error prediction equivalent to an absolute error estimation given in the units for that element, i.e., oxides are in wt.% and trace elements are in parts per million (ppm) weight
3	Predicted Elemental Abundances	Double	Predicted oxide abundance for the average spectrum in the units of that element, i.e., oxides are in wt.% and trace elements are in parts per million (ppm) weight

Preceding the data will be several rows of comments. The number of oxides may vary in future data releases, but the major elements that are reported are: SiO_2, TiO_2, Al_2O_3, Fe_2O_3T, MgO, CaO, Na_2O, K_2O.

5.3 RDR Product Format

5.3.1 Image RDRs

The image RDR data products covered by this SIS are listed in Tables 5.3.1.1 and 5.3.1.2 below. Products listed as "1-3" bands are generally 1-band products but could have 3 RGB bands for certain MMM camera modes.

Table 5.3.1.1 - MSL Camera Image RDR Binary Formats

Description	Product Identifier	# Bands	Data Type	DERIVED_IMAGE_TYPE Keyword value
CAHV-linearized	LIN	1-3	16-bit signed integer or 8-bit unsigned byte	"IMAGE"

Description	Product Identifier	# Bands	Data Type	DERIVED_IMAGE_TYPE Keyword value
Bayer pattern	BAY	1-3	16-bit signed integer	“IMAGE”
Inverse lookup table (ILUT)	ILT	1-3	16-bit signed integer	“IMAGE”
	ILC	1-3	16-bit signed integer	“IMAGE”
	ILP	1-3	16-bit signed integer	“IMAGE”
Radiometrically corrected	RAD	1-3	16-bit signed integer	“IMAGE”
	RAS	1-3	16-bit signed integer	“IMAGE”
	RAF	1-3	Float	“IMAGE”
Radiometrically corrected for Instrument Effects	RIE	1-3	16-bit signed integer	“IMAGE”
	RIF	1-3	Float	“IMAGE”
Radiometrically corrected IOF radiance factor	IOI	1-3	16-bit signed integer	“IMAGE”
	IOF	1-3	16-bit signed integer	“IMAGE”
MMM-produced Radiometrically corrected	DRX	1-3	16-bit signed integer	“IMAGE”
	DRW	1-3	16-bit signed integer	“IMAGE”
Stereo Disparity	DSP	2	Float	“DISPARITY_MAP”
	DSR	2	Float	“DISPARITY_MAP”
	DSL	1	Float	“DISPARITY_LINE_MAP”
	DSS	1	Float	“DISPARITY_SAMPLE_MAP”
	DSG	1	8-bit unsigned byte	“DISPARITY_MAP”
	DSE	TBD	Float	“DISPARITY_ERROR_MAP”
Stereo Disparity Mask File	MDS	1	8-bit unsigned byte	“MASK”
Stereo Delta Disparity	DDD	2	Float	“DELTA_DISPARITY_MAP”
	DDL	1	Float	“DELTA_DISPARITY_LINE_MAP”
	DDS	1	Float	“DELTA_DISPARITY_SAMPLE_MAP”
Stereo First-stage Disparity	DFF	2	Float	“DISPARITY_MAP”
	DFL	1	Float	“DISPARITY_LINE_MAP”
	DFS	1	Float	“DISPARITY_SAMPLE_MAP”
XYZ	XYZ	3	Float	“XYZ_MAP”
	XYF	3	Float	“XYZ_MAP”
	XYO	3	8-bit unsigned byte	“XYZ_MAP”
	XYE	3	Float	“XYZ_ERROR_MAP”
	XYM	3	Float	“XYZ_MAP”
XYZ Mask File	MXY	1	8-bit unsigned byte	“MASK”
XYZ X-band	XXX	1	Float	“X_MAP”
	XXF	1	Float	“X_MAP”
XYZ Y-band	YYY	1	Float	“Y_MAP”
	YYF	1	Float	“Y_MAP”
	ZZZ	1	Float	“Z_MAP”
XYZ Z-band	ZZF	1	Float	“Z_MAP”
	ZZO	3	8-bit unsigned byte	“Z_MAP”
Surface Normal (UVW)	UVW	3	Float	“UVW_MAP”
	UVS	3	Float	“UVW_MAP”
	UVP	3	Float	“UVW_MAP”
	UVT	1	Float	“ANGLE_MAP”
	UVO	3	8-bit unsigned byte	“UVW_MAP”
	UVF	3	Float	“UVW_MAP”
Surface Normal U-band	UUU	1	Float	“U_MAP”
	UUF	1	Float	“U_MAP”
Surface Normal V-band	VVV	1	Float	“V_MAP”
	VVF	1	Float	“V_MAP”
Surface Normal W-band	WWW	1	Float	“W_MAP”
	WWF	1	Float	“W_MAP”

Description	Product Identifier	# Bands	Data Type	DERIVED_IMAGE_TYPE Keyword value
Surface Roughness	RUD	3	Float	“ROUGHNESS_MAP”
	RUT	2	Float	“ROUGHNESS_MAP”
Range	RNG	1	Float	“RANGE_MAP”
	RNM	1	Float	“RANGE_MAP”
	RNO	3	8-bit unsigned byte	“RANGE_MAP”
	RNF	1	Float	“RANGE_MAP”
	RNE	3	Float	“RANGE_ERROR_MAP”
Arm Reachability	ARM	5	16-bit signed integer	“REACHABILITY_MAP”
	ARO	3	8-bit unsigned byte	“REACHABILITY_MAP”
	ARK	5	16-bit signed integer	“REACHABILITY_MAP”
Arm Reachability Mask File	MAR	1	8-bit unsigned byte	“MASK”
Arm Preload Values	ARP	2	16-bit signed integer	“PRELOAD_MAP”
Slope	SLP	1	Float	“SLOPE_MAP”
	SLO	3	8-bit unsigned byte	“SLOPE_MAP”
	SRD	1	Float	“RADIAL_SLOPE_MAP”
	SRO	3	8-bit unsigned byte	“RADIAL_SLOPE_MAP”
	SHD	1	Float	“SLOPE_HEADING_MAP”
	SHO	3	8-bit unsigned byte	“SLOPE_HEADING_MAP”
	SMG	1	Float	“SLOPE_MAGNITUDE_MAP”
	SMO	3	8-bit unsigned byte	“SLOPE_MAGNITUDE_MAP”
	SNT	1	Float	“NORTHERLY_TILT_MAP”
	SNO	3	8-bit unsigned byte	“NORTHERLY_TILT_MAP”
Solar Energy	SEN	1	Float	“SOLAR_ENERGY_MAP”
	SEO	3	8-bit unsigned byte	“SOLAR_ENERGY_MAP”
Incidence, Emission, Phase	IEP	3	Float	“IEP_MAP”
	IEF	3	Float	“IEP_MAP”
ChemCam Finder	MCR	1	8-bit unsigned byte	“MASK”
	MCE	1	8-bit unsigned byte	“MASK”
	MCF	1	8-bit unsigned byte	“MASK”
ChemCam LIBS Range and Exclusion Mask	CXR	1	Float	“RANGE_MAP”
	CXM	1	8-bit unsigned byte	“MASK”
AEGIS	AGS	1	16-bit unsigned integer	“AEGIS_MAP”
Mosaic-to-Image Coregistration Map	ICM	2	16-bit unsigned integer	“ICM_MAP”
	IDX	1	16-bit unsigned integer	“IDX_MAP”
Terrain Classifier	TEN	N	8-bit unsigned byte	“TERRAIN_CLASSIFICATION_MAP”
	TER	2	8-bit unsigned byte	“TERRAIN_CLASSIFICATION_MAP”

Table 5.3.1.2 - MSL Camera Non-image RDR Binary Formats

Description	Product ID	# Bands	Data Type	Data Structure
Terrain Mesh	Terrain Mesh RDR	N/A	Inventor (IV)	IV
JPEG compressed	JPEG compressed	3	8-bit unsigned	JPEG, no label
AEGIS Metadata	“AGM”	N/A	Text	JSON

5.3.2 Spectroscopy RDRs

Each ChemCam LIBS RDR data file is a comma delimited ASCII text table. Each line or record in the files is terminated with a two-character sequence of carriage return (<CR>, ASCII 13) and line feed

(<LF>, ASCII 10) to comply with PDS standards. This line terminator sequence will allow the data files and labels to be easily read on most computers, which recognize either the carriage return, the line feed, or the <CR>/<LF> sequence as an ASCII record terminator.

Table 5.3.2 - MSL LIBS Spectroscopy RDR Binary Formats

Name	Product Identifier	Data Type	Standard / Special
LIBS Spectrum	RDR	Double	Standard
Intermediate Clean Calibrated Spectra	CCS	Double	Special
Multivariate Prediction of Oxide Composition	MOC	Double, ASCII	Standard

5.4 RDR Product Structure

RDR products will have three possible structures. RDRs generated by MIPL will have a VICAR label wrapped by an ODL label, see Figure 5.4, Diagram A). This is the same as the EDR format, with the exception that the binary header data are eliminated. RDR products not generated by MIPL may look the same, or they may omit the VICAR label, containing only an ODL label (Figure 5.4, Diagram B). Or, RDR products conforming to a standard other than PDS, such as JPEG compressed or certain Terrain products (Figure 5.4, Diagram C), may contain no additional labels, instead following the other standard's formatting. In any case, RDRs will also have a detached PDS label. For a description of the PDS, ODL, and VICAR labels, and the mapping between PDS/ODL and VICAR, see Section 3.2.

The RDR data product is comprised of processed versions of the raw camera data, in both single and multi-frame (mosaic or mesh) form. Most RDR data products will be in "image" format, having detached PDS labels, as well as attached ODL labels or, if generated by MIPL (OPGS), dual attached ODL/VICAR labels. Non-labeled RDRs include JPEG compressed products and the Terrain products.

For Mosaic RDRs, the detached PDS label is shared amongst four types of archiveable files: a) the ".IMG" Mosaic image file, b) the ".NAV" navigation file, c), the ".BRT" brightness correction file, and d) the ".LIS" list file identifying the mosaic component images. For a description of "b" and "c", see Section 5.2.1.13.3. To support the four types of Mosaic files, the PDS label contains four FILE Objects in the structure. See Appendix B for an example of a Mosaic RDR detached PDS label.

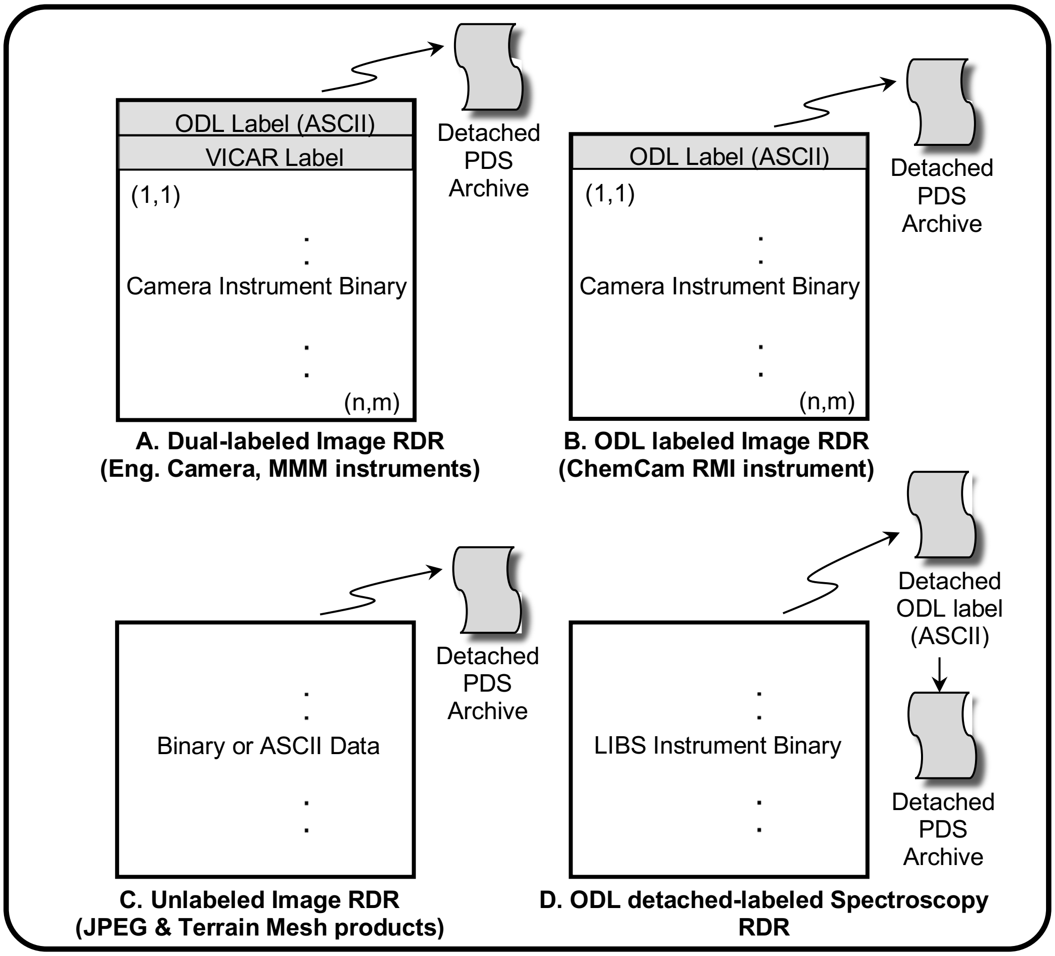


Figure 5.4 - RDR Product Structures

6. STANDARDS USED IN GENERATING PRODUCTS

6.1 File Naming Standards

The file naming schemes adhere to the Level II 36.3 filename standard approved by PDS in 2009. This is a change from the 27.3 convention that MER and PHX were constrained to using. Use of three- character extensions, such as ".IMG" for image EDRs and RDRs and ".DAT" for spectrum EDRs and state-of-health EDRs, is consistent with the PDS standard.

There are three file naming schemes adapted for the MSL image and non-image data products. The first applies to the EDR data product and all Single-frame RDR data products. The second applies to all Mosaic RDR data products. The third applies to Terrain products.

The primary attributes of the filename nomenclature are:

a) Uniqueness - It must be unique unto itself without the file system's directory path. This protects against product overwrite as files are copied/moved within the file system and external to the file system, if managed correctly.

b) Metadata - It should be comprised of metadata fields that keep file bookkeeping and sorting intuitive to the human user. Even though autonomous file processing will be managed via databases, there will always be human-in-the-loop that puts a premium on filename intuition. Secondly, the metadata fields should be smartly selected based on their value to ground processing tools, as it is less CPU-intensive to extract information from the filename than from the label.

NOTE: Metadata information in the filename also resides in the product label.

The metadata fields have been selected based on MER and PHX lessons learned. In general, the metadata fields are arranged to achieve:

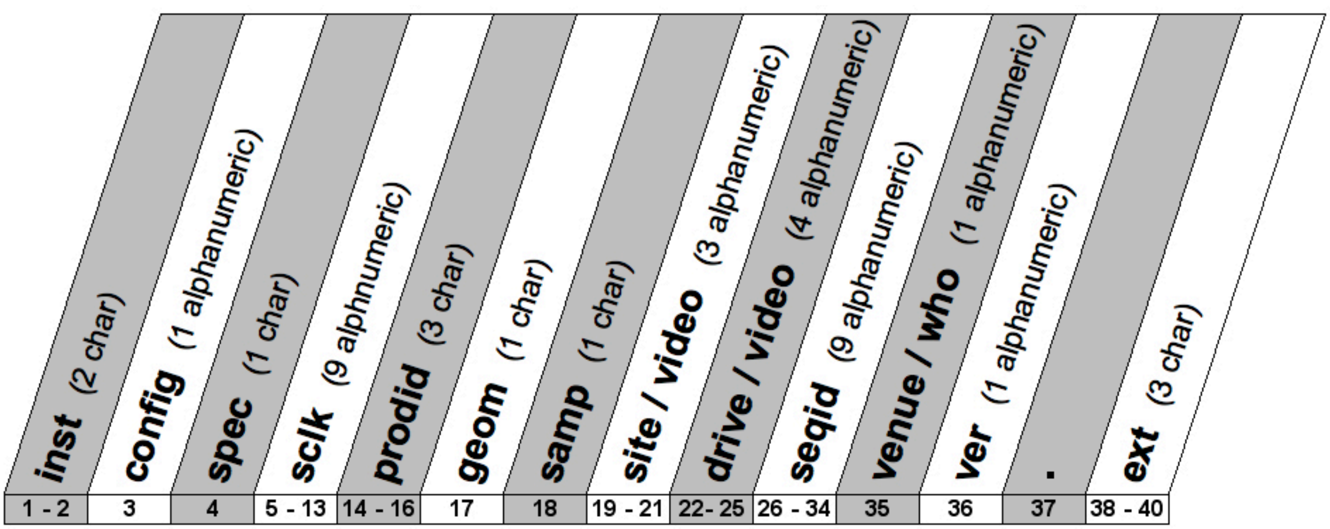
a) Sortability - At the beginning of the filename resides a primary time oriented field such as Spacecraft Clock Start Count (SCLK). This allows for sorting of files on the MSL file system by spacecraft data acquisition time as events occurred on Mars.

b) Readability - An effort is made to alternate Integer fields with ASCII character fields to optimize differentiation of field boundaries for the human user.

6.1.1 EDR and Single-frame RDR Filename

Each MSL EDR and Single-frame RDR data product can be uniquely identified by incorporating into the product filename at minimum the Instrument ID, SCLK, Product Type identifier, Site, Drive and Command Sequence identifier. The convention is illustrated in Figure 6.1.1 below.

The Single-frame RDR data products that share the naming scheme with the EDR data product are numerous. They are listed in the description of the Product Type field found in the filename



convention definition, which follows:

Figure 6.1.1 – EDR and Single-frame RDR Filename Convention

inst = (2 alpha character) Instrument ID, denoting the source MSL science or engineering instrument that acquired the data.

Valid values for Instrument IDs are:

"**FL**" -	Front Hazcam Left	"**MR**" -	Mastcam Right
"**FR**" -	Front Hazcam Right	"**MS**" -	Mastcam Stereo (2-banded)

“**FS**” - Front Hazcam Stereo (2-banded)
“**FA**” - Front Hazcam Anaglyph (3-banded)
“**RL**” - Rear Hazcam Left
“**RR**” - Rear Hazcam Right
“**RS**” - Rear Hazcam Stereo (2-banded)
“**RA**” - Front Hazcam Anaglyph (3-banded)
“**NL**” - Navcam Left
“**NR**” - Navcam Right
“**NS**” - Navcam Stereo (2-banded)
“**NA**” - Navcam Anaglyph (3-banded)
“**ML**” - Mastcam Left
“**MA**” - Mastcam Anaglyph (3-banded)
“**MG**” - Mastcam Colorglyph (Lr,Rg,Rb)
“**MH**” - MAHLI
“**HL**” - MAHLI used as Stereo Left
“**HR**” - MAHLI used as Stereo Right
“**HS**” - MAHLI Stereo (2-banded)
“**HA**” - MAHLI Anaglyph (3-banded)
“**HG**” - MAHLI Colorglyph (Lr,Rg,Rb)
“**MD**” - MARDI
“**CR**” - ChemCam RMI
“**CL**” - ChemCam LIBS
“**CC**” - ChemCam generic

Valid values for Instrument IDs not described in this SIS:

“**AP**” - APXS
“**CM**” - CheMin
“**DN**” - DAN
“**RD**” - RAD
“**RM**” - REMS
“**SM**” - SAM
“**SP**” - SA/SPaH

config = (1 alphanumeric) Instrument Configuration, an operational attribute of the Instrument that assists in characterizing the data.

Valid config values for MSL camera instruments:

Instrument	Configuration	
	Values	Description
Front Hazcam Left (“FL”)	“**A**”, “**B**”	A-side configuration, B-side configuration
Front Hazcam Right (“FR”)	“**A**”, “**B**”	A-side configuration, B-side configuration
Front Hazcam Stereo (“FS”)	“**A**”, “**B**”	A-side configuration, B-side configuration
Front Hazcam Anaglyph (“FA”)	“**A**”, “**B**”	A-side configuration, B-side configuration
Rear Hazcam Left (“RL”)	“**A**”, “**B**”	A-side configuration, B-side configuration
Rear Hazcam Right (“RR”)	“**A**”, “**B**”	A-side configuration, B-side configuration
Rear Hazcam Stereo (“RS”)	“**A**”, “**B**”	A-side configuration, B-side configuration
Rear Hazcam Anaglyph (“RA”)	“**A**”, “**B**”	A-side configuration, B-side configuration
Navcam Left (“NL”)	“**A**”, “**B**”	A-side configuration, B-side configuration
Navcam Right (“NR”)	“**A**”, “**B**”	A-side configuration, B-side configuration
Navcam Stereo (“NS”)	“**A**”, “**B**”	A-side configuration, B-side configuration
Navcam Anaglyph (“NA”)	“**A**”, “**B**”	A-side configuration, B-side configuration
ChemCam RMI (“CR”)	“**0**”	For ChemCam images, the types are: 0 = Image data (Fullframe, Subframe, Downsampled, Thumbnail, Reference Pixel, Row-summed, Column-summed, Histogram)
ChemCam LIBS (“CL”)	“**0**” - “**9**”	For ChemCam spectra, the types are: 0 = 1-D Average Spectra, no laser 1 = 1-D Single Spectra, no laser 2 = Spectra Stats (Mean & Std Dev), no laser 3 = 2-D Diagnostic, no laser 4 = 1-D Average Spectra Laser 5 = 1-D Single Spectra Laser

		6 = Spectra Stats (Mean & Std Dev), laser capable 7 = 2-D Diagnostic Laser 8 = Spectra Stats (all), laser capable 9 = Spectra Stats (all), no laser
ChemCam generic ("CC")	"**0**" - "**5**"	For ChemCam Generic, the types are: 0 = SOH 1 = Params 2 = Memory Dump 3 = Debug Dump 4 = Move Focus 5 = Util Test
Mastcam Left ("ML")	"**0**" - "**7**"	For all EDRs and those RDRs that are NOT "Bayer-pattern" or "color-split": Filters 0 thru 7
	"**R**", "**G**", "**B**", "**F**", "**U**", "**L**", "**D**", "**C**", "**A**"	**R** = **Red** Bayer cells only, or Red band of JPEG'd product **G** = **Green** Bayer cells only, or Green band of JPEG'd product **B** = **Blue** Bayer cells only, or Blue band of JPEG'd product **F** = **Full** color de-Bayered RGB merged into single 3-band image **U** = **Upper** half of Green cells only **L** = **Lower** half of Green cells only **D** = All Green cells (**double** resolution in one dimension) **C** = All cells, **corrected** for responsivity with the filter **A** = **All** cells, with 2x2 averaging
Mastcam Right ("MR")	"**0**" - "**7**"	For all EDRs and those RDRs that are NOT "Bayer-pattern" or "color-split": Filters 0 thru 7
	"**R**", "**G**", "**B**", "**F**", "**U**", "**L**", "**D**", "**C**", "**A**"	For "Bayer-pattern" or "color-split" RDRs: (same as for Mastcam Left)
Mastcam Stereo ("MS")	"**0**" - "**7**"	For all EDRs and those RDRs that are NOT "Bayer-pattern" or "color-split": Filters 0 thru 7
	"**R**", "**G**", "**B**", "**F**", "**U**", "**L**", "**D**", "**C**", "**A**"	For "Bayer-pattern" or "color-split" RDRs: (same as for Mastcam Left)
Mastcam Anaglyph ("MA")	"**0**" - "**7**"	For all EDRs and those RDRs that are NOT "Bayer-pattern" or "color-split": Filters 0 thru 7
	"**R**", "**G**", "**B**", "**F**", "**U**", "**L**", "**D**", "**C**", "**A**"	For "Bayer-pattern" or "color-split" RDRs: (same as for Mastcam Left)
MAHLI ("MH")	"**0**" - "**3**"	For all EDRs and those RDRs that are NOT "Bayer-pattern" or "color-split": 0 = cover closed, LEDs off 1 = cover open, LEDs off 2 = cover closed, LEDs on 3 = cover open, LEDs on
	"**R**", "**G**", "**B**", "**F**", "**U**", "**L**", "**D**", "**C**", "**A**"	For "Bayer-pattern" or "color-split" RDRs: (same as for Mastcam Left)

MARDI (“MD”)	“**0**”	For all EDRs and those RDRs that are NOT “Bayer-pattern” or “color-split”: 0 = no Filters
	“**R**”, “**G**”, “**B**”, “**F**”, “**U**”, “**L**”, “**D**”, “**C**”, “**A**”	For “Bayer-pattern” or “color-split” RDRs: (same as for Mastcam Left)

spec = (1 character) Special Processing flag, applicable to RDRs only.

The Special Processing character is used to indicate off-nominal or special processing of the image. Examples include a) use of different correlation parameters, b) special stretches to eliminate shadows, c) reprocessing with different camera pointing, etc.

The meaning of any individual character in this field will be defined on an ad-hoc basis as needed during the mission. Within one Sol or a range of Sol’s, the character will be used consistently. So, this field can be used to group together all derived products resulting in one kind of special processing. An attempt will be made to maintain consistency across different Sol's as well, but this may not always be possible; thus the meaning of characters may change across different individual or ranges of Sol's, depending on the definition.

A “.txt” ASCII text file will be maintained containing all special processing designators that are used, the Sol's they relate to, and a description of the special processing that was done. This file will be included in the PDS archive.

This field has the following rule-of-thumb:

Best Tactical - If value is character “**T**”, it indicates "best tactical" if other than nominal processing. The intent of this is to hold a copy of the special product best suited for tactical planning (at the discretion of OPGS ops personnel in consultation with tactical planners). Such products should have an ordinary special processing flag documented as described here, but be copied to flag "**T**" (incrementing version if necessary) if they are to be used for tactical planning.
If there is no "**T**", then the nominal "**_**" should be used for tactical planning.

Valid spec values are:

Special Processing	EDR Value	RDR Value
None	“_”	“_”
Special method types A-S and U-Z	n/a	“**A**” - “**S**”, “**U**” - “**Z**”
Best tactical Special method	n/a	“**T**”

sclk = (9 alphanumeric) Spacecraft Clock Start Count, in units of seconds.

Which specific SCLK is used depends on the instrument but is generally expected to be the time the data was acquired. For the engineering cameras and ChemCam, the SCLK in the IDPH is used. For MMM cameras, the SCLK in the MMM mini-header is used. It is not guaranteed that this SCLK match the DVT (Data Validity Time) used for operational data management.

The valid SCLK values, in their progression, are as follows (non-Hex):

Range 000000000 thru 999999999 - “**000000000**”, “**000000001**”, **…** “**999999999**”
Range 1000000000 thru 1099999999 - “**A00000000**”, “**A00000001**”, **…** “**A99999999**”
Range 1100000000 thru 1199999999 - “**B00000000**”, “**B00000001**”, **…** “**B99999999**”
•
•
•
Range 3500000000 thru 3599999999 - “**Z00000000**”, “**Z00000001**”, **…** “**Z99999999**”

prodid = (3 characters) Product Type identifier.
This field has the following rule-of-thumb:
Beginning "**E**" or "**N**" - Type of EDR, which is the first order product with no processing applied, such as geometric correction ("linearization") or radiometric correction. A beginning "N" denotes the EDR as a type of NavMap product (see Section 4.2.4.2). If no beginning "E" or "N", then the product is an RDR.

The various image EDR and RDR Product Type values, which are many and can be somewhat confusing, are color-coded for easier interpretation according to the following themes:

a) *Primary to End-user:* Most important, most used, most popular products

b) *Secondary:* Intermediate product, or final product not commonly used

c) *Special:* Generated outside of Pipeline as special request and not nominal

Valid values for Product identifiers for Image EDRs are listed below in Table 6.1.1.1.

See Table 6.1.1.2 for a list of valid prodid's for Image RDRs, and Table 6.1.1.3 for Spectroscopy (LIBS) RDRs.

Table 6.1.1.1: Valid Product Identifiers ("prodid") for EDRs

EDR Product Type Description	Value
First order product	**"EDR"**
Reference Pixel	**"ERP"**
Row-summed	**"ERS"**
Column-summed	**"ECS"**
Histogram	**"EHG"**
IDPH-only	**"EID"**
State-of-Health Initialize (ChemCam SOH only)	**"EIN"**
State-of-Health Power On (ChemCam SOH only)	**"EPW"**
State-of-Health Power Off (ChemCam SOH only)	**"EPO"**
State-of-Health Warmup (ChemCam SOH only)	**"EWU"**
State-of-Health Sun Safe (ChemCam SOH only)	**"ESS"**
Original JPEG as received from rover (MMM cameras only)	**"EJP"**
Z-stack combined image (MMM cameras only)	**"EZS"**
Depth map image (MMM cameras only)	**"EDM"**
Video image (MMM cameras only)	**"EVD"**
Recovered EDR (MMM cameras only)	**"ERD"**
Nav Map Goodness	**"NGD"**
Nav Map Certainty	**"NCE"**
Nav Map Idles	**"NID"**
Nav Map Minimum Count	**"NMN"**
Nav Map Maximum Count	**"NMX"**
Nav Map Elevation	**"NEL"**
Nav Map Normal X	**"NNX"**

EDR Product Type Description	Value
Nav Map Normal Y	**"NNY"**
Nav Map Normal Z	**"NNZ"**
Nav Map Tilt	**"NTL"**
Nav Map Residual	**"NRS"**
Nav Map Offset	**"NOF"**
Nav Map Moments	**"NMO"**
Nav Map Footprint	**"NFP"**
Nav Map Rock	**"NRK"**
Nav Map Minimum Cells	**"NMC"**
Nav Map FOV	**"NFV"**
Nav Map FOV Edge	**"NFE"**
Nav Map Path Information	**"NPI"**
Nav Map Layer Certainty	**"NLY"**
Nav Map Elevation Difference	**"NED"**
Nav Map Layer 0	**"NL0"**
Nav Map Layer 1	**"NL1"**
Nav Map Layer 2	**"NL2"**
Nav Map Layer 3	**"NL3"**
Nav Map Layer 4	**"NL4"**
Nav Map Layer 5	**"NL5"**
Nav Map Layer 6	**"NL6"**
Nav Map Layer 7	**"NL7"**
Nav Map Layer 8	**"NL8"**
Nav Map Layer 9	**"NL9"**
Nav Map Layer 10	**"NLA"**
Nav Map Layer 11	**"NLB"**
Nav Map Layer 12	**"NLC"**
Nav Map Layer 13	**"NLD"**
Nav Map Layer 14	**"NLE"**
Nav Map Layer 15	**"NLF"**
Nav Map Temporary Storage Layer 0	**"NT0"**
Nav Map Temporary Storage Layer 1	**"NT1"**
Nav Map Temporary Storage Layer 2	**"NT2"**
Nav Map Temporary Storage Layer 3	**"NT3"**
Nav Map Temporary Storage Layer 4	**"NT4"**
Nav Map Temporary Storage Layer 5	**"NT5"**
Nav Map Temporary Storage Layer 6	**"NT6"**
Nav Map Temporary Storage Layer 7	**"NT7"**
Nav Map Temporary Storage Layer 8	**"NT8"**
Nav Map Temporary Storage Layer 9	**"NT9"**

EDR Product Type Description	Value
Nav Map Temporary Storage Layer 10	**"NTA"**
Nav Map Temporary Storage Layer 11	**"NTB"**
Nav Map Temporary Storage Layer 12	**"NTC"**
Nav Map Temporary Storage Layer 13	**"NTD"**
Nav Map Temporary Storage Layer 14	**"NTE"**
Nav Map Temporary Storage Layer 15	**"NTF"**
Nav Map Wraparound	**"NWR"**
Nav Map Stereo Filters	**"NSF"**
Nav Map Stereo Disparity	**"NSD"**
Nav Map Left Rectified	**"NLR"**
Nav Map Right Rectified	**"NRR"**
Nav Map Second Stereo Filters	**"N2F"**
Nav Map Second Stereo Disparity	**"N2D"**
Nav Map Second Left Rectified	**"N2L"**
Nav Map Second Right Rectified	**"N2R"**
Nav Map VO Features	**"NVF"**
Nav Map Memory Manager Status	**"NMS"**
Nav Map D-star Cost	**"NDC"**
Nav Map D-star Planning Cost	**"NPC"**
Nav Map D-star Layer Cost	**"NDR"**
Nav Map D-star Field	**"NDF"**
Nav Map D-star Look Ahead	**"NDL"**
Nav Map Keepout	**"NKO"**
Nav Map Keepout Site	**"NKS"**
Nav Map Keepout Path Site	**"NKP"**
Nav Map IDPH	**"NMI"**
Nav Map Step Goodness	**"NSG"**
Nav Map Tilt Goodness	**"NTG"**
Nav Map Roughness Goodness	**"NRG"**
Nav Map Good Keep	**"NGK"**
Nav Map Stereo Points	**"NSP"**
Nav Map Num Entries	**"NNE"**

Table 6.1.1.2: Valid Product Identifiers ("prodid") for Image RDRs

Image RDR Product Type Description	Value
CAHV-linearized (identical to "EDR" except for Geometry type)	**"LIN"**
Bayer pattern (identical to "EDR" except for Bayer pattern extract)	**"BAY"**
Inverse lookup table (ILUT)	**"ILT"**
Inverse lookup table (ILUT) with corrections such as despike	**"ILC"**

Image RDR Product Type Description	Value
Inverse lookup table (ILUT) with pointing correction applied	**“ILP”**
Image Mask File	**“MSK”**
Rad-corrected absolute radiance units, integer	**“RAD”**
Rad-corrected absolute radiance units, scaled to 12-bit	**“RAS”**
Rad-corrected absolute radiance units, float	**“RAF”**
Rad-corrected for Instrument Effects only, integer DN	**“RIE”**
Rad-corrected for Instrument Effects only, float	**“RIF”**
Rad-corrected IOF radiance factor, integer	**“IOI”**
Rad-corrected IOF radiance factor, float	**“IOF”**
ChemCam RMI Partially Rad-corrected	**“PRC”**
MMM-produced Rad-corrected	**“DRX”**
MMM-produced Rad-corrected and White Balanced	**“DRW”**
Stereo Disparity Final	**“DSP”**
Stereo Disparity Raw	**“DSR”**
Stereo Disparity of Lines (single-band)	**“DSL”**
Stereo Disparity of Samples (single-band)	**“DSS”**
Stereo Disparity Grid	**“DSG”**
Stereo Disparity Error Metric	**“DSE”**
Stereo Disparity Mask File	**“MDS”**
Stereo Delta Disparity (2-band, true disparity offset)	**“DDD”**
Stereo Delta Disparity Line (single-band)	**“DDL”**
Stereo Delta Disparity Sample (single-band)	**“DDS”**
Stereo First-stage Disparity Final	**“DFF”**
Stereo First-stage Disparity Line	**“DFL”**
Stereo First-stage Disparity Sample	**“DFS”**
XYZ expressed in Site frame	**“XYZ”**
XYZ Error Metric	**“XYE”**
XYZ expressed in Rover Nav frame	**“XYR”**
XYZ Masked	**“XYM”**
XYZ Mask File	**“MXY”**
XYZ Filled	**“XYF”**
XYZ with Overlay	**“XYO”**
XYZ X-band	**“XXX”**
XYZ X-band Filled	**“XXF”**
XYZ Y-band	**“YYY”**
XYZ Y-band Filled	**“YYF”**
XYZ Z-band	**“ZZZ”**
XYZ Z-band Filled	**“ZZF”**
XYZ Z-band with Overlay	**“ZZO”**
Surface Normal (UVW)	**“UVW”**

Image RDR Product Type Description	Value
Surface Normal (UVW) for Slope computations	**"UVS"**
Surface Normal (UVW) Projected onto Plane	**"UVP"**
Surface Normal (UVW) Angle ('T' for theta) between Normal and Plane	**"UVT"**
Surface Normal (UVW) with Overlay	**"UVO"**
Surface Normal (UVW) Filled	**"UVF"**
Surface Normal (UVW) U-band	**"UUU"**
Surface Normal (UVW) V-band	**"VVV"**
Surface Normal (UVW) W-band	**"WWW"**
Surface Normal (UVW) U-band Filled	**"UUF"**
Surface Normal (UVW) V-band Filled	**"VVF"**
Surface Normal (UVW) W-band Filled	**"WWF"**
Surface Roughness (Drill)	**"RUD"**
Surface Roughness (DRT)	**"RUT"**
Range from Camera	**"RNG"**
Range from ChemCam RMI	**"CXR"**
Range from Camera, Masked	**"RNM"**
Range from Rover Nav frame origin	**"RNR"**
Range with Overlay	**"RNO"**
Range Filled	**"RNF"**
Range Error Metric	**"RNE"**
Arm Reachability	**"ARM"**
Arm Reachability with Overlay	**"ARO"**
Arm Reachability Masked	**"ARK"**
Arm Reachability Mask File	**"MAR"**
Arm Preload Values	**"ARP"**
Slope	**"SLP"**
Slope with Overlay	**"SLO"**
Slope Rover Direction	**"SRD"**
Slope Rover Direction with Overlay	**"SRO"**
Slope Heading	**"SHD"**
Slope Heading with Overlay	**"SHO"**
Slope Magnitude	**"SMG"**
Slope Magnitude with Overlay	**"SMO"**
Slope Northerly Tilt	**"SNT"**
Slope Northerly Tilt with Overlay	**"SNO"**
Solar Energy	**"SEN"**
Solar Energy with Overlay	**"SEO"**
Incidence, Emission, Phase angles	**"IEP"**
Incidence, Emission, Phase angles Filled	**"IEF"**
Terrain Classifier Probability Map	**"TEN"**

Image RDR Product Type Description	Value
Terrain Classification Map	**"TER"**
Mosaic-to-Image Coregistration Map	**"ICM"**
Mosaic-to-Image Coregistration Index	**"IDX"**
ChemCam Finder Error Mask	**"MCE"**
ChemCam Finder Risk Mask	**"MCR"**
ChemCam Finder Combo Risk/Error Mask	**"MCF"**
ChemCam Exclusion Zone Mask	**"CXM"**
AEGIS Target Map (with "AGM" JSON target file)	**"AGS"**
AEGIS JSON Target File	**"AGM"**

Table 6.1.1.3: Valid Product Identifiers ("prodid") for Spectroscopy (LIBS) RDRs

Spectroscopy (LIBS) RDR Product Type Description	Value
Initial LIBS Spectrum	**"RDR"**
Multivariate Prediction of Oxide Composition	**"MOC"**
Intermediate Clean Calibrated Spectra	**"CCS"**

geom = (1 alphanumeric) Geometry type, used to differentiate products as having camera-induced distortion removed ("linearized") or not removed ("non-linearized"). For MMM camera products, also serves a secondary function by flagging changes in compression mode across iterations of the same data to distinguish between MMM product types.

Products from stereo cameras are normally linearized using the nominal stereo partner (whether the partner image was actually acquired or not). Products such as long-baseline or re-pointed stereo pairs, however, are often linearized using the actual stereo partner, resulting in different geometry.

This field has the following rules-of-thumb:

a) Linearization - If value is any alpha character "**A** - **Z**", then product is "linearized" using one of the two modes (nominal or actual) according to the table below. If value is <u>not</u> any alpha character, then product is "non-linearized".

b) First Compression Mode - For all cameras, if value is an underscore ("_"), "**L**" (for "linearized-nominal") or "**A**" (for "linearized-actual"), it denotes that compression mode is for the <u>first</u> instance of the product data processed on the ground. The identity of the mode type, whether "uncompressed" or a type of compression, is not provided here.

c) Changed Compression Modes - For MMM cameras, if value is an integer "**1** - **9, 0**" (iterating numerically in order of 1, 2 … 9, 0), or in the character ranges "**M** - **V**" (iterating alphabetically after "L" for "linearized-nominal") or "**B** - **K**" (iterating alphabetically after "A" for "linearized-actual"), it indicates a <u>different</u> compression mode than all previous modes noted. Again, the identity of the mode type, whether it's "uncompressed" or a type of compression, is not provided here.

These flags are necessary because operation of MMM cameras includes the possibility that the same image be requested multiple times from the rover, each with different compression modes. In these cases, the compression mode for the first product processed on the ground is noted. Thereafter, if the compression mode for any subsequent product processed on the ground is different than all previous modes, it will be noted and indexed incrementally (using alpha characters for "linearized" and integers for "non-linearized").

NOTE: Retransmissions or reprocessing of the same compression mode will reuse the appropriate mode character, with filename's Version field incrementing for each retransmission.

See the following table of valid values:

Camera Types			Non-linearized	Linearized: Nominal	Linearized: Actual
Camera Types	All	1st compression mode received	**“_”**	**“L”**	**“A”**
	MMM Only	2nd compression mode, different than any previous mode	**“1”**	**“M”**	**“B”**
		3rd compression mode, different than any previous mode	**“2”**	**“N”**	**“C”**
		• • •	• • •		
		10th compression mode, different than any previous mode	**“9”**	**“U”**	**“J”**
		11th compression mode or higher, different than any previous mode	**“0”**	**“V”**	**“K”**

NOTE: The EDR controls the compression mode flag. All RDRs derived from a given EDR share the same row in the table above. So, “2” and “N” and “C” will always correspond, for example.

samp = (1 character) Sample type, indicating how the data was sampled.

Valid values for Sample type are:

- “**F**” - Full frame raster data, full resolution
- “**S**” - Subframed raster data, full resolution
- “**D**” - Downsampled raster data, reduced resolution
- “**M**” - Mixed (Subframe and Downsampled) raster data, mixed resolution
- “**T**” - Thumbnail raster data, reduced resolution
- “**B**” - Bayer extraction subsampling (MMM only) raster data
- “**Y**” - Thumbnail Bayer extraction subsampling (MMM only) raster data
- “**N**” - Non-raster data

site / video = (3 alphanumeric) Site location count, from the RMC.

This field has the following rules-of-thumb:

a) Site - If value is any 3 alphanumeric characters, or 3 underscores (denoting value is out-of-range), then content represents Site index extracted from RMC.

b) Video Subframe / Recovered Data - If first character is underscore (“_”) followed by integers in remaining positions, then value occupies Site and Drive fields and represents MMM data in either of two scenarios: “Video Subframe” or “Recovered Data”. In total, this consists of a leading underscore followed by 5 integers followed by a trailing underscore (example: "_00000_"). The 5 integers are 5 bytes in HEX representation. They are used to disambiguate multiple frames occurring in the same second. The meaning of the values slightly differ according to the scenario:

- Video Subframe - The 4 leftmost of these are the lowest 4 bytes of the Product ID entity in the MMM mini-header. The 5th rightmost integer is the Frame Number entity in the MMM mini-header, indicating which frame the image is (out of a possible 16 that can be packed into a single video data product).
- Recovered Data - The 5 Hex values denote the Image ID to uniquely identify products that share the same SCLKs.

The valid Site values, in their progression, are as follows (non-Hex):

Range 000 thru 999 - "**000**", "**001**", … "**999**"
Range 1000 thru 1099 - "**A00**", "**A01**", … "**A99**"
Range 1100 thru 1199 - "**B00**", "**B01**", … "**B99**"
•
•
•
Range 3500 thru 3599 - "**Z00**", "**Z01**", … "**Z99**"
Range 3600 thru 3609 - "**AA0**", "**AA1**", … "**AA9**"
Range 3610 thru 3619 - "**AB0**", "**AB1**", … "**AB9**"
•
•
•
Range 3850 thru 3859 - "**AZ0**", "**AZ1**", … "**AZ9**"
Range 3860 thru 3869 - "**BA0**", "**BA1**", … "**BA9**"
Range 3870 thru 3879 - "**BB0**", "**BB1**", … "**BB9**"
•
•
•
Range 10350 thru 10359 - "**ZZ0**", "**ZZ1**", … "**ZZ9**"
Range 10360 thru 10385 - "**AAA**", "**AAB**", … "**AAZ**"
Range 10386 thru 10411 - "**ABA**", "**ABB**", … "**ABZ**"
•
•
•
Range 27910 thru 27935 - "**ZZA**", "**ZZB**", … "**ZZZ**"
Range 27936 thru 27961 - "**0AA**", "**0AB**", … "**0AZ**"
Range 27962 thru 27987 - "**0BA**", "**0BB**", … "**0BZ**"
•
•
•
Range 32720 thru 32745 - "**7CA**", "**7CB**", … "**7CZ**"
Range 32746 thru 32767 - "**7DA**", "**7DB**", … "**7DV**"
Value is out of range - "**_ _ _**" (3 consecutive underscores)

drive / video = (4 alphanumeric) Drive (position-within-Site) location count, from the RMC.

This field has the following rules-of-thumb:

a) Drive - If value is any 4 alphanumeric characters, or 4 underscores (denoting value is out-of-range), then content represents Drive index extracted from RMC.
b) Video Subframe / Recovered Data - If last character is underscore ("_") preceded by integers in remaining positions, then value occupies Site and Drive fields and represents MMM data in either of two scenarios: "Video Subframe" or "Recovered Data". See "b" in Site field description above for more details.

The valid Drive values, in their progression, are as follows (non-Hex):

Range 0000 thru 9999 - "**0000**", "**0001**", … "**9999**"
Range 10000 thru 10999 - "**A000**", "**A001**", … "**A999**"
•
•
•
Range 35000 thru 35999 - "**Z000**", "**Z001**", … "**Z999**"
Range 36000 thru 36099 - "**AA00**", "**AA01**", … "**AA99**"
•
•
•

Range 38500 thru 38599 - “**AZ00**”, “**AZ01**”, **…** “**AZ99**”
Range 38600 thru 38699 - “**BA00**”, “**BA01**”, **…** “**BA99**”
•
•
•
Range 65500 thru 65535 - “**LJ00**”, “**LJ01**”, **…** “**LJ35**”
Value is out of range - “**_ _ _ _**” (4 consecutive underscores)

seqid = (9 alphanumeric) Sequence identifier. Composed of a 4-char subfield and a 5-digit numeric subfield representing the 6-bit “Category” and 14-bit numeric components of the commanded Sequence ID, respectively.

venue / who = (1 character) Venue and Product Producer ID shared in the same field.

Venue denotes Flight Model versus Engineering Model in data acquisition. Product Producer ID identifies the institution that generated the product.

This field has the following rules-of-thumb:

a) Venue - A value in the range “**A** - **P**” indicates Flight Model rover. A value in the range “**Q** - **Z**” indicates Engineering (testbed) rover. The range “N - O” is not used.
b) Producer - If value is “**P**” (for Flight) or “**Y**” (for Engineering), the provider of the product is the Principal Investigator. Except for MIPL as the provider (“**M**” for Flight or “**Z**” for Engineering), the remaining characters are assigned to Co-investigator providers at the discretion of the P.I. and will be identified in due time. Within the instrument of the P.I., characters are unique. Across instruments, characters are reusable.

See the following table of valid values:

Venue		**by Producer**
Flight Model	Eng. Model	
“**M**”	“**Z**”	**M**IPL (OPGS at JPL)
“**P**”	“**Y**”	**P**rincipal Investigator of Instrument … <u>Instrument</u> — <u>Principal Investigator</u> MMM Cameras — MSSS (San Diego, CA) ChemCam LIBS & SOH — LANL (Los Alamos, NM) ChemCam RMI — IRAP (France)
“**A**” -	“**Q**” - “**X**”	Co-Investigators (to be identified by P.I. per instrument)

See the following table of valid values for Instruments not covered by this SIS:

Venue		**by Producer**
Flight Model	Eng. Model	
“**M**”	“**Z**”	**M**IPL (OPGS at JPL)
“**P**”	“**Y**”	**P**rincipal Investigator of Instrument … <u>Instrument</u> — <u>Principal Investigator</u> SAM — GSFC (Goddard, MD) REMS — Ministry of Education & Science (Spain) DAN — Federal Space Agency (Russia) RAD — SwRI (Boulder, CO) CheMin — Ames Research Center (Mountain View, CA) APXS — Max-Planck Institute (Germany) SA/SPaH — JPL
“**A**” -	“**Q**” - “**X**”	Co-Investigators (to be identified by P.I. per instrument)

ver = (1 alphanumeric) Version identifier. The Version number increments by one whenever an otherwise-identical filename would be produced.

The valid values, in their progression that <u>excludes</u> "**0**" altogether, are as follows (non-Hex):

Range 1 thru 9 - "**1**", "**2**", **…** "**9**"
Range 10 thru 35 - "**A**", "**B**", **…** "**Z**"
Range 36 and higher - "**_**" (underscore)

Note that not every version need exist, e.g. versions 1, 2 and 4 may exist but not 3. In general, the highest-numbered Version represents the "best" version of that product.

NOTE: To be clear, this field increments independently of all fields, including the Special Processing field.

ext = (2 to 3 characters) Product type extension.

Valid values for nominal operations camera data products:

"**IMG**" - Image EDRs / RDRs (ODL label, may include embedded VICAR label)
"**VIC**" - Temporary image EDR / RDRs with only VICAR label (no ODL label)
"**iv**" - per-XYZ Terrain Mesh RDR in Inventor binary format (no label)
"**ht**" - per-XYZ Height Map RDR (VICAR label)
"**rgb**" - per-XYZ Terrain Mesh texture RDR in RGB format (no label)
"**LBL**" - Detached label file in PDS or ODL format
"**JPG**" - JPEG compressed (no label)
"**TIF**" - TIFF format (no label)
"**PNG**" - PNG format (no label)
"**TXT**" - Text file associated with Specially-processed files (no label)
"**tar**" - Tar file containing per-XYZ Terrain Mesh vertices tile files (no label)

Valid values for nominal operations non-camera data products:

"**QUB**" - Multi-layer spectral cube data
"**CSV**" - Comma-separated-value text file, used for LIBS RDRs
"**DAT**" - Non-imaging instrument data, including LIBS and ChemCam state-of-health
"**LBL**" - Detached label in PDS or ODL format
"**TAB**" - Table data

Of the above, only "IMG", "LBL", "JPG", "TXT", "QUB", "CSV", "DAT" and "TAB" are currently supported by PDS.

Example #1: NRA_013760215EDR_F0930008NCAM22103M1.IMG
where,

instr	=	"**NR**"	=	Navcam Right
config	=	"**A**"	=	"A-side" configuration
spec	=	"**_**"	=	No special processing
sclk	=	"**013760215**"	=	Spacecraft Clock Start Count of 13760215 secs
prod	=	"**EDR**"	=	Image EDR
geom	=	"**_**"	=	Raw (non-linearized)
samp	=	"**F**"	=	Full frame
site	=	"**093**"	=	Site 93
drive	=	"**0008**"	=	Drive (Position-within-Site) 8
seqid	=	"**NCAM22103**"	=	Command Sequence ncam22103
venue / who	=	"**M**"	=	Flight Model data / produced by MIPL (at JPL)
ver	=	"**1**"	=	Version 1
ext	=	"**IMG**"	=	Image product with ODL label

Example #2: FLBA012885634XYZLS0320154FHAZ00348Z1.IMG
where,

instr	=	"**FL**"	=	Front Hazcam Left
config	=	"**B**"	=	"B-side" configuration
spec	=	"**A**"	=	Special processing method "A" (defined in a text file)
sclk	=	"**012885634**"	=	Spacecraft Clock Start Count of 12885634 secs
prod	=	"**XYZ**"	=	XYZ RDR
geom	=	"**L**"	=	Linearized
samp	=	"**S**"	=	Subframe
site	=	"**032**"	=	Site 32
drive	=	"**0154**"	=	Drive (Position-within-Site) 154
seqid	=	"**FHAZ00348**"	=	Command Sequence fhaz00348
venue / who	=	"**Z**"	=	Eng. Model (testbed) data / produced by MIPL (at JPL)
ver	=	"**1**"	=	Version 1
ext	=	"**IMG**"	=	Image product with ODL label

Example #3: MR1_012500462EDR3F_00002_MCAM00117P1.JPG
where,

instr	=	"**MR**"	=	Mastcam Right
config	=	"**1**"	=	Filter 1
spec	=	"**_**"	=	No special processing
sclk	=	"**012500462**"	=	Spacecraft Clock Start Count of 12500462 secs
prod	=	"**EDR**"	=	Image EDR
geom	=	"**3**"	=	4th different compression of same data, non-linearized
samp	=	"**F**"	=	Full frame
video	=	"**00002**"	=	Video Subframe count 00002 (occupies Site/Drive content)
seqid	=	"**MCAM00117**"	=	Command Sequence mcam00117
venue / who	=	"**P**"	=	Flight Model data / produced by P.I. of Mastcam (MSSS)
ver	=	"**1**"	=	Version 1
ext	=	"**JPG**"	=	JPEG product

6.1.2 Mosaic RDR Filename

The MSL camera Mosaic RDR data products are usually derived from multiple EDR or RDR data products mosaicked together, although they can also be derived from single data products. They are uniquely identified by incorporating into the product filename the Lander mission identifier, the "primary" Instrument identifier, the "secondary" Instrument identifier, the starting Sol denoting the start of mosaic data, the geometric Projection type, the Product Type ingested to build the mosaic, the starting Site location, the rover's starting Position within the site, the camera "Eye", the spectral Filter, the product Creator identifier and a Version number. The convention is illustrated in Figure 6.1.2 below.

The filename convention follows:

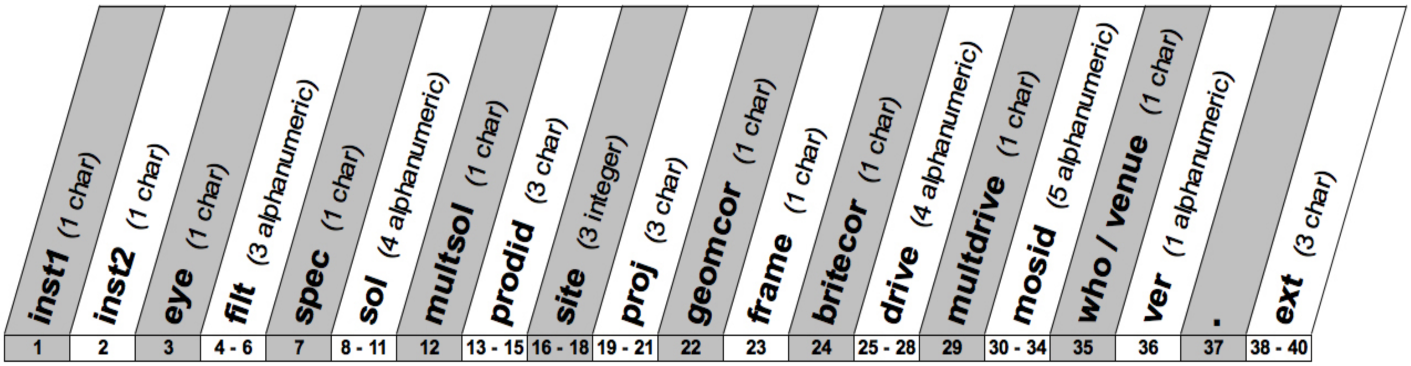


Figure 6.1.2 – Mosaic RDR Filename Convention

inst1 = (1 character) Instrument ID primary, denoting the source MSL science or engineering instrument that acquired the data. Valid values are:

- "**F**" - Front Hazcam
- "**R**" - Rear Hazcam
- "**N**" - Navcam
- "**M**" - Mastcam
- "**H**" - MAHLI
- "**D**" - MARDI
- "**C**" - ChemCam RMI

inst2 = (1 character) Instrument ID secondary, denoting the source MSL science or engineering instrument that acquired the data. Valid values are:

- "**F**" - Front Hazcam
- "**R**" - Rear Hazcam
- "**N**" - Navcam
- "**M**" - Mastcam
- "**H**" - MAHLI
- "**D**" - MARDI
- "**C**" - ChemCam RMI
- "**X**" - flag for more than 2 instruments
- "**_**" - flag for no 2nd instrument

eye = (1 character) Camera eye. Valid values are:

- "**L**" - Left camera eye
- "**R**" - Right camera eye
- "**M**" - Monoscopic (non-stereo camera)
- "**S**" - Stereo (2-band)
- "**C**" - Color stereo (6-band)
- "**A**" - Anaglyph
- "**G**" - Colorglyph (Lr, Rg, Rb)
- "**N**" - Not applicable
- "**X**" - Mixed

filt = (3 alphanumeric) Spectral filter position. Valid values are:

- "**0**" - Non-filter camera
- "**0 - 8**" - Filter for MMM cameras
- "**R**", "**G**", "**B**" - sRGB color space or de-Bayered results
- "**X**", "**Y**", "**I**" - xyY color space
- "**A**" - Alternative (define alternative color spaces if needed)

spec = (1 character) Special processing flag. See description and valid values for field "spec" in Section 6.1.1.

Valid values are:

Special Processing	Value
none	"_"
Special method types A-S and U-Z	"**A**" - "**S**", "**U**" - "**Z**"
Best tactical Special method	"**T**"

sol = (4 alphanumeric) Primary Sol (for simulated or nominal surface ops) or Day of Year (DOY, for some testbed activities) included in mosaic. Nominally, it is the Sol or DOY of the last (in time order, the highest SCLK) input image built into the mosaic. This can change in off-nominal cases when special circumstances prevail.

The valid values, in their progression, are as follows (non-Hex):

Sol / DOY	Range	Values
Sol	0000 thru 9999	"**0000**", "**0001**", … "**9999**"
	10000 thru 10999	"**A000**", "**A001**", … "**A999**"
	11000 thru 11999	"**B000**", "**B001**", … "**B999**"
	• • •	
	33000 thru 33999	"**X000**", "**X001**", … "**X999**"
	Value is out of range	"_ _ _ _" (4 consecutive underscores)
DOY	001 thru 365 (2012)	"**Y001**", "**Y002**", … "**Y365**"
	001 thru 365 (2013)	"**Z001**", "**Z002**", … "**Z365**"
	Value is out of range	"_ _ _ _" (4 consecutive underscores)

multsol = (1 character) Flag indicating that a "significant" percentage of the data content was acquired on more than a single Sol, i.e., across multiple Sols. Specification of this flag is at the discretion of the Mosaic product's provider, who determines what percentage is "significant". Therefore, it is possible for a Mosaic product to contain data acquired across multiple Sols, but unless the provider determines that the portions are significant, this flag may not be activated.

Valid values are:

"**_**" - Flags field Sol as solitary
"**X**" - Flags field Sol as one of multiple Sols

prodid = (3 characters) Input Product type identifier. See description and valid values for field "prodid" in Section 6.1.1.

site = (3 alphanumeric) Site location count, from the RMC. See description and valid values for field "site" in Section 6.1.1.

proj = (3 characters) Projection type. Valid values are:

"**CYL**" - Cylindrical
"**PER**" - Perspective
"**CYP**" - Cylindrical-Perspective
"**POL**" - Polar
"**ORT**" - Orthographic
"**ORR**" - Orthorectified
"**VRT**" - Vertical (special case of Orthographic)

geomcor = (1 character) Geometric correction type indicator. Specifies the correction type that was applied to the largest percentage of data content.

Valid values are:

"**_**" - No correction (raw pointing)
"**A**" - Auto-correction via tiepointing
"**T**" - Manual tiepointing
"**R**" - Manual tiepointing & auto-registration

- "**F**" - Auto-correction via tiepointing & auto-registration with fiducials
- "**G**" - Auto-correction via tiepointing & manual registration with fiducials
- "**M**" - Manual tiepointing & manual registration with fiducials
- "**O**" - Other correction not listed above

frame = (1 character) Coordinate system (frame) type. Valid values are:

- "**S**" - Site frame
- "**L**" - Local Level frame
- "**R**" - Rover frame
- "**U**" - Untilt (CYP only, is Rover frame with rotation)
- "**O**" - Other

britecor = (1 character) Brightness correction type indicator. Specifies the correction type that was applied to the largest percentage of data content.

Valid values are:

- "**_**" - No correction
- "**B**" - Automatic brightness adjustment (multiplicative and/or additive factor applied to each frame)
- "**M**" - Manual brightness adjustment (same factors as "B")
- "**V**" - Anti-vignetting adjustment applied to some or all frames
- "**G**" - General brightness correction (manual)
- "**A**" - General brightness correction that can vary across the frame (automatic)
- "**O**" - Other correction not listed above

drive = (4 alphanumeric) Drive location count, from the RMC. See description and valid values for field "drive" in Section 6.1.1.

multdrive = (1 character) Flag indicating data content from multiple Drives (Positions). Valid values are:

- "**_**" - Flags field Drive as solitary.
- "**X**" - Flags field Drive as <u>last</u> of multiple Drives.

mosid = (5 alphanumeric) General purpose mosaic identifier. Can be set to anything to help identify the mosaic, such as target name, panorama name, theme name, etc. Valid values include "**A** - **Z**", "**0** - **9**" and underscore. Must always pad to 5 characters, using underscores as necessary.

venue / who = (1 character) Venue and Product Producer ID shared in the same field. See description and valid values for field "venue / who" in Section 6.1.1.

ver = (1 alphanumeric) Version identifier. The Version number increments by one whenever an otherwise-identical filename would be produced.

The valid values, in their progression that <u>excludes</u> "**0**" altogether, are as follows (non-Hex):

- Range 1 thru 9 - "**1**", "**2**", **…** "**9**"
- Range 10 thru 35 - "**A**", "**B**", **…** "**Z**"
- Range 36 and higher - "**_**" (underscore)

Note that not every version need exist, e.g. versions 1, 2 and 4 may exist but not 3. In general, the highest-numbered Version represents the "best" version of that product.

NOTE: To be clear, this field increments independently of all fields, including the Special Processing field.

ext = (2 to 3 characters) Product type extension.

Valid values for nominal operations camera data products:

- "**IMG**" - Image EDRs / RDRs (ODL label, may include embedded VICAR label)
- "**JPG**" - JPEG compressed (detached label)
- "**LIS**" - ASCII filelist of component EDR / RDR filenames comprising the mosaic
- "**NAV**" - Navigation (geometric correction) file

“**BRT**” - Brightness correction file
“**TIE**” - Tiepoint file
“**OVR**” - Overlap file

Example: N_L000_0060XILT005CYLTSG0104_DRIVEM1.IMG
where,

instr1	=	“**N**”	=	Navcam
instr2	=	“**_**”	=	No 2nd instrument
eye	=	“**L**”	=	Left camera eye
filt	=	“**000**”	=	Non-filter camera
spec	=	“**_**”	=	No special processing
sol	=	“**0060**”	=	Sol 60 as Sol of last (highest SCLK) input product
multsol	=	“**X**”	=	Flags Sol 60 as last of multiple Sols
prodid	=	“**ILT**”	=	Input products are ILUT RDRs
site	=	“**005**”	=	Site 5
proj	=	“**CYL**”	=	Cylindrical projection
geomcor	=	“**T**”	=	Geometric correction method is “manual tiepointing”
frame	=	“**S**”	=	Coordinate system (frame) is Site
britecor	=	“**G**”	=	Brightness correction method is “General” (manual)
drive	=	“**0104**”	=	Drive (Position-within-Site) 104
multdrive	=	“**_**”	=	Flags Drive 104 as only Drive
mosid	=	“**DRIVE**”	=	Arbitrary string identifying product as type of Drive mosaic
venue / who	=	“**M**”	=	Identifies MIPL/OPGS as product provider
ver	=	“**1**”	=	Version 1
ext	=	“**IMG**”	=	Image product with ODL label

6.1.3 Unified Terrain Mesh RDR Filename

Each MSL unified Terrain Mesh RDR product can be uniquely identified by incorporating into the product filename the Instrument identifier, the Starting Sol and Ending Sol, the Product Type of the mesh surface texture (“skin”), the Geometry type (linearized vs. nonlinearized), the Site and Drive, the camera Eye and a Version number. The filename complies with the PDS 36.3 standard and is variable length, due to the variable length of the Instrument identifier field. The convention is illustrated in Figure 6.1.3 below.

The filename convention follows:

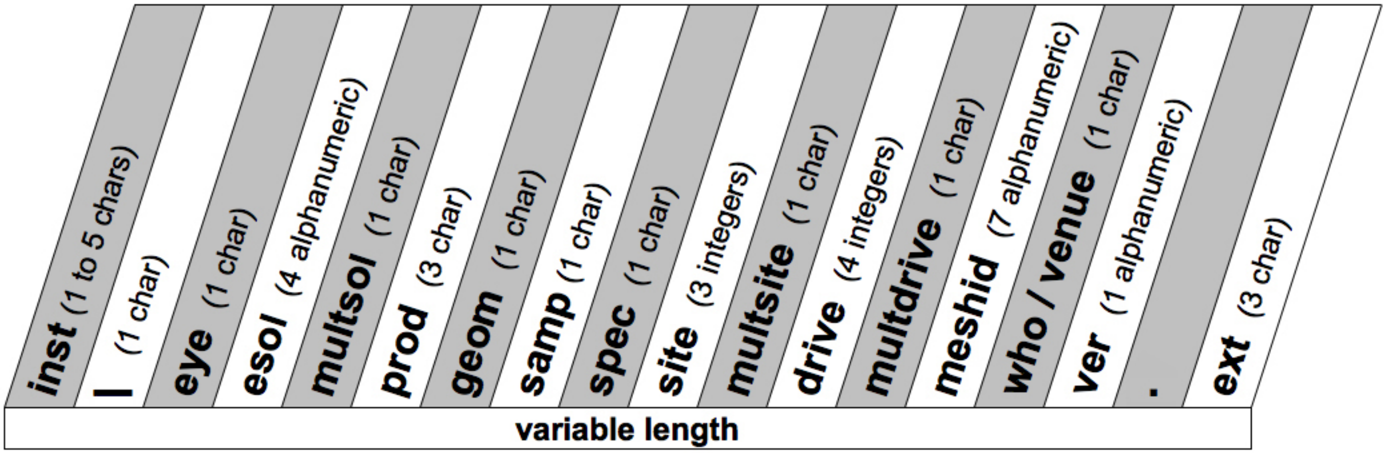


Figure 6.1.3 – "Unified" Terrain Mesh RDR Filename Convention

inst = (1 to 5 characters) One or more Instrument ID's, denoting the source MSL science or engineering instrument(s) and/or orbiter that acquired the data. Note that if data were acquired by all six instruments, a single "A" character will appear instead. Valid values are:

- "**F**" - Front Hazcam
- "**R**" - Rear Hazcam
- "**N**" - Navcam
- "**M**" - Mastcam
- "**H**" - MAHLI
- "**O**" - Orbiter
- "**A**" - All six instruments

eye = (1 character) Indicates Camera eye that was referenced during stereo correlation. Valid values are:

- "**L**" - Left-to-right stereo correlation
- "**R**" - Right-to-left stereo correlation
- "**M**" - Monoscopic (non-stereo camera)
- "**N**" - Not applicable
- "**X**" - Mixed

esol = (4 alphanumeric) Ending Sol (for simulated or nominal surface ops) or Day of Year (DOY, for some testbed activities) included in mesh. Indicates the Sol or DOY of the last (in time order, the highest SCLK) input image built into the mesh.

The valid values, in their progression, are as follows (non-Hex):

Sol / DOY	Range	Values
Sol	0000 thru 9999	"**0000**", "**0001**", … "**9999**"
	10000 thru 10999	"**A000**", "**A001**", … "**A999**"
	11000 thru 11999	"**B000**", "**B001**", … "**B999**"
	• • •	
	33000 thru 33999	"**X000**", "**X001**", … "**X999**"
	Value is out of range	"_ _ _ _" (4 consecutive underscores)
DOY	001 thru 365 (2012)	"**Y001**", "**Y002**", … "**Y365**"
	001 thru 365 (2013)	"**Z001**", "**Z002**", … "**Z365**"
	Value is out of range	"_ _ _ _" (4 consecutive underscores)

multsol = (1 character) Flag indicating data content from multiple Sols. Valid values are:
- "**_**" - Flags field Esol as solitary.
- "**X**" - Flags field Esol as last of multiple Sols.

prodid = (3 characters) Product type identifier of the mesh surface texture ("skin"). See description and valid values for field "prodid" in Section 6.1.1.

geom = (1 alphanumeric) Geometry type, used to differentiate products as having camera-induced distortion removed ("linearized") or not removed ("non-linearized"). See description for field "geom" in Section 6.1.1.

Valid values are:

Non-linearized	**Linearized**	
	Nominal	Actual
"**_**"	"**L**" - "**V**"	"**A**" - "**K**"

samp = (1 character) Sample type, indicating how the data was sampled.

Valid values for Sample type are:
- "**F**" - Full frame raster data, full resolution
- "**S**" - Subframed raster data, full resolution
- "**D**" - Downsampled raster data, reduced resolution
- "**M**" - Mixed (Subframe and Downsampled) raster data, mixed resolution
- "**T**" - Thumbnail raster data, reduced resolution
- "**B**" - Bayer extraction subsampling (MMM only) raster data
- "**Y**" - Thumbnail Bayer extraction subsampling (MMM only) raster data
- "**N**" - Non-raster data

spec = (1 character) Special processing flag. See description for field "spec" in Section 6.1.1.

Valid values are:

Special Processing	**Value**
None	"**_**"
Special method types A-S and U-Z	"**A**" - "**S**", "**U**" - "**Z**"
Best tactical Special method	"**T**"

site = (3 alphanumeric) Site location count, from the RMC. See description and valid values for field "site" in Section 6.1.1.

multsite = (1 character) Flag indicating data content from multiple Sites. Valid values are:
- "**_**" - Flags field Site as solitary.
- "**X**" - Flags field Site as last of multiple Sites.

drive = (4 alphanumeric) Drive location count, from the RMC. See description and valid values for field "drive" in Section 6.1.1.

multdrive = (1 character) Flag indicating data content from multiple Drives (Positions). Valid values are:
- "**_**" - Flags field Drive as solitary.
- "**X**" - Flags field Drive as last of multiple Drives.

meshid = (7 alphanumeric) General purpose Mesh identifier. Can be set to anything to help identify the Mesh, such as target name, panorama name, theme name, etc. Valid values include "**A**

- **Z**", "**0** - **9**" and underscore. For Meshes, this field must ensure uniqueness in like-named Mesh filenames of identical Site, Drive and Instrument specification. This varies from Mosid field used in Mosaics. Must always pad to 7 characters, using underscores as necessary.

Pipeline (automated) Use Case	Value
Hazcam content only	Last 7 digits of XYZ's SCLK
All other Instrument content (including Hazcam combined with other)	"**AUTOGEN**"

venue / who = (1 character) Venue and Product Producer ID shared in the same field. See description and valid values for field "venue / who" in Section 6.1.1.

ver = (1 alphanumeric) Version identifier. The Version number increments by one when the current version is locked (in-use) by the operations team.

The valid values, in their progression that <u>excludes</u> "**0**" altogether, are as follows (non-Hex):

Range 1 thru 9 - "**1**", "**2**", **…** "**9**"
Range 10 thru 35 - "**A**", "**B**", **…** "**Z**"
Range 36 and higher - "**_**" (underscore)

Note that not every version need exist, e.g. versions 1, 2 and 4 may exist but not 3. In general, the highest-numbered Version represents the most recent version of that product.

ext = (2 to 3 characters) Product type extension.

Valid values for nominal operations camera data products:

"**iv**" - Unified Terrain Mesh RDR in Inventor ASCII format (no label)
"**mod**" - ASCII index list of Mesh component ".ht" Height Map and "tile.iv" vertices files

Example: NFR_L0060_RASLF_005_0104_AUTOGENM1.iv

where,

inst	=	"**NFR**"	=	Mesh comprised of Navcam, Front and Rear Hazcam data
delimiter	=	"**_**"	=	Delimiter
eye	=	"**L**"	=	Left camera eye
esol	=	"**0060**"	=	Sol 60 as Sol of last (highest SCLK) input product
multsol	=	"**_**"	=	Flags Sol 60 as only Sol
prodid	=	"**RAS**"	=	Terrain texture are contrast enhanced Rad-corrected RDRs
geom	=	"**L**"	=	"Linearized" input products without camera distortion
samp	=	"**F**"	=	Full frame, full resolution input products
spec	=	"**_**"	=	No special processing
site	=	"**005**"	=	Site 5
multsite	=	"**_**"	=	Flags Site 5 as only Site
drive	=	"**0104**"	=	Drive (Position-within-Site) 104
multdrive	=	"**_**"	=	Flags Drive 104 as only Drive
meshid	=	"**AUTOGEN**"	=	Default string identifying nominal pipeline as the process
venue / who	=	"**M**"	=	Identifies MIPL/OPGS as product provider
ver	=	"**1**"	=	Version 1
ext	=	"**iv**"	=	Unified Terrain Mesh in Open Inventor format

6.2 PDS Standards

The MSL camera instrument EDR data product complies with Planetary Data System standards for file formats and labels, as specified in the PDS Standards Reference [Ref 14]. See Section 4.2 for a description of the PDS Label and the specific conventions adopted by MSL.

6.3 Time Standards

The EDR PDS label uses keywords containing time values. Each time value standard is defined according to the keyword description. See Appendix F.

6.4 Coordinate Frame Standards

The MSL Frame Manager defines several dozen coordinate frames, which can be used for commanding pointing among other things. Refer to the Pointing, Positioning, Phasing and Coordinate Systems (PPPCS) document [Ref 1] or the Surface Attitude, Positioning and Pointing (SAPP) Functional Design Description (FDD) [Ref 2] for more details on all these coordinate frames. This section describes in detail the subset of Frames used by the products and processes in this SIS. The INSTRUMENT_COORD_FRAME_ID label (a command echo) is the only place in this SIS where the full set of frames can appear.

A subset of these frames needed for a specific image or data set are defined by the *_COORDINATE_SYSTEM groups.

Note that the PLACES database [Ref 10] maintains both telemetered and re-localized versions of the Site and Rover Nav frames at every available index.

Table 6.4 - Coordinate Frames Used for MSL Surface Operations

Frame Name (Label Keyword Value)	Short Name (SAPP FDD)	Reference Frame (Used to Define)	Coordinate Frame	
			Origin	Orientation
ROVER_NAV_FRAME	RNAV	Enclosing SITE_FRAME	Attached to rover	Aligned with rover
ROVER_MECH_FRAME	RMECH	Enclosing SITE_FRAME	Attached to rover	Aligned with rover
LOCAL_LEVEL_FRAME	LL	Enclosing SITE_FRAME	Attached to rover (coincident with Rover Nav Frame)	North/East/Nadir
SITE_FRAME	SITE(n)	Previous SITE_FRAME	Attached to surface	North/East/Nadir
RSM_HEAD_FRAME	RSM_HEAD	ROVER_NAV_FRAME	Attached to mast head	Aligned with pointing of mast head. This corresponds to RSM_HEAD in the Frame Manager
Arm Frames: ARM_TURRET_FRAME ARM_DRILL_FRAME ARM_DRT_FRAME ARM_MAHLI_FRAME ARM_APXS_FRAME ARM_PORTION_FRAME ARM_SCOOP_TIP_FRAME ARM_SCOOP_TCP_FRAME	Arm Frames: TURRET DRILL DRT MAHLI APXS PORTION SCOOP_TIP SCOOP_TCP	ROVER_NAV_FRAME	Attached to the tool; see PPPCS for the specific tool frame.	Aligned with tool in some way; see PPPCS [Ref 1] for the specific tool Frame.

6.4.1 Rover Navigation (Rover Nav) Frame

The Rover Nav frame (RNAV) is the one used for surface navigation and mobility. By definition, the frame is attached to the rover, and moves with it when the rover moves while on the surface. Its Y origin is centered on the rover and the X origin is aligned with the middle wheels' rotation axis for the deployed rover and suspension system on a flat plane. The Z origin is defined to be at the nominal surface, which is a fixed position with respect to the rover body. The actual surface will likely not be at exactly Z=0 due to the effects of suspension sag, rover tilt, rocker bogie angles, etc. The +X axis points to the front of the rover, +Y to the right side, and +Z down (perpendicular to the chassis deck). See Figure 6.4.1.1.

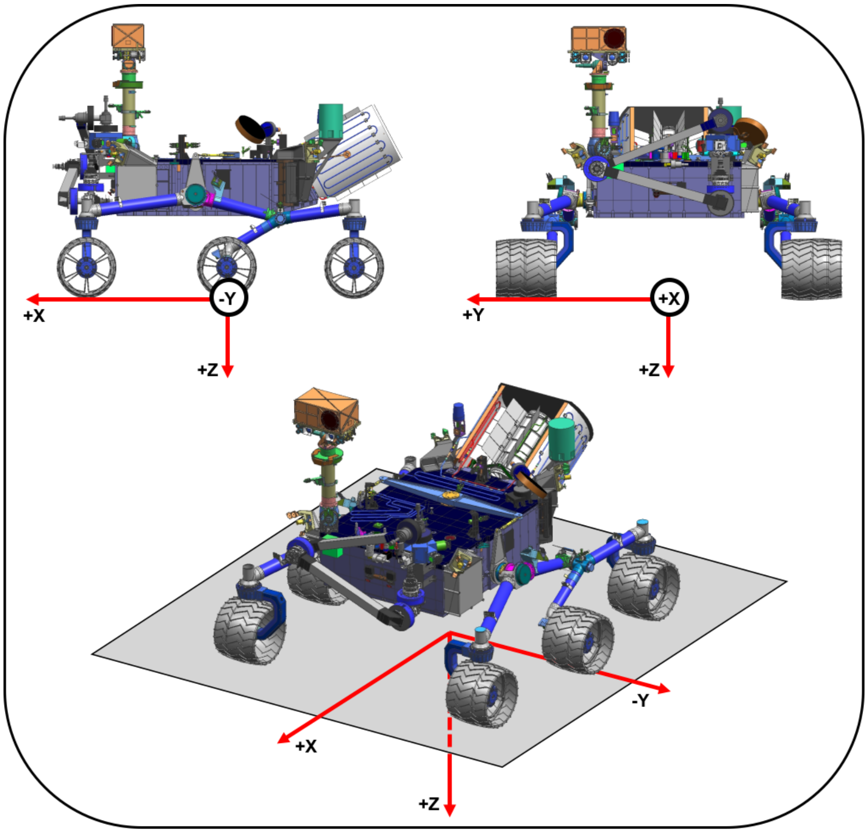


Figure 6.4.1.1 – Rover Navigation (RNAV) Coordinate Frame

The Rover Nav frame is specified via an offset from the current Site frame, and a quaternion that represents the rotation between the two. A new instance of the Rover Nav frame, with a potentially unique offset/quaternion, is created every time the ROVER_MOTION_COUNTER increments.

Orientation of the rover (and thus Rover Nav) with respect to Local Level or Site is also sometimes described by Euler angles as shown in Figure 6.4.1.2, where ψ is heading, θ is attitude or pitch, and ϕ is bank or roll.

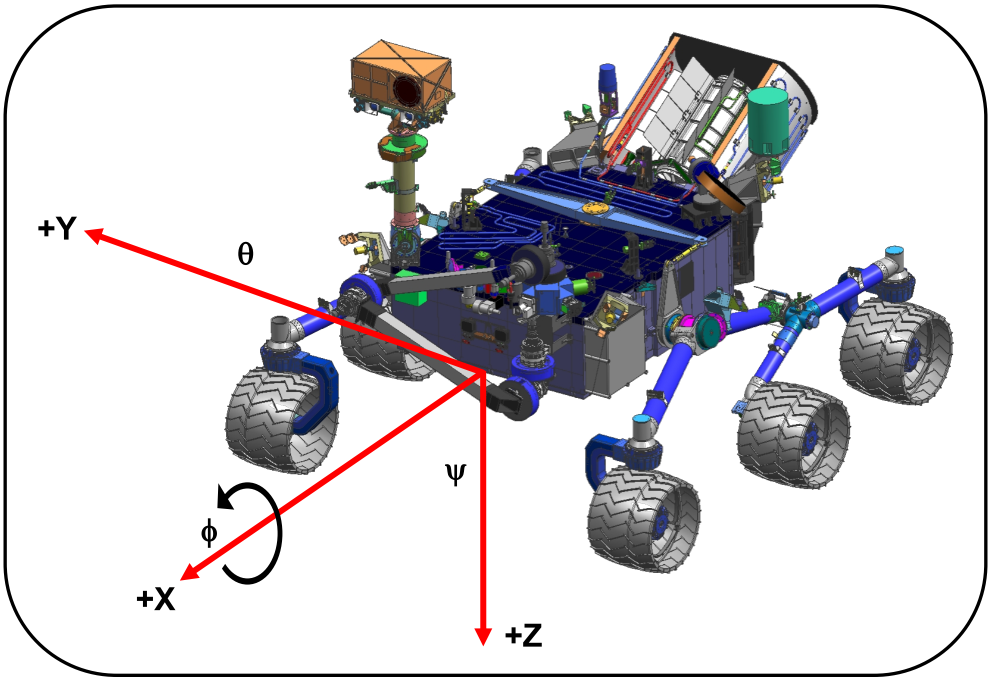


Figure 6.4.1.2 – Yaw, Pitch and Roll Definitions

6.4.2 Rover Mechanical (Rover Mech) Frame

The Rover Mechanical (RMECH) frame is oriented identically to the Rover Nav frame. The origin is forward of Rover Nav by x=0.09002 meters. In other words, given a point expressed in Rover Mech, if you add (0.09002, 0.0, -1.1205) you will get the same point expressed in Rover Nav. Rover Mech is not used by any nominal products (EDR or RDR) but could appear in certain special products, generally having to do with arm kinematics.

6.4.3 Local Level Frame

The Local Level frame is coincident with the Rover Nav frame, i.e. they share the same origin at all times. The orientation is different, however. The +X axis points North, +Z points down to nadir along the local gravity vector, and +Y completes the right-handed system. Thus the orientation matches the orientation of Site frames.

Local Level frames are defined by an offset from the current Site frame, with an identity quaternion.

6.4.4 Site Frame

Site frames are used to reduce accumulation of rover localization error. They are used to provide a common reference point for all operations within a local area. Rover Nav and Local Level frames are specified using an offset from this origin. When a new Site is declared, that becomes the new reference, and the offset is zeroed. In this way, long-term localization error is relegated to the offset between Sites, becoming irrelevant to local operations, because the positions are reset with each new Site.

When a Site frame is declared, it is identical to the Local Level frame, sharing both orientation and position. However, the Site frame is fixed to the Mars surface; when the rover moves, Local Level moves with it but Site stays put. Therefore, for the Site frame, +X points North, +Z points down to nadir along the local gravity vector, and +Y completes the right-handed system.

Sites are indexed, meaning there are multiple instances. Site 1 by definition represents the landing location. New Sites are declared as needed during operations, as the rover moves away from the local area. See Figure 6.4.4.

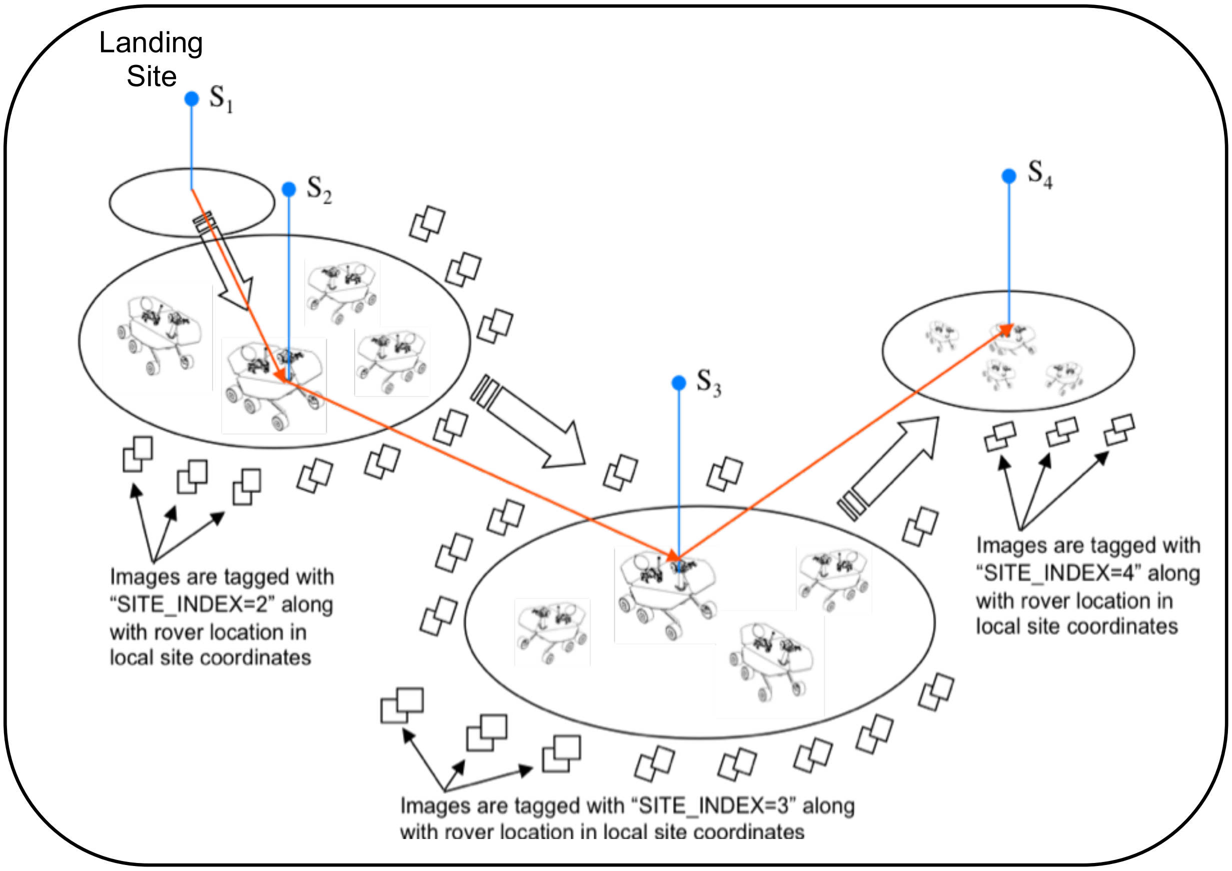


Figure 6.4.4 – Site and Rover Frames

The PLACES database [Ref 10] stores the set of all site-to-site offsets; such offsets are not in every image label.

6.4.5 RSM Frame

The RSM frame is attached to the Remote Sensing Mast (RSM) camera head, and moves with it. See the PPPCS for specific definition. It is expressed as an offset and quaternion from the Rover Nav frame.

6.4.6 Arm Frames

The frame representing the currently selected arm tool is reported in the arm coordinate system group. The selected tool, given by ARTICULATION_DEV_INSTRUMENT_ID, is arbitrary for any given image and may be surprising; for example, MAHLI may not be the selected tool for a MAHLI image. The various tool frames are attached to and aligned with the tool in some manner specific to that tool. See the PPPCS [Ref 1] for actual frame definitions.

7. APPLICABLE SOFTWARE

The instrument data downlink processing software is focused on rapid reduction, calibration, and visualization (in the case of images) of products in order to make discoveries, to accurately and expeditiously characterize the geologic environment around the rover, and to provide timely input for operational decisions concerning rover navigation and Arm target selection. Key software tools have been developed at JPL as part of the OPGS and APSS subsystems, and at LANL/IRAP/CNES by the ChemCam team. These toolsets can be used to process data to yield substantial scientific potential in addition to their operational importance.

7.1 Utility Programs

Table 7.1 lists (in no particular order) the primary software tools that will be used to process and manipulate downlinked MSL instrument payload data. Instrument data processing software executed by teams working the OPGS and APSS subsystems at JPL will be capable of reading and writing image and spectra data in PDS format. Within OPGS, the “msledrgen” program will generate EDRs and the Mars Program Suite of VICAR programs will generate RDRs in PDS format. An OPGS pipeline system will deliver the products to the FEI server for transfer to MSL’s ODS as rapidly as possible after receipt of telemetry.

Table 7.1 - Key Software Toolsets

Name	Description	Primary Development Responsibility
ChemCam Ops Software	LIBS software development is primarily performed in IDL code, with LANL (New Mexico) as lead institute.	Dot DeLapp (LANL)
	RMI software development is primarily performed in IDL code, with IRAP (France) as lead institute.	David Baratoux (IRAP)
msledrgen	Fetches the image Data Product Object (DPO) records from MSL Data Product (DP) files, reconstructing the image file from the telemetry data into a PDS-labeled image EDR data product. VICAR code.	Alice Stanboli, Costin Radulescu (JPL/MIPL)
Mars Program Suite	Stereo image processing software using EDRs or calibrated images (RDRs), image mosaicking software, 3-D terrain building software. VICAR code: • MARSCAHV – Generates a geometrically corrected version of the EDR, applying the C, A, H and V camera model vectors. • MARSRAD – Generates a radiometrically corrected image from a single input EDR. • MARSJPLSTEREO – Generates a disparity map from a stereo pair of input EDRs, applying a 1-D correlator (fast). • MARSCOR3 – Generates a disparity map from a stereo pair of input EDRs, applying a 2-D correlator (more robust). • MARSXYZ – Generates an XYZ image from an input disparity map. • MARSRANGE – Generates a range image from an input XYZ map. • MSLREACH – Generates an arm reachability map from an input XYZ map. • MARSINVERTER – Generates inverse lookup table (ILUT) products. • MARSDEBAYER – Generates de-Bayered images. • MSLROUGH – Generates roughness maps.	Bob Deen (JPL/MIPL)

Name	Description	Primary Development Responsibility
	• MARSERROR – Generates XYZ and range error maps. • MARSSLOPE – Generates slope maps. • MARSBRT – Generates brightness/contrast correction file for mosaic processing. • MSLFILTER – Generates XML file for image mask files. • MARSFILTER – Generates image mask files. • MARSMASK – Applies image mask files to image files. • MARSDISPCOMPARE – Checks consistency for left-to-right and right-to-left stereo image correlations. • MARSUVW - Generates a surface normal image, wherein XYZ is computed normal to the surface. • MARSMAP – Generates a Cylindrical, Polar or Vertical projection mosaic from a list of input EDRs. • MARSMOS – Produces pinhole camera mosaics using uncorrected input images and CAHVOR camera model. • MARSMCAULEY – Generates a combination Cylindrical-Perspective projection mosaic from a list of input EDRs. • MARSTIE – Generates pointing corrections (tiepoint file) from an overlapping set of input EDRs. • MARSNAV – Generates an updated azimuth and elevation file based on comparison with existing image data that can be directly compared. • XVD – De facto image reader software capable of displaying VICAR-labeled image files.	
	• CRUMBS – 3-D terrain building software	Oleg Pariser (JPL/MIPL)
APSS / MSLICE	Visualization and planning software for creation of science products and candidate observations for presentation at Ops planning meetings, and then Sol activity list at end of planning meetings. Java code.	Mark Powell (JPL)
APSS / RSVP	Visualization, planning, and sequence generation software for use by Sequence Team to create Sol sequences based on activity lists generated by PSI during planning meetings. Java, C and C++ code.	Brian Cooper (JPL)

7.2 Applicable PDS Software Tools

PDS-labeled images and tables can be viewed with the program NASAView, developed by the PDS and available for a variety of computer platforms from the PDS web site http://pds.jpl.nasa.gov/tools/software_download.cfm. There is no charge for NASAView.

7.3 Software Distribution and Update Procedures

The FEI distribution tool and Mars Image Processing Program Suite are available to researchers and academic institutions. Refer to the MIPL Web site at http://www-mipl.jpl.nasa.gov for contact information. FEI is described in detail at http://www-mipl.jpl.nasa.gov/MDMS.html

APPENDIX A - Composite EDR / RDR Label (ODL / PDS format)

PDS Label Symbols (displayed next to Keyword or Value)

{MSL:} = Keyword present in detached PDS label, but with "MSL:" prepended to name

Θ = Keyword not present in detached PDS label

[ ] = Specifies the Keyword or Value or Group in detached PDS label, when different than ODL label

NOTE: If no symbol is displayed, then Keyword is present in detached PDS label as shown for the ODL label. OPGS-generated products with MMM content have no PDS label.

ODL & VICAR Primary Label Symbols (displayed in matrix)

X = Keyword not present in any label

• = Keyword present in all labels that exist. For RDRs, keyword matches source product (if not in source product, then keyword is not present)

ODL & VICAR Secondary Label Symbols (displayed in matrix)

o = Keyword present in attached/detached **ODL** label only, and not present in VICAR label (if it exists)

a = Keyword present in **attached** ODL label only, and not present in detached ODL label or VICAR label (if it exists)

p = Keyword present in detached **PDS** label only; no existing ODL label or VICAR label

d = Keyword present in all labels, with value **different** from source EDR

r = Keyword present only if **relevant** and therefore optional (such as camera model vector keywords MODEL_COMPONENT_n in Group GEOMETRIC_CAMERA_MODEL)

m = Keyword present in labels for **MMM** products only

UDR Groups

MMM: Mcam*, Mhli*, Mrdi*

EDR Groups

RMI: CcamRmi*	**LIBS A:** CcamSpectra	**LIBS B:** CcamSpectraStats	**SOH:** CcamSoh* (no CcamSohSunSafe)
Sun Safe: CcamSohSunSafe	**Util Test:** CcamUtilTest	**Eng. Cameras:** Img*	**MMM A:** Mcam*, Mhli*, Mrdi* (no *Video or *Zstack)
MMM B: Mcam*Video, Mhli*Video, Mrdi*Video			**MMM C:** Mcam*Zstack, Mhli*Zstack, Mrdi*Zstack

RDR Groups

RMI: CcamRmi* (No AGS)	**LIBS A:** CcamSpectra	**LIBS B:** CcamSpectraStats	**AEGIS:** CcamRmiImage or EngCam Mask w/ AEGIS ROIs
XYZ: XYZ, Surface Normal, Surface Roughness		**Arm Reach:** Reachability, Pre-load	**Mosaic ORT:** Orthographic, Orthorectified

TC RDR: Terrain Classification Map ("TER") & Probability Map ("TEN")

Ops Labelless: Mosaic Ancillary (".LIS", ".NAV", ".BRT") & Terrain Per-XYZ (".iv", ".ht", ".rgb") & Terrain Unified (".iv", ".mod")

	PDS Label	Keyword		Example Value/Comment	UDR	EDR										RDR																			
						Ccam						Eng. Cams & MMM				Ccam				Eng. Cameras & MMM															
					MMM Group	RMI Group	LIBS Group A	LIBS Group B	SOH Group	Sun Safe Group	Util Test Group	Eng. Camera Grp	MMM Group A	MMM Group B	MMM Group C	RMI Group	LIBS Group A	LIBS Group B	AEGIS	Inverse LUT	Rad Correction	Geom Correction	Disparity	Range	XYZ Group	Slope	Arm Reach Group	TC RDR Grp	Mosaic (CYL)	Mosaic (PER)	Mosaic (CYP)	Mosaic (POL)	Mosaic (VRT)	Mosaic (ORT) Grp	Ops Labelless Grp
1	⊖	ODL_VERSION_ID [PDS_VERSION_ID]	=	ODL3 [PDS3]	•	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	p
2		/* FILE DATA ELEMENTS */			X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X
3		RECORD_TYPE	=	FIXED_LENGTH	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X
4		RECORD_BYTES	=	2048	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X
5		FILE_RECORDS	=	1038	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X
6		LABEL_RECORDS	=	8	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X
7		/* POINTERS TO DATA OBJECTS */			X	o	o	o	o	o	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X
8		^CCAM_LIBS_TABLE	=	12	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
9		^CCAM_SOH_DPO_TABLE	=	12	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
10		^IMAGE_HEADER	=	12	X	o	X	X	X	X	X	o	o	o	o	o	X	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X
11		^ANCILLARY_TABLE	=	18	X	o	X	X	X	X	X	X	X	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
12		^SOH_BEFORE_TABLE	=	19	X	o	X	X	X	X	X	X	X	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
13		^SOH_AFTER_TABLE	=	20	X	o	X	X	X	X	X	X	X	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
14		^IMAGE_REPLY_TABLE	=	21	X	o	X	X	X	X	X	X	X	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
15		^IMAGE_HEADER_FOOTER_TABLE	=	22	X	o	X	X	X	X	X	X	X	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
16		^IMAGE	=	23	X	o	X	X	X	X	X	o	o	o	o	o	X	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X
17		/* IDENTIFICATION DATA ELEMENTS */			•	o	o	o	o	o	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X
18	{MSL:}	ACTIVE_FLIGHT_STRING_ID	=	"A"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X
19		DATA_SET_ID	=	"MSL-M-NAVCAM-2-EDR-V1.0"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	d	d	d	d	d	d	d	d	X	d	d	d	d	d	d	X
20		DATA_SET_NAME	=	"MSL MARS NAVIGATION CAMERA 2 EDR V1.0"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	d	d	d	d	d	d	d	d	X	d	d	d	d	d	d	X
21		COMMAND_SEQUENCE_NUMBER	=	33	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
22		FRAME_ID	=	"RIGHT"	•	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	d	d	d	d	d	d	X
23		FRAME_TYPE	=	STEREO	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	d	d	d	d	d	d	X
24		GEOMETRY_PROJECTION_TYPE	=	RAW	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	d	•	•	•	•	•	X	X	X	X	X	X	X	X
25		IMAGE_ID	=	"44160017"	X	•	X	X	X	X	X	•	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
26		IMAGE_TYPE	=	REGULAR	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
27	{MSL:}	IMAGE_ACQUIRE_MODE	=	"IMAGE"	X	•	X	X	X	X	X	•	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
28		INSTRUMENT_HOST_ID	=	"MSL"	•	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	d	d	d	d	d	d	X
29		INSTRUMENT_HOST_NAME	=	"MARS SCIENCE LABORATORY"	•	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	d	d	d	d	d	d	X
30		INSTRUMENT_ID	=	"NAV_RIGHT_A"	•	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	d	d	d	d	d	d	X
31		INSTRUMENT_NAME	=	"NAVIGATION CAMERA RIGHT STRING A"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	d	d	d	d	d	d	X
32		INSTRUMENT_SERIAL_NUMBER	=	"54"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
33		INSTRUMENT_TYPE	=	"IMAGING CAMERA"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	d	d	d	d	d	d	X
34		INSTRUMENT_VERSION_ID	=	"FM"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
35	{MSL:}	LOCAL_MEAN_SOLAR_TIME	=	"Sol-00039M20:27:48.280"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
36		LOCAL_TRUE_SOLAR_TIME	=	"12:22:24"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
37	{MSL:}	LOCAL_TRUE_SOLAR_TIME_SOL	=	39	X	•	•	•	•	•	X	•	•	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X

	PDS Label	Keyword	Example Value/Comment	UDR	EDR										RDR																			
					Ccam						Eng. Cams & MMM				Ccam				Eng. Cameras & MMM															
				MMM Group	RMI Group	LIBS Group A	LIBS Group B	SOH Group	Sun Safe Group	Util Test Group	Eng. Camera Grp	MMM Group A	MMM Group B	MMM Group C	RMI Group	LIBS Group A	LIBS Group B	AEGIS	Inverse LUT	Rad Correction	Geom Correction	Disparity	Range	XYZ Group	Slope	Arm Reach Group	TC RDR Grp	Mosaic (CYL)	Mosaic (PER)	Mosaic (CYP)	Mosaic (POL)	Mosaic (VRT)	Mosaic (ORT) Grp	Ops Labelless Grp
38		MISSION_NAME	= "MARS SCIENCE LABORATORY"	•	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	d	d	d	d	d	d	X
39		MISSION_PHASE_NAME	= "PRIMARY SURFACE MISSION"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
40		OBSERVATION_ID	= "UNK"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
41		PLANET_DAY_NUMBER	= 39	•	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
42		PRODUCER_INSTITUTION_NAME	= "MULTIMISSION INSTRUMENT PROCESSING LAB, JET PROPULSION LAB"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	d	d	d	d	d	d	d	d	•	d	d	d	d	d	d	X
43		PRODUCT_CREATION_TIME	= 2009-06-04T08:39:29.000	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
44		PRODUCT_ID	= "NRA_411689704EDR_F0000000001015808M1"	•	•	•	•	•	•	X	•	•	•	•	•	•	•	•	d	d	d	d	d	d	d	d	•	d	d	d	d	d	d	X
45		PRODUCT_VERSION_ID	= "V2.0"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	p	p	p	p	p	p	X
46		RELEASE_ID	= "0001"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
47	{MSL:}	REQUEST_ID	= "0"	•	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
48		ROVER_MOTION_COUNTER	= (1, 2, 0, 0, 0, 10, 6, 12, 163, 0)	•	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
49		ROVER_MOTION_COUNTER_NAME	= ("SITE", "DRIVE", "POSE", "ARM", "CHIMRA", "DRILL", "RSM", "HGA", "DRT", "IC")	•	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
50		SEQUENCE_ID	= "NCAM10362"	•	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
51		SEQUENCE_VERSION_ID	= "2"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
52		SOLAR_LONGITUDE	= 102.844	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
53		SOURCE_PRODUCT_ID	= "NRA_411689704EDR_F0000000001015808M1"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
54		SPACECRAFT_CLOCK_CNT_PARTITION	= 1	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
55		SPACECRAFT_CLOCK_START_COUNT	= "4876293673.256"	•	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	d	d	d	d	d	d	X
56		SPACECRAFT_CLOCK_STOP_COUNT	= "4876293679.381"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	d	d	d	d	d	d	X
57		START_TIME	= 2009-01-28T20:27:48.049	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	d	d	d	d	d	d	X
58		STOP_TIME	= 2009-01-28T20:27:48.280	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	d	d	d	d	d	d	X
59		TARGET_NAME	= "MARS"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	d	d	d	d	d	d	X
60		TARGET_TYPE	= PLANET	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	d	d	d	d	d	d	X
61		/* TELEMETRY DATA ELEMENTS */		X	o	o	o	o	o	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X	X	X	X	X	X	X
62		APPLICATION_PROCESS_ID	= 22	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
63		APPLICATION_PROCESS_NAME	= "ImgImageNr"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
64	{MSL:}	AUTO_DELETE_FLAG	= "FALSE"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
65	{MSL:}	COMMUNICATION_SESSION_ID	= "54321"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
66		DOWNLOAD_PRIORITY	= 0	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
67		EARTH_RECEIVED_START_TIME	= 2009-01-28T22:06:00.901	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
68		EARTH_RECEIVED_STOP_TIME	= 2009-01-28T22:06:04.018	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
69		EXPECTED_PACKETS	= 76	X	X	X	X	X	X	X	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
70		RECEIVED_PACKETS	= 76	X	X	X	X	X	X	X	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
71	{MSL:}	EXPECTED_TRANSMISSION_PATH	= "3845"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
72	{MSL:}	FLIGHT_SOFTWARE_MODE	= "1"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
73		FLIGHT_SOFTWARE_VERSION_ID	= "97208714"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
74	{MSL:}	PRODUCT_COMPLETION_STATUS	= "COMPLETE_CHECKSUM_PASS"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
75	{MSL:}	PRODUCT_TAG	= "0"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
76	{MSL:}	SEQUENCE_EXECUTION_COUNT	= 0	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
77		SPICE_FILE_NAME	= "chronos.msl_cs0n45e120806_v2"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
78	{MSL:}	STRIPING_COUNT	= 0	X	•	X	X	X	X	X	•	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X

	PDS Label	Keyword	Example Value/Comment	UDR	EDR											RDR																			
					Ccam						Eng. Cams & MMM					Ccam				Eng. Cameras & MMM															
				MMM Group	RMI Group	LIBS Group A	LIBS Group B	SOH Group	Sun Safe Group	Util Test Group	Eng. Camera Grp	MMM Group A	MMM Group B	MMM Group C	RMI Group	LIBS Group A	LIBS Group B	AEGIS	Inverse LUT	Rad Correction	Geom Correction	Disparity	Range	XYZ Group	Slope	Arm Reach Group	TC RDR Grp	Mosaic (CYL)	Mosaic (PER)	Mosaic (CYP)	Mosaic (POL)	Mosaic (VRT)	Mosaic (ORT) Grp	Ops Labelless Grp
79	{MSL:}	STRIPING_OVERLAP_ROWS	= 0	X	•	X	X	X	X	X	•	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
80		TELEMETRY_PROVIDER_ID	= "MPCS_MSL_DP"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
81	{MSL:}	TELEMETRY_SOURCE_CHECKSUM	= 59736	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
82	{MSL:}	TELEMETRY_SOURCE_HOST_NAME	= 59736	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
83		TELEMETRY_SOURCE_NAME	= "mslgdsdev2"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
84	{MSL:}	TELEMETRY_SOURCE_SCLK_START	= "022_001_p1001-002-0001_005_0126467896-255.dat"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
85	{MSL:}	TELEMETRY_SOURCE_START_TIME	= "4876293673.256"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
86	{MSL:}	TELEMETRY_SOURCE_SIZE	= 2009-243T40:27:48.280	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
87	{MSL:}	TELEMETRY_SOURCE_TYPE	= 574464	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
88	{MSL:}	TRANSMISSION_PATH	= "DATA PRODUCT"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
89	{MSL:}	VIRTUAL_CHANNEL_ID	= "65535"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
90		/* HISTORY DATA ELEMENTS */		X	o	o	o	o	o	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X	X	X	X	X	X	X
91		GROUP	= MSLEDRGEN_HISTORY_PARMS	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
92		SOFTWARE_NAME	= "MSLEDRGEN"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
93		SOFTWARE_VERSION_ID	= "V1.24.46"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
94		PROCESSING_HISTORY_TEXT	= "CODMAC LEVEL 1 TO LEVEL 2 CONVERSION VIA JPL/MIPL MSLEDRGEN"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
95		END_GROUP	= MSLEDRGEN_HISTORY_PARMS	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
96		/* CAMERA_MODEL DATA ELEMENTS */		•	o	o	o	o	o	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X
97		GROUP	= GEOMETRIC_CAMERA_MODEL [GEOMETRIC_CAMERA_MODEL_PARMS]	•	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X
98		CALIBRATION_SOURCE_ID	= "59"	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	d	•	•	•	•	•	•	X	d	r	X	X	X	X
99	Θ	CAMERA_SERIAL_NUMBER	= "54"	X	•	X	X	X	X	X	•	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	X	X	X	r	X	X	X	X
100		FILTER_NAME	= "NONE"	X	X	X	X	X	X	X	X	•	X	X	X	X	X	X	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X	X
101		FILTER_NUMBER	= 0	X	X	X	X	X	X	X	X	•	X	X	X	X	X	X	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X	X
102	{MSL:}	INTERPOLATION_METHOD	= "TEMPERATURE"	•	r	X	X	X	X	X	r	r	r	r	r	X	X	r	r	r	r	r	r	r	r	r	X	X	d	r	X	X	X	X
103	{MSL:}	INTERPOLATION_VALUE	= -26.4723	•	r	X	X	X	X	X	r	r	r	r	r	X	X	r	r	r	r	r	r	r	r	r	X	X	d	r	X	X	X	X
104		^MODEL_DESC	= "GEOMETRIC_CM.TXT"	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	X	•	•	r	•	•	•	X
105		MODEL_TYPE	= CAHVORE	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	d	•	•	•	•	•	•	X	d	r	X	X	X	X
106		MODEL_COMPONENT_ID	= ("C", "A", "H", "V", "O", "R", "E", "T", "P")	•	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	d	•	•	•	•	•	•	X	d	r	X	X	X	X
107		MODEL_COMPONENT_NAME	= ("CENTER", "AXIS", "HORIZONTAL", "VERTICAL", "OPTICAL", "RADIAL", "ENTRANCE", "MTYPE", "MPARM")	•	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	d	•	•	•	•	•	•	X	d	r	X	X	X	X
108		MODEL_COMPONENT_1	= (0.0230152, -0.076101, 0.874005)	•	r	X	X	X	X	X	r	r	r	r	r	X	X	r	r	r	d	r	r	r	r	r	•	X	d	r	X	X	X	X
109		MODEL_COMPONENT_2	= (0.0602658, 0.945477, -0.304335)	•	r	X	X	X	X	X	r	r	r	r	r	X	X	r	r	r	d	r	r	r	r	r	•	X	d	r	X	X	X	X
110		MODEL_COMPONENT_3	= (1011.61, 62.6302, -20.1324)	•	r	X	X	X	X	X	r	r	r	r	r	X	X	r	r	r	d	r	r	r	r	r	•	X	d	r	X	X	X	X
111		MODEL_COMPONENT_4	= (7.72579, -183.499, -995.739)	•	r	X	X	X	X	X	r	r	r	r	r	X	X	r	r	r	d	r	r	r	r	r	•	X	d	r	X	X	X	X
112		MODEL_COMPONENT_5	= (0.0602619, 0.945474, -0.304348)	•	r	X	X	X	X	X	r	r	r	r	r	X	X	r	r	r	r	r	r	r	r	r	•	X	d	r	X	X	X	X
113		MODEL_COMPONENT_6	= (0.0, -0.001377, -0.027648)	•	r	X	X	X	X	X	r	r	r	r	r	X	X	r	r	r	r	r	r	r	r	r	•	X	d	r	X	X	X	X
114		MODEL_COMPONENT_7	= (0.0, -0.001356, -0.027693)	•	r	X	X	X	X	X	r	r	r	r	r	X	X	r	r	r	r	r	r	r	r	r	•	X	d	r	X	X	X	X

#	PDS Label	Keyword	Example Value/Comment	UDR: MMM Group	EDR Ccam: RMI Group	EDR Ccam: LIBS Group A	EDR Ccam: LIBS Group B	EDR Ccam: SOH Group	EDR Ccam: Sun Safe Group	EDR Ccam: Util Test Group	EDR Eng. Cams & MMM: Eng. Camera Grp	EDR Eng. Cams & MMM: MMM Group A	EDR Eng. Cams & MMM: MMM Group B	EDR Eng. Cams & MMM: MMM Group C	RDR Ccam: RMI Group	RDR Ccam: LIBS Group A	RDR Ccam: LIBS Group B	RDR Ccam: AEGIS	RDR Eng. Cameras & MMM: Inverse LUT	Rad Correction	Geom Correction	Disparity	Range	XYZ Group	Slope	Arm Reach Group	TC RDR Grp	Mosaic (CYL)	Mosaic (PER)	Mosaic (CYP)	Mosaic (POL)	Mosaic (VRT)	Mosaic (ORT) Grp	Ops Labelless Grp
115		MODEL_COMPONENT_8	= 3.0	•	r	X	X	X	X	X	r	r	r	r	r	X	X	r	r	r	r	r	r	r	r	r	•	X	d	r	X	X	X	X
116		MODEL_COMPONENT_9	= 0.27741	•	r	X	X	X	X	X	r	r	r	r	r	X	X	r	r	r	r	r	r	r	r	r	•	X	d	r	X	X	X	X
117	{MSL:}	MODEL_TRANSFORM_QUATERNION	= (1.875, 0.0, 0.0, 0.0)	•	r	X	X	X	X	X	r	r	r	r	r	X	X	r	r	r	r	r	r	r	r	r	X	X	d	r	X	X	X	X
118	{MSL:}	MODEL_TRANSFORM_VECTOR	= (0.0, 0.0, 0.0)	•	r	X	X	X	X	X	r	r	r	r	r	X	X	r	r	r	r	r	r	r	r	r	X	X	d	r	X	X	X	X
119		REFERENCE_COORD_SYSTEM_NAME	= "ROVER_NAV_FRAME"	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	d	•	•	•	•	•	•	X	d	r	X	X	X	X
120		REFERENCE_COORD_SYSTEM_INDEX	= (1, 2, 0, 0, 0, 10, 6, 12, 163, 0)	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	d	•	•	•	•	•	•	X	d	r	X	X	X	X
121	{MSL:}	REFERENCE_COORD_SYSTEM_SOLN_ID	= "mipl_rgd_sol3nav_5"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	r	r	r	r	r	X	X	r	r	X	X	X	X
122		END_GROUP	= GEOMETRIC_CAMERA_MODEL [GEOMETRIC_CAMERA_MODEL_PARMS]	•	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X
123		/* COORDINATE SYSTEM STATE: ROVER */		•	o	o	o	o	o	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	r	r	r	r	r	r	X
124		GROUP	= ROVER_COORDINATE_SYSTEM [ROVER_COORD_SYSTEM_PARMS]	•	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	r	r	r	r	r	r	X
125	{MSL:}	SOLUTION_ID	= "telemetry"	X	X	X	X	X	X	X	•	•	•	•	X	X	X	•	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	X
126		COORDINATE_SYSTEM_NAME	= "ROVER_NAV_FRAME"	•	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	r	r	r	r	r	r	X
127		COORDINATE_SYSTEM_INDEX	= (1, 2, 0, 0, 0, 10, 6, 12, 163, 0)	•	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	r	r	r	r	r	r	X
128		COORDINATE_SYSTEM_INDEX_NAME	= ("SITE", "DRIVE", "POSE", "ARM", "CHIMRA", "DRILL", "RSM", "HGA", "DRT", "IC")	•	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	r	r	r	r	r	r	X
129		ORIGIN_OFFSET_VECTOR	= (0.0230152, -0.076101, 0.874005)	•	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	r	r	r	r	r	r	X
130		ORIGIN_ROTATION_QUATERNION	= (0.922297, -0.0165226, -0.0413094, 0.382304)	•	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	r	r	r	r	r	r	X
131		POSITIVE_AZIMUTH_DIRECTION	= CLOCKWISE	•	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	r	r	r	r	r	r	X
132		POSITIVE_ELEVATION_DIRECTION	= UP	•	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	r	r	r	r	r	r	X
133		QUATERNION_MEASUREMENT_METHOD	= "FINE"	•	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	r	r	r	r	r	r	X
134		REFERENCE_COORD_SYSTEM_NAME	= "SITE_FRAME"	•	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	r	r	r	r	r	r	X
135		REFERENCE_COORD_SYSTEM_INDEX	= 1	•	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	r	r	r	r	r	r	X
136	{MSL:}	REFERENCE_COORD_SYSTEM_SOLN_ID	= "mipl_rgd_sol3nav_5"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	X
137		END_GROUP	= ROVER_COORDINATE_SYSTEM [ROVER_COORD_SYSTEM_PARMS]	•	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	r	r	r	r	r	r	X
138		/* COORDINATE SYSTEM STATE: SITE */		X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	X
139		GROUP	= SITE_COORDINATE_SYSTEM [SITE_COORD_SYSTEM_PARMS]	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	X
140	{MSL:}	SOLUTION_ID	= "telemetry"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	X
141		COORDINATE_SYSTEM_NAME	= "SITE_FRAME"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	X
142		COORDINATE_SYSTEM_INDEX	= (1, 2, 0, 0, 0, 10, 6, 12, 163, 0)	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	X
143		COORDINATE_SYSTEM_INDEX_NAME	= ("SITE", "DRIVE", "POSE", "ARM", "CHIMRA", "DRILL", "RSM", "HGA", "DRT", "IC")	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	X
144		ORIGIN_OFFSET_VECTOR	= (0.0230152, -0.076101, 0.874005)	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	X
145		ORIGIN_ROTATION_QUATERNION	= (0.922297, -0.0165226, -0.0413094, 0.382304)	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	X
146		POSITIVE_AZIMUTH_DIRECTION	= CLOCKWISE	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	X
147		POSITIVE_ELEVATION_DIRECTION	= UP	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	X
148		REFERENCE_COORD_SYSTEM_NAME	= "SITE_FRAME"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	X
149		REFERENCE_COORD_SYSTEM_INDEX	= 1	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	X

#	PDS Label	Keyword	Example Value/Comment	UDR: MMM Group	EDR Ccam: RMI Group	EDR Ccam: LIBS Group A	EDR Ccam: LIBS Group B	EDR Ccam: SOH Group	EDR Ccam: Sun Safe Group	EDR Ccam: Util Test Group	EDR Eng. Cams & MMM: Eng. Camera Grp	EDR Eng. Cams & MMM: MMM Group A	EDR Eng. Cams & MMM: MMM Group B	EDR Eng. Cams & MMM: MMM Group C	RDR Ccam: RMI Group	RDR Ccam: LIBS Group A	RDR Ccam: LIBS Group B	RDR Ccam: AEGIS	RDR Eng. Cameras & MMM: Inverse LUT	Rad Correction	Geom Correction	Disparity	Range	XYZ Group	Slope	Arm Reach Group	TC RDR Grp	Mosaic (CYL)	Mosaic (PER)	Mosaic (CYP)	Mosaic (POL)	Mosaic (VRT)	Mosaic (ORT) Grp	Ops Labelless Grp
150	{MSL:}	REFERENCE_COORD_SYSTEM_SOLN_ID	= "mipl_rgd_sol3nav_5"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	X
151		END_GROUP	= SITE_COORDINATE_SYSTEM [SITE_COORD_SYSTEM_PARMS]	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	X
152		/* COORDINATE SYSTEM STATE: LOCAL LEVEL */		X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	X
153		GROUP	= LOCAL_LEVEL_COORDINATE_SYSTEM [LOCAL_LEVEL_COORD_SYSTEM_PARMS]	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	X
154	{MSL:}	SOLUTION_ID	= "telemetry"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	X
155		COORDINATE_SYSTEM_NAME	= "LOCAL_LEVEL_FRAME"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	X
156		COORDINATE_SYSTEM_INDEX	= (1, 2, 0, 0, 0, 10, 6, 12, 163, 0)	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	X
157		COORDINATE_SYSTEM_INDEX_NAME	= ("SITE", "DRIVE", "POSE", "ARM", "CHIMRA", "DRILL", "RSM", "HGA", "DRT", "IC")	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	X
158		ORIGIN_OFFSET_VECTOR	= (0.0230152, -0.076101, 0.874005)	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	X
159		ORIGIN_ROTATION_QUATERNION	= (0.922297, -0.0165226, -0.0413094, 0.382304)	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	X
160		POSITIVE_AZIMUTH_DIRECTION	= CLOCKWISE	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	X
161		POSITIVE_ELEVATION_DIRECTION	= UP	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	X
162		REFERENCE_COORD_SYSTEM_NAME	= "SITE_FRAME"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	X
163		REFERENCE_COORD_SYSTEM_INDEX	= 1	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	X
164	{MSL:}	REFERENCE_COORD_SYSTEM_SOLN_ID	= "mipl_rgd_sol3nav_5"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	X
165		END_GROUP	= LOCAL_LEVEL_COORDINATE_SYSTEM [LOCAL_LEVEL_COORD_SYSTEM_PARMS]	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	X
166		/* COORDINATE SYSTEM STATE: REMOTE SENSING MAST */		X	o	X	X	X	X	X	o	o	o	o	o	X	X	o	o	o	o	o	o	o	o	o	o	X	X	X	X	X	X	X
167		GROUP	= RSM_COORDINATE_SYSTEM [RSM_COORD_SYSTEM_PARMS]	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
168	{MSL:}	SOLUTION_ID	= "telemetry"	X	X	X	X	X	X	X	•	•	•	•	X	X	X	r	r	r	r	r	r	r	r	r	r	X	X	X	X	X	X	X
169		COORDINATE_SYSTEM_NAME	= "MAST_FRAME"	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
170		COORDINATE_SYSTEM_INDEX	= (1, 2, 0, 0, 0, 10, 6, 12, 163, 0)	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
171		COORDINATE_SYSTEM_INDEX_NAME	= ("SITE", "DRIVE", "POSE", "ARM", "CHIMRA", "DRILL", "RSM", "HGA", "DRT", "IC")	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
172		ORIGIN_OFFSET_VECTOR	= (0.0230152, -0.076101, 0.874005)	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
173		ORIGIN_ROTATION_QUATERNION	= (0.922297, -0.0165226, -0.0413094, 0.382304)	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
174		POSITIVE_AZIMUTH_DIRECTION	= CLOCKWISE	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
175		POSITIVE_ELEVATION_DIRECTION	= UP	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
176		REFERENCE_COORD_SYSTEM_NAME	= "ROVER_NAV_FRAME"	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
177		REFERENCE_COORD_SYSTEM_INDEX	= (1, 2, 0, 0, 0, 10, 6, 12, 163, 0)	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
178	{MSL:}	REFERENCE_COORD_SYSTEM_SOLN_ID	= "mipl_rgd_sol3nav_5"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	r	r	r	r	r	r	X	X	X	X	X	X	X
179		END_GROUP	= RSM_COORDINATE_SYSTEM [RSM_COORD_SYSTEM_PARMS]	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
180		/* COORDINATE SYSTEM STATE: ROBOTIC ARM */		X	X	X	X	X	X	X	o	o	o	o	X	X	X	o	o	o	o	o	o	o	o	o	o	X	X	X	X	X	X	X

	PDS Label	Keyword	Example Value/Comment	UDR	EDR										RDR																				
					Ccam						Eng. Cams & MMM				Ccam				Eng. Cameras & MMM															
				MMM Group	RMI Group	LIBS Group A	LIBS Group B	SOH Group	Sun Safe Group	Util Test Group	Eng. Camera Grp	MMM Group A	MMM Group B	MMM Group C	RMI Group	LIBS Group A	LIBS Group B	AEGIS	Inverse LUT	Rad Correction	Geom Correction	Disparity	Range	XYZ Group	Slope	Arm Reach Group	TC RDR Grp	Mosaic (CYL)	Mosaic (PER)	Mosaic (CYP)	Mosaic (POL)	Mosaic (VRT)	Mosaic (ORT) Grp	Ops Labelless Grp
181		GROUP	= ARM_COORDINATE_SYSTEM [ARM_COORD_SYSTEM_PARMS]	X	X	X	X	X	X	X	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
182	{MSL:}	SOLUTION_ID	= "telemetry"	X	X	X	X	X	X	X	•	•	•	•	X	X	X	r	r	r	r	r	r	r	r	r	r	X	X	X	X	X	X	X
183		COORDINATE_SYSTEM_NAME	= "ARM_FRAME"	X	X	X	X	X	X	X	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
184		COORDINATE_SYSTEM_INDEX	= (1, 2, 0, 0, 0, 10, 6, 12, 163, 0)	X	X	X	X	X	X	X	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
185		COORDINATE_SYSTEM_INDEX_NAME	= ("SITE", "DRIVE", "POSE", "ARM", "CHIMRA", "DRILL", "RSM", "HGA", "DRT", "IC")	X	X	X	X	X	X	X	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
186		ORIGIN_OFFSET_VECTOR	= (0.0230152, -0.076101, 0.874005)	X	X	X	X	X	X	X	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
187		ORIGIN_ROTATION_QUATERNION	= (0.922297, -0.0165226, -0.0413094, 0.382304)	X	X	X	X	X	X	X	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
188		POSITIVE_AZIMUTH_DIRECTION	= CLOCKWISE	X	X	X	X	X	X	X	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
189		POSITIVE_ELEVATION_DIRECTION	= UP	X	X	X	X	X	X	X	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
190		REFERENCE_COORD_SYSTEM_NAME	= "ROVER_NAV_FRAME"	X	X	X	X	X	X	X	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
191		REFERENCE_COORD_SYSTEM_INDEX	= (1, 2, 0, 0, 0, 10, 6, 12, 163, 0)	X	X	X	X	X	X	X	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
192	{MSL:}	REFERENCE_COORD_SYSTEM_SOLN_ID	= "mipl_rgd_sol3nav_5"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	r	r	r	r	r	r	X	X	X	X	X	X	X
193		END_GROUP	= ARM_COORDINATE_SYSTEM [ARM_COORD_SYSTEM_PARMS]	X	X	X	X	X	X	X	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
194		/* ARTICULATION DEVICE STATE: REMOTE SENSING MAST */		X	o	o	o	o	o	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X	X	X	X	X	X	X
195		GROUP	= RSM_ARTICULATION_STATE [RSM_ARTICULATION_STATE_PARMS]	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
196	{MSL:}	SOLUTION_ID	= "telemetry"	X	X	X	X	X	X	X	•	•	•	•	X	X	X	X	r	r	r	r	r	r	r	r	r	X	X	X	X	X	X	X
197		ARTICULATION_DEVICE_ID	= "RSM"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
198		ARTICULATION_DEVICE_NAME	= "REMOTE SENSING MAST"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
199		ARTICULATION_DEVICE_ANGLE	= (0.0230152 <rad>, -0.076101 <rad>, 0.874005 <rad>, 9.4095 <rad>, 0.3467 <rad>, 0.922297 <rad>)	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
200		ARTICULATION_DEVICE_ANGLE_NAME	= ("AZIMUTH-MEASURED", "ELEVATION-MEASURED", "AZIMUTH-REQUESTED", "ELEVATION-REQUESTED", "AZIMUTH-INITIAL", "ELEVATION-INITIAL", "AZIMUTH-FINAL", "ELEVATION-FINAL")	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
201		ARTICULATION_DEVICE_MODE	= "DEPLOYED"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
202		END_GROUP	= RSM_ARTICULATION_STATE [RSM_ARTICULATION_STATE_PARMS]	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
203		/* ARTICULATION DEVICE STATE: ROBOTIC ARM */		X	X	X	X	X	X	X	o	o	o	o	X	X	X	r	o	o	o	o	o	o	o	o	o	X	X	X	X	X	X	X
204		GROUP	= ARM_ARTICULATION_STATE [ARM_ARTICULATION_STATE_PARMS]	X	X	X	X	X	X	X	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
205		ARTICULATION_DEVICE_ID	= "ARM"	X	X	X	X	X	X	X	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
206		ARTICULATION_DEVICE_NAME	= "ROBOTIC ARM"	X	X	X	X	X	X	X	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
207		ARTICULATION_DEVICE_ANGLE	= (0.0230152 <rad>, -0.076101 <rad>, 0.874005 <rad>, 9.4095 <rad>, 0.3467 <rad>, 0.922297 <rad>, -0.0165226 <rad>, -0.0413094 <rad>, 0.38230 <rad>, 0.456 <rad>)	X	X	X	X	X	X	X	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X

PDS Label	Keyword		Example Value/Comment	UDR	EDR Ccam						EDR Eng. Cams & MMM				RDR Ccam				RDR Eng. Cameras & MMM															
				MMM Group	RMI Group	LIBS Group A	LIBS Group B	SOH Group	Sun Safe Group	Util Test Group	Eng. Camera Grp	MMM Group A	MMM Group B	MMM Group C	RMI Group	LIBS Group A	LIBS Group B	AEGIS	Inverse LUT	Rad Correction	Geom Correction	Disparity	Range	XYZ Group	Slope	Arm Reach Group	TC RDR Grp	Mosaic (CYL)	Mosaic (PER)	Mosaic (CYP)	Mosaic (POL)	Mosaic (VRT)	Mosaic (ORT) Grp	Ops Labelless Grp
208	ARTICULATION_DEVICE_ANGLE_NAME	=	("JOINT 1 AZIMUTH-ENCODER", "JOINT 2 ELEVATION-ENCODER", "JOINT 3 ELBOW-ENCODER", "JOINT 4 WRIST-ENCODER", "JOINT 5 TURRET-ENCODER", "JOINT 1 AZIMUTH-RESOLVER", "JOINT 2 ELEVATION-RESOLVER", "JOINT 3 ELBOW-RESOLVER", "JOINT 4 WRIST-RESOLVER", "JOINT 5 TURRET-RESOLVER")	X	X	X	X	X	X	X	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
209	ARTICULATION_DEVICE_MODE	=	"STOWED"	X	X	X	X	X	X	X	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
210	ARTICULATION_DEVICE_TEMP	=	(0.922297 <degC>, -0.0165226 <degC>)	X	X	X	X	X	X	X	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
211	ARTICULATION_DEVICE_TEMP_NAME	=	("AZIMUTH JOINT 1", "TURRET JOINT 5")	X	X	X	X	X	X	X	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
212	ARTICULATION_DEV_INSTRUMENT_ID	=	"ARM"	X	X	X	X	X	X	X	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
213	ARTICULATION_DEV_VECTOR	=	(1.23456, 3.4567, 23.456)	X	X	X	X	X	X	X	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
214	ARTICULATION_DEV_VECTOR_NAME	=	"GRAVITY"	X	X	X	X	X	X	X	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
215	CONTACT_SENSOR_STATE	=	("NO CONTACT", "NO CONTACT", "NO CONTACT", "NO CONTACT", "NO CONTACT", "NO CONTACT", "NO CONTACT", "NO CONTACT")	X	X	X	X	X	X	X	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
216	CONTACT_SENSOR_STATE_NAME	=	("APXS CONTACT SWITCH 1", "APXS CONTACT SWITCH 2", "DRILL SWITCH 1", "DRILL SWITCH 2", "MAHLI SWITCH 1A", "MAHLI SWITCH 1B", "MAHLI SWITCH 2A", "MAHLI SWITCH 2B")	X	X	X	X	X	X	X	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
217	END_GROUP	=	ARM_ARTICULATION_STATE [ARM_ARTICULATION_STATE_PARMS]	X	X	X	X	X	X	X	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
218	/* ARTICULATION DEVICE STATE: MOBILITY CHASSIS */			X	o	X	X	X	X	X	o	o	o	o	o	X	X	o	o	o	o	o	o	o	o	o	o	X	X	X	X	X	X	X
219	GROUP	=	CHASSIS_ARTICULATION_STATE [CHASSIS_ARTICULATION_ST_PARMS]	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
220	ARTICULATION_DEVICE_ID	=	"CHASSIS"	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
221	ARTICULATION_DEVICE_NAME	=	"MOBILITY CHASSIS"	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
222	ARTICULATION_DEVICE_ANGLE	=	(0.0230152 <rad>, -0.076101 <rad>, 0.874005 <rad>, 9.4095 <rad>, 0.3467 <rad>, 0.922297 <rad>, -0.0165226 <rad>)	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
223	ARTICULATION_DEVICE_ANGLE_NAME	=	("LEFT FRONT WHEEL", "RIGHT FRONT WHEEL", "LEFT REAR WHEEL", "RIGHT REAR WHEEL", "RSM AZIMUTH", "RSM ELEVATION", "LEFT BOGIE", "RIGHT BOGIE", "LEFT DIFFERENTIAL", "RIGHT DIFFERENTIAL")	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
224	ARTICULATION_DEVICE_MODE	=	"DEPLOYED"	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
225	END_GROUP	=	CHASSIS_ARTICULATION_STATE [CHASSIS_ARTICULATION_ST_PARMS]	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
226	/* ARTICULATION DEVICE STATE: HIGH GAIN ANTENNA */			X	X	X	X	X	X	X	o	o	o	o	X	X	X	o	o	o	o	o	o	o	o	o	o	X	X	X	X	X	X	X

	PDS Label	Keyword	Example Value/Comment	UDR	EDR										RDR																			
					Ccam						Eng. Cams & MMM				Ccam				Eng. Cameras & MMM															
				MMM Group	RMI Group	LIBS Group A	LIBS Group B	SOH Group	Sun Safe Group	Util Test Group	Eng. Camera Grp	MMM Group A	MMM Group B	MMM Group C	RMI Group	LIBS Group A	LIBS Group B	AEGIS	Inverse LUT	Rad Correction	Geom Correction	Disparity	Range	XYZ Group	Slope	Arm Reach Group	TC RDR Grp	Mosaic (CYL)	Mosaic (PER)	Mosaic (CYP)	Mosaic (POL)	Mosaic (VRT)	Mosaic (ORT) Grp	Ops Labelless Grp
227		GROUP	= HGA_ARTICULATION_STATE [HGA_ARTICULATION_STATE_PARMS]	X	X	X	X	X	X	X	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
228		ARTICULATION_DEVICE_ID	= "HGA"	X	X	X	X	X	X	X	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
229		ARTICULATION_DEVICE_NAME	= "HIGH GAIN ANTENNA"	X	X	X	X	X	X	X	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
230		ARTICULATION_DEVICE_ANGLE	= (0.0230152 <rad>, -0.076101 <rad>)	X	X	X	X	X	X	X	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
231		ARTICULATION_DEVICE_ANGLE_NAME	= ("AZIMUTH", "ELEVATION")	X	X	X	X	X	X	X	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
232		ARTICULATION_DEVICE_MODE	= "DEPLOYED"	X	X	X	X	X	X	X	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
233		END_GROUP	= HGA_ARTICULATION_STATE [HGA_ARTICULATION_STATE_PARMS]	X	X	X	X	X	X	X	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
234		/* OBSERVATION REQUEST */		X	o	o	o	o	o	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X	X	X	X	X	X	X
235		GROUP	= OBSERVATION_REQUEST_PARMS	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
236		SOURCE_ID	= "GROUND COMMANDED"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
237		AUTO_EXPOSURE_DATA_CUT	= 1024	X	•	X	X	X	X	X	•	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
238		AUTO_EXPOSURE_PERCENT	= 20.0	X	•	X	X	X	X	X	•	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
239		AUTO_EXPOSURE_PIXEL_FRACTION	= 50.0	X	•	X	X	X	X	X	•	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
240		BAD_PIXEL_REPLACEMENT_FLAG	= "FALSE"	X	•	X	X	X	X	X	•	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
241		COMMAND_INSTRUMENT_ID	= "NAVCAM_RIGHT"	X	•	X	X	X	X	X	•	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
242	{MSL:}	DARK_SPECTRA_MODE	= "PRE_AND_POST"	X	X	•	•	X	X	X	X	X	X	X	X	•	•	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
243		DETECTOR_ERASE_COUNT	= 10	X	•	X	X	X	X	X	•	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
244		EARLY_PIXEL_SCALE_FLAG	= "TRUE"	X	•	X	X	X	X	X	•	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
245		EARLY_IMAGE_RETURN_FLAG	= "TRUE"	X	•	X	X	X	X	X	•	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
246		EXPOSURE_TYPE	= AUTO	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
247		EXPOSURE_SCALE_FACTOR	= 4.2135	X	•	X	X	X	X	X	•	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
248		EXPOSURE_DURATION_COUNT	= 129	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
249		EXPOSURE_TABLE_ID	= "IMG_CAMCONFIG_NAVCAM_R"	X	•	X	X	X	X	X	•	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
250		EXPOSURE_TBL_UPDATE_FLAG	= "TRUE"	X	•	X	X	X	X	X	•	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
251		FILTER_NAME	= "NONE"	X	X	X	X	X	X	X	X	•	•	•	X	X	X	X	m	m	m	m	m	m	m	m	X	X	X	X	X	X	X	X
252		FILTER_NUMBER	= 0	X	X	X	X	X	X	X	X	•	•	•	X	X	X	X	m	m	m	m	m	m	m	m	X	X	X	X	X	X	X	X
253		FLAT_FIELD_CORRECTION_FLAG	= "TRUE"	X	•	X	X	X	X	X	•	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
254		GAIN_NUMBER	= 0	X	•	•	•	X	X	X	X	X	X	X	•	•	•	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
255		GROUP_APPLICABILITY_FLAG	= "FALSE"	X	•	•	•	X	X	X	X	X	X	X	•	•	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
256	{MSL:}	ICT_DIVIDER	= 300	X	X	•	•	X	X	X	X	X	X	X	X	•	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
257	{MSL:}	IPBC_DIVIDER	= 330	X	X	•	•	X	X	X	X	X	X	X	X	•	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
258	{MSL:}	INSTRUMENT_COORD_FRAME_ID	= "CAMERA_BAR"	X	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
259	{MSL:}	INSTRUMENT_COORD_FRAME_INDEX	= "N/A"	X	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
260		INSTRUMENT_COORDINATE	= (3.4589 <rad>, 38.90734 <rad>)	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
261		INSTRUMENT_COORDINATE_NAME	= "MAST AZEL"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
262	{MSL:}	INSTRUMENT_COORDINATE_TYPE	= "XYZ"	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X	X
263	{MSL:}	INSTRUMENT_FOCUS_DISTANCE	= 20	X	•	•	•	X	X	X	X	X	X	X	•	•	•	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
264	{MSL:}	INSTRUMENT_FOCUS_INIT_FLAG	= "FALSE"	X	•	X	X	X	X	X	X	X	X	X	•	X	X	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
265	{MSL:}	INSTRUMENT_FOCUS_MODE	= "NO_FOCUS"	X	•	•	•	X	X	X	X	X	X	X	•	•	•	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
266	{MSL:}	INSTRUMENT_FOCUS_POSITION_CNT	= "UNK"	X	•	X	X	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
267	{MSL:}	INSTRUMENT_FOCUS_STEPS	= "UNK"	X	•	X	X	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

	PDS Label	Keyword	Example Value/Comment	UDR	EDR										RDR																				
					Ccam						Eng. Cams & MMM				Ccam				Eng. Cameras & MMM															
				MMM Group	RMI Group	LIBS Group A	LIBS Group B	SOH Group	Sun Safe Group	Util Test Group	Eng. Camera Grp	MMM Group A	MMM Group B	MMM Group C	RMI Group	LIBS Group A	LIBS Group B	AEGIS	Inverse LUT	Rad Correction	Geom Correction	Disparity	Range	XYZ Group	Slope	Arm Reach Group	TC RDR Grp	Mosaic (CYL)	Mosaic (PER)	Mosaic (CYP)	Mosaic (POL)	Mosaic (VRT)	Mosaic (ORT) Grp	Ops Labelless Grp
268	{MSL:}	INSTRUMENT_FOCUS_STEP_SIZE	= "UNK"	X	•	X	X	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
269		INSTRUMENT_IDLE_TIMEOUT	= 1000 <s>	X	•	X	X	X	X	X	•	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
270		INSTRUMENT_MODE_ID	= "SPECTRAL_DATA"	X	•	•	•	X	X	X	X	X	X	X	•	•	•	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
271	{MSL:}	LASER_MODE	= "YES"	X	X	•	•	X	X	X	X	X	X	X	X	•	•	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
272		MAX_AUTO_EXPOS_ITERATION_COUNT	= 5	X	•	X	X	X	X	X	•	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
273	{MSL:}	N_SHOTS	= 30	X	X	•	•	X	X	X	X	X	X	X	X	•	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
274	{MSL:}	N_SHOTS_2_AVG	= 0	X	X	•	•	X	X	X	X	X	X	X	X	•	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
275	{MSL:}	N_SHOTS_2_IGNORE	= 0	X	X	•	•	X	X	X	X	X	X	X	X	•	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
276	{MSL:}	OBS_FROM_LIMIT_SWITCH	= 0	X	X	•	•	X	X	X	X	X	X	X	X	•	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
277		OFFSET_MODE_ID	= "4095"	X	•	•	•	X	X	X	X	X	X	X	•	•	•	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
278		SHUTTER_CORRECTION_MODE_ID	= "CONDITIONAL"	X	•	X	X	X	X	X	•	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
279		SHUTTER_CORRECT_THRESH_COUNT	= 10	X	•	X	X	X	X	X	•	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
280	{MSL:}	SPEC_AD_CONVERTUV	= 5	X	X	•	•	X	X	X	X	X	X	X	X	•	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
281	{MSL:}	SPEC_AD_CONVERTVIS	= 5	X	X	•	•	X	X	X	X	X	X	X	X	•	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
282	{MSL:}	SPEC_AD_CONVERTVNIR	= 5	X	X	•	•	X	X	X	X	X	X	X	X	•	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
283	{MSL:}	SPEC_IMAGE_TYPE	= 4	X	X	•	•	X	X	X	X	X	X	X	X	•	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
284	{MSL:}	SPEC_VERT_CLK	= 2	X	X	•	•	X	X	X	X	X	X	X	X	•	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
285	{MSL:}	SPECTROMETER_CONTROL_BYTE	= 2	X	X	•	•	X	X	X	X	X	X	X	X	•	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
286	{MSL:}	SPECTROMETER_SELECT	= 14	X	X	•	•	X	X	X	X	X	X	X	X	•	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
287	{MSL:}	SPECTROMETER_SERIAL_CLOCK	= 1	X	X	•	•	X	X	X	X	X	X	X	X	•	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
288	{MSL:}	STACK_1_LEVEL	= 100	X	X	•	•	X	X	X	X	X	X	X	X	•	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
289	{MSL:}	STACK_2_LEVEL	= 95	X	X	•	•	X	X	X	X	X	X	X	X	•	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
290	{MSL:}	STACK_3_LEVEL	= 95	X	X	•	•	X	X	X	X	X	X	X	X	•	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
291	{MSL:}	STACK_DURATION	= 145	X	X	•	•	X	X	X	X	X	X	X	X	•	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
292	{MSL:}	START_IMAGE_ID	= 3	X	•	X	X	X	X	X	X	X	X	X	•	X	X	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
293	{MSL:}	START_ROW_UV	= 100	X	X	•	•	X	X	X	X	X	X	X	X	•	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
294	{MSL:}	STOP_ROW_UV	= 300	X	X	•	•	X	X	X	X	X	X	X	X	•	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
295	{MSL:}	START_ROW_VIS	= 75	X	X	•	•	X	X	X	X	X	X	X	X	•	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
296	{MSL:}	STOP_ROW_VIS	= 275	X	X	•	•	X	X	X	X	X	X	X	X	•	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
297	{MSL:}	START_ROW_VNIR	= 100	X	X	•	•	X	X	X	X	X	X	X	X	•	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
298	{MSL:}	STOP_ROW_VNIR	= 300	X	X	•	•	X	X	X	X	X	X	X	X	•	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
299	{MSL:}	TIME_BETWEEN_SHOTS [CCAM_TIME_BETWEEN_SHOTS]	= 244	X	X	•	•	X	X	X	X	X	X	X	X	•	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
300	{MSL:}	VALID_MINIMUM_PIXEL	= 0	X	•	X	X	X	X	X	X	X	X	X	•	X	X	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
301	{MSL:}	VALID_MAXIMUM_PIXEL	= 1023	X	•	X	X	X	X	X	X	X	X	X	•	X	X	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
302		END_GROUP	= OBSERVATION_REQUEST_PARMS	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
303		/* IMAGE REQUEST */		•	o	X	X	X	X	X	o	o	o	o	o	X	X	o	o	o	o	o	o	o	o	o	o	X	X	X	X	X	X	X
304		GROUP	= IMAGE_REQUEST_PARMS	•	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
305		SOURCE_ID	= "GROUND COMMANDED"	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
306		GROUP_APPLICABILITY_FLAG	= "FALSE"	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
307		DOWNLOAD_PRIORITY	= 0	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
308		PIXEL_DOWNSAMPLE_OPTION	= SW_MEAN	•	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X

	PDS Label	Keyword	Example Value/Comment	UDR: MMM Group	EDR Ccam: RMI Group	EDR Ccam: LIBS Group A	EDR Ccam: LIBS Group B	EDR Ccam: SOH Group	EDR Ccam: Sun Safe Group	EDR Ccam: Util Test Group	EDR Eng. Cams & MMM: Eng. Camera Grp	EDR Eng. Cams & MMM: MMM Group A	EDR Eng. Cams & MMM: MMM Group B	EDR Eng. Cams & MMM: MMM Group C	RDR Ccam: RMI Group	RDR Ccam: LIBS Group A	RDR Ccam: LIBS Group B	RDR Ccam: AEGIS	RDR Eng. Cameras & MMM: Inverse LUT	Rad Correction	Geom Correction	Disparity	Range	XYZ Group	Slope	Arm Reach Group	TC RDR Grp	Mosaic (CYL)	Mosaic (PER)	Mosaic (CYP)	Mosaic (POL)	Mosaic (VRT)	Mosaic (ORT) Grp	Ops Labelless Grp
309		PIXEL_AVERAGING_HEIGHT	= 3	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
310		PIXEL_AVERAGING_WIDTH	= 3	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
311		SAMPLE_BIT_MODE_ID	= "LUT6"	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
312	Θ	INST_CMPRS_DEFERRED_FLAG	= "TRUE"	X	X	X	X	X	X	X	X	•	•	•	X	X	X	X	m	m	m	m	m	m	m	m	X	X	X	X	X	X	X	X
313		INST_CMPRS_MODE	= 1	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
314		INST_CMPRS_RATE	= 19.30945	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
315		INST_CMPRS_QUALITY	= 3	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
316		INST_CMPRS_FILTER	= A	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
317		INST_CMPRS_SEGMENTS	= 32	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
318		INST_DECOMP_STAGES	= 3	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
319		END_GROUP	= IMAGE_REQUEST_PARMS	•	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
320		/* SUBFRAME REQUEST */		•	o	X	X	X	X	X	o	o	o	o	o	X	X	o	o	o	o	o	o	o	o	o	o	X	X	X	X	X	X	X
321		GROUP	= SUBFRAME_REQUEST_PARMS	•	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
322		SOURCE_ID	= "GROUND COMMANDED"	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
323		GROUP_APPLICABILITY_FLAG	= "TRUE"	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
324		SUBFRAME_TYPE	= HW_SW	•	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
325		FIRST_LINE	= 1	•	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
326		FIRST_LINE_SAMPLE	= 1	•	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
327		LINES	= 100	•	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
328		LINE_SAMPLES	= 100	•	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
329		END_GROUP	= SUBFRAME_REQUEST_PARMS	•	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
330		/* THUMBNAIL REQUEST */		X	o	X	X	X	X	X	o	o	o	o	o	X	X	o	o	o	o	o	o	o	o	o	o	X	X	X	X	X	X	X
331		GROUP	= THUMBNAIL_REQUEST_PARMS	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
332		SOURCE_ID	= "GROUND COMMANDED"	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
333		GROUP_APPLICABILITY_FLAG	= "FALSE"	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
334		DOWNLOAD_PRIORITY	= 0	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
335		LINES	= 256	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
336		LINE_SAMPLES	= 256	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
337		SAMPLE_BIT_MODE_ID	= "LUT3"	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
338	Θ	INST_CMPRS_DEFERRED_FLAG	= "TRUE"	X	X	X	X	X	X	X	X	•	•	•	X	X	X	X	m	m	m	m	m	m	m	m	X	X	X	X	X	X	X	X
339		INST_CMPRS_MODE	= 1	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
340		INST_CMPRS_RATE	= 19.30945	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
341		INST_CMPRS_QUALITY	= 3	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
342		INST_CMPRS_FILTER	= A	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
343		INST_CMPRS_SEGMENTS	= 32	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
344		INST_DECOMP_STAGES	= 3	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
345		END_GROUP	= THUMBNAIL_REQUEST_PARMS	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
346		/* ROW SUMMATION REQUEST */		X	X	X	X	X	X	X	o	X	X	X	X	X	X	X	o	o	o	o	o	o	o	o	X	X	X	X	X	X	X	X

	PDS Label	Keyword	Example Value/Comment	UDR: MMM Group	EDR Ccam: RMI Group	EDR Ccam: LIBS Group A	EDR Ccam: LIBS Group B	EDR Ccam: SOH Group	EDR Ccam: Sun Safe Group	EDR Ccam: Util Test Group	EDR Eng. Cams & MMM: Eng. Camera Grp	EDR Eng. Cams & MMM: MMM Group A	EDR Eng. Cams & MMM: MMM Group B	EDR Eng. Cams & MMM: MMM Group C	RDR Ccam: RMI Group	RDR Ccam: LIBS Group A	RDR Ccam: LIBS Group B	RDR Ccam: AEGIS	RDR Eng. Cameras & MMM: Inverse LUT	Rad Correction	Geom Correction	Disparity	Range	XYZ Group	Slope	Arm Reach Group	TC RDR Grp	Mosaic (CYL)	Mosaic (PER)	Mosaic (CYP)	Mosaic (POL)	Mosaic (VRT)	Mosaic (ORT) Grp	Ops Labelless Grp
347		GROUP	= ROW_SUM_REQUEST_PARMS	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X	X
348		SOURCE_ID	= "GROUND COMMANDED"	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X	X
349		GROUP_APPLICABILITY_FLAG	= "TRUE"	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X	X
350		DOWNLOAD_PRIORITY	= 9	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X	X
351		END_GROUP	= ROW_SUM_REQUEST_PARMS	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X	X
352		/* COLUMN SUMMATION REQUEST */		X	X	X	X	X	X	X	o	X	X	X	X	X	X	X	o	o	o	o	o	o	o	o	o	X	X	X	X	X	X	X
353		GROUP	= COLUMN_SUM_REQUEST_PARMS	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
354		SOURCE_ID	= "GROUND COMMANDED"	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
355		GROUP_APPLICABILITY_FLAG	= "TRUE"	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
356		DOWNLOAD_PRIORITY	= 8	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
357		END_GROUP	= COLUMN_SUM_REQUEST_PARMS	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
358		/* REFERENCE PIXEL REQUEST */		X	o	X	X	X	X	X	o	X	X	X	o	X	X	o	o	o	o	o	o	o	o	o	o	X	X	X	X	X	X	X
359		GROUP	= REFERENCE_PIXEL_REQUEST_PARMS	X	•	X	X	X	X	X	•	X	X	X	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
360		SOURCE_ID	= "GROUND COMMANDED"	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
361		GROUP_APPLICABILITY_FLAG	= "FALSE"	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
362		DOWNLOAD_PRIORITY	= 0	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
363	Θ	INST_CMPRS_DEFERRED_FLAG	= "TRUE"	X	X	X	X	X	X	X	X	•	•	•	X	X	X	X	m	m	m	m	m	m	m	m	X	X	X	X	X	X	X	X
364		INST_CMPRS_MODE	= 1	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
365		INST_CMPRS_RATE	= 19.30945	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
366		INST_CMPRS_QUALITY	= 3	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
367		INST_CMPRS_FILTER	= A	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
368		INST_CMPRS_SEGMENTS	= 32	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
369		INST_DECOMP_STAGES	= 3	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
370		END_GROUP	= REFERENCE_PIXEL_REQUEST_PARMS	X	•	X	X	X	X	X	•	X	X	X	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
371		/* HISTOGRAM REQUEST */		X	X	X	X	X	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	o	X	X	X	X	X	X	X
372		GROUP	= HISTOGRAM_REQUEST_PARMS	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X
373		SOURCE_ID	= "GROUND COMMANDED"	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X
374		GROUP_APPLICABILITY_FLAG	= "TRUE"	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X
375		DOWNLOAD_PRIORITY	= 10	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X
376		END_GROUP	= HISTOGRAM_REQUEST_PARMS	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X
377		/* INSTRUMENT STATE RESULTS */		X	o	o	o	o	o	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X	X	X	X	X	X	X
378		GROUP	= INSTRUMENT_STATE_PARMS	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
379		AZIMUTH_FOV	= 14.0032 <deg>	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
380		ELEVATION_FOV	= 13.5656 <deg>	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
381		BAD_PIXEL_REPLACEMENT_ID	= "3"	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
382	Θ	BAYER_MODE	= MIPL_INTERPOLATED	X	X	X	X	X	X	X	X	•	•	•	X	X	X	X	m	m	m	m	m	m	m	m	X	X	X	X	X	X	X	X

	PDS Label	Keyword		Example Value/Comment	UDR	EDR										RDR																			
						Ccam						Eng. Cams & MMM				Ccam				Eng. Cameras & MMM															
					MMM Group	RMI Group	LIBS Group A	LIBS Group B	SOH Group	Sun Safe Group	Util Test Group	Eng. Camera Grp	MMM Group A	MMM Group B	MMM Group C	RMI Group	LIBS Group A	LIBS Group B	AEGIS	Inverse LUT	Rad Correction	Geom Correction	Disparity	Range	XYZ Group	Slope	Arm Reach Group	TC RDR Grp	Mosaic (CYL)	Mosaic (PER)	Mosaic (CYP)	Mosaic (POL)	Mosaic (VRT)	Mosaic (ORT) Grp	Ops Labelless Grp
383		DETECTOR_FIRST_LINE	=	1	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
384		DETECTOR_LINES	=	430	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
385		DETECTOR_TO_IMAGE_ROTATION	=	90	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
386		DOWNSAMPLE_METHOD	=	HARDWARE	X	•	X	X	X	X	X	•	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
387		EXPOSURE_COUNT	=	33	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
388		EXPOSURE_DURATION	=	216.0 <ms>	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
389		EXPOSURE_DURATION_COUNT	=	3	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
390		FILTER_NAME	=	"NONE"	X	X	X	X	X	X	X	X	•	X	X	X	X	X	X	m	m	m	m	m	m	m	m	X	X	X	X	X	X	X	X
391		FILTER_NUMBER	=	0	X	X	X	X	X	X	X	X	•	X	X	X	X	X	X	m	m	m	m	m	m	m	m	X	X	X	X	X	X	X	X
392		FLAT_FIELD_CORRECTION_FLAG	=	"TRUE"	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
393		FLAT_FIELD_CORRECTION_PARM	=	(-10.4344, -11.4673,-10.4344, -11.4673, -10.4344)	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
394		INSTRUMENT_MODE_ID	=	"FULL_FRAME"	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
395		INSTRUMENT_TEMPERATURE	=	(-10.4344 <degC>, -11.4673 <degC>, -10.4344 <degC>, -11.4673 <degC>, -10.4344 <degC>, -11.4673 <degC>, - 5.0345 <degC>, -16.7433 <degC>, -10.4344 <degC>, - 11.4673 <degC>, -10.4344 <degC>, -11.4673 <degC>, - 10.4344 <degC>, -11.4673 <degC>, 5.0345 <degC>, - 16.7433 <degC>, -10.4344 <degC>, -11.4673 <degC>, - 10.4344 <degC>, -11.4673 <degC>, -10.4344 <degC>, - 11.4673 <degC>, -5.0345 <degC>, -16.7433 <degC>)	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
396		INSTRUMENT_TEMPERATURE_NAME	=	(“A_FRONT_LEFT_HAZ_CCD”, “A_FRONT_LEFT_HAZ_ELECTRONICS”, “A_FRONT_RIGHT_HAZ_CCD”, “A_FRONT_RIGHT_HAZ_ELECTRONICS”, “A_REAR_LEFT_HAZ_CCD”, “A_REAR_LEFT_HAZ_ELECTRONICS”, “A_REAR_RIGHT_HAZ_CCD”, “A_REAR_RIGHT_HAZ_ELECTRONICS”, “A_LEFT_NAV_CCD”, “A_LEFT_NAV_ELECTRONICS”, “A_RIGHT_NAV_CCD”, “A_RIGHT_NAV_ELECTRONICS”, “B_FRONT_LEFT_HAZ_CCD”, “B_FRONT_LEFT_HAZ_ELECTRONICS”, “B_FRONT_RIGHT_HAZ_CCD”, “B_FRONT_RIGHT_HAZ_ELECTRONICS”, “B_REAR_LEFT_HAZ_CCD”, “B_REAR_LEFT_HAZ_ELECTRONICS”, “B_REAR_RIGHT_HAZ_CCD”, “B_REAR_RIGHT_HAZ_ELECTRONICS”, “B_LEFT_NAV_CCD”, “B_LEFT_NAV_ELECTRONICS”, “B_RIGHT_NAV_CCD”, “B_RIGHT_NAV_ELECTRONICS”)	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
397	{MSL:}	INSTRUMENT_TEMPERATURE_STATUS	=	(0, 0, 0, 0, 0, 0, 0, 0)	X	•	•	•	•	•	X	X	X	X	X	•	•	•	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
398		OFFSET_MODE_ID	=	"4095"	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
399		PIXEL_AVERAGING_HEIGHT	=	3	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
400		PIXEL_AVERAGING_WIDTH	=	3	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
401		SAMPLE_BIT_METHOD	=	HARDWARE	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
402		SAMPLE_BIT_MODE_ID	=	"LUT6"	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
403		SHUTTER_EFFECT_CORRECTION_FLAG	=	"TRUE"	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
404	{MSL:}	SPECIAL_LINE	=	497	X	•	X	X	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
405	{MSL:}	SPECIAL_SAMPLE	=	532	X	•	X	X	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

#	PDS Label	Keyword	Example Value/Comment	UDR: MMM Group	EDR Ccam: RMI Group	EDR Ccam: LIBS Group A	EDR Ccam: LIBS Group B	EDR Ccam: SOH Group	EDR Ccam: Sun Safe Group	EDR Ccam: Util Test Group	EDR Eng. Cams & MMM: Eng. Camera Grp	EDR Eng. Cams & MMM: MMM Group A	EDR Eng. Cams & MMM: MMM Group B	EDR Eng. Cams & MMM: MMM Group C	RDR Ccam: RMI Group	RDR Ccam: LIBS Group A	RDR Ccam: LIBS Group B	RDR Ccam: AEGIS	RDR Eng. Cameras & MMM: Inverse LUT	Rad Correction	Geom Correction	Disparity	Range	XYZ Group	Slope	Arm Reach Group	TC RDR Grp	Mosaic (CYL)	Mosaic (PER)	Mosaic (CYP)	Mosaic (POL)	Mosaic (VRT)	Mosaic (ORT) Grp	Ops Labelless Grp
406	{MSL:}	SPECIAL_NAME	= "Location of the LIBS laser spot in the RMI image described by the camera model"	X	•	X	X	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
407		END_GROUP	= INSTRUMENT_STATE_PARMS	X	•	•	•	•	•	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
408		/* INSTRUMENT INITIAL STATE RESULTS */		X	o	X	X	X	X	X	o	o	o	o	o	X	X	X	o	o	o	o	o	o	o	o	o	X	X	X	X	X	X	X
409		GROUP	= INITIAL_STATE_PARMS	X	•	X	X	X	X	X	•	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
410		ROVER_MOTION_COUNTER	= (1, 2, 0, 0, 0, 10, 6, 12, 163, 0)	X	•	X	X	X	X	X	•	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
411		ROVER_MOTION_COUNTER_NAME	= ("SITE", "DRIVE", "POSE", "ARM", "CHIMRA", "DRILL", "RSM", "HGA", "DRT", "IC")	X	•	X	X	X	X	X	•	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
412		ORIGIN_OFFSET_VECTOR	= (0.0230152, -0.076101, 0.874005)	X	•	X	X	X	X	X	•	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
413		ORIGIN_ROTATION_QUATERNION	= (0.922297, -0.0165226, -0.0413094, 0.382304)	X	•	X	X	X	X	X	•	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
414		QUATERNION_MEASUREMENT_METHOD	= FINE	X	•	X	X	X	X	X	•	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
415		END_GROUP	= INITIAL_STATE_PARMS	X	•	X	X	X	X	X	•	•	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
416		/* COMPRESSION RESULTS */		X	o	X	X	X	X	X	o	o	o	o	o	X	X	o	o	o	o	o	o	o	o	o	o	X	X	X	X	X	X	X
417		GROUP	= COMPRESSION_PARMS	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
418		ERROR_PIXELS	= 0	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
419		INST_CMPRS_FILTER	= A	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
420	⊖	INST_CMPRS_COLOR_MODE	= COLOR_MODE_GRAY	X	X	X	X	X	X	X	X	•	•	•	X	X	X	X	m	m	m	m	m	m	m	m	X	X	X	X	X	X	X	X
421	⊖	INST_CMPRS_DEFERRED_FLAG	= "TRUE"	X	X	X	X	X	X	X	X	•	•	•	X	X	X	X	m	m	m	m	m	m	m	m	X	X	X	X	X	X	X	X
422		INST_CMPRS_MODE	= 1	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
423		INST_CMPRS_NAME	= ICER	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
424		INST_CMPRS_QUALITY	= 0	X	X	X	X	X	X	X	X	•	•	•	X	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
425		INST_CMPRS_RATE	= 7.1787	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
426		INST_CMPRS_RATIO	= 1.6716	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
427		INST_CMPRS_SEGMENTS	= 4	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
428		INST_CMPRS_SEGMENT_QUALITY	= (0,0,0,0)	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
429		INST_CMPRS_SEGMENT_STATUS	= (1,7,11,21)	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
430		INST_CMPRS_SEG_FIRST_LINE	= (10,24,13,60)	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
431		INST_CMPRS_SEG_FIRST_LINE_SAMP	= (5,5,5,5)	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
432		INST_CMPRS_SEG_LINES	= (20,20,20,20)	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
433		INST_CMPRS_SEG_SAMPLES	= (10,10,30,30)	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
434		INST_CMPRS_SEG_MISSING_PIXELS	= "N/A"	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
435		INST_DECOMP_STAGES	= 4	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
436		END_GROUP	= COMPRESSION_PARMS	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
437		/* DERIVED GEOMETRY DATA ELEMENTS: ROVER FRAME */		X	o	o	o	o	o	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X	X	X	X	X	X	X
438		GROUP	= ROVER_DERIVED_GEOMETRY_PARMS	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
439		INSTRUMENT_AZIMUTH	= 92.0847 <deg>	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
440		INSTRUMENT_ELEVATION	= 0.894 <deg>	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
441		SOLAR_AZIMUTH	= 6.5029 <deg>	X	X	X	X	X	X	X	•	X	X	X	X	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
442		SOLAR_ELEVATION	= 69.1396 <deg>	X	X	X	X	X	X	X	•	X	X	X	X	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X

#	PDS Label	Keyword	Example Value/Comment	UDR: MMM Group	EDR Ccam: RMI Group	EDR Ccam: LIBS Group A	EDR Ccam: LIBS Group B	EDR Ccam: SOH Group	EDR Ccam: Sun Safe Group	EDR Ccam: Util Test Group	EDR Eng. Cams & MMM: Eng. Camera Grp	EDR Eng. Cams & MMM: MMM Group A	EDR Eng. Cams & MMM: MMM Group B	EDR Eng. Cams & MMM: MMM Group C	RDR Ccam: RMI Group	RDR Ccam: LIBS Group A	RDR Ccam: LIBS Group B	RDR Ccam: AEGIS	RDR Eng. Cameras & MMM: Inverse LUT	Rad Correction	Geom Correction	Disparity	Range	XYZ Group	Slope	Arm Reach Group	TC RDR Grp	Mosaic (CYL)	Mosaic (PER)	Mosaic (CYP)	Mosaic (POL)	Mosaic (VRT)	Mosaic (ORT) Grp	Ops Labelless Grp
443		SUN_VIEW_DIRECTION	= (-0.392242,-0.283997,-0.874924)	X	X	X	X	X	X	X	•	X	X	X	X	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
444		REFERENCE_COORD_SYSTEM_NAME	= "ROVER_NAV_FRAME"	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
445		REFERENCE_COORD_SYSTEM_INDEX	= (1, 2, 0, 0, 0, 10, 6, 12, 163, 0)	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
446	{MSL:}	REFERENCE_COORD_SYSTEM_SOLN_ID	= "mipl_rgd_sol3nav_5"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	r	r	r	r	r	r	X	X	X	X	X	X	X
447		END_GROUP	= ROVER_DERIVED_GEOMETRY_PARMS	X	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
448		/* DERIVED GEOMETRY DATA ELEMENTS: SITE FRAME */		X	X	X	X	X	X	X	o	o	o	o	X	X	X	r	o	o	o	o	o	o	o	o	o	X	X	X	X	X	X	X
449		GROUP	= SITE_DERIVED_GEOMETRY_PARMS	X	X	X	X	X	X	X	•	•	•	•	X	X	X	r	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
450		INSTRUMENT_AZIMUTH	= 131.808 <deg>	X	X	X	X	X	X	X	•	•	•	•	X	X	X	r	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
451		INSTRUMENT_ELEVATION	= -18.2877 <deg>	X	X	X	X	X	X	X	•	•	•	•	X	X	X	r	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
452		START_AZIMUTH	= 360 <deg>	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X	X
453		STOP_AZIMUTH	= 360 <deg>	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X	X
454		SOLAR_AZIMUTH	= 102.347 <deg>	X	X	X	X	X	X	X	•	X	X	X	X	X	X	r	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
455		SOLAR_ELEVATION	= 69.2342 <deg>	X	X	X	X	X	X	X	•	X	X	X	X	X	X	r	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
456		REFERENCE_COORD_SYSTEM_NAME	= "SITE_FRAME"	X	X	X	X	X	X	X	•	•	•	•	X	X	X	r	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
457		REFERENCE_COORD_SYSTEM_INDEX	= 1	X	X	X	X	X	X	X	•	•	•	•	X	X	X	r	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
458	{MSL:}	REFERENCE_COORD_SYSTEM_SOLN_ID	= "mipl_rgd_sol3nav_5"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	r	r	r	r	r	r	X	X	X	X	X	X	X
459		END_GROUP	= SITE_DERIVED_GEOMETRY_PARMS	X	X	X	X	X	X	X	•	•	•	•	X	X	X	r	•	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X
460		/* DERIVED IMAGE DATA ELEMENTS */		X	X	X	X	X	X	X	X	X	X	X	X	X	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X
461		GROUP	= DERIVED_IMAGE_PARMS	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X
462	{MSL:}	BIAS_COEFFS_FILE_DESC	= "Bias coefficients file."	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	X	X	X	X	X	X	X	X	X	X	X	X	X
463	{MSL:}	BIAS_COEFFS_FILE_NAME	= "MSL_ccd_115_bias_coeffs_01.dat"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	X	X	X	X	X	X	X	X	X	X	X	X	X
464	{MSL:}	BRIGHTNESS_CORRECTION_FILE	= "MSL_ccd_115_bias_coeffs_01.dat"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	X	X	X	X	X	X	X	X	X	X	X	X	X
465	{MSL:}	BRIGHTNESS_CORRECTION_TYPE	= "MSL_ccd_115_brightness_corr_01.xml"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	X	X	X	X	X	X	X	X	X	X	X	X	X
466	⊖	CLASSIFIER_BAND_HUE	= ("#9933FF", "#33CC00", "#3366CC", "#FFFF33", "#FF6600", "#FF0033", "#006600")	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X
467	⊖	CLASSIFIER_BAND_INDEX	= (0, 1, 2, 3, 4, 5, 6) *[TENL]* ("CLASSIFIER_LABEL_INDEX", "CONFIDENCE") *[TERL]*	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X
468	⊖	CLASSIFIER_BAND_INDEX_NAME	= ("Ripples", "Smooth", "Smooth with rocks", "Murray smooth or caprock smooth", "Murray rough", "Caprock or caprock rough", "Rover tracks")	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X
469	⊖	CLASSIFIER_LABEL_HUE	= ("#9933FF", "#33CC00", "#3366CC", "#FFFF33", "#FF6600", "#FF0033", "#006600", "#FFFFFF")	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X
470	⊖	CLASSIFIER_LABEL_INDEX	= (0, 1, 2, 3, 4, 5, 6, 255)	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X
471	⊖	CLASSIFIER_LABEL_INDEX_NAME	= ("Ripples", "Smooth", "Smooth with rocks", "Murray smooth or caprock smooth", "Murray rough", "Caprock or caprock rough", "Rover tracks", "None")	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X
472	⊖	CLASSIFIER_MAX_RANGE	= 22 <m>	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X
473	⊖	CLASSIFIER_PROJECTION_SCALE	= 0.008789 <m/px>	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X

#	PDS Label	Keyword	Example Value/Comment	UDR: MMM Group	EDR Ccam: RMI Group	EDR Ccam: LIBS Group A	EDR Ccam: LIBS Group B	EDR Ccam: SOH Group	EDR Ccam: Sun Safe Group	EDR Ccam: Util Test Group	EDR Eng. Cams & MMM: Eng. Camera Grp	EDR Eng. Cams & MMM: MMM Group A	EDR Eng. Cams & MMM: MMM Group B	EDR Eng. Cams & MMM: MMM Group C	RDR Ccam: RMI Group	RDR Ccam: LIBS Group A	RDR Ccam: LIBS Group B	RDR Ccam: AEGIS	RDR Eng. Cameras & MMM: Inverse LUT	Rad Correction	Geom Correction	Disparity	Range	XYZ Group	Slope	Arm Reach Group	TC RDR Grp	Mosaic (CYL)	Mosaic (PER)	Mosaic (CYP)	Mosaic (POL)	Mosaic (VRT)	Mosaic (ORT) Grp	Ops Labelless Grp
474	⊖	CLASSIFIER_VERSION	= 2.1	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X
475	{MSL:}	CONFIGURATION_BIT_ID	= ("ELBOW_UP_WRIST_UP", "ELBOW_UP_WRIST_DOWN", "ELBOW_DOWN_WRIST_UP", "ELBOW_DOWN_WRIST_DOWN")	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	•	•	•	•	•	d	d	X	X	X	X	X	X	X	X
476		DARK_CURRENT_FILE_DESC	= "Active dark coefficients image."	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	•	X	X	X	X	X	X	X	X	X	X	X	X	X
477		DARK_CURRENT_FILE_NAME	= "MSL_ccd_115_dark_active_coeffs_01.img"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	•	X	X	X	X	X	X	X	X	X	X	X	X	X
478		DERIVED_IMAGE_TYPE	= XYZ_MAP	X	X	X	X	X	X	X	X	X	X	X	X	X	X	d	•	d	•	d	d	d	d	d	d	X	X	X	X	X	X	X
479		FLAT_FIELD_FILE_DESC	= "Flat field derived from sky flat sequence NCAM00565 produced by Mark Lemmon at Texas A and M"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	X	X	X	X	X	X	X	X	X	X	X	X	X
480		FLAT_FIELD_FILE_NAME	= "MSL_FLAT_SN_0215_V03"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	X	X	X	X	X	X	X	X	X	X	X	X	X
481	{MSL:}	IMAGE_RADIANCE_FACTOR	= "UNK"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	X	X	X	X	X	X	X	X	X	X	X	X	X
482	{MSL:}	IMAGE_RADIANCE_OFFSET	= "UNK"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	X	X	X	X	X	X	X	X	X	X	X	X	X
483	{MSL:}	INPUT_PRODUCT_ID	= "NRA_411689704EDR_F0000000001015808M1"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	•	•	•	•	•	•	•	d	X	X	X	X	X	X	X
484		INSTRUMENT_BAND_ID	= ("DRILL", "DRT", "MAHLI", "APXS", "PORTIONER", "SCOOP")	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X	X
485	{MSL:}	INVERSE_LUT_FILE_NAME	= "MSL_inverse_lut_2.txt"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X	X
486	{MSL:}	LINEARIZATION_MODE	= "NOMINAL"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X	X
487	⊖	LINEARIZATION_PRODUCT_ID	= TBD	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X	X
488	{MSL:}	NUM_SOFTWARE_KEYWORDS	= 1	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	X	X	X	X	X	X	X	X	X	X	X	X	X
489	{MSL:}	NUM_SOFTWARE_PARAMETERS	= 1	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	X	X	X	X	X	X	X	X	X	X	X	X	X
490	{MSL:}	POINTING_CORRECTION_FILE_NAME	= "MSL_ccd_115_pointing_corr_01.xml"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X	X
491	{MSL:}	POINTING_MODEL_NAME	= "NONE"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X	X
492	{MSL:}	POINTING_MODEL_PARAMS	= "UNK"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X	X
493	{MSL:}	POINTING_MODEL_PARAMS_NAME	= ("AZIMUTH", "ELEVATION")	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X	X
494	{MSL:}	RADIANCE_SCALING_FACTOR	= 5 <W/m^2/nm/sr>	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	X	X	X	X	X	•	X	X	X	X	X	X	X
495	{MSL:}	RADIANCE_OFFSET	= 1000 <W/m^2/nm/sr>	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	X	X	X	X	X	•	X	X	X	X	X	X	X
496		RADIOMETRIC_CORRECTION_TYPE	= NAVCAL	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	d	•	•	•	•	•	•	•	d	d	d	d	d	d	X
497	{MSL:}	RANGE_ORIGIN_VECTOR	= (0.0230152, -0.076101, 0.874005)	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	•	•	•	•	d	•	•	X	•	•	•	•	•	•	X
498		REFERENCE_COORD_SYSTEM_NAME	= "SITE_FRAME"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	•	•	•	d	•	d	d	X	•	•	•	•	•	•	X
499		REFERENCE_COORD_SYSTEM_INDEX	= 1	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	•	•	•	d	•	d	d	X	•	•	•	•	•	•	X
500	{MSL:}	REFERENCE_COORD_SYSTEM_SOLN_ID	= "mipl_rgd_sol3nav_5"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	r	r	r	r	r	X	X	X	X	X	X	X	X
501	{MSL:}	REFERENCE_PIXEL_IMAGE_ID	= "NRA_411689704EDR_F0000000001015808M1"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	X	X	X	X	X	X	X	X	X	X	X	X	X
502	⊖	REGION_COUNT	62	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
503	{MSL:}	RESPONSIVITY_CONSTANTS	= (4.750000000000000E-06, 3.607000000000000E-09, 0.000000000000000)	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	•	X	X	X	X	X	X	X	X	X	X	X	X	X
504	{MSL:}	RESPONSIVITY_CONSTANTS_FILE	= "default_responsivity_constants.txt"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	•	X	X	X	X	X	X	X	X	X	X	X	X	X
505	{MSL:}	SOFTWARE_LANGUAGE	= "IDL"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	X	X	X	X	X	X	X	X	X	X	X	X	X
506	{MSL:}	SOFTWARE_MODULE_NAME	= "MSL_NAVCAL_IMAGE"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	X	X	X	X	X	X	X	X	X	X	X	X	X
507	{MSL:}	SOFTWARE_MODULE_TYPE	= FUNCTION	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	X	X	X	X	X	X	X	X	X	X	X	X	X
508	{MSL:}	SOFTWARE_KEYWORD_NAME	= "AMBIENT"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	X	X	X	X	X	X	X	X	X	X	X	X	X
509	{MSL:}	SOFTWARE_KEYWORD_TYPE	= INT	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	X	X	X	X	X	X	X	X	X	X	X	X	X
510	{MSL:}	SOFTWARE_KEYWORD_VALUE	= 0	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	X	X	X	X	X	X	X	X	X	X	X	X	X
511	{MSL:}	SOFTWARE_PARAMETER_NAME	= "FILENAME"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	X	X	X	X	X	X	X	X	X	X	X	X	X

#	PDS Label	Keyword	Example Value/Comment	UDR: MMM Group	EDR Ccam: RMI Group	EDR Ccam: LIBS Group A	EDR Ccam: LIBS Group B	EDR Ccam: SOH Group	EDR Ccam: Sun Safe Group	EDR Ccam: Util Test Group	EDR Eng. Cams & MMM: Eng. Camera Grp	EDR Eng. Cams & MMM: MMM Group A	EDR Eng. Cams & MMM: MMM Group B	EDR Eng. Cams & MMM: MMM Group C	RDR Ccam: RMI Group	RDR Ccam: LIBS Group A	RDR Ccam: LIBS Group B	RDR Ccam: AEGIS	RDR Eng. Cameras & MMM: Inverse LUT	Rad Correction	Geom Correction	Disparity	Range	XYZ Group	Slope	Arm Reach Group	TC RDR Grp	Mosaic (CYL)	Mosaic (PER)	Mosaic (CYP)	Mosaic (POL)	Mosaic (VRT)	Mosaic (ORT) Grp	Ops Labelless Grp
512	{MSL:}	SOFTWARE_PARAMETER_TYPE	= STRING	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	X	X	X	X	X	X	X	X	X	X	X	X	X
513	{MSL:}	SOFTWARE_PARAMETER_VALUE	= "NRA_411689704EDR_F0000000001015808M1.gz"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	X	X	X	X	X	X	X	X	X	X	X	X	X
514	{MSL:}	SOURCE_PRODUCT_WAVELENGTH	= 0	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	X	X	X	X	X	X	X	X	X	X	X	X	X
515	{MSL:}	STEREO_PRODUCT_ID	= "UNK"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X	X
516	⊖	STEREO_BASELINE	= 1.332	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	•	•	•	•	•	•	•	X	X	X	X	X	X	X	X
517	{MSL:}	ZERO_EXPOSURE_IMAGE_ID	= "NRA_411689704EDR_F0000000001015808M1"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	X	X	X	X	X	X	X	X	X	X	X	X	X
518		^MODEL_DESC	= "MSL_ccd_115_pointing_model_01.txt"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	o	o	o	o	o	o	o	o	X	o	o	o	o	o	o	X
519		^MOSAIC_DESC	= "MSL_MOSAIC_DESC.TXT"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	o	o	o	o	o	o	X
520		END_GROUP	= DERIVED_IMAGE_PARMS	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	•	X
521		/* SURFACE PROJECTION DATA ELEMENTS */		X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	o	o	o	o	o	o	X
522		GROUP	= SURFACE_PROJECTION_PARMS	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	•	•	•	•	•	X
523	{MSL:}	CAMERA_ROTATION_AXIS_VECTOR	= (0.0, 0.0, 0.0)	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	X	X	X	X
524		LINE_CAMERA_MODEL_OFFSET	= -11.7843 <pixel>	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	d	X	X	X	X	X
525		SAMPLE_CAMERA_MODEL_OFFSET	= -17.1418 <pixel>	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	d	X	X	X	X	X
526		LINE_PROJECTION_OFFSET	= 749.0 <pixel>	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	d	d	d	X
527		SAMPLE_PROJECTION_OFFSET	= 749.0 <pixel>	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	d	d	d	X
528		MAP_PROJECTION_TYPE	= CYLINDRICAL	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	d	d	d	d	d	d	X
529		MAP_RESOLUTION	= (17.7607 <pix/deg>, 17.7619 <pix/deg>)	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	d	d	d	d	X	X	X
530		MAP_SCALE	= (0.01 <m/pixel>, 0.01 <m/pixel>)	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	d	d	X
531	{MSL:}	MAXIMUM_ELEVATION	= -33.5129 <deg>	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	d	r	r	d	X	X	X
532	{MSL:}	MINIMUM_ELEVATION	= -74.5294 <deg>	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	d	r	r	X	X	X	X
533		PROJECTION_AZIMUTH	= -10.6724 <deg>	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	d	r	X	X	X	X
534		PROJECTION_ELEVATION	= -22.0756 <deg>	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	d	d	X	X	X	X
535		PROJECTION_ELEVATION_LINE	= 1395.52 <pixel>	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	d	X	X	X	X
536		PROJECTION_ORIGIN_VECTOR	= (-0.334 <m>, 0.202 <m>, -1.816 <m>)	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	d	X	r	d	X	d	X
537	{MSL:}	PROJECTION_AXIS_OFFSET	= 0.0 <m>	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	X	X	•	X
538	{MSL:}	PROJECTION_X_AXIS_VECTOR	= 0.0 <m>	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	X
539	{MSL:}	PROJECTION_Y_AXIS_VECTOR	= 0.0 <m>	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	X
540	{MSL:}	PROJECTION_Z_AXIS_VECTOR	= 0.0 <m>	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	X	X	X	X
541		REFERENCE_AZIMUTH	= 0.0 <deg>	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	d	X	X	X
542		REFERENCE_COORD_SYSTEM_NAME	= "LOCAL_LEVEL_FRAME"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	d	d	d	d	d	d	X
543		REFERENCE_COORD_SYSTEM_INDEX	= 1	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	d	d	d	d	d	d	X
544	{MSL:}	REFERENCE_COORD_SYSTEM_SOLN_ID	= "mipl_rgd_sol2nav_5"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	r	r	r	X
545		START_AZIMUTH	= 360 <deg>	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	d	X	d	X	X	X	X
546		STOP_AZIMUTH	= 360 <deg>	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	d	X	d	X	X	X	X
547		X_AXIS_MAXIMUM	= 5.0 <m>	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	d	d	d	X
548		X_AXIS_MINIMUM	= -5.0 <m>	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	d	d	d	X
549		Y_AXIS_MAXIMUM	= -5.0 <m>	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	d	d	d	X
550		Y_AXIS_MINIMUM	= -5.0 <m>	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	d	d	d	X
551		ZERO_ELEVATION_LINE	= -443.019 <pixel>	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	d	X	X	X	X	X	X
552		END_GROUP	= SURFACE_PROJECTION_PARMS	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	•	•	•	•	•	X

	PDS Label	Keyword	Example Value/Comment	UDR: MMM Group	EDR Ccam: RMI Group	EDR Ccam: LIBS Group A	EDR Ccam: LIBS Group B	EDR Ccam: SOH Group	EDR Ccam: Sun Safe Group	EDR Ccam: Util Test Group	EDR Eng. Cams & MMM: Eng. Camera Grp	EDR Eng. Cams & MMM: MMM Group A	EDR Eng. Cams & MMM: MMM Group B	EDR Eng. Cams & MMM: MMM Group C	RDR Ccam: RMI Group	RDR Ccam: LIBS Group A	RDR Ccam: LIBS Group B	RDR Ccam: AEGIS	RDR Eng. Cameras & MMM: Inverse LUT	Rad Correction	Geom Correction	Disparity	Range	XYZ Group	Slope	Arm Reach Group	TC RDR Grp	Mosaic (CYL)	Mosaic (PER)	Mosaic (CYP)	Mosaic (POL)	Mosaic (VRT)	Mosaic (ORT) Grp	Ops Labelless Grp
553		/* SURFACE MODEL DATA ELEMENTS */		X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	o	o	o	o	o	o	X
554		GROUP	= SURFACE_MODEL_PARMS	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	•	•	•	•	•	X
555	{MSL:}	SURFACE_MODEL_FILE_NAME	= "1N863294624XYL411100171L0M1"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	r	r	r	X
556		SURFACE_MODEL_TYPE	= PLANE	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	d	d	d	d	d	d	X
557		SURFACE_NORMAL_VECTOR	= (0.0, 0.0, -1.0)	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	d	d	d	d	d	d	X
558		SURFACE_GROUND_LOCATION	= (0.0, 0.0, 0.0)	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	d	d	d	d	d	d	X
559		REFERENCE_COORD_SYSTEM_NAME	= "LOCAL_LEVEL_FRAME"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	d	d	d	d	d	d	X
560		REFERENCE_COORD_SYSTEM_INDEX	= (1, 2, 0)	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	d	d	d	d	d	d	X
561	{MSL:}	REFERENCE_COORD_SYSTEM_SOLN_ID	= "mipl_rgd_sol2nav_5"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	r	r	r	r	r	r	X
562		END_GROUP	= SURFACE_MODEL_PARMS	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	•	•	•	•	•	•	X
563		/* IMAGE HEADER DATA ELEMENTS */		X	o	X	X	X	X	X	o	o	o	o	o	X	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X
564		OBJECT	= IMAGE_HEADER	X	o	X	X	X	X	X	o	o	o	o	o	X	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X
565		HEADER_TYPE	= VICAR	X	o	X	X	X	X	X	o	o	o	o	o	X	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X
566		INTERCHANGE_FORMAT	= ASCII	X	o	X	X	X	X	X	o	o	o	o	o	X	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X
567		BYTES	= 12288	X	o	X	X	X	X	X	o	o	o	o	o	X	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X
568		DESCRIPTION	= "VICAR2.TXT"	X	o	X	X	X	X	X	o	o	o	o	o	X	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X
569		END_OBJECT	= IMAGE_HEADER	X	o	X	X	X	X	X	o	o	o	o	o	X	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X
570		/* ODL HEADER DATA ELEMENTS */		X	o	X	X	X	X	X	o	o	o	o	o	X	X	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X
571		OBJECT	= ODL_HEADER	X	o	X	X	X	X	X	o	o	o	o	o	X	X	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X
572		HEADER_TYPE	= ODL	X	o	X	X	X	X	X	o	o	o	o	o	X	X	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X
573		INTERCHANGE_FORMAT	= ASCII	X	o	X	X	X	X	X	o	o	o	o	o	X	X	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X
574		BYTES	= 12288	X	o	X	X	X	X	X	o	o	o	o	o	X	X	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X
575		DESCRIPTION	= "ODL.TXT"	X	o	X	X	X	X	X	o	o	o	o	o	X	X	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X
576		END_OBJECT	= ODL_HEADER	X	o	X	X	X	X	X	o	o	o	o	o	X	X	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X
577		/* IMAGE DATA ELEMENTS */		•	o	X	X	X	X	X	o	o	o	o	o	X	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X
578		OBJECT	= IMAGE	•	o	X	X	X	X	X	o	o	o	o	o	X	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X
579		INTERCHANGE_FORMAT	= BINARY	•	o	X	X	X	X	X	o	o	o	o	o	X	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X
580		LINES	= 1024	d	o	X	X	X	X	X	o	o	o	o	o	X	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X
581		LINE_SAMPLES	= 1024	d	o	X	X	X	X	X	o	o	o	o	o	X	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X
582		SAMPLE_TYPE	= MSB_INTEGER	d	o	X	X	X	X	X	o	o	o	o	o	X	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X
583		SAMPLE_BITS	= 16	d	o	X	X	X	X	X	o	o	o	o	o	X	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X
584		SAMPLE_BIT_MASK	= 2#0000111111111111#	d	•	X	X	X	X	X	•	•	•	•	•	X	X	•	X	•	•	X	X	X	X	X	X	•	•	•	•	•	•	X
585		BANDS	= 1	d	o	X	X	X	X	X	o	o	o	o	o	X	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X
586		BAND_STORAGE_TYPE	= BAND_SEQUENTIAL	d	o	X	X	X	X	X	o	o	o	o	o	X	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X
587		FIRST_LINE	= 3	d	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	X	•	•	•	•	•	•	X
588		FIRST_LINE_SAMPLE	= 1	d	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	•	•	•	•	•	X	•	•	•	•	•	•	X

	PDS Label Keyword	Example Value/Comment	UDR	EDR										RDR																			
				Ccam						Eng. Cams & MMM				Ccam				Eng. Cameras & MMM															
			MMM Group	RMI Group	LIBS Group A	LIBS Group B	SOH Group	Sun Safe Group	Util Test Group	Eng. Camera Grp	MMM Group A	MMM Group B	MMM Group C	RMI Group	LIBS Group A	LIBS Group B	AEGIS	Inverse LUT	Rad Correction	Geom Correction	Disparity	Range	XYZ Group	Slope	Arm Reach Group	TC RDR Grp	Mosaic (CYL)	Mosaic (PER)	Mosaic (CYP)	Mosaic (POL)	Mosaic (VRT)	Mosaic (ORT) Grp	Ops Labelless Grp
589	MEAN	= 755.973	X	o	X	X	X	X	X	o	o	o	o	o	X	X	X	o	r	r	r	r	r	r	r	X	r	r	r	r	r	r	X
590	MEDIAN	= 99.0	X	o	X	X	X	X	X	o	o	o	o	o	X	X	X	o	r	r	r	r	r	r	r	X	r	r	r	r	r	r	X
591	MAXIMUM	= 3265.0	X	o	X	X	X	X	X	o	o	o	o	o	X	X	X	o	r	r	r	r	r	r	r	X	r	r	r	r	r	r	X
592	MINIMUM	= 13.0	X	o	X	X	X	X	X	o	o	o	o	o	X	X	X	o	r	r	r	r	r	r	r	X	r	r	r	r	r	r	X
593	STANDARD_DEVIATION	= 1012.48	X	o	X	X	X	X	X	o	o	o	o	o	X	X	X	o	r	r	r	r	r	r	r	X	r	r	r	r	r	r	X
594	INVALID_CONSTANT	= 0.0	d	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	d	d	d	d	•	X	•	•	•	•	•	•	X
595	MISSING_CONSTANT	= 0.0	d	•	X	X	X	X	X	•	•	•	•	•	X	X	•	•	•	•	d	d	d	d	•	X	•	•	•	•	•	•	X
596	END_OBJECT	= IMAGE	•	o	X	X	X	X	X	o	o	o	o	o	X	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X
597	/* MOSAIC ANCILLARY */		X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
598	OBJECT	= FILE	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
599	RECORD_TYPE	= UNDEFINED	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
600	RECORD_BYTES	= "N/A"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
601	FILE_RECORD	= "N/A"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
602	LABEL_RECORDS	= "N/A"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
603	DATA_SET_ID	= "MSL-M-NAVCAM-5-RDR-MOSAIC-V1.0"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
604	DATA_SET_NAME	= "MSL MARS NAVIGATION CAMERA 5 RDR MOSAIC V1.0"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
605	INSTRUMENT_HOST_NAME	= "MARS SCIENCE LABORATORY"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
606	INSTRUMENT_NAME	= "NAVIGATION CAMERA LEFT STRING A"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
607	MISSION_PHASE_NAME	= "PRIMARY SURFACE MISSION"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
608	PLANET_DAY_NUMBER	= 24	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
609	PRODUCT_ID	= "N_L111_0121_ILT013CYLASB0111_DRIVEM1"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
610	RELEASE_ID	= "0001"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
611	TARGET_NAME	= "MARS"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
612	START_TIME	= 2012-08-27T18:48:09.058Z	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
613	STOP_TIME	= 2012-08-27T18:50:49.233Z	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
614	SPACECRAFT_CLOCK_START_COUNT	= "399365141.167"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
615	SPACECRAFT_CLOCK_STOP_COUNT	= "399365301.011"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
616	PRODUCT_CREATION_TIME	= 2012-08-27T22:16:50.000Z	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
617	FILE_NAME	= "N_L111_0121_ILT013CYLASB0111_DRIVEM1.LIS"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
618	^DESCRIPTION	= "MSL_CAMERA_SIS.PDF"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
619	END_OBJECT	= FILE	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
620	OBJECT	= FILE	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
621	RECORD_TYPE	= UNDEFINED	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
622	RECORD_BYTES	= "N/A"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
623	FILE_RECORD	= "N/A"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
624	LABEL_RECORDS	= "N/A"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
625	DATA_SET_ID	= "MSL-M-NAVCAM-5-RDR-MOSAIC-V1.0"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
626	DATA_SET_NAME	= "MSL MARS NAVIGATION CAMERA 5 RDR MOSAIC V1.0"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
627	INSTRUMENT_HOST_NAME	= "MARS SCIENCE LABORATORY"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p

	PDS Label / Keyword	Example Value/Comment	UDR	EDR										RDR																			
				Ccam						Eng. Cams & MMM				Ccam				Eng. Cameras & MMM															
			MMM Group	RMI Group	LIBS Group A	LIBS Group B	SOH Group	Sun Safe Group	Util Test Group	Eng. Camera Grp	MMM Group A	MMM Group B	MMM Group C	RMI Group	LIBS Group A	LIBS Group B	AEGIS	Inverse LUT	Rad Correction	Geom Correction	Disparity	Range	XYZ Group	Slope	Arm Reach Group	TC RDR Grp	Mosaic (CYL)	Mosaic (PER)	Mosaic (CYP)	Mosaic (POL)	Mosaic (VRT)	Mosaic (ORT) Grp	Ops Labelless Grp
628	INSTRUMENT_NAME	= "NAVIGATION CAMERA LEFT STRING A"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
629	MISSION_PHASE_NAME	= "PRIMARY SURFACE MISSION"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
630	PLANET_DAY_NUMBER	= 24	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
631	PRODUCT_ID	= "N_L111_0121_ILT013CYLASB0111_DRIVEM1"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
632	RELEASE_ID	= "0001"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
633	TARGET_NAME	= "MARS"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
634	START_TIME	= 2012-08-27T18:48:09.058Z	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
635	STOP_TIME	= 2012-08-27T18:50:49.233Z	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
636	SPACECRAFT_CLOCK_START_COUNT	= "399365141.167"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
637	SPACECRAFT_CLOCK_STOP_COUNT	= "399365301.011"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
638	PRODUCT_CREATION_TIME	= 2012-08-27T22:16:50.000Z	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
639	FILE_NAME	= "N_L111_0121_ILT013CYLASB0111_DRIVEM1.NAV"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
640	^DESCRIPTION	= "MSL_CAMERA_SIS.PDF"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
641	END_OBJECT	= FILE	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
642	/* TERRAIN PER-XYZ */		X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
643	OBJECT	= FILE	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
644	RECORD_TYPE	= UNDEFINED	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
645	RECORD_BYTES	= "N/A"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
646	FILE_RECORD	= "N/A"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
647	LABEL_RECORDS	= "N/A"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
648	DATA_SET_ID	= "MSL-M-NAVCAM-5-RDR-MESH-V1.0"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
649	DATA_SET_NAME	= "MSL MARS NAVCAM CAMERA 5 RDR TERRAIN MESH V1.0"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
650	INSTRUMENT_HOST_NAME	= "MARS SCIENCE LABORATORY"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
651	INSTRUMENT_NAME	= "NAVIGATION CAMERA LEFT STRING A"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
652	MISSION_PHASE_NAME	= "PRIMARY SURFACE MISSION"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
653	PLANET_DAY_NUMBER	= 24	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
654	PRODUCT_ID	= "NLA_401151138RASLF0141916NCAM11421M1"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
655	RELEASE_ID	= "0001"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
656	TARGET_NAME	= "MARS"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
657	START_TIME	= 2012-08-27T18:48:09.058Z	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
658	STOP_TIME	= 2012-08-27T18:50:49.233Z	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
659	SPACECRAFT_CLOCK_START_COUNT	= "399365141.167"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
660	SPACECRAFT_CLOCK_STOP_COUNT	= "399365301.011"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
661	PRODUCT_CREATION_TIME	= 2012-08-27T22:16:50.000Z	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
662	FILE_NAME	= "NLA_401151138RASLF0141916NCAM11421M1.iv"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
663	^DESCRIPTION	= "MSL_CAMERA_SIS.PDF"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
664	END_OBJECT	= FILE	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
665	OBJECT	= FILE	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
666	RECORD_TYPE	= UNDEFINED	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p

	PDS Label	Keyword	Example Value/Comment	UDR	EDR										RDR																			
					Ccam						Eng. Cams & MMM				Ccam				Eng. Cameras & MMM															
				MMM Group	RMI Group	LIBS Group A	LIBS Group B	SOH Group	Sun Safe Group	Util Test Group	Eng. Camera Grp	MMM Group A	MMM Group B	MMM Group C	RMI Group	LIBS Group A	LIBS Group B	AEGIS	Inverse LUT	Rad Correction	Geom Correction	Disparity	Range	XYZ Group	Slope	Arm Reach Group	TC RDR Grp	Mosaic (CYL)	Mosaic (PER)	Mosaic (CYP)	Mosaic (POL)	Mosaic (VRT)	Mosaic (ORT) Grp	Ops Labelless Grp
667		RECORD_BYTES	= "N/A"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
668		FILE_RECORD	= "N/A"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
669		LABEL_RECORDS	= "N/A"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
670		DATA_SET_ID	= "MSL-M-NAVCAM-5-RDR-MESH-V1.0"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
671		DATA_SET_NAME	= "MSL MARS NAVCAM CAMERA 5 RDR TERRAIN MESH V1.0"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
672		INSTRUMENT_HOST_NAME	= "MARS SCIENCE LABORATORY"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
673		INSTRUMENT_NAME	= "NAVIGATION CAMERA LEFT STRING A"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
674		MISSION_PHASE_NAME	= "PRIMARY SURFACE MISSION"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
675		PLANET_DAY_NUMBER	= 24	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
676		PRODUCT_ID	= "NLA_401151138RASLF0141916NCAM11421M1"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
677		RELEASE_ID	= "0001"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
678		TARGET_NAME	= "MARS"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
679		START_TIME	= 2012-08-27T18:48:09.058Z	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
680		STOP_TIME	= 2012-08-27T18:50:49.233Z	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
681		SPACECRAFT_CLOCK_START_COUNT	= "399365141.167"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
682		SPACECRAFT_CLOCK_STOP_COUNT	= "399365301.011"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
683		PRODUCT_CREATION_TIME	= 2012-08-27T22:16:50.000Z	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
684		FILE_NAME	= "NLA_401151138RASLF0141916NCAM11421M1.ht"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
685		^DESCRIPTION	= "MSL_CAMERA_SIS.PDF"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
686		END_OBJECT	= FILE	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
687		OBJECT	= FILE	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
688		RECORD_TYPE	= UNDEFINED	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
689		RECORD_BYTES	= "N/A"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
690		FILE_RECORD	= "N/A"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
691		LABEL_RECORDS	= "N/A"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
692		DATA_SET_ID	= "MSL-M-NAVCAM-5-RDR-MESH-V1.0"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
693		DATA_SET_NAME	= "MSL MARS NAVCAM CAMERA 5 RDR TERRAIN MESH V1.0"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
694		INSTRUMENT_HOST_NAME	= "MARS SCIENCE LABORATORY"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
695		INSTRUMENT_NAME	= "NAVIGATION CAMERA LEFT STRING A"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
696		MISSION_PHASE_NAME	= "PRIMARY SURFACE MISSION"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
697		PLANET_DAY_NUMBER	= 24	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
698		PRODUCT_ID	= "NLA_401151138RASLF0141916NCAM11421M1"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
699		RELEASE_ID	= "0001"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
700		TARGET_NAME	= "MARS"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
701		START_TIME	= 2012-08-27T18:48:09.058Z	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
702		STOP_TIME	= 2012-08-27T18:50:49.233Z	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
703		SPACECRAFT_CLOCK_START_COUNT	= "399365141.167"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
704		SPACECRAFT_CLOCK_STOP_COUNT	= "399365301.011"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
705		PRODUCT_CREATION_TIME	= 2012-08-27T22:16:50.000Z	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
706		FILE_NAME	= "NLA_401151138RASLF0141916NCAM11421M1.rgb"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p

	PDS Label	Keyword	Example Value/Comment	UDR	EDR										RDR																			
					Ccam						Eng. Cams & MMM				Ccam				Eng. Cameras & MMM															
				MMM Group	RMI Group	LIBS Group A	LIBS Group B	SOH Group	Sun Safe Group	Util Test Group	Eng. Camera Grp	MMM Group A	MMM Group B	MMM Group C	RMI Group	LIBS Group A	LIBS Group B	AEGIS	Inverse LUT	Rad Correction	Geom Correction	Disparity	Range	XYZ Group	Slope	Arm Reach Group	TC RDR Grp	Mosaic (CYL)	Mosaic (PER)	Mosaic (CYP)	Mosaic (POL)	Mosaic (VRT)	Mosaic (ORT) Grp	Ops Labelless Grp
707		^DESCRIPTION	= "MSL_CAMERA_SIS.PDF"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
708		END_OBJECT	= FILE	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
709		/* TERRAIN UNIFIED */		X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
710		OBJECT	= FILE	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
711		RECORD_TYPE	= UNDEFINED	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
712		RECORD_BYTES	= "N/A"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
713		FILE_RECORD	= "N/A"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
714		LABEL_RECORDS	= "N/A"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
715		DATA_SET_ID	= "MSL-M-NAVCAM-5-RDR-MESH-V1.0"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
716		DATA_SET_NAME	= "MSL MARS NAVCAM CAMERA 5 RDR TERRAIN MESH V1.0"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
717		INSTRUMENT_HOST_NAME	= "MARS SCIENCE LABORATORY"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
718		INSTRUMENT_NAME	= "NAVIGATION CAMERA LEFT STRING A"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
719		MISSION_PHASE_NAME	= "PRIMARY SURFACE MISSION"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
720		PLANET_DAY_NUMBER	= 24	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
721		PRODUCT_ID	= "FNR_L0146XRASLF_014_2111_AUTOGENM1"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
722		RELEASE_ID	= "0001"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
723		TARGET_NAME	= "MARS"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
724		START_TIME	= 2012-08-27T18:48:09.058Z	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
725		STOP_TIME	= 2012-08-27T18:50:49.233Z	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
726		SPACECRAFT_CLOCK_START_COUNT	= "399365141.167"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
727		SPACECRAFT_CLOCK_STOP_COUNT	= "399365301.011"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
728		PRODUCT_CREATION_TIME	= 2012-08-27T22:16:50.000Z	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
729		FILE_NAME	= "FNR_L0146XRASLF_014_2111_AUTOGENM1.iv"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
730		^DESCRIPTION	= "MSL_CAMERA_SIS.PDF"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
731		END_OBJECT	= FILE	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
732		OBJECT	= FILE	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
733		RECORD_TYPE	= UNDEFINED	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
734		RECORD_BYTES	= "N/A"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
735		FILE_RECORD	= "N/A"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
736		LABEL_RECORDS	= "N/A"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
737		DATA_SET_ID	= "MSL-M-NAVCAM-5-RDR-MESH-V1.0"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
738		DATA_SET_NAME	= "MSL MARS NAVCAM CAMERA 5 RDR TERRAIN MESH V1.0"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
739		INSTRUMENT_HOST_NAME	= "MARS SCIENCE LABORATORY"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
740		INSTRUMENT_NAME	= "NAVIGATION CAMERA LEFT STRING A"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
741		MISSION_PHASE_NAME	= "PRIMARY SURFACE MISSION"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
742		PLANET_DAY_NUMBER	= 24	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
743		PRODUCT_ID	= "FNR_L0146XRASLF_014_2111_AUTOGENM1"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
744		RELEASE_ID	= "0001"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
745		TARGET_NAME	= "MARS"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p

			UDR	EDR										RDR																		
				Ccam						Eng. Cams & MMM				Ccam				Eng. Cameras & MMM														
PDS Label	Keyword	Example Value/Comment	MMM Group	RMI Group	LIBS Group A	LIBS Group B	SOH Group	Sun Safe Group	Util Test Group	Eng. Camera Grp	MMM Group A	MMM Group B	MMM Group C	RMI Group	LIBS Group A	LIBS Group B	AEGIS	Inverse LUT	Rad Correction	Geom Correction	Disparity	Range	XYZ Group	Slope	Arm Reach Group	TC RDR Grp	Mosaic (CYL)	Mosaic (PER)	Mosaic (CYP)	Mosaic (POL)	Mosaic (VRT)	Mosaic (ORT) Grp	Ops Labelless Grp
746	START_TIME	= 2012-08-27T18:48:09.058Z	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
747	STOP_TIME	= 2012-08-27T18:50:49.233Z	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
748	SPACECRAFT_CLOCK_START_COUNT	= "399365141.167"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
749	SPACECRAFT_CLOCK_STOP_COUNT	= "399365301.011"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
750	PRODUCT_CREATION_TIME	= 2012-08-27T22:16:50.000Z	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
751	FILE_NAME	= "FNR_L0146XRASLF_014_2111_AUTOGENM1.mod"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
752	^DESCRIPTION	= "MSL_CAMERA_SIS.PDF"	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
753	END_OBJECT	= FILE	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	p
754	/* CHEMCAM RMI TABLE ELEMENTS */		X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
755	OBJECT	= ANCILLARY_TABLE	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
756	^PDS_OBJECT	= 1	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
757	BYTES	= 138	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
758	INTERCHANGE_FORMAT	= BINARY	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
759	ROWS	= 1	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
760	COLUMNS	= 45	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
761	ROW_BYTES	= 138	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
762	DESCRIPTION	= "Ancillary DPO for MSL ChemCam"	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
763	^STRUCTURE	= "ANCILLARY.FMT"	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
764	END_OBJECT	= ANCILLARY_TABLE	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
765	OBJECT	= SOH_BEFORE_TABLE	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
766	^PDS_OBJECT	= 2	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
767	BYTES	= 172	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
768	INTERCHANGE_FORMAT	= BINARY	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
769	ROWS	= 1	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
770	COLUMNS	= 31	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
771	ROW_BYTES	= 172	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
772	DESCRIPTION	= "SOH Sci Data Before DPO for MSL ChemCam"	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
773	^STRUCTURE	= "SOH_BEFORE.FMT"	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
774	END_OBJECT	= SOH_BEFORE_TABLE	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
775	OBJECT	= SOH_AFTER_TABLE	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
776	^PDS_OBJECT	= 3	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
777	BYTES	= 172	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
778	INTERCHANGE_FORMAT	= BINARY	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
779	ROWS	= 1	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
780	COLUMNS	= 31	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
781	ROW_BYTES	= 372	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
782	DESCRIPTION	= "SOH Sci Data After DPO for MSL ChemCam"	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
783	^STRUCTURE	= "SOH_AFTER.FMT"	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
784	END_OBJECT	= SOH_AFTER_TABLE	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

	PDS Label Keyword	Example Value/Comment	UDR	EDR Ccam						EDR Eng. Cams & MMM				RDR Ccam				RDR Eng. Cameras & MMM															
			MMM Group	RMI Group	LIBS Group A	LIBS Group B	SOH Group	Sun Safe Group	Util Test Group	Eng. Camera Grp	MMM Group A	MMM Group B	MMM Group C	RMI Group	LIBS Group A	LIBS Group B	AEGIS	Inverse LUT	Rad Correction	Geom Correction	Disparity	Range	XYZ Group	Slope	Arm Reach Group	TC RDR Grp	Mosaic (CYL)	Mosaic (PER)	Mosaic (CYP)	Mosaic (POL)	Mosaic (VRT)	Mosaic (ORT) Grp	Ops Labelless Grp
785	OBJECT	= IMAGE_REPLY_TABLE	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
786	^PDS_OBJECT	= 4	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
787	BYTES	= 12	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
788	INTERCHANGE_FORMAT	= BINARY	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
789	ROWS	= 1	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
790	COLUMNS	= 2	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
791	ROW_BYTES	= 12	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
792	DESCRIPTION	= "Xmit Image Cmd Reply DPO for MSL ChemCam"	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
793	^STRUCTURE	= "IMAGE_REPLY.FMT"	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
794	END_OBJECT	= IMAGE_REPLY_TABLE	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
795	OBJECT	= IMAGE_HEADER_FOOTER_TABLE	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
796	^PDS_OBJECT	= 5	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
797	BYTES	= 36	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
798	INTERCHANGE_FORMAT	= BINARY	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
799	ROWS	= 1	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
800	COLUMNS	= 13	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
801	ROW_BYTES	= 36	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
802	DESCRIPTION	= "Image Header and Footer for MSL ChemCam"	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
803	^STRUCTURE	= "IMAGE_HEADER_FOOTER.FMT"	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
804	END_OBJECT	= IMAGE_HEADER_FOOTER_TABLE	X	o	X	X	X	X	X	X	X	X	X	o	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
805	/* CHEMCAM LIBS TABLE ELEMENTS */		X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
806	OBJECT	= CCAM_LIBS_TABLE	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
807	COLUMNS	= 381	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
808	ROWS	= 1	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
809	ROW_BYTES	= 387866	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
810	INTERCHANGE_FORMAT	= BINARY	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
811	DESCRIPTION	= "ChemCam LIBS"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
812	OBJECT	= CONTAINER	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
813	NAME	= "CCAM_LIBS_ANCILLARY"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
814	^STRUCTURE	= "CCAM_LIBS_ANCILLARY.FMT"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
815	START_BYTE	= 1	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
816	BYTES	= 150	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
817	REPETITIONS	= 1	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
818	DESCRIPTION	= "ChemCam LIBS Ancillary Data"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
819	END_OBJECT	= CONTAINER	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
820	OBJECT	= COLUMN	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
821	NAME	= "CCAM_REPLY_BYTESIZE"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
822	^STRUCTURE	= MSB_UNSIGNED_INTEGER	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

PDS Label	Keyword	Example Value/Comment	UDR	EDR										RDR																			
				Ccam						Eng. Cams & MMM				Ccam				Eng. Cameras & MMM															
			MMM Group	RMI Group	LIBS Group A	LIBS Group B	SOH Group	Sun Safe Group	Util Test Group	Eng. Camera Grp	MMM Group A	MMM Group B	MMM Group C	RMI Group	LIBS Group A	LIBS Group B	AEGIS	Inverse LUT	Rad Correction	Geom Correction	Disparity	Range	XYZ Group	Slope	Arm Reach Group	TC RDR Grp	Mosaic (CYL)	Mosaic (PER)	Mosaic (CYP)	Mosaic (POL)	Mosaic (VRT)	Mosaic (ORT) Grp	Ops Labelless Grp
823	START_BYTE	= 151	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
824	BYTES	= 4	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
825	REPETITIONS	= 1	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
826	DESCRIPTION	= "Size of CMD_REPLY packet"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
827	END_OBJECT	= COLUMN	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
828	OBJECT	= CONTAINER	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
829	NAME	= "CCAM_CMD_REPLY"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
830	^STRUCTURE	= "CCAM_CMD_REPLY.FMT"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
831	START_BYTE	= 155	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
832	BYTES	= 4	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
833	REPETITIONS	= 1	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
834	DESCRIPTION	= "ChemCam CMD_REPLY frame"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
835	END_OBJECT	= CONTAINER	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
836	OBJECT	= COLUMN	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
837	NAME	= "CCAM_REPLY_CHECKSUM"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
838	^STRUCTURE	= MSB_UNSIGNED_INTEGER	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
839	START_BYTE	= 159	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
840	BYTES	= 4	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
841	REPETITIONS	= 1	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
842	DESCRIPTION	= "Second part of CMD_REPLY data: MD5 checksum"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
843	END_OBJECT	= COLUMN	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
844	OBJECT	= CONTAINER	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
845	NAME	= "CCAM_SOH_SCIDATA_COLS"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
846	^STRUCTURE	= "CCAM_SOH_SCIDATA_COLS.FMT"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
847	START_BYTE	= 163	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
848	BYTES	= 68	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
849	REPETITIONS	= 1	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
850	DESCRIPTION	= "ChemCam SOH header"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
851	END_OBJECT	= CONTAINER	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
852	OBJECT	= CONTAINER	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
853	NAME	= "CCAM_SOH_TO_RCE_CONTAINER"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
854	^STRUCTURE	= "CCAM_SOH_TO_RCE_CONTAINER.FMT"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
855	START_BYTE	= 231	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
856	BYTES	= 100	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
857	REPETITIONS	= 1	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
858	DESCRIPTION	= "ChemCam LIBS to RCE structure. Includes arrays ccam_DPU_LIBS and ccam_MU_LIBS_struct"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
859	END_OBJECT	= CONTAINER	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

PDS Label	Keyword	Example Value/Comment	UDR	EDR										RDR																			
				Ccam						Eng. Cams & MMM				Ccam				Eng. Cameras & MMM															
			MMM Group	RMI Group	LIBS Group A	LIBS Group B	SOH Group	Sun Safe Group	Util Test Group	Eng. Camera Grp	MMM Group A	MMM Group B	MMM Group C	RMI Group	LIBS Group A	LIBS Group B	AEGIS	Inverse LUT	Rad Correction	Geom Correction	Disparity	Range	XYZ Group	Slope	Arm Reach Group	TC RDR Grp	Mosaic (CYL)	Mosaic (PER)	Mosaic (CYP)	Mosaic (POL)	Mosaic (VRT)	Mosaic (ORT) Grp	Ops Labelless Grp
860	OBJECT	= COLUMN	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
861	NAME	= "CCAM_SOH_BEFORE_CHECKSUM"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
862	DATA_BYTE	= MSB_UNSIGNED_INTEGER	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
863	START_BYTE	= 331	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
864	BYTES	= 4	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
865	DESCRIPTION	= "MD5 checksum"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
866	END_OBJECT	= COLUMN	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
867	OBJECT	= COLUMN	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
868	NAME	= "CMD_REPLY_BYTESIZE"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
869	DATA_BYTE	= MSB_UNSIGNED_INTEGER	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
870	START_BYTE	= 335	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
871	BYTES	= 4	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
872	DESCRIPTION	= "Size of CMD_REPLY packet"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
873	END_OBJECT	= COLUMN	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
874	OBJECT	= CONTAINER	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
875	NAME	= "CCAM_CMD_REPLY"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
876	^STRUCTURE	= "CCAM_CMD_REPLY.FMT"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
877	START_BYTE	= 339	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
878	BYTES	= 4	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
879	REPETITIONS	= 1	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
880	DESCRIPTION	= "ChemCam CMD_REPLY frame"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
881	END_OBJECT	= CONTAINER	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
882	OBJECT	= COLUMN	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
883	NAME	= "CMD_REPLY_CHECKSUM"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
884	DATA_BYTE	= MSB_UNSIGNED_INTEGER	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
885	START_BYTE	= 343	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
886	BYTES	= 4	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
887	DESCRIPTION	= "Second part of CMD_REPLY data: MD5 checksum"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
888	END_OBJECT	= COLUMN	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
889	OBJECT	= CONTAINER	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
890	NAME	= "CCAM_SOH_SCIDATA_COLS"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
891	^STRUCTURE	= "CCAM_SOH_SCIDATA_COLS.FMT"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
892	START_BYTE	= 347	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
893	BYTES	= 268	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
894	REPETITIONS	= 1	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
895	DESCRIPTION	= "ChemCam SOH header"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
896	END_OBJECT	= CONTAINER	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
897	OBJECT	= CONTAINER	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

	PDS Label	Keyword	Example Value/Comment	UDR	EDR Ccam						EDR Eng. Cams & MMM				RDR Ccam				RDR Eng. Cameras & MMM															
				MMM Group	RMI Group	LIBS Group A	LIBS Group B	SOH Group	Sun Safe Group	Util Test Group	Eng. Camera Grp	MMM Group A	MMM Group B	MMM Group C	RMI Group	LIBS Group A	LIBS Group B	AEGIS	Inverse LUT	Rad Correction	Geom Correction	Disparity	Range	XYZ Group	Slope	Arm Reach Group	TC RDR Grp	Mosaic (CYL)	Mosaic (PER)	Mosaic (CYP)	Mosaic (POL)	Mosaic (VRT)	Mosaic (ORT) Grp	Ops Labelless Grp
898		NAME	= "CCAM_SOH_TO_RCE_CONTAINER"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
899		^STRUCTURE	= "CCAM_SOH_TO_RCE_CONTAINER.FMT"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
900		START_BYTE	= 415	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
901		BYTES	= 100	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
902		REPETITIONS	= 3	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
903		DESCRIPTION	= "ChemCam LIBS to RCE structure. Includes arrays ccam_DPU_LIBS and ccam_MU_LIBS_struct"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
904		END_OBJECT	= CONTAINER	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
905		OBJECT	= COLUMN	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
906		NAME	= "CCAM_SOH_CHECKSUM"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
907		DATA_BYTE	= MSB_UNSIGNED_INTEGER	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
908		START_BYTE	= 715	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
909		BYTES	= 4	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
910		DESCRIPTION	= "MD5 checksum"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
911		END_OBJECT	= COLUMN	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
912		OBJECT	= COLUMN	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
913		NAME	= "CMD_REPLY_BYTESIZE"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
914		DATA_BYTE	= MSB_UNSIGNED_INTEGER	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
915		START_BYTE	= 719	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
916		BYTES	= 4	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
917		DESCRIPTION	= "Size of CMD_REPLY packet"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
918		END_OBJECT	= COLUMN	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
919		OBJECT	= CONTAINER	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
920		NAME	= "CCAM_CMD_REPLY"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
921		^STRUCTURE	= "CCAM_CMD_REPLY.FMT"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
922		START_BYTE	= 723	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
923		BYTES	= 4	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
924		REPETITIONS	= 1	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
925		DESCRIPTION	= "ChemCam CMD_REPLY frame"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
926		END_OBJECT	= CONTAINER	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
927		OBJECT	= COLUMN	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
928		NAME	= "CMD_REPLY_CHECKSUM"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
929		DATA_BYTE	= MSB_UNSIGNED_INTEGER	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
930		START_BYTE	= 727	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
931		BYTES	= 4	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
932		DESCRIPTION	= "Second part of CMD_REPLY data: MD5 checksum"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
933		END_OBJECT	= COLUMN	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
934		OBJECT	= CONTAINER	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

	PDS Label	Keyword	Example Value/Comment	UDR	EDR										RDR																			
					Ccam						Eng. Cams & MMM				Ccam				Eng. Cameras & MMM															
				MMM Group	RMI Group	LIBS Group A	LIBS Group B	SOH Group	Sun Safe Group	Util Test Group	Eng. Camera Grp	MMM Group A	MMM Group B	MMM Group C	RMI Group	LIBS Group A	LIBS Group B	AEGIS	Inverse LUT	Rad Correction	Geom Correction	Disparity	Range	XYZ Group	Slope	Arm Reach Group	TC RDR Grp	Mosaic (CYL)	Mosaic (PER)	Mosaic (CYP)	Mosaic (POL)	Mosaic (VRT)	Mosaic (ORT) Grp	Ops Labelless Grp
935		NAME	= "CCAM_LASER_HEADER_COLS"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
936		^STRUCTURE	= "CCAM_LASER_HEADER_COLS.FMT"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
937		START_BYTE	= 731	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
938		BYTES	= 24	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
939		REPETITIONS	= 1	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
940		DESCRIPTION	= "ChemCam Laser data header"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
941		END_OBJECT	= CONTAINER	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
942		OBJECT	= CONTAINER	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
943		NAME	= "CCAM_LASER_SCIDATA_COLS"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
944		^STRUCTURE	= "CCAM_LASER_SCIDATA_COLS.FMT"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
945		START_BYTE	= 755	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
946		BYTES	= 420	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
947		REPETITIONS	= 30	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
948		DESCRIPTION	= "ChemCam Laser data"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
949		END_OBJECT	= CONTAINER	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
950		OBJECT	= COLUMN	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
951		NAME	= "CCAM_LASER_CHECKSUM"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
952		DATA_BYTE	= MSB_UNSIGNED_INTEGER	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
953		START_BYTE	= 1175	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
954		BYTES	= 4	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
955		DESCRIPTION	= "MD5 checksum"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
956		END_OBJECT	= COLUMN	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
957		OBJECT	= COLUMN	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
958		NAME	= "CMD_REPLY_BYTESIZE"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
959		DATA_BYTE	= MSB_UNSIGNED_INTEGER	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
960		START_BYTE	= 1179	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
961		BYTES	= 4	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
962		DESCRIPTION	= "Size of CMD_REPLY packet"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
963		END_OBJECT	= COLUMN	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
964		OBJECT	= CONTAINER	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
965		NAME	= "CCAM_CMD_REPLY"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
966		^STRUCTURE	= "CCAM_CMD_REPLY.FMT"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
967		START_BYTE	= 1183	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
968		BYTES	= 4	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
969		REPETITIONS	= 1	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
970		DESCRIPTION	= "ChemCam CMD_REPLY frame"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
971		END_OBJECT	= CONTAINER	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
972		OBJECT	= COLUMN	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
973		NAME	= "CMD_REPLY_CHECKSUM"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

PDS Label	Keyword	Example Value/Comment	UDR	EDR										RDR																				
				Ccam						Eng. Cams & MMM				Ccam				Eng. Cameras & MMM																
			MMM Group	RMI Group	LIBS Group A	LIBS Group B	SOH Group	Sun Safe Group	Util Test Group	Eng. Camera Grp	MMM Group A	MMM Group B	MMM Group C	RMI Group	LIBS Group A	LIBS Group B	AEGIS	Inverse LUT	Rad Correction	Geom Correction	Disparity	Range	XYZ Group	Slope	Arm Reach Group	TC RDR Grp	Mosaic (CYL)	Mosaic (PER)	Mosaic (CYP)	Mosaic (POL)	Mosaic (VRT)	Mosaic (ORT) Grp	Ops Labelless Grp
974	DATA_BYTE	= MSB_UNSIGNED_INTEGER	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
975	START_BYTE	= 1187	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
976	BYTES	= 4	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
977	DESCRIPTION	= "Second part of CMD_REPLY data: MD5 checksum"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
978	END_OBJECT	= COLUMN	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
979	OBJECT	= CONTAINER	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
980	NAME	= "CCAM_LIBS_HEADER"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
981	^STRUCTURE	= "CCAM_LIBS_HEADER.FMT"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
982	START_BYTE	= 1191	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
983	BYTES	= 33	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
984	REPETITIONS	= 1	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
985	DESCRIPTION	= "ChemCam LIBS header"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
986	END_OBJECT	= CONTAINER	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
987	OBJECT	= CONTAINER	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
988	NAME	= "CCAM_LIBS_DATA_CONTAINER"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
989	^STRUCTURE	= "CCAM_LIBS_DATA_CONTAINER.FMT"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
990	START_BYTE	= 1224	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
991	BYTES	= 386639	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
992	REPETITIONS	= 30	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
993	DESCRIPTION	= "ChemCam LIBS data"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
994	END_OBJECT	= CONTAINER	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
995	OBJECT	= COLUMN	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
996	NAME	= "CCAM_LIBS_CHECKSUM"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
997	DATA_BYTE	= MSB_UNSIGNED_INTEGER	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
998	START_BYTE	= 387863	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
999	BYTES	= 4	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1000	DESCRIPTION	= "MD5 checksum"	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1001	END_OBJECT	= COLUMN	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1002	END_OBJECT	CCAM_LIBS_TABLE	X	X	o	o	X	X	X	X	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1003	/* CHEMCAM SOH TABLE ELEMENTS */		X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1004	OBJECT	= CCAM_SHO_DPO_TABLE	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1005	COLUMNS	= 80	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1006	ROWS	= 1	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1007	ROW_BYTES	= 240	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1008	INTERCHANGE_FORMAT	= BINARY	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1009	DESCRIPTION	= "ChemCam State of Health Structure"	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1010	OBJECT	= CONTAINER	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X

PDS Label	Keyword	Example Value/Comment	UDR MMM Group	EDR Ccam RMI Group	EDR Ccam LIBS Group A	EDR Ccam LIBS Group B	EDR Ccam SOH Group	EDR Ccam Sun Safe Group	EDR Ccam Util Test Group	EDR Eng. Cams & MMM Eng. Camera Grp	EDR Eng. Cams & MMM MMM Group A	EDR Eng. Cams & MMM MMM Group B	EDR Eng. Cams & MMM MMM Group C	RDR Ccam RMI Group	RDR Ccam LIBS Group A	RDR Ccam LIBS Group B	RDR Ccam AEGIS	RDR Eng. Cameras & MMM Inverse LUT	Rad Correction	Geom Correction	Disparity	Range	XYZ Group	Slope	Arm Reach Group	TC RDR Grp	Mosaic (CYL)	Mosaic (PER)	Mosaic (CYP)	Mosaic (POL)	Mosaic (VRT)	Mosaic (ORT) Grp	Ops Labelless Grp
1011	NAME	= "CCAM_LIBS_ANCILLARY"	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1012	^STRUCTURE	= "CCAM_LIBS_ANCILLARY.FMT"	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1013	START_BYTE	= 1	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1014	BYTES	= 68	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1015	REPETITIONS	= 1	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1016	DESCRIPTION	= "ChemCam LIBS Ancillary Data"	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1017	END_OBJECT	= CONTAINER	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1018	OBJECT	= CONTAINER	X	X	X	X	o	o	X	X	X	X	X	X	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1019	NAME	= "CCAM_SOH_SCIDATA_COLS"	X	X	X	X	o	o	X	X	X	X	X	X	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1020	^STRUCTURE	= "CCAM_SOH_SCIDATA_COLS.FMT"	X	X	X	X	o	o	X	X	X	X	X	X	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1021	START_BYTE	= 69	X	X	X	X	o	o	X	X	X	X	X	X	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1022	BYTES	= 68	X	X	X	X	o	o	X	X	X	X	X	X	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1023	REPETITIONS	= 1	X	X	X	X	o	o	X	X	X	X	X	X	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1024	DESCRIPTION	= "ChemCam SOH header"	X	X	X	X	o	o	X	X	X	X	X	X	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1025	END_OBJECT	= CONTAINER	X	X	X	X	o	o	X	X	X	X	X	X	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1026	OBJECT	= CONTAINER	X	X	X	X	o	o	X	X	X	X	X	X	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1027	NAME	= "CCAM_SOH_TO_RCE_CONTAINER"	X	X	X	X	o	o	X	X	X	X	X	X	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1028	^STRUCTURE	= "CCAM_SOH_TO_RCE_CONTAINER.FMT"	X	X	X	X	o	o	X	X	X	X	X	X	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1029	START_BYTE	= 137	X	X	X	X	o	o	X	X	X	X	X	X	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1030	BYTES	= 100	X	X	X	X	o	o	X	X	X	X	X	X	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1031	REPETITIONS	= 1	X	X	X	X	o	o	X	X	X	X	X	X	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1032	DESCRIPTION	= "ChemCam SOH to RCE structure"	X	X	X	X	o	o	X	X	X	X	X	X	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1033	END_OBJECT	= CONTAINER	X	X	X	X	o	o	X	X	X	X	X	X	X	X	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1034	OBJECT	= CONTAINER	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1035	NAME	= "CCAM_SOH_EDR_ANCILLARY"	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1036	^STRUCTURE	= "CCAM_SOH_EDR_ANCILLARY_V4.FMT"	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1037	START_BYTE	= 5	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1038	BYTES	= 83	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1039	REPETITIONS	= 1	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1040	DESCRIPTION	= "ChemCam SOH Ancillary Data"	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1041	END_OBJECT	= CONTAINER	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1042	OBJECT	= COLUMN	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1043	NAME	= "CCAM_SOH_CHECKSUM"	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1044	DATA_BYTE	= MSB_UNSIGNED_INTEGER	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1045	START_BYTE	= 237	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1046	BYTES	= 4	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1047	DESCRIPTION	= "MD5 checksum"	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1048	END_OBJECT	= COLUMN	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1049	END_OBJECT	= CCAM_SHO_DPO_TABLE	X	X	X	X	o	o	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
1050	END		X	o	o	o	o	o	X	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	X

APPENDIX B – Example Mosaic RDR Detached PDS Label

```
PDS_VERSION_ID                   = PDS3

/* IDENTIFICATION DATA ELEMENTS THAT APPLY TO ALL REFERENCED DATA FILES */

DATA_SET_ID                      = "MSL-M-NAVCAM-5-RDR-MOSAIC-V1.0"
DATA_SET_NAME                    = "MSL MARS NAVIGATION CAMERA 5 RDR
                                    MOSAIC V1.0"
INSTRUMENT_HOST_NAME             = "MARS SCIENCE LABORATORY"
INSTRUMENT_NAME                  = "NAVIGATION CAMERA LEFT STRING B"
MISSION_PHASE_NAME               = "PRIMARY SURFACE MISSION"
PLANET_DAY_NUMBER                = 351
PRODUCT_ID                       = "N_L000_0351_EDR011CYLTSB0302_DRIVEM1"
RELEASE_ID                       = "0001"
TARGET_NAME                      = "MARS"
SPACECRAFT_CLOCK_START_COUNT     = "428653753.119"
SPACECRAFT_CLOCK_STOP_COUNT      = "428654464.090"
PRODUCT_CREATION_TIME            = 2013-08-02T19:41:34.000Z
START_TIME                       = 2013-08-01T18:36:12.827Z
STOP_TIME                        = 2013-08-01T18:48:03.804Z

OBJECT                           = FILE

/* FILE DATA ELEMENTS */

  RECORD_TYPE                    = FIXED_LENGTH
  RECORD_BYTES                   = 7696
  FILE_RECORDS                   = 984
  LABEL_RECORDS                  = 2

/* POINTERS TO DATA OBJECTS */

  ^ODL_HEADER                    = ("N_L000_0351_EDR011CYLTSB0302_DRIVEM1.IMG"
                                    ,1)
  ^IMAGE_HEADER                  = ("N_L000_0351_EDR011CYLTSB0302_DRIVEM1.IMG"
                                    ,3)
  ^IMAGE                         = ("N_L000_0351_EDR011CYLTSB0302_DRIVEM1.IMG"
                                    ,4)

/* IDENTIFICATION DATA ELEMENTS */

  SOURCE_PRODUCT_ID              = ("NLB_428654463EDR_F0110302NCAM00263M1"
                                    ,"NLB_428654432EDR_F0110302NCAM00263M1"
                                    ,"NLB_428654371EDR_F0110302NCAM00263M1"
                                    ,"NLB_428654401EDR_F0110302NCAM00263M1"
                                    ,"NLB_428654280EDR_F0110302NCAM00263M1"
                                    ,"NLB_428654249EDR_F0110302NCAM00263M1"
                                    ,"NLB_428654226EDR_F0110302NCAM00263M1"
                                    ,"NLB_428653864EDR_F0110302NCAM05114M2"
                                    ,"NLB_428653840EDR_F0110302NCAM05114M2"
                                    ,"NLB_428653815EDR_F0110302NCAM05114M2"
                                    ,"NLB_428653753EDR_F0110302NCAM05114M2"
                                    ,"NLB_428653784EDR_F0110302NCAM05114M2")
  FRAME_ID                       = "LEFT"
  INSTRUMENT_HOST_ID             = "MSL"
  INSTRUMENT_ID                  = "NAV_LEFT_B"
  INSTRUMENT_TYPE                = "IMAGING CAMERA"
  MISSION_NAME                   = "MARS SCIENCE LABORATORY"

/* DERIVED IMAGE DATA ELEMENTS */

  GROUP                          = DERIVED_IMAGE_PARMS
    MSL:INPUT_PRODUCT_ID         = ("NLB_428654463EDR_F0110302NCAM00263M1"
```

```
                                    ,"NLB_428654432EDR_F0110302NCAM00263M1"
                                    ,"NLB_428654371EDR_F0110302NCAM00263M1"
                                    ,"NLB_428654401EDR_F0110302NCAM00263M1"
                                    ,"NLB_428654280EDR_F0110302NCAM00263M1"
                                    ,"NLB_428654249EDR_F0110302NCAM00263M1"
                                    ,"NLB_428654226EDR_F0110302NCAM00263M1"
                                    ,"NLB_428653864EDR_F0110302NCAM05114M2"
                                    ,"NLB_428653840EDR_F0110302NCAM05114M2"
                                    ,"NLB_428653815EDR_F0110302NCAM05114M2"
                                    ,"NLB_428653753EDR_F0110302NCAM05114M2"
                                    ,"NLB_428653784EDR_F0110302NCAM05114M2")
    MSL:RADIANCE_OFFSET            = 0.0 <W.m**-2.sr**-1.nm**-1>
    MSL:RADIANCE_SCALING_FACTOR    = 1.0e-05 <W.m**-2.sr**-1.nm**-1>
    RADIOMETRIC_CORRECTION_TYPE    = MIPLRAD
    MSL:IMAGE_RADIANCE_FACTOR      = (1.00845,0.994785,0.982771,0.985547,
                                      0.978813,0.979495,0.9849250000000001,
                                      1.004488,1.013195,1.024389,1.017953,
                                      1.025188)
    MSL:IMAGE_RADIANCE_OFFSET      = (-5.001228,-2.605817,6.997507,-4.32027,
                                      31.696735,39.821237,41.257677,
                                      -15.218787,-18.662335,-36.227389,
                                      -9.655215999999999,-28.082114)
    MSL:BRIGHTNESS_CORRECTION_TYPE = LINEAR
  END_GROUP                        = DERIVED_IMAGE_PARMS

/* SURFACE PROJECTION DATA ELEMENTS */

  GROUP                            = SURFACE_PROJECTION_PARMS
    MAP_PROJECTION_TYPE            = CYLINDRICAL
    MAP_RESOLUTION                 = (21.3782 <pixel/degree>,
                                      21.3782 <pixel/degree>)
    MSL:MAXIMUM_ELEVATION          = 5.74066 <deg>
    MSL:MINIMUM_ELEVATION          = -40.1625 <deg>
    PROJECTION_ORIGIN_VECTOR       = (-34.5096,-77.4266,-1.11058)
    REFERENCE_COORD_SYSTEM_INDEX   = 11
    REFERENCE_COORD_SYSTEM_NAME    = "SITE_FRAME"
    START_AZIMUTH                  = 0.0 <deg>
    STOP_AZIMUTH                   = 360.0 <deg>
    ZERO_ELEVATION_LINE            = 123.725
  END_GROUP                        = SURFACE_PROJECTION_PARMS

/* SURFACE MODEL DATA ELEMENTS */

  GROUP                           = SURFACE_MODEL_PARMS
    REFERENCE_COORD_SYSTEM_INDEX  = 11
    REFERENCE_COORD_SYSTEM_NAME   = "SITE_FRAME"
    SURFACE_MODEL_TYPE            = PLANE
    SURFACE_NORMAL_VECTOR         = (0.0133647,0.00176187,-0.999909)
    SURFACE_GROUND_LOCATION       = (-36.6343,-73.9069,0.726934)
  END_GROUP                       = SURFACE_MODEL_PARMS

/* VICAR IMAGE HEADER DATA ELEMENTS */

  OBJECT                          = IMAGE_HEADER
    HEADER_TYPE                   = VICAR2
    INTERCHANGE_FORMAT            = ASCII
    BYTES                         = 7696
    ^DESCRIPTION                  = "VICAR2.TXT"
  END_OBJECT                      = IMAGE_HEADER

/* ODL IMAGE HEADER DATA ELEMENTS */

  OBJECT                          = ODL_HEADER
    HEADER_TYPE                   = ODL
    INTERCHANGE_FORMAT            = ASCII
    BYTES                         = 15392
```

```
    ^DESCRIPTION                  = "ODL.TXT"
  END_OBJECT                      = ODL_HEADER

/* IMAGE DATA ELEMENTS */
  OBJECT                          = IMAGE
    INTERCHANGE_FORMAT            = BINARY
    LINES                         = 981
    LINE_SAMPLES                  = 7696
    SAMPLE_TYPE                   = UNSIGNED_INTEGER
    SAMPLE_BITS                   = 8
    BANDS                         = 1
    BAND_STORAGE_TYPE             = BAND_SEQUENTIAL
  END_OBJECT                      = IMAGE
END_OBJECT                        = FILE

OBJECT                            = FILE
  RECORD_TYPE                     = UNDEFINED
  RECORD_BYTES                    = "N/A"
  FILE_RECORDS                    = "N/A"
  LABEL_RECORDS                   = "N/A"
  ^FILE                           = "N_L000_0351_EDR011CYLTSB0302_DRIVEM1.LIS"
  FILE_NAME                       = "N_L000_0351_EDR011CYLTSB0302_DRIVEM1.LIS"
  ^DESCRIPTION                    = "MSL_CAMERA_SIS.PDF"
END_OBJECT                        = FILE

OBJECT                            = FILE
  RECORD_TYPE                     = UNDEFINED
  RECORD_BYTES                    = "N/A"
  FILE_RECORDS                    = "N/A"
  LABEL_RECORDS                   = "N/A"
  ^FILE                           = "N_L000_0351_EDR011CYLTSB0302_DRIVEM1.NAV"
  FILE_NAME                       = "N_L000_0351_EDR011CYLTSB0302_DRIVEM1.NAV"
  ^DESCRIPTION                    = "MSL_CAMERA_SIS.PDF"
END_OBJECT                        = FILE

OBJECT                            = FILE
  RECORD_TYPE                     = UNDEFINED
  RECORD_BYTES                    = "N/A"
  FILE_RECORDS                    = "N/A"
  LABEL_RECORDS                   = "N/A"
  ^FILE                           = "N_L000_0351_EDR011CYLTSB0302_DRIVEM1.BRT"
  FILE_NAME                       = "N_L000_0351_EDR011CYLTSB0302_DRIVEM1.BRT"
  ^DESCRIPTION                    = "MSL_CAMERA_SIS.PDF"
END_OBJECT                        = FILE
END
```

APPENDIX C – Example LIBS RDR Detached PDS Labels

a) LIBS RDR Product Type “RDR”

```
PDS_VERSION_ID                   = PDS3

/* FILE DATA ELEMENTS */

RECORD_TYPE                      = FIXED_LENGTH
RECORD_BYTES                    =  660
FILE_RECORDS                    =  6161

/* POINTERS TO DATA OBJECTS */

^HEADER              = ("CL5_531897243RDR_F0591596CCAM01514P3.CSV",1)
^SPREADSHEET         = ("CL5_531897243RDR_F0591596CCAM01514P3.CSV",1856 <BYTES>)

/* IDENTIFICATION DATA ELEMENTS */

MSL:ACTIVE_FLIGHT_STRING_ID      = B
DATA_SET_ID                     = "MSL-M-CHEMCAM-LIBS-4/5-RDR-V1.0"
PRODUCT_TYPE                    = "CHEMCAM-RDR"
COMMAND_SEQUENCE_NUMBER          = 9
FRAME_ID                         = "N/A"
FRAME_TYPE                       = "N/A"
GEOMETRY_PROJECTION_TYPE         = RAW
INSTRUMENT_HOST_ID               = "MSL"
INSTRUMENT_HOST_NAME             = "MARS SCIENCE LABORATORY"
INSTRUMENT_ID                    = "CHEMCAM_LIBS"
INSTRUMENT_NAME                  = "CHEMISTRY CAMERA LASER INDUCED
                                    BREAKDOWN SPECTROMETER"
INSTRUMENT_SERIAL_NUMBER         = "UNK"
INSTRUMENT_TYPE                  = "SPECTROMETER"
INSTRUMENT_VERSION_ID            = "FM"
MSL:LOCAL_MEAN_SOLAR_TIME        = "Sol-01514M12:31:20.360"
LOCAL_TRUE_SOLAR_TIME            = "12:31:56"
MISSION_NAME                     = "MARS SCIENCE LABORATORY"
MISSION_PHASE_NAME               = "EXTENDED SURFACE MISSION"
OBSERVATION_ID                   = "UNK"
PLANET_DAY_NUMBER                = 1514
MSL:LOCAL_TRUE_SOLAR_TIME_SOL    = 1514
PRODUCER_INSTITUTION_NAME        = "LOS ALAMOS NATIONAL LABORATORY"
PRODUCT_CREATION_TIME            =  2017-01-26T17:18:42.000
PRODUCT_ID                       = "CL5_531897243RDR_F0591596CCAM01514P3"
PRODUCT_VERSION_ID               = "3"
RELEASE_ID                       = "0014"
MSL:REQUEST_ID                   = "0"
SOURCE_PRODUCT_ID                = "CL5_531897243EDR_F0591596CCAM01514M1"
ROVER_MOTION_COUNTER             = (59,1596,22,54,0,0,180,188,0,0)
ROVER_MOTION_COUNTER_NAME        = (SITE,DRIVE,POSE,ARM,CHIMRA,DRILL,RSM,
                                    HGA,DRT,IC)
SEQUENCE_ID                      = "ccam01514"
SEQUENCE_VERSION_ID              = "10"
SOLAR_LONGITUDE                  = 257.618
SPACECRAFT_CLOCK_CNT_PARTITION   = 1
SPACECRAFT_CLOCK_START_COUNT     = "531897243.240"
SPACECRAFT_CLOCK_STOP_COUNT      = "UNK"
START_TIME                       = 2016-11-08T17:37:17.092
STOP_TIME                        = UNK
TARGET_NAME                      = MARS
TARGET_TYPE                      = PLANET

/* TELEMETRY DATA ELEMENTS */
```

```
APPLICATION_PROCESS_ID            = 140
APPLICATION_PROCESS_NAME          = "CcamSpectra"
MSL:AUTO_DELETE_FLAG              = "FALSE"
DOWNLOAD_PRIORITY                 = 56
EARTH_RECEIVED_START_TIME         = 2016-11-08T20:00:52.439
EARTH_RECEIVED_STOP_TIME          = 2016-11-08T19:56:10.232
SPICE_FILE_NAME                   = "chronos.msl"
TELEMETRY_PROVIDER_ID             = "MPCS_MSL_DP"
MSL:TELEMETRY_SOURCE_HOST_NAME    = "mslsmsampcs1"
TELEMETRY_SOURCE_NAME             = "CcamSpectra_0531897243-24098-2.dat"
MSL:TELEMETRY_SOURCE_TYPE         = "DATA PRODUCT"
MSL:COMMUNICATION_SESSION_ID      = "35142"
MSL:EXPECTED_TRANSMISSION_PATH    = "3851"
MSL:FLIGHT_SOFTWARE_MODE          = "8"
FLIGHT_SOFTWARE_VERSION_ID        = "208889339"
MSL:PRODUCT_COMPLETION_STATUS     = "COMPLETE_CHECKSUM_PASS"
MSL:PRODUCT_TAG                   = "0"
MSL:SEQUENCE_EXECUTION_COUNT      = 1
MSL:TELEMETRY_SOURCE_START_TIME   = 2016-11-08T17:37:17.092
MSL:TELEMETRY_SOURCE_SCLK_START   = "531897243.240"
MSL:TELEMETRY_SOURCE_CHECKSUM     = 56232
MSL:TELEMETRY_SOURCE_SIZE         = 388325
MSL:TRANSMISSION_PATH             = "65535"
MSL:VIRTUAL_CHANNEL_ID            = "32"

/* HISTORY DATA ELEMENTS */

GROUP                             = MSLEDRGEN_HISTORY_PARMS
  SOFTWARE_NAME                   = MSLEDRGEN
  SOFTWARE_VERSION_ID             = "V1.0 04-01-2011"
  PROCESSING_HISTORY_TEXT         = "CODMAC LEVEL 1 TO LEVEL 2 CONVERSION
                                     VIA JPL/MIPL MSLEDRGEN"
END_GROUP                         = MSLEDRGEN_HISTORY_PARMS

/* CAMERA MODEL DATA ELEMENTS */

GROUP                             = GEOMETRIC_CAMERA_MODEL_PARMS
  CALIBRATION_SOURCE_ID           = "3"
  ^MODEL_DESC                     = "GEOMETRIC_CM.TXT"
  MODEL_TYPE                      = "CAHV"
  MODEL_COMPONENT_ID              = ("C","A","H","V")
  MODEL_COMPONENT_NAME            = ("CENTER","AXIS","HORIZONTAL","VERTICAL")
  MODEL_COMPONENT_1               = (1.0266,0.541721,-2.02478)
  MODEL_COMPONENT_2               = (0.73867,0.515791,0.433948)
  MODEL_COMPONENT_3               = (-20649.6,44272.3,142.727)
  MODEL_COMPONENT_4               = (-27578.8,-12999.8,38047.5)
  REFERENCE_COORD_SYSTEM_NAME     = "ROVER_NAV_FRAME"
  ROVER_MOTION_COUNTER            = (59,1596,22,54,0,0,180,188,0,0)
  MSL:MODEL_TRANSFORM_VECTOR      = (0.804374,0.559364,-1.90608)
  MSL:MODEL_TRANSFORM_QUATERNION  = (0.919328,0.0745237,-0.326214,0.20704)
END_GROUP                         = GEOMETRIC_CAMERA_MODEL_PARMS

/* COORDINATE SYSTEM STATE: ROVER */

GROUP                             = ROVER_COORD_SYSTEM_PARMS
  COORDINATE_SYSTEM_NAME          = "ROVER_NAV_FRAME"
  COORDINATE_SYSTEM_INDEX         = (59,1596,22,54,0,0,180,188,0,0)
  COORDINATE_SYSTEM_INDEX_NAME    = (SITE,DRIVE,POSE,ARM,CHIMRA,DRILL,RSM,
                                     HGA,DRT,IC)
  ORIGIN_OFFSET_VECTOR            = (-188.87,-18.0913,-5.93937)
  ORIGIN_ROTATION_QUATERNION      = (0.163375,-0.0673011,0.0182682,0.984096)
  QUATERNION_MEASUREMENT_METHOD   = FINE
  POSITIVE_AZIMUTH_DIRECTION      = CLOCKWISE
  POSITIVE_ELEVATION_DIRECTION    = UP
  REFERENCE_COORD_SYSTEM_NAME     = SITE_FRAME
  REFERENCE_COORD_SYSTEM_INDEX    = 59
```

```
END_GROUP                         = ROVER_COORD_SYSTEM_PARMS

/* COORDINATE SYSTEM STATE: SITE */

GROUP                             = SITE_COORD_SYSTEM_PARMS
  MSL:SOLUTION_ID                 = "telemetry"
  COORDINATE_SYSTEM_NAME          = "SITE_FRAME"
  COORDINATE_SYSTEM_INDEX         = 59
  COORDINATE_SYSTEM_INDEX_NAME    = "SITE"
  ORIGIN_OFFSET_VECTOR            = (-312.707,134.942,-25.5103)
  ORIGIN_ROTATION_QUATERNION      = (1,0,0,0)
  POSITIVE_AZIMUTH_DIRECTION      = "CLOCKWISE"
  POSITIVE_ELEVATION_DIRECTION    = "UP"
  REFERENCE_COORD_SYSTEM_NAME     = "SITE_FRAME"
  REFERENCE_COORD_SYSTEM_INDEX    = 58
END_GROUP                         = SITE_COORD_SYSTEM_PARMS

/* COORDINATE SYSTEM STATE: REMOTE SENSING MAST */

GROUP                             = RSM_COORD_SYSTEM_PARMS
  COORDINATE_SYSTEM_NAME          = "MAST_FRAME"
  COORDINATE_SYSTEM_INDEX         = (59,1596,22,54,0,0,180,188,0,0)
  COORDINATE_SYSTEM_INDEX_NAME    = (SITE,DRIVE,POSE,ARM,CHIMRA,DRILL,RSM,
                                     HGA,DRT,IC)
  ORIGIN_OFFSET_VECTOR            = (0.804374,0.559364,-1.90608)
  ORIGIN_ROTATION_QUATERNION      = (0.919328,0.0745237,-0.326214,0.20704)
  POSITIVE_AZIMUTH_DIRECTION      = CLOCKWISE
  POSITIVE_ELEVATION_DIRECTION    = UP
  REFERENCE_COORD_SYSTEM_NAME     = ROVER_NAV_FRAME
  REFERENCE_COORD_SYSTEM_INDEX    = (59,1596,22,54,0,0,180,188)
END_GROUP                         = RSM_COORD_SYSTEM_PARMS

/* COORDINATE SYSTEM STATE: ROBOTIC ARM */

GROUP                             = ARM_COORD_SYSTEM_PARMS
  COORDINATE_SYSTEM_NAME          = "ARM_FRAME"
  COORDINATE_SYSTEM_INDEX         = (59,1596,22,54,0,0,180,188,0,0)
  COORDINATE_SYSTEM_INDEX_NAME    = (SITE,DRIVE,POSE,ARM,CHIMRA,DRILL,RSM,
                                     HGA,DRT,IC)
  ORIGIN_OFFSET_VECTOR            = (0.824981,-0.703072,-0.24517)
  ORIGIN_ROTATION_QUATERNION      = (0.270667,-0.477234,0.719953,0.425034)
  REFERENCE_COORD_SYSTEM_NAME     = ROVER_NAV_FRAME
  REFERENCE_COORD_SYSTEM_INDEX    = (59,1596,22,54,0,0,180,188)
END_GROUP                         = ARM_COORD_SYSTEM_PARMS

/* ARTICULATION DEVICE STATE: REMOTE SENSING MAST */

GROUP                             = RSM_ARTICULATION_STATE_PARMS
  ARTICULATION_DEVICE_ID          = RSM
  ARTICULATION_DEVICE_NAME        = "REMOTE SENSING MAST"
  ARTICULATION_DEVICE_ANGLE       = (3.61023 <rad>,0.905672 <rad>,
                                     3.61291 <rad>,0.909407 <rad>,
                                     3.61498 <rad>,0.90957 <rad>,
                                     3.61299 <rad>,0.9094 <rad>)
  ARTICULATION_DEVICE_ANGLE_NAME  = ("AZIMUTH-MEASURED",
                                     "ELEVATION-MEASURED",
                                     "AZIMUTH-REQUESTED",
                                     "ELEVATION-REQUESTED","AZIMUTH-INITIAL"
                                     ,"ELEVATION-INITIAL","AZIMUTH-FINAL",
                                     "ELEVATION-FINAL")
  ARTICULATION_DEVICE_MODE        = DEPLOYED
END_GROUP                         = RSM_ARTICULATION_STATE_PARMS

/* ARTICULATION DEVICE STATE: ROBOTIC ARM */

GROUP                             = ARM_ARTICULATION_STATE_PARMS
```

```
  ARTICULATION_DEVICE_ID          = ARM
  ARTICULATION_DEVICE_ANGLE       = (1.57222 <rad>,-0.277767 <rad>,
                                     -2.81629 <rad>,3.12107 <rad>,
                                     0.593842 <rad>,1.56839 <rad>,
                                     -0.277768 <rad>,-2.82549 <rad>,
                                     3.11656 <rad>,0.593551 <rad>)
  ARTICULATION_DEVICE_ANGLE_NAME  = ("JOINT 1 SHOULDER AZIMUTH",
                                     "JOINT 2 SHOULDER ELEVATION",
                                     "JOINT 3 ELBOW-ENCODER",
                                     "JOINT 4 WRIST-ENCODER",
                                     "JOINT 5 TURRET-ENCODER",
                                     "JOINT 1 SHOULDER AZIMUTH-RESOLVER",
                                     "JOINT 2 SHOULDER ELEVATION-RESOLVER",
                                     "JOINT 3 ELBOW-RESOLVER",
                                     "JOINT 4 WRIST-RESOLVER",
                                     "JOINT 5 TURRET-RESOLVER")
  ARTICULATION_DEVICE_TEMP        = (-8.36346 <degC>,-2.12067 <degC>,
                                     3.53258 <degC>,4.37844 <degC>,
                                     -12.6872 <degC>)
  ARTICULATION_DEVICE_TEMP_NAME   = ("AZIMUTH JOINT","ELEVATION JOINT",
                                     "ELBOW JOINT","WRIST JOINT",
                                     "TURRET JOINT")
  ARTICULATION_DEV_VECTOR         = (-0.138431,0.0139646,0.990274)
  CONTACT_SENSOR_STATE            = "NO CONTACT"
  ARTICULATION_DEV_INSTRUMENT_ID  = "MAHLI"
  ARTICULATION_DEVICE_MODE        = "FREE SPACE"
END_GROUP                         = ARM_ARTICULATION_STATE_PARMS

/* ARTICULATION DEVICE STATE: MOBILITY CHASSIS */

GROUP                             = CHASSIS_ARTICULATION_ST_PARMS
  ARTICULATION_DEVICE_ID          = CHASSIS
  ARTICULATION_DEVICE_NAME        = "MOBILITY CHASSIS"
  ARTICULATION_DEVICE_ANGLE       = (-0 <rad>,4.26106e-05 <rad>,-0 <rad>,
                                     -0 <rad>,3.61023 <rad>,0.905672 <rad>,
                                     -0.0504155 <rad>,-0.0428046 <rad>,
                                     -0.0191213 <rad>,0.0142404 <rad>)
  ARTICULATION_DEVICE_ANGLE_NAME  = ("LEFT FRONT WHEEL","RIGHT FRONT WHEEL"
                                     ,"LEFT REAR WHEEL","RIGHT REAR WHEEL",
                                     "RSM AZIMUTH","RSM ELEVATION",
                                     "LEFT BOGIE","RIGHT BOGIE",
                                     "LEFT DIFFERENTIAL",
                                     "RIGHT DIFFERENTIAL")
  ARTICULATION_DEVICE_MODE        = DEPLOYED
END_GROUP                         = CHASSIS_ARTICULATION_ST_PARMS

/* ARTICULATION DEVICE STATE: HIGH GAIN ANTENNA */

GROUP                             = HGA_ARTICULATION_STATE_PARMS
  ARTICULATION_DEVICE_ID          = HGA
  ARTICULATION_DEVICE_NAME        = "HIGH GAIN ANTENNA"
  ARTICULATION_DEVICE_ANGLE       = (-3.33928e-05 <rad>,-0.784975 <rad>)
  ARTICULATION_DEVICE_ANGLE_NAME  = ("AZIMUTH","ELEVATION")
  ARTICULATION_DEVICE_MODE        = DEPLOYED
END_GROUP                         = HGA_ARTICULATION_STATE_PARMS

/* OBSERVATION REQUEST */

GROUP                             = OBSERVATION_REQUEST_PARMS
  SOURCE_ID                       = "GROUND COMMANDED"
  MSL:INSTRUMENT_COORD_FRAME_INDEX= 0
  MSL:INSTRUMENT_COORD_FRAME_ID   = "NAV"
  INSTRUMENT_COORDINATE_NAME      = ("AZIMUTH","ELEVATION")
  INSTRUMENT_COORDINATE           = (-0.002 <rad>,0 <rad>)
  MSL:DARK_SPECTRA_MODE           = "NONE"
  GAIN_NUMBER                     = 0
```

```
  GROUP_APPLICABILITY_FLAG         = "TRUE"
  MSL:INSTRUMENT_FOCUS_MODE        = "NO_FOCUS"
  MSL:INSTRUMENT_FOCUS_DISTANCE    = 3250
  INSTRUMENT_MODE_ID               = "SPECTRAL_DATA"
  MSL:LASER_MODE                   = "YES"
  OFFSET_MODE_ID                   = "0"
 MSL:N_SHOTS_2_IGNORE            =  0
 MSL:N_SHOTS_2_AVG               =  0
 MSL:N_SHOTS                     =  30
 MSL:SPECTROMETER_CONTROL_BYTE   =  2
 MSL:SPEC_IMAGE_TYPE             =  4
 MSL:SPECTROMETER_SERIAL_CLOCK   =  1
 MSL:SPECTROMETER_SELECT         =  "UNK"
 MSL:ICT_DIVIDER                 =  300
 MSL:IPBC_DIVIDER                =  330
 MSL:START_ROW_UV                =  100
 MSL:STOP_ROW_UV                 =  300
 MSL:START_ROW_VIS               =  75
 MSL:STOP_ROW_VIS                =  275
 MSL:START_ROW_VNIR              =  100
 MSL:STOP_ROW_VNIR               =  300
 MSL:OBS_FROM_LIMIT_SWITCH       =  0
 MSL:STACK_1_LEVEL               =  100
 MSL:STACK_2_LEVEL               =  95
 MSL:STACK_3_LEVEL               =  95
 MSL:STACK_DURATION              =  145
 MSL:CCAM_TIME_BETWEEN_SHOTS          =  244
 MSL:SPEC_AD_CONVERTVNIR         =  5
 MSL:SPEC_AD_CONVERTVIS          =  5
 MSL:SPEC_AD_CONVERTUV           =  5
 MSL:SPEC_VERT_CLK               =  2
END_GROUP                        = OBSERVATION_REQUEST_PARMS

/* INSTRUMENT STATE RESULTS */

GROUP                             = INSTRUMENT_STATE_PARMS
  MSL:SPECIAL_LINE                = 497
  MSL:SPECIAL_SAMPLE              = 532
  MSL:SPECIAL_NAME                = "Location of the LIBS laser spot in the
                                     RMI image described by the camera
                                    model"
  INSTRUMENT_TEMPERATURE          = (-21.9229 <degC>,-21.9229 <degC>,
                                    -14.8855 <degC>,-14.8855 <degC>,
                                    -14.8855 <degC>,-14.8855 <degC>,
                                    0.392304 <degC>,8.78029 <degC>,
                                    17.877 <degC>,17.877 <degC>)
  INSTRUMENT_TEMPERATURE_NAME     = ("MU_OBOX_TELESCOPE","MU_LASER_IF",
                                    "MU_EBOX_HEATSINK","MU_EBOX_FPGA",
                                    "BU_CCD_VNIR_B","BU_SPEC_B",
                                    "BU_CCD_UV_A","BU_SPEC_A","BU_DEMUX_A",
                                    "BU_DEMUX_B")
  MSL:INSTRUMENT_TEMPERATURE_STATUS= (0,-42,0,-42,-42,-42,0,0,0,-42)
END_GROUP                         = INSTRUMENT_STATE_PARMS

/* DERIVED GEOMETRY DATA ELEMENTS: ROVER FRAME */

GROUP                             = ROVER_DERIVED_GEOMETRY_PARMS
  INSTRUMENT_AZIMUTH              = 25.6447
  INSTRUMENT_ELEVATION            = -39.2685
  REFERENCE_COORD_SYSTEM_NAME     = "ROVER_NAV_FRAME"
  REFERENCE_COORD_SYSTEM_INDEX    = (59,1596,22,54,0,0,180,188)
  SUN_VIEW_DIRECTION              = (0.408491,0.216446,-0.886728)
  SOLAR_AZIMUTH                   = 27.9176 <deg>
  SOLAR_ELEVATION                 = 62.4649 <deg>
END_GROUP                         = ROVER_DERIVED_GEOMETRY_PARMS
```

```
/* DERIVED GEOMETRY DATA ELEMENTS: SITE FRAME */

GROUP                              = SITE_DERIVED_GEOMETRY_PARMS
  INSTRUMENT_AZIMUTH               = 183.907
  INSTRUMENT_ELEVATION             = -32.3346
  REFERENCE_COORD_SYSTEM_NAME      = "SITE_FRAME"
  REFERENCE_COORD_SYSTEM_INDEX     = (59,1596,22,54,0,0,180,188)
  SOLAR_AZIMUTH                    = 200.302 <deg>
  SOLAR_ELEVATION                  = 68.6462 <deg>
  POSITIVE_AZIMUTH_DIRECTION       = CLOCKWISE
END_GROUP                          = SITE_DERIVED_GEOMETRY_PARMS

/* HEADER DESCRIPTION */

OBJECT                           = HEADER
  BYTES                          = 1855
  RECORDS                        = 17
  HEADER_TYPE                    = "TEXT"
END_OBJECT                       = HEADER

/* DATA DESCRIPTION */

OBJECT                           = SPREADSHEET
  FIELD_DELIMITER                = "COMMA"
  FIELDS                         = 33
  ROW_BYTES                      = 660
  ROWS                           = 6144
  DESCRIPTION                    = "Clean calibrated spectra from LIBS in
      radiance as a function of wavelength.  The number of spectra in the
      file depends on the number of shots plus the median and average."

  OBJECT                       = FIELD
    FIELD_NUMBER               = 1
    NAME                       = "WAVELENGTH"
    BYTES                      = 20
    DATA_TYPE                  = ASCII_REAL
    UNIT                       = NANOMETER
    DESCRIPTION                = "Calibrated wavelength"
  END_OBJECT                   = FIELD

  OBJECT                       = FIELD
    FIELD_NUMBER               = 2
    NAME                       = "SPECTRA_001"
    BYTES                      = 20
    DATA_TYPE                  = ASCII_REAL
    UNIT                       = DIGITAL_NUMBER
    DESCRIPTION                = "SHOT 1"
  END_OBJECT                   = FIELD


  OBJECT                       = FIELD
    FIELD_NUMBER               = 3
    NAME                       = "SPECTRA_002"
    BYTES                      = 20
    DATA_TYPE                  = ASCII_REAL
    UNIT                       = DIGITAL_NUMBER
    DESCRIPTION                = "SHOT 2"
  END_OBJECT                   = FIELD


  OBJECT                       = FIELD
    FIELD_NUMBER               = 4
    NAME                       = "SPECTRA_003"
    BYTES                      = 20
    DATA_TYPE                  = ASCII_REAL
    UNIT                       = DIGITAL_NUMBER
```

```
  DESCRIPTION                 = "SHOT 3"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  FIELD_NUMBER                = 5
  NAME                        = "SPECTRA_004"
  BYTES                       = 20
  DATA_TYPE                   = ASCII_REAL
  UNIT                        = DIGITAL_NUMBER
  DESCRIPTION                 = "SHOT 4"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  FIELD_NUMBER                = 6
  NAME                        = "SPECTRA_005"
  BYTES                       = 20
  DATA_TYPE                   = ASCII_REAL
  UNIT                        = DIGITAL_NUMBER
  DESCRIPTION                 = "SHOT 5"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  FIELD_NUMBER                = 7
  NAME                        = "SPECTRA_006"
  BYTES                       = 20
  DATA_TYPE                   = ASCII_REAL
  UNIT                        = DIGITAL_NUMBER
  DESCRIPTION                 = "SHOT 6"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  FIELD_NUMBER                = 8
  NAME                        = "SPECTRA_007"
  BYTES                       = 20
  DATA_TYPE                   = ASCII_REAL
  UNIT                        = DIGITAL_NUMBER
  DESCRIPTION                 = "SHOT 7"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  FIELD_NUMBER                = 9
  NAME                        = "SPECTRA_008"
  BYTES                       = 20
  DATA_TYPE                   = ASCII_REAL
  UNIT                        = DIGITAL_NUMBER
  DESCRIPTION                 = "SHOT 8"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  FIELD_NUMBER                = 10
  NAME                        = "SPECTRA_009"
  BYTES                       = 20
  DATA_TYPE                   = ASCII_REAL
  UNIT                        = DIGITAL_NUMBER
  DESCRIPTION                 = "SHOT 9"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  FIELD_NUMBER                = 11
```

```
  NAME                        = "SPECTRA_010"
  BYTES                       = 20
  DATA_TYPE                   = ASCII_REAL
  UNIT                        = DIGITAL_NUMBER
  DESCRIPTION                 = "SHOT 10"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  FIELD_NUMBER                = 12
  NAME                        = "SPECTRA_011"
  BYTES                       = 20
  DATA_TYPE                   = ASCII_REAL
  UNIT                        = DIGITAL_NUMBER
  DESCRIPTION                 = "SHOT 11"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  FIELD_NUMBER                = 13
  NAME                        = "SPECTRA_012"
  BYTES                       = 20
  DATA_TYPE                   = ASCII_REAL
  UNIT                        = DIGITAL_NUMBER
  DESCRIPTION                 = "SHOT 12"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  FIELD_NUMBER                = 14
  NAME                        = "SPECTRA_013"
  BYTES                       = 20
  DATA_TYPE                   = ASCII_REAL
  UNIT                        = DIGITAL_NUMBER
  DESCRIPTION                 = "SHOT 13"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  FIELD_NUMBER                = 15
  NAME                        = "SPECTRA_014"
  BYTES                       = 20
  DATA_TYPE                   = ASCII_REAL
  UNIT                        = DIGITAL_NUMBER
  DESCRIPTION                 = "SHOT 14"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  FIELD_NUMBER                = 16
  NAME                        = "SPECTRA_015"
  BYTES                       = 20
  DATA_TYPE                   = ASCII_REAL
  UNIT                        = DIGITAL_NUMBER
  DESCRIPTION                 = "SHOT 15"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  FIELD_NUMBER                = 17
  NAME                        = "SPECTRA_016"
  BYTES                       = 20
  DATA_TYPE                   = ASCII_REAL
  UNIT                        = DIGITAL_NUMBER
  DESCRIPTION                 = "SHOT 16"
END_OBJECT                    = FIELD
```

```
OBJECT                        = FIELD
  FIELD_NUMBER                = 18
  NAME                        = "SPECTRA_017"
  BYTES                       = 20
  DATA_TYPE                   = ASCII_REAL
  UNIT                        = DIGITAL_NUMBER
  DESCRIPTION                 = "SHOT 17"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  FIELD_NUMBER                = 19
  NAME                        = "SPECTRA_018"
  BYTES                       = 20
  DATA_TYPE                   = ASCII_REAL
  UNIT                        = DIGITAL_NUMBER
  DESCRIPTION                 = "SHOT 18"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  FIELD_NUMBER                = 20
  NAME                        = "SPECTRA_019"
  BYTES                       = 20
  DATA_TYPE                   = ASCII_REAL
  UNIT                        = DIGITAL_NUMBER
  DESCRIPTION                 = "SHOT 19"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  FIELD_NUMBER                = 21
  NAME                        = "SPECTRA_020"
  BYTES                       = 20
  DATA_TYPE                   = ASCII_REAL
  UNIT                        = DIGITAL_NUMBER
  DESCRIPTION                 = "SHOT 20"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  FIELD_NUMBER                = 22
  NAME                        = "SPECTRA_021"
  BYTES                       = 20
  DATA_TYPE                   = ASCII_REAL
  UNIT                        = DIGITAL_NUMBER
  DESCRIPTION                 = "SHOT 21"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  FIELD_NUMBER                = 23
  NAME                        = "SPECTRA_022"
  BYTES                       = 20
  DATA_TYPE                   = ASCII_REAL
  UNIT                        = DIGITAL_NUMBER
  DESCRIPTION                 = "SHOT 22"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  FIELD_NUMBER                = 24
  NAME                        = "SPECTRA_023"
  BYTES                       = 20
```

```
  DATA_TYPE                   = ASCII_REAL
  UNIT                        = DIGITAL_NUMBER
  DESCRIPTION                 = "SHOT 23"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  FIELD_NUMBER                = 25
  NAME                        = "SPECTRA_024"
  BYTES                       = 20
  DATA_TYPE                   = ASCII_REAL
  UNIT                        = DIGITAL_NUMBER
  DESCRIPTION                 = "SHOT 24"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  FIELD_NUMBER                = 26
  NAME                        = "SPECTRA_025"
  BYTES                       = 20
  DATA_TYPE                   = ASCII_REAL
  UNIT                        = DIGITAL_NUMBER
  DESCRIPTION                 = "SHOT 25"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  FIELD_NUMBER                = 27
  NAME                        = "SPECTRA_026"
  BYTES                       = 20
  DATA_TYPE                   = ASCII_REAL
  UNIT                        = DIGITAL_NUMBER
  DESCRIPTION                 = "SHOT 26"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  FIELD_NUMBER                = 28
  NAME                        = "SPECTRA_027"
  BYTES                       = 20
  DATA_TYPE                   = ASCII_REAL
  UNIT                        = DIGITAL_NUMBER
  DESCRIPTION                 = "SHOT 27"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  FIELD_NUMBER                = 29
  NAME                        = "SPECTRA_028"
  BYTES                       = 20
  DATA_TYPE                   = ASCII_REAL
  UNIT                        = DIGITAL_NUMBER
  DESCRIPTION                 = "SHOT 28"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  FIELD_NUMBER                = 30
  NAME                        = "SPECTRA_029"
  BYTES                       = 20
  DATA_TYPE                   = ASCII_REAL
  UNIT                        = DIGITAL_NUMBER
  DESCRIPTION                 = "SHOT 29"
END_OBJECT                    = FIELD
```

```
  OBJECT                        = FIELD
    FIELD_NUMBER                = 31
    NAME                        = "SPECTRA_030"
    BYTES                       = 20
    DATA_TYPE                   = ASCII_REAL
    UNIT                        = DIGITAL_NUMBER
    DESCRIPTION                 = "SHOT 30"
  END_OBJECT                    = FIELD

  OBJECT                        = FIELD
    FIELD_NUMBER                = 32
    NAME                        = "MEDIAN"
    BYTES                       = 20
    DATA_TYPE                   = ASCII_REAL
    UNIT                        = DIGITAL_NUMBER
    DESCRIPTION                 = "Median of 30 shots"
  END_OBJECT                    = FIELD

  OBJECT                        = FIELD
    FIELD_NUMBER                = 33
    NAME                        = "AVERAGE"
    BYTES                       = 20
    DATA_TYPE                   = ASCII_REAL
    UNIT                        = DIGITAL_NUMBER
    DESCRIPTION                 = "Average of 30 shots"
  END_OBJECT                    = FIELD
END_OBJECT                      = SPREADSHEET

END
```

b) LIBS RDR Product Type "CCS"

```
PDS_VERSION_ID                    = PDS3

/* FILE DATA ELEMENTS */

RECORD_TYPE                       = FIXED_LENGTH
RECORD_BYTES                    =  660
FILE_RECORDS                    =  6161

/* POINTERS TO DATA OBJECTS */

^HEADER            = ("CL5_531897243CCS_F0591596CCAM01514P3.CSV",1)
^SPREADSHEET       = ("CL5_531897243CCS_F0591596CCAM01514P3.CSV",1856 <BYTES>)

/* IDENTIFICATION DATA ELEMENTS */

MSL:ACTIVE_FLIGHT_STRING_ID       = B
DATA_SET_ID                     = "MSL-M-CHEMCAM-LIBS-4/5-RDR-V1.0"
PRODUCT_TYPE                    = "CHEMCAM-CCS"
COMMAND_SEQUENCE_NUMBER           = 9
FRAME_ID                          = "N/A"
FRAME_TYPE                        = "N/A"
GEOMETRY_PROJECTION_TYPE          = RAW
INSTRUMENT_HOST_ID                = "MSL"
INSTRUMENT_HOST_NAME              = "MARS SCIENCE LABORATORY"
INSTRUMENT_ID                     = "CHEMCAM_LIBS"
INSTRUMENT_NAME                   = "CHEMISTRY CAMERA LASER INDUCED
                                     BREAKDOWN SPECTROMETER"
INSTRUMENT_SERIAL_NUMBER          = "UNK"
INSTRUMENT_TYPE                   = "SPECTROMETER"
INSTRUMENT_VERSION_ID             = "FM"
MSL:LOCAL_MEAN_SOLAR_TIME         = "Sol-01514M12:31:20.360"
LOCAL_TRUE_SOLAR_TIME             = "12:31:56"
MISSION_NAME                      = "MARS SCIENCE LABORATORY"
```

```
MISSION_PHASE_NAME                 = "EXTENDED SURFACE MISSION"
OBSERVATION_ID                     = "UNK"
PLANET_DAY_NUMBER                  = 1514
MSL:LOCAL_TRUE_SOLAR_TIME_SOL      = 1514
PRODUCER_INSTITUTION_NAME          = "LOS ALAMOS NATIONAL LABORATORY"
PRODUCT_CREATION_TIME              =  2017-01-26T17:20:18.000
PRODUCT_ID                         = "CL5_531897243CCS_F0591596CCAM01514P3"
PRODUCT_VERSION_ID                 = "3"
RELEASE_ID                         = "0014"
MSL:REQUEST_ID                     = "0"
SOURCE_PRODUCT_ID                  = "CL5_531897243EDR_F0591596CCAM01514M1"
ROVER_MOTION_COUNTER               = (59,1596,22,54,0,0,180,188,0,0)
ROVER_MOTION_COUNTER_NAME          = (SITE,DRIVE,POSE,ARM,CHIMRA,DRILL,RSM,
                                      HGA,DRT,IC)
SEQUENCE_ID                        = "ccam01514"
SEQUENCE_VERSION_ID                = "10"
SOLAR_LONGITUDE                    = 257.618
SPACECRAFT_CLOCK_CNT_PARTITION     = 1
SPACECRAFT_CLOCK_START_COUNT       = "531897243.240"
SPACECRAFT_CLOCK_STOP_COUNT        = "UNK"
START_TIME                         = 2016-11-08T17:37:17.092
STOP_TIME                          = UNK
TARGET_NAME                        = MARS
TARGET_TYPE                        = PLANET

/* TELEMETRY DATA ELEMENTS */

APPLICATION_PROCESS_ID             = 140
APPLICATION_PROCESS_NAME           = "CcamSpectra"
MSL:AUTO_DELETE_FLAG               = "FALSE"
DOWNLOAD_PRIORITY                  = 56
EARTH_RECEIVED_START_TIME          = 2016-11-08T20:00:52.439
EARTH_RECEIVED_STOP_TIME           = 2016-11-08T19:56:10.232
SPICE_FILE_NAME                    = "chronos.msl"
TELEMETRY_PROVIDER_ID              = "MPCS_MSL_DP"
MSL:TELEMETRY_SOURCE_HOST_NAME     = "mslsmsampcs1"
TELEMETRY_SOURCE_NAME              = "CcamSpectra_0531897243-24098-2.dat"
MSL:TELEMETRY_SOURCE_TYPE          = "DATA PRODUCT"
MSL:COMMUNICATION_SESSION_ID       = "35142"
MSL:EXPECTED_TRANSMISSION_PATH     = "3851"
MSL:FLIGHT_SOFTWARE_MODE           = "8"
FLIGHT_SOFTWARE_VERSION_ID         = "208889339"
MSL:PRODUCT_COMPLETION_STATUS      = "COMPLETE_CHECKSUM_PASS"
MSL:PRODUCT_TAG                    = "0"
MSL:SEQUENCE_EXECUTION_COUNT       = 1
MSL:TELEMETRY_SOURCE_START_TIME    = 2016-11-08T17:37:17.092
MSL:TELEMETRY_SOURCE_SCLK_START    = "531897243.240"
MSL:TELEMETRY_SOURCE_CHECKSUM      = 56232
MSL:TELEMETRY_SOURCE_SIZE          = 388325
MSL:TRANSMISSION_PATH              = "65535"
MSL:VIRTUAL_CHANNEL_ID             = "32"

/* HISTORY DATA ELEMENTS */

GROUP                              = MSLEDRGEN_HISTORY_PARMS
  SOFTWARE_NAME                    = MSLEDRGEN
  SOFTWARE_VERSION_ID              = "V1.0 04-01-2011"
  PROCESSING_HISTORY_TEXT          = "CODMAC LEVEL 1 TO LEVEL 2 CONVERSION
                                      VIA JPL/MIPL MSLEDRGEN"
END_GROUP                          = MSLEDRGEN_HISTORY_PARMS

/* CAMERA MODEL DATA ELEMENTS */

GROUP                              = GEOMETRIC_CAMERA_MODEL_PARMS
  CALIBRATION_SOURCE_ID            = "3"
  ^MODEL_DESC                      = "GEOMETRIC_CM.TXT"
```

```
  MODEL_TYPE                        = "CAHV"
  MODEL_COMPONENT_ID                = ("C","A","H","V")
  MODEL_COMPONENT_NAME              = ("CENTER","AXIS","HORIZONTAL","VERTICAL")
  MODEL_COMPONENT_1                 = (1.0266,0.541721,-2.02478)
  MODEL_COMPONENT_2                 = (0.73867,0.515791,0.433948)
  MODEL_COMPONENT_3                 = (-20649.6,44272.3,142.727)
  MODEL_COMPONENT_4                 = (-27578.8,-12999.8,38047.5)
  REFERENCE_COORD_SYSTEM_NAME       = "ROVER_NAV_FRAME"
  ROVER_MOTION_COUNTER              = (59,1596,22,54,0,0,180,188,0,0)
  MSL:MODEL_TRANSFORM_VECTOR        = (0.804374,0.559364,-1.90608)
  MSL:MODEL_TRANSFORM_QUATERNION    = (0.919328,0.0745237,-0.326214,0.20704)
END_GROUP                           = GEOMETRIC_CAMERA_MODEL_PARMS

/* COORDINATE SYSTEM STATE: ROVER */

GROUP                               = ROVER_COORD_SYSTEM_PARMS
  COORDINATE_SYSTEM_NAME            = "ROVER_NAV_FRAME"
  COORDINATE_SYSTEM_INDEX           = (59,1596,22,54,0,0,180,188,0,0)
  COORDINATE_SYSTEM_INDEX_NAME      = (SITE,DRIVE,POSE,ARM,CHIMRA,DRILL,RSM,
                                       HGA,DRT,IC)
  ORIGIN_OFFSET_VECTOR              = (-188.87,-18.0913,-5.93937)
  ORIGIN_ROTATION_QUATERNION        = (0.163375,-0.0673011,0.0182682,0.984096)
  QUATERNION_MEASUREMENT_METHOD     = FINE
  POSITIVE_AZIMUTH_DIRECTION        = CLOCKWISE
  POSITIVE_ELEVATION_DIRECTION      = UP
  REFERENCE_COORD_SYSTEM_NAME       = SITE_FRAME
  REFERENCE_COORD_SYSTEM_INDEX      = 59
END_GROUP                           = ROVER_COORD_SYSTEM_PARMS

/* COORDINATE SYSTEM STATE: SITE */

GROUP                               = SITE_COORD_SYSTEM_PARMS
  MSL:SOLUTION_ID                   = "telemetry"
  COORDINATE_SYSTEM_NAME            = "SITE_FRAME"
  COORDINATE_SYSTEM_INDEX           = 59
  COORDINATE_SYSTEM_INDEX_NAME      = "SITE"
  ORIGIN_OFFSET_VECTOR              = (-312.707,134.942,-25.5103)
  ORIGIN_ROTATION_QUATERNION        = (1,0,0,0)
  POSITIVE_AZIMUTH_DIRECTION        = "CLOCKWISE"
  POSITIVE_ELEVATION_DIRECTION      = "UP"
  REFERENCE_COORD_SYSTEM_NAME       = "SITE_FRAME"
  REFERENCE_COORD_SYSTEM_INDEX      = 58
END_GROUP                           = SITE_COORD_SYSTEM_PARMS

/* COORDINATE SYSTEM STATE: REMOTE SENSING MAST */

GROUP                               = RSM_COORD_SYSTEM_PARMS
  COORDINATE_SYSTEM_NAME            = "MAST_FRAME"
  COORDINATE_SYSTEM_INDEX           = (59,1596,22,54,0,0,180,188,0,0)
  COORDINATE_SYSTEM_INDEX_NAME      = (SITE,DRIVE,POSE,ARM,CHIMRA,DRILL,RSM,
                                       HGA,DRT,IC)
  ORIGIN_OFFSET_VECTOR              = (0.804374,0.559364,-1.90608)
  ORIGIN_ROTATION_QUATERNION        = (0.919328,0.0745237,-0.326214,0.20704)
  POSITIVE_AZIMUTH_DIRECTION        = CLOCKWISE
  POSITIVE_ELEVATION_DIRECTION      = UP
  REFERENCE_COORD_SYSTEM_NAME       = ROVER_NAV_FRAME
  REFERENCE_COORD_SYSTEM_INDEX      = (59,1596,22,54,0,0,180,188)
END_GROUP                           = RSM_COORD_SYSTEM_PARMS

/* COORDINATE SYSTEM STATE: ROBOTIC ARM */

GROUP                               = ARM_COORD_SYSTEM_PARMS
  COORDINATE_SYSTEM_NAME            = "ARM_FRAME"
  COORDINATE_SYSTEM_INDEX           = (59,1596,22,54,0,0,180,188,0,0)
  COORDINATE_SYSTEM_INDEX_NAME      = (SITE,DRIVE,POSE,ARM,CHIMRA,DRILL,RSM,
                                       HGA,DRT,IC)
```

```
  ORIGIN_OFFSET_VECTOR            = (0.824981,-0.703072,-0.24517)
  ORIGIN_ROTATION_QUATERNION      = (0.270667,-0.477234,0.719953,0.425034)
  REFERENCE_COORD_SYSTEM_NAME     = ROVER_NAV_FRAME
  REFERENCE_COORD_SYSTEM_INDEX    = (59,1596,22,54,0,0,180,188)
END_GROUP                         = ARM_COORD_SYSTEM_PARMS

/* ARTICULATION DEVICE STATE: REMOTE SENSING MAST */

GROUP                             = RSM_ARTICULATION_STATE_PARMS
  ARTICULATION_DEVICE_ID          = RSM
  ARTICULATION_DEVICE_NAME        = "REMOTE SENSING MAST"
  ARTICULATION_DEVICE_ANGLE       = (3.61023 <rad>,0.905672 <rad>,
                                     3.61291 <rad>,0.909407 <rad>,
                                     3.61498 <rad>,0.90957 <rad>,
                                     3.61299 <rad>,0.9094 <rad>)
  ARTICULATION_DEVICE_ANGLE_NAME  = ("AZIMUTH-MEASURED",
                                     "ELEVATION-MEASURED",
                                     "AZIMUTH-REQUESTED",
                                     "ELEVATION-REQUESTED","AZIMUTH-INITIAL"
                                     ,"ELEVATION-INITIAL","AZIMUTH-FINAL",
                                     "ELEVATION-FINAL")
  ARTICULATION_DEVICE_MODE        = DEPLOYED
END_GROUP                         = RSM_ARTICULATION_STATE_PARMS

/* ARTICULATION DEVICE STATE: ROBOTIC ARM */

GROUP                             = ARM_ARTICULATION_STATE_PARMS
  ARTICULATION_DEVICE_ID          = ARM
  ARTICULATION_DEVICE_ANGLE       = (1.57222 <rad>,-0.277767 <rad>,
                                     -2.81629 <rad>,3.12107 <rad>,
                                     0.593842 <rad>,1.56839 <rad>,
                                     -0.277768 <rad>,-2.82549 <rad>,
                                     3.11656 <rad>,0.593551 <rad>)
  ARTICULATION_DEVICE_ANGLE_NAME  = ("JOINT 1 SHOULDER AZIMUTH",
                                     "JOINT 2 SHOULDER ELEVATION",
                                     "JOINT 3 ELBOW-ENCODER",
                                     "JOINT 4 WRIST-ENCODER",
                                     "JOINT 5 TURRET-ENCODER",
                                     "JOINT 1 SHOULDER AZIMUTH-RESOLVER",
                                     "JOINT 2 SHOULDER ELEVATION-RESOLVER",
                                     "JOINT 3 ELBOW-RESOLVER",
                                     "JOINT 4 WRIST-RESOLVER",
                                     "JOINT 5 TURRET-RESOLVER")
  ARTICULATION_DEVICE_TEMP        = (-8.36346 <degC>,-2.12067 <degC>,
                                     3.53258 <degC>,4.37844 <degC>,
                                     -12.6872 <degC>)
  ARTICULATION_DEVICE_TEMP_NAME   = ("AZIMUTH JOINT","ELEVATION JOINT",
                                     "ELBOW JOINT","WRIST JOINT",
                                     "TURRET JOINT")
  ARTICULATION_DEV_VECTOR         = (-0.138431,0.0139646,0.990274)
  CONTACT_SENSOR_STATE            = "NO CONTACT"
  ARTICULATION_DEV_INSTRUMENT_ID  = "MAHLI"
  ARTICULATION_DEVICE_MODE        = "FREE SPACE"
END_GROUP                         = ARM_ARTICULATION_STATE_PARMS

/* ARTICULATION DEVICE STATE: MOBILITY CHASSIS */

GROUP                             = CHASSIS_ARTICULATION_ST_PARMS
  ARTICULATION_DEVICE_ID          = CHASSIS
  ARTICULATION_DEVICE_NAME        = "MOBILITY CHASSIS"
  ARTICULATION_DEVICE_ANGLE       = (-0 <rad>,4.26106e-05 <rad>,-0 <rad>,
                                     -0 <rad>,3.61023 <rad>,0.905672 <rad>,
                                     -0.0504155 <rad>,-0.0428046 <rad>,
                                     -0.0191213 <rad>,0.0142404 <rad>)
  ARTICULATION_DEVICE_ANGLE_NAME  = ("LEFT FRONT WHEEL","RIGHT FRONT WHEEL"
                                     ,"LEFT REAR WHEEL","RIGHT REAR WHEEL",
```

```
                                  "RSM AZIMUTH","RSM ELEVATION",
                                  "LEFT BOGIE","RIGHT BOGIE",
                                  "LEFT DIFFERENTIAL",
                                  "RIGHT DIFFERENTIAL")
  ARTICULATION_DEVICE_MODE        = DEPLOYED
END_GROUP                         = CHASSIS_ARTICULATION_ST_PARMS

/* ARTICULATION DEVICE STATE: HIGH GAIN ANTENNA */

GROUP                             = HGA_ARTICULATION_STATE_PARMS
  ARTICULATION_DEVICE_ID          = HGA
  ARTICULATION_DEVICE_NAME        = "HIGH GAIN ANTENNA"
  ARTICULATION_DEVICE_ANGLE       = (-3.33928e-05 <rad>,-0.784975 <rad>)
  ARTICULATION_DEVICE_ANGLE_NAME  = ("AZIMUTH","ELEVATION")
  ARTICULATION_DEVICE_MODE        = DEPLOYED
END_GROUP                         = HGA_ARTICULATION_STATE_PARMS

/* OBSERVATION REQUEST */

GROUP                             = OBSERVATION_REQUEST_PARMS
  SOURCE_ID                       = "GROUND COMMANDED"
  MSL:INSTRUMENT_COORD_FRAME_INDEX= 0
  MSL:INSTRUMENT_COORD_FRAME_ID   = "NAV"
  INSTRUMENT_COORDINATE_NAME      = ("AZIMUTH","ELEVATION")
  INSTRUMENT_COORDINATE           = (-0.002 <rad>,0 <rad>)
  MSL:DARK_SPECTRA_MODE           = "NONE"
  GAIN_NUMBER                     = 0
  GROUP_APPLICABILITY_FLAG        = "TRUE"
  MSL:INSTRUMENT_FOCUS_MODE       = "NO_FOCUS"
  MSL:INSTRUMENT_FOCUS_DISTANCE   = 3250
  INSTRUMENT_MODE_ID              = "SPECTRAL_DATA"
  MSL:LASER_MODE                  = "YES"
  OFFSET_MODE_ID                  = "0"
 MSL:N_SHOTS_2_IGNORE            =  0
 MSL:N_SHOTS_2_AVG               =  0
 MSL:N_SHOTS                     =  30
 MSL:SPECTROMETER_CONTROL_BYTE   =  2
 MSL:SPEC_IMAGE_TYPE             =  4
 MSL:SPECTROMETER_SERIAL_CLOCK   =  1
 MSL:SPECTROMETER_SELECT         =  "UNK"
 MSL:ICT_DIVIDER                 =  300
 MSL:IPBC_DIVIDER                =  330
 MSL:START_ROW_UV                =  100
 MSL:STOP_ROW_UV                 =  300
 MSL:START_ROW_VIS               =  75
 MSL:STOP_ROW_VIS                =  275
 MSL:START_ROW_VNIR              =  100
 MSL:STOP_ROW_VNIR               =  300
 MSL:OBS_FROM_LIMIT_SWITCH       =  0
 MSL:STACK_1_LEVEL               =  100
 MSL:STACK_2_LEVEL               =  95
 MSL:STACK_3_LEVEL               =  95
 MSL:STACK_DURATION              =  145
 MSL:CCAM_TIME_BETWEEN_SHOTS          =  244
 MSL:SPEC_AD_CONVERTVNIR         =  5
 MSL:SPEC_AD_CONVERTVIS          =  5
 MSL:SPEC_AD_CONVERTUV           =  5
 MSL:SPEC_VERT_CLK               =  2
END_GROUP                        = OBSERVATION_REQUEST_PARMS

/* INSTRUMENT STATE RESULTS */

GROUP                             = INSTRUMENT_STATE_PARMS
  MSL:SPECIAL_LINE                = 497
  MSL:SPECIAL_SAMPLE              = 532
  MSL:SPECIAL_NAME                = "Location of the LIBS laser spot in the
```

```
                                   RMI image described by the camera
                                  model"
  INSTRUMENT_TEMPERATURE         = (-21.9229 <degC>,-21.9229 <degC>,
                                  -14.8855 <degC>,-14.8855 <degC>,
                                  -14.8855 <degC>,-14.8855 <degC>,
                                  0.392304 <degC>,8.78029 <degC>,
                                  17.877 <degC>,17.877 <degC>)
  INSTRUMENT_TEMPERATURE_NAME    = ("MU_OBOX_TELESCOPE","MU_LASER_IF",
                                  "MU_EBOX_HEATSINK","MU_EBOX_FPGA",
                                  "BU_CCD_VNIR_B","BU_SPEC_B",
                                  "BU_CCD_UV_A","BU_SPEC_A","BU_DEMUX_A",
                                  "BU_DEMUX_B")
  MSL:INSTRUMENT_TEMPERATURE_STATUS= (0,-42,0,-42,-42,-42,0,0,0,-42)
END_GROUP                        = INSTRUMENT_STATE_PARMS

/* DERIVED GEOMETRY DATA ELEMENTS: ROVER FRAME */

GROUP                            = ROVER_DERIVED_GEOMETRY_PARMS
  INSTRUMENT_AZIMUTH             = 25.6447
  INSTRUMENT_ELEVATION           = -39.2685
  REFERENCE_COORD_SYSTEM_NAME    = "ROVER_NAV_FRAME"
  REFERENCE_COORD_SYSTEM_INDEX   = (59,1596,22,54,0,0,180,188)
  SUN_VIEW_DIRECTION             = (0.408491,0.216446,-0.886728)
  SOLAR_AZIMUTH                  = 27.9176 <deg>
  SOLAR_ELEVATION                = 62.4649 <deg>
END_GROUP                        = ROVER_DERIVED_GEOMETRY_PARMS

/* DERIVED GEOMETRY DATA ELEMENTS: SITE FRAME */

GROUP                            = SITE_DERIVED_GEOMETRY_PARMS
  INSTRUMENT_AZIMUTH             = 183.907
  INSTRUMENT_ELEVATION           = -32.3346
  REFERENCE_COORD_SYSTEM_NAME    = "SITE_FRAME"
  REFERENCE_COORD_SYSTEM_INDEX   = (59,1596,22,54,0,0,180,188)
  SOLAR_AZIMUTH                  = 200.302 <deg>
  SOLAR_ELEVATION                = 68.6462 <deg>
  POSITIVE_AZIMUTH_DIRECTION     = CLOCKWISE
END_GROUP                        = SITE_DERIVED_GEOMETRY_PARMS

/* HEADER DESCRIPTION */

OBJECT                          = HEADER
  BYTES                         = 1855
  RECORDS                       = 17
  HEADER_TYPE                   = "TEXT"
END_OBJECT                      = HEADER

/* DATA DESCRIPTION */

OBJECT                          = SPREADSHEET
  FIELD_DELIMITER               = "COMMA"
  FIELDS                        = 33
  ROW_BYTES                     = 660
  ROWS                          = 6144
  DESCRIPTION                   = "Clean calibrated spectra from LIBS in
      radiance as a function of wavelength.  The number of spectra in the
      file depends on the number of shots plus the median and average."

  OBJECT                     = FIELD
    FIELD_NUMBER             = 1
    NAME                     = "WAVELENGTH"
    BYTES                    = 20
    DATA_TYPE                = ASCII_REAL
    UNIT                     = NANOMETER
    DESCRIPTION              = "Calibrated wavelength"
  END_OBJECT                 = FIELD
```

```
OBJECT                       = FIELD
  FIELD_NUMBER               = 2
  NAME                       = "SPECTRA_001"
  BYTES                      = 20
  DATA_TYPE                  = ASCII_REAL
  UNIT                       = RADIANCE
  DESCRIPTION                = "SHOT 1"
END_OBJECT                   = FIELD

OBJECT                       = FIELD
  FIELD_NUMBER               = 3
  NAME                       = "SPECTRA_002"
  BYTES                      = 20
  DATA_TYPE                  = ASCII_REAL
  UNIT                       = RADIANCE
  DESCRIPTION                = "SHOT 2"
END_OBJECT                   = FIELD

OBJECT                       = FIELD
  FIELD_NUMBER               = 4
  NAME                       = "SPECTRA_003"
  BYTES                      = 20
  DATA_TYPE                  = ASCII_REAL
  UNIT                       = RADIANCE
  DESCRIPTION                = "SHOT 3"
END_OBJECT                   = FIELD

OBJECT                       = FIELD
  FIELD_NUMBER               = 5
  NAME                       = "SPECTRA_004"
  BYTES                      = 20
  DATA_TYPE                  = ASCII_REAL
  UNIT                       = RADIANCE
  DESCRIPTION                = "SHOT 4"
END_OBJECT                   = FIELD

OBJECT                       = FIELD
  FIELD_NUMBER               = 6
  NAME                       = "SPECTRA_005"
  BYTES                      = 20
  DATA_TYPE                  = ASCII_REAL
  UNIT                       = RADIANCE
  DESCRIPTION                = "SHOT 5"
END_OBJECT                   = FIELD

OBJECT                       = FIELD
  FIELD_NUMBER               = 7
  NAME                       = "SPECTRA_006"
  BYTES                      = 20
  DATA_TYPE                  = ASCII_REAL
  UNIT                       = RADIANCE
  DESCRIPTION                = "SHOT 6"
END_OBJECT                   = FIELD

OBJECT                       = FIELD
  FIELD_NUMBER               = 8
  NAME                       = "SPECTRA_007"
  BYTES                      = 20
  DATA_TYPE                  = ASCII_REAL
```

```
  UNIT                        = RADIANCE
  DESCRIPTION                 = "SHOT 7"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  FIELD_NUMBER                = 9
  NAME                        = "SPECTRA_008"
  BYTES                       = 20
  DATA_TYPE                   = ASCII_REAL
  UNIT                        = RADIANCE
  DESCRIPTION                 = "SHOT 8"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  FIELD_NUMBER                = 10
  NAME                        = "SPECTRA_009"
  BYTES                       = 20
  DATA_TYPE                   = ASCII_REAL
  UNIT                        = RADIANCE
  DESCRIPTION                 = "SHOT 9"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  FIELD_NUMBER                = 11
  NAME                        = "SPECTRA_010"
  BYTES                       = 20
  DATA_TYPE                   = ASCII_REAL
  UNIT                        = RADIANCE
  DESCRIPTION                 = "SHOT 10"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  FIELD_NUMBER                = 12
  NAME                        = "SPECTRA_011"
  BYTES                       = 20
  DATA_TYPE                   = ASCII_REAL
  UNIT                        = RADIANCE
  DESCRIPTION                 = "SHOT 11"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  FIELD_NUMBER                = 13
  NAME                        = "SPECTRA_012"
  BYTES                       = 20
  DATA_TYPE                   = ASCII_REAL
  UNIT                        = RADIANCE
  DESCRIPTION                 = "SHOT 12"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  FIELD_NUMBER                = 14
  NAME                        = "SPECTRA_013"
  BYTES                       = 20
  DATA_TYPE                   = ASCII_REAL
  UNIT                        = RADIANCE
  DESCRIPTION                 = "SHOT 13"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
```

```
  FIELD_NUMBER            = 15
  NAME                    = "SPECTRA_014"
  BYTES                   = 20
  DATA_TYPE               = ASCII_REAL
  UNIT                    = RADIANCE
  DESCRIPTION             = "SHOT 14"
END_OBJECT                = FIELD

OBJECT                    = FIELD
  FIELD_NUMBER            = 16
  NAME                    = "SPECTRA_015"
  BYTES                   = 20
  DATA_TYPE               = ASCII_REAL
  UNIT                    = RADIANCE
  DESCRIPTION             = "SHOT 15"
END_OBJECT                = FIELD

OBJECT                    = FIELD
  FIELD_NUMBER            = 17
  NAME                    = "SPECTRA_016"
  BYTES                   = 20
  DATA_TYPE               = ASCII_REAL
  UNIT                    = RADIANCE
  DESCRIPTION             = "SHOT 16"
END_OBJECT                = FIELD

OBJECT                    = FIELD
  FIELD_NUMBER            = 18
  NAME                    = "SPECTRA_017"
  BYTES                   = 20
  DATA_TYPE               = ASCII_REAL
  UNIT                    = RADIANCE
  DESCRIPTION             = "SHOT 17"
END_OBJECT                = FIELD

OBJECT                    = FIELD
  FIELD_NUMBER            = 19
  NAME                    = "SPECTRA_018"
  BYTES                   = 20
  DATA_TYPE               = ASCII_REAL
  UNIT                    = RADIANCE
  DESCRIPTION             = "SHOT 18"
END_OBJECT                = FIELD

OBJECT                    = FIELD
  FIELD_NUMBER            = 20
  NAME                    = "SPECTRA_019"
  BYTES                   = 20
  DATA_TYPE               = ASCII_REAL
  UNIT                    = RADIANCE
  DESCRIPTION             = "SHOT 19"
END_OBJECT                = FIELD

OBJECT                    = FIELD
  FIELD_NUMBER            = 21
  NAME                    = "SPECTRA_020"
  BYTES                   = 20
  DATA_TYPE               = ASCII_REAL
  UNIT                    = RADIANCE
  DESCRIPTION             = "SHOT 20"
```

```
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  FIELD_NUMBER                = 22
  NAME                        = "SPECTRA_021"
  BYTES                       = 20
  DATA_TYPE                   = ASCII_REAL
  UNIT                        = RADIANCE
  DESCRIPTION                 = "SHOT 21"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  FIELD_NUMBER                = 23
  NAME                        = "SPECTRA_022"
  BYTES                       = 20
  DATA_TYPE                   = ASCII_REAL
  UNIT                        = RADIANCE
  DESCRIPTION                 = "SHOT 22"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  FIELD_NUMBER                = 24
  NAME                        = "SPECTRA_023"
  BYTES                       = 20
  DATA_TYPE                   = ASCII_REAL
  UNIT                        = RADIANCE
  DESCRIPTION                 = "SHOT 23"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  FIELD_NUMBER                = 25
  NAME                        = "SPECTRA_024"
  BYTES                       = 20
  DATA_TYPE                   = ASCII_REAL
  UNIT                        = RADIANCE
  DESCRIPTION                 = "SHOT 24"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  FIELD_NUMBER                = 26
  NAME                        = "SPECTRA_025"
  BYTES                       = 20
  DATA_TYPE                   = ASCII_REAL
  UNIT                        = RADIANCE
  DESCRIPTION                 = "SHOT 25"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  FIELD_NUMBER                = 27
  NAME                        = "SPECTRA_026"
  BYTES                       = 20
  DATA_TYPE                   = ASCII_REAL
  UNIT                        = RADIANCE
  DESCRIPTION                 = "SHOT 26"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  FIELD_NUMBER                = 28
  NAME                        = "SPECTRA_027"
```

```
    BYTES                      = 20
    DATA_TYPE                  = ASCII_REAL
    UNIT                       = RADIANCE
    DESCRIPTION                = "SHOT 27"
  END_OBJECT                   = FIELD


  OBJECT                       = FIELD
    FIELD_NUMBER               = 29
    NAME                       = "SPECTRA_028"
    BYTES                      = 20
    DATA_TYPE                  = ASCII_REAL
    UNIT                       = RADIANCE
    DESCRIPTION                = "SHOT 28"
  END_OBJECT                   = FIELD


  OBJECT                       = FIELD
    FIELD_NUMBER               = 30
    NAME                       = "SPECTRA_029"
    BYTES                      = 20
    DATA_TYPE                  = ASCII_REAL
    UNIT                       = RADIANCE
    DESCRIPTION                = "SHOT 29"
  END_OBJECT                   = FIELD


  OBJECT                       = FIELD
    FIELD_NUMBER               = 31
    NAME                       = "SPECTRA_030"
    BYTES                      = 20
    DATA_TYPE                  = ASCII_REAL
    UNIT                       = RADIANCE
    DESCRIPTION                = "SHOT 30"
  END_OBJECT                   = FIELD

  OBJECT                       = FIELD
    FIELD_NUMBER               = 32
    NAME                       = "MEDIAN"
    BYTES                      = 20
    DATA_TYPE                  = ASCII_REAL
    UNIT                       = RADIANCE
    DESCRIPTION                = "Median of 30 shots"
  END_OBJECT                   = FIELD

  OBJECT                       = FIELD
    FIELD_NUMBER               = 33
    NAME                       = "AVERAGE"
    BYTES                      = 20
    DATA_TYPE                  = ASCII_REAL
    UNIT                       = RADIANCE
    DESCRIPTION                = "Average of 30 shots"
  END_OBJECT                   = FIELD
END_OBJECT                     = SPREADSHEET

END
```

c) LIBS RDR Product Type “MOC” (Concatenated Products)

```
PDS_VERSION_ID                = PDS3

/* FILE DATA ELEMENTS */

RECORD_TYPE                   = STREAM
RECORD_BYTES                  = 272
```

```
FILE_RECORDS                  = 1094

/* POINTERS TO DATA OBJECTS */

^HEADER                       = ("MOC_1418_1514.CSV", 1)
^SPREADSHEET                  = ("MOC_1418_1514.CSV", 1057 <BYTES>)

DATA_SET_ID                   = "MSL-M-CHEMCAM-LIBS-4/5-RDR-V1.0"
PRODUCT_ID                    = "MOC_1418_1514.CSV"
PRODUCT_VERSION_ID            = "3"
PRODUCT_TYPE                  = "CHEMCAM-MOC"
PRODUCT_CREATION_TIME         = 2017-02-10T14:52:00
START_TIME                    = 2016-08-02T00:52:38.470
STOP_TIME                     = 2016-11-08T22:36:45.979
PLANET_DAY_NUMBER             = "N/A"
RELEASE_ID                    = "0014"
INSTRUMENT_HOST_NAME          = "MARS SCIENCE LABORATORY"
MISSION_NAME                  = "MARS SCIENCE LABORATORY"
INSTRUMENT_ID                 = "CHEMCAM_LIBS"
TARGET_NAME                   = "MARS"
PRODUCER_INSTITUTION_NAME     = "LOS ALAMOS NATIONAL LABORATORY"
DESCRIPTION                   = "Concatenated Multivariant Oxides Composition
  Table for sol range 1418 to 1514"

/* HEADER DESCRIPTION */

OBJECT                            = HEADER
  BYTES                           = 1056
  RECORDS                         = 7
  HEADER_TYPE                     = "TEXT"
  DESCRIPTION                     = "The header records in this file give
     information about the version of the software used and the
     quartiles of the test sets used to calculate RMSE."
END_OBJECT                        = HEADER

/* DATA DESCRIPTION */

OBJECT                            = SPREADSHEET
  FIELD_DELIMITER                 = "COMMA"
  FIELDS                          = 38
  ROW_BYTES                       = 272
  ROWS                            = 1087
  DESCRIPTION                     = "This data product is a ChemCam
    Multivariant Oxides Composition Table and RMSE for the major elements
    SiO2, TiO2, Al2O3, FeOT, MgO, CaO, Na2O, and K2O."

  OBJECT                          = FIELD
    NAME                          = "PRODUCT_NAME"
    FIELD_NUMBER                  = 1
    DATA_TYPE                     = CHARACTER
    BYTES                         = 40
    DESCRIPTION                   = "Filename of the data that produced this
      result."
  END_OBJECT                      = FIELD

  OBJECT                          = FIELD
    NAME                          = "TARGET_NAME"
    FIELD_NUMBER                  = 2
    DATA_TYPE                     = CHARACTER
    BYTES                         = 30
    DESCRIPTION                   = "Target"
  END_OBJECT                      = FIELD

  OBJECT                          = FIELD
    NAME                          = "SIO2"
    FIELD_NUMBER                  = 3
```

```
  DATA_TYPE                     = ASCII_REAL
  BYTES                         = 5
  DESCRIPTION                   = "Predicted composition SiO2"
END_OBJECT                      = FIELD

OBJECT                          = FIELD
  NAME                          = "ERR_PLUS_MINUS"
  FIELD_NUMBER                  = 4
  DATA_TYPE                     = CHARACTER
  BYTES                         = 4
  DESCRIPTION                   = "+/- SiO2 err"
END_OBJECT                      = FIELD

OBJECT                          = FIELD
  NAME                          = "SIO2_RMSEP"
  FIELD_NUMBER                  = 5
  DATA_TYPE                     = ASCII_REAL
  BYTES                         = 5
  DESCRIPTION                   = "Predicted SiO2_RMSEP"
END_OBJECT                      = FIELD

OBJECT                          = FIELD
  NAME                          = "SIO2_SHOTS_STDEV"
  FIELD_NUMBER                  = 6
  DATA_TYPE                     = ASCII_REAL
  BYTES                         = 5
  DESCRIPTION                   = "SiO2 standard deviation"
END_OBJECT                      = FIELD

OBJECT                          = FIELD
  NAME                          = "TIO2"
  FIELD_NUMBER                  = 7
  DATA_TYPE                     = ASCII_REAL
  BYTES                         = 5
  DESCRIPTION                   = "Predicted composition TiO2"
END_OBJECT                      = FIELD

OBJECT                          = FIELD
  NAME                          = "ERR_PLUS_MINUS"
  FIELD_NUMBER                  = 8
  DATA_TYPE                     = CHARACTER
  BYTES                         = 4
  DESCRIPTION                   = "+/- TiO2 err"
END_OBJECT                      = FIELD

OBJECT                          = FIELD
  NAME                          = "TIO2_RMSEP"
  FIELD_NUMBER                  = 9
  DATA_TYPE                     = ASCII_REAL
  BYTES                         = 5
  DESCRIPTION                   = "Predicted TiO2_RMSEP"
END_OBJECT                      = FIELD

OBJECT                          = FIELD
  NAME                          = "TIO2_SHOTS_STDEV"
  FIELD_NUMBER                  = 10
  DATA_TYPE                     = ASCII_REAL
  BYTES                         = 5
  DESCRIPTION                   = "TiO2 standard deviation"
END_OBJECT                      = FIELD

OBJECT                          = FIELD
  NAME                          = "AL2O3"
  FIELD_NUMBER                  = 11
  DATA_TYPE                     = ASCII_REAL
  BYTES                         = 12
```

```
  DESCRIPTION                    = "Predicted composition Al2O3"
END_OBJECT                       = FIELD

OBJECT                           = FIELD
  NAME                           = "ERR_PLUS_MINUS"
  FIELD_NUMBER                   = 12
  DATA_TYPE                      = CHARACTER
  BYTES                          = 4
  DESCRIPTION                    = "+/- Al203 err"
END_OBJECT                       = FIELD

OBJECT                           = FIELD
  NAME                           = "AL203_RMSEP"
  FIELD_NUMBER                   = 13
  DATA_TYPE                      = ASCII_REAL
  BYTES                          = 5
  DESCRIPTION                    = "Predicted Al203_RMSEP"
END_OBJECT                       = FIELD

OBJECT                           = FIELD
  NAME                           = "AL203_SHOTS_STDEV"
  FIELD_NUMBER                   = 14
  DATA_TYPE                      = ASCII_REAL
  BYTES                          = 5
  DESCRIPTION                    = "Al203 standard deviation"
END_OBJECT                       = FIELD

OBJECT                           = FIELD
  NAME                           = "FEOT"
  FIELD_NUMBER                   = 15
  DATA_TYPE                      = ASCII_REAL
  BYTES                          = 12
  DESCRIPTION                    = "Predicted composition FeOT"
END_OBJECT                       = FIELD

OBJECT                           = FIELD
  NAME                           = "ERR_PLUS_MINUS"
  FIELD_NUMBER                   = 16
  DATA_TYPE                      = CHARACTER
  BYTES                          = 4
  DESCRIPTION                    = "+/- FeOT err"
END_OBJECT                       = FIELD

OBJECT                           = FIELD
  NAME                           = "FEOT_RMSEP"
  FIELD_NUMBER                   = 17
  DATA_TYPE                      = ASCII_REAL
  BYTES                          = 5
  DESCRIPTION                    = "Predicted FeOT_RMSEP"
END_OBJECT                       = FIELD

OBJECT                           = FIELD
  NAME                           = "FEOT_SHOTS_STDEV"
  FIELD_NUMBER                   = 18
  DATA_TYPE                      = ASCII_REAL
  BYTES                          = 5
  DESCRIPTION                    = "FeOT standard deviation"
END_OBJECT                       = FIELD

OBJECT                           = FIELD
  NAME                           = "MGO"
  FIELD_NUMBER                   = 19
  DATA_TYPE                      = ASCII_REAL
  BYTES                          = 12
  DESCRIPTION                    = "Predicted composition MgO"
END_OBJECT                       = FIELD
```

```
OBJECT                        = FIELD
  NAME                        = "ERR_PLUS_MINUS"
  FIELD_NUMBER                = 20
  DATA_TYPE                   = CHARACTER
  BYTES                       = 4
  DESCRIPTION                 = "+/- MgO err"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  NAME                        = "MGO_RMSEP"
  FIELD_NUMBER                = 21
  DATA_TYPE                   = ASCII_REAL
  BYTES                       = 5
  DESCRIPTION                 = "Predicted MgO_RMSEP"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  NAME                        = "MGO_SHOTS_STDEV"
  FIELD_NUMBER                = 22
  DATA_TYPE                   = ASCII_REAL
  BYTES                       = 5
  DESCRIPTION                 = "MgO standard deviation"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  NAME                        = "CAO"
  FIELD_NUMBER                = 23
  DATA_TYPE                   = ASCII_REAL
  BYTES                       = 12
  DESCRIPTION                 = "Predicted composition CaO"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  NAME                        = "ERR_PLUS_MINUS"
  FIELD_NUMBER                = 24
  DATA_TYPE                   = CHARACTER
  BYTES                       = 4
  DESCRIPTION                 = "+/- CaO err"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  NAME                        = "CAO_RMSEP"
  FIELD_NUMBER                = 25
  DATA_TYPE                   = ASCII_REAL
  BYTES                       = 5
  DESCRIPTION                 = "Predicted CaO_RMSEP"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  NAME                        = "CAO_SHOTS_STDEV"
  FIELD_NUMBER                = 26
  DATA_TYPE                   = ASCII_REAL
  BYTES                       = 5
  DESCRIPTION                 = "CaO standard deviation"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
  NAME                        = "NA2O"
  FIELD_NUMBER                = 27
  DATA_TYPE                   = ASCII_REAL
  BYTES                       = 12
  DESCRIPTION                 = "Predicted composition Na2O"
END_OBJECT                    = FIELD

OBJECT                        = FIELD
```

```
    NAME                          = "ERR_PLUS_MINUS"
    FIELD_NUMBER                  = 28
    DATA_TYPE                     = CHARACTER
    BYTES                         = 4
    DESCRIPTION                   = "+/- Na2O err"
  END_OBJECT                      = FIELD

  OBJECT                          = FIELD
    NAME                          = "NA2O_RMSEP"
    FIELD_NUMBER                  = 29
    DATA_TYPE                     = ASCII_REAL
    BYTES                         = 5
    DESCRIPTION                   = "Predicted Na2O_RMSEP"
  END_OBJECT                      = FIELD

  OBJECT                          = FIELD
    NAME                          = "NA2O_SHOTS_STDEV"
    FIELD_NUMBER                  = 30
    DATA_TYPE                     = ASCII_REAL
    BYTES                         = 5
    DESCRIPTION                   = "Na2O standard deviation"
  END_OBJECT                      = FIELD

  OBJECT                          = FIELD
    NAME                          = "K2O"
    FIELD_NUMBER                  = 31
    DATA_TYPE                     = ASCII_REAL
    BYTES                         = 12
    DESCRIPTION                   = "Predicted composition K2O"
  END_OBJECT                      = FIELD

  OBJECT                          = FIELD
    NAME                          = "ERR_PLUS_MINUS"
    FIELD_NUMBER                  = 32
    DATA_TYPE                     = CHARACTER
    BYTES                         = 4
    DESCRIPTION                   = "+/- K2O err"
  END_OBJECT                      = FIELD

  OBJECT                          = FIELD
    NAME                          = "K2O_RMSEP"
    FIELD_NUMBER                  = 33
    DATA_TYPE                     = ASCII_REAL
    BYTES                         = 5
    DESCRIPTION                   = "Predicted K2O_RMSEP"
  END_OBJECT                      = FIELD

  OBJECT                          = FIELD
    NAME                          = "K2O_SHOTS_STDEV"
    FIELD_NUMBER                  = 34
    DATA_TYPE                     = ASCII_REAL
    BYTES                         = 5
    DESCRIPTION                   = "K2O standard deviation"
  END_OBJECT                      = FIELD

  OBJECT                          = FIELD
    NAME                          = "SUM_OF_OXIDES"
    FIELD_NUMBER                  = 35
    DATA_TYPE                     = ASCII_REAL
    BYTES                         = 10
    DESCRIPTION                   = "The sum of the major oxides"
  END_OBJECT                      = FIELD

  OBJECT                          = FIELD
    NAME                          = "DISTANCE_TO_TARGET"
    FIELD_NUMBER                  = 36
```

```
    DATA_TYPE                    = ASCII_REAL
    BYTES                        = 12
    DESCRIPTION                  = "Distance (m)"
  END_OBJECT                     = FIELD

  OBJECT                         = FIELD
    NAME                         = "LASER_POWER"
    FIELD_NUMBER                 = 37
    DATA_TYPE                    = CHARACTER
    BYTES                        = 16
    DESCRIPTION                  = "Laser power all three spectrometers"
  END_OBJECT                     = FIELD

  OBJECT                         = FIELD
    NAME                         = "SPECTRUM_TOTAL"
    FIELD_NUMBER                 = 38
    DATA_TYPE                    = ASCII_REAL
    BYTES                        = 12
    DESCRIPTION                  = "The total integrated signal before
       normalization is applied, but after the mask is applied."
  END_OBJECT                     = FIELD

END_OBJECT                     = SPREADSHEET

END
```

APPENDIX D – ".FMT" Files for ChemCam DPOs

a) "CCAM_RMI_HEADER"

```
/* CCAM_RMI_HEADER_V4.FMT.                                          */

  OBJECT                = COLUMN
    COLUMN_NUMBER       = 1
    NAME                = "MU_HDR_0"
    START_BYTE          = 1
    BYTES               = 2
    DATA_TYPE           = MSB_INTEGER
    DESCRIPTION         = "0EB"
  END_OBJECT            = COLUMN

  OBJECT                = COLUMN
    COLUMN_NUMBER       = 2
    NAME                = "MU_HDR_1"
    START_BYTE          = 3
    BYTES               = 2
    DATA_TYPE           = MSB_INTEGER
    DESCRIPTION         = "090"
  END_OBJECT            = COLUMN

  OBJECT                = COLUMN
    COLUMN_NUMBER       = 3
    NAME                = "MU_HDR_2"
    START_BYTE          = 5
    BYTES               = 2
    DATA_TYPE           = MSB_INTEGER
    DESCRIPTION         = "000"
  END_OBJECT            = COLUMN

  OBJECT                = COLUMN
    COLUMN_NUMBER       = 4
    NAME                = "MU_HDR_3"
    START_BYTE          = 7
    BYTES               = 2
    DATA_TYPE           = MSB_INTEGER
    DESCRIPTION         = "022"
  END_OBJECT            = COLUMN

  OBJECT                = COLUMN
    COLUMN_NUMBER       = 5
    NAME                = "MU_HDR_4"
    START_BYTE          = 9
    BYTES               = 2
    DATA_TYPE           = MSB_INTEGER
    DESCRIPTION         = "08C"
  END_OBJECT            = COLUMN

  OBJECT                = COLUMN
    COLUMN_NUMBER       = 6
    NAME                = "MU_HDR_5"
    START_BYTE          = 11
```

```
    BYTES             = 2
    DATA_TYPE         = MSB_INTEGER
    DESCRIPTION       = "00A"
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 7
    NAME              = "MU_HDR_6"
    START_BYTE        = 13
    BYTES             = 2
    DATA_TYPE         = MSB_INTEGER
    DESCRIPTION       = "0C9"
  END_OBJECT          = COLUMN

/* End of format file CCAM_RMI_HEADER_V4.FMT. */
```

b) "CCAM_RMI_FOOTER"

```
/* CCAM_RMI_FOOTER.FMT.                                              */

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 1
    NAME              = "CMD_ID"
    START_BYTE        = 1
    BYTES             = 2
    DATA_TYPE         = MSB_INTEGER
    DESCRIPTION       = "Command ID"
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 2
    NAME              = "EXPOSURE"
    START_BYTE        = 3
    BYTES             = 4
    DATA_TYPE         = MSB_INTEGER
    DESCRIPTION       = "Exposure"
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 3
    NAME              = "AD_OFFSET"
    START_BYTE        = 7
    BYTES             = 2
    DATA_TYPE         = MSB_INTEGER
    DESCRIPTION       = "AD offset"
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 9
    NAME              = "ADC_GAIN"
    START_BYTE        = 9
    BYTES             = 2
    DATA_TYPE         = MSB_INTEGER
    DESCRIPTION       = "ADC gain"
  END_OBJECT          = COLUMN
```

```
  OBJECT              = COLUMN
    COLUMN_NUMBER     = 5
    NAME              = "CS_RMI"
    START_BYTE        = 11
    BYTES             = 2
    DATA_TYPE         = MSB_INTEGER
    DESCRIPTION       = "CS RMI"
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 6
    NAME              = "IMAGE_N"
    START_BYTE        = 13
    BYTES             = 2
    DATA_TYPE         = MSB_INTEGER
    DESCRIPTION       = "imageN"
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 7
    NAME              = "FOOTER_CHECKSUM"
    START_BYTE        = 15
    BYTES             = 4
    DATA_TYPE         = MSB_INTEGER
    DESCRIPTION       = "Checksum footer"
  END_OBJECT          = COLUMN

/* End of format file CCAM_RMI_FOOTER_V4.FMT. */
```

c) "CCAM_RMI_ANCILLARY"

```
/* CCAM_RMI_ANCILLARY_V4.FMT.                                           */

  /* This format file defines the 20 ancillary temps data columns.   */
  ^STRUCTURE                     = "CCAM_ANCILLARY_TMPS_V4.FMT"

  /* This format file defines the 15 command arguments and    */
  /* 25 command Parameters                                    */
  ^STRUCTURE                     = "CCAM_RMI_CMD_ARG_PARAMS_V4.FMT"

/* End of format file CCAM_RMI_ANCILLARY_V4.FMT */
```

d) "CCAM_RMI_CMD_ARG_PARMS"

```
/* CCAM_RMI_CMD_ARG_PARMS_V4.FMT                                             */

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 21
    NAME              = "FRAME_ID"
    START_BYTE        = 81
    BYTES             = 4
    DATA_TYPE         = MSB_INTEGER
    DESCRIPTION       = "FRAME in which the RSM pointing coords are specified.
                         This argument is irrelevant if coord_type = JOINTS_*."
  END_OBJECT          = COLUMN
```

```
OBJECT              = COLUMN
  COLUMN_NUMBER     = 22
  NAME              = "FRAME_INDEX"
  START_BYTE        = 85
  BYTES             = 2
  DATA_TYPE         = MSB_INTEGER
  DESCRIPTION       = "Index of the chosen frame(n/a for many frame_IDs)."
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 23
  NAME              = "COORD_TYPE"
  START_BYTE        = 87
  BYTES             = 4
  DATA_TYPE         = MSB_INTEGER
  DESCRIPTION       = "Types of coordinates specified by coord1, coord2,
                       coord3 args."
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 24
  NAME              = "COORD_1"
  START_BYTE        = 91
  BYTES             = 4
  DATA_TYPE         = IEEE_REAL
  DESCRIPTION       = "X or AZ coordinate for pointing in frame"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 25
  NAME              = "COORD_2"
  START_BYTE        = 95
  BYTES             = 4
  DATA_TYPE         = IEEE_REAL
  DESCRIPTION       = "Y or EL coordinate for pointing in frame"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 26
  NAME              = "COORD_3"
  START_BYTE        = 99
  BYTES             = 4
  DATA_TYPE         = IEEE_REAL
  DESCRIPTION       = "Z or N/A coordinate for pointing in frame"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 27
  NAME              = "FOCUS"
  START_BYTE        = 103
  BYTES             = 4
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Focus algorithm:
                  0 = NO_FOCUS - Dosen't move focus (others mark CCAM unsafe)
                  1 = BASELINE - use CWL to find optimal focus position
                  3 = MANUAL - positions focus based on range argument
```

```
                 4 = AF_OFFSET - applies RMI offset
                     from last autofocus solution"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 28
  NAME             = "RANGE"
  START_BYTE       = 107
  BYTES            = 2
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Distance to target for MANUAL focus, seed for BASELINE"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 29
  NAME             = "EXPOSURE_TYPE"
  START_BYTE       = 109
  BYTES            = 4
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "AUTO; MANUAL MANUAL uses exposure time argument"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 30
  NAME             = "EXPOSURE_TIME"
  START_BYTE       = 113
  BYTES            = 2
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Exposure time for MANUAL exposure,
                      seed time for BASELINE"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 31
  NAME             = "START_C_PIXEL"
  START_BYTE       = 115
  BYTES            = 2
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Starting coord for sub-framing (ROI)"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 32
  NAME             = "START_R_PIXEL"
  START_BYTE       = 117
  BYTES            = 2
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Starting coord for sub-framing (ROI)"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 33
  NAME             = "C_HEIGHT"
  START_BYTE       = 119
  BYTES            = 2
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Vertical size for sub-framing (ROI)"
```

```
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 34
  NAME             = "R_HEIGHT"
  START_BYTE       = 121
  BYTES            = 2
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Horizontal size for sub-framing (ROI)"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 35
  NAME             = "COMPRESSION"
  START_BYTE       = 123
  BYTES            = 4
  DATA_TYPE        = MSB_INTEGER
  DESCRIPTION      = "Compression to apply to the sub-framed image:
                      0 = No RMI compression
                      1 = LOCO
                      2 = ICER_1BPP"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 36
  NAME             = "LINK_TO_USE"
  START_BYTE       = 127
  BYTES            = 1
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "CCMU-CCBU link:
                      0 = synchronous
                      1 = asynchronous
                      currently unavailable for RMI image data transfer"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 37
  NAME             = "UPPER_THRESHOLD"
  START_BYTE       = 128
  BYTES            = 2
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Pixel values outside this range are not included in
                      good pixel count"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 38
  NAME             = "LOWER_THRESHOLD"
  START_BYTE       = 130
  BYTES            = 2
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Pixel values outside this range are not included in
                      good pixel count"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 39
```

```
    NAME              = "START_IMAGE_ID"
    START_BYTE        = 132
    BYTES             = 1
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    DESCRIPTION       = "Selectable in case the SRAM in MU goes bad."
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 40
    NAME              = "AD_OFFSET"
    START_BYTE        = 133
    BYTES             = 1
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    DESCRIPTION       = "Controls an analog offset in CCMU"
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 41
    NAME              = "AD_GAIN"
    START_BYTE        = 134
    BYTES             = 1
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    DESCRIPTION       = "Controls gain value in CCMU"
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 42
    NAME              = "CCD_CLEAN_COUNT"
    START_BYTE        = 135
    BYTES             = 1
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    DESCRIPTION       = "N images before transferring image to FPGA"
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 43
    NAME              = "OBS_FROM_LIMIT_SWITCH"
    START_BYTE        = 136
    BYTES             = 1
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    DESCRIPTION       = "0 = default
                         1 = brings focus stage to limit switch 1st to
                             initiliaze position"
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 44
    NAME              = "THUMBNAIL_SIZE"
    START_BYTE        = 137
    BYTES             = 4
    DATA_TYPE         = MSB_INTEGER
    DESCRIPTION       = "0 = NO_THUMB turns off thumbnail creation
                         1 = THUMB_64 creates 64x64 down sample"
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 45
```

```
  NAME             = "THUMBNAIL_COMPRESSION"
  START_BYTE       = 141
  BYTES            = 4
  DATA_TYPE        = MSB_INTEGER
  DESCRIPTION      = "0 = NO Compression
                      1 = LOCO
                      2 = ICER_1BPP"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 46
  NAME             = "THUMBNAIl_PRIORITY"
  START_BYTE       = 145
  BYTES            = 1
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Data priority of the thumbnail DP if any 0 = default"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 47
  NAME             = "RMI_REF_PIX_PRIORITY"
  START_BYTE       = 146
  BYTES            = 1
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Priority for the reference pixels data product"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 48
  NAME             = "RMI_REF_PIX_DP"
  START_BYTE       = 147
  BYTES            = 4
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Indicates if reference pixels should be collected and
                      packaged into a Data Product:
                      0 = FALSE - (default)
                      1 = TRUE"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 49
  NAME             = "THUMB_ICER_SEGMENTS"
  START_BYTE       = 151
  BYTES            = 1
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Thumbnail icer segments"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 50
  NAME             = "THUMB_ICER_DECOMPS"
  START_BYTE       = 152
  BYTES            = 1
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Thumbnail icer decomps"
END_OBJECT         = COLUMN
```

```
OBJECT              = COLUMN
  COLUMN_NUMBER     = 51
  NAME              = "THUMB_ICER_MIN_LOSS"
  START_BYTE        = 153
  BYTES             = 1
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Thumbnail icer minimum loss"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 52
  NAME              = "THUMB_ICER_BPP"
  START_BYTE        = 154
  BYTES             = 4
  DATA_TYPE         = MSB_INTEGER
  DESCRIPTION       = "Thumbnail icer bits per pixel"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 53
  NAME              = "THUMB_LOCO_SEGMENTS"
  START_BYTE        = 158
  BYTES             = 1
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Thumbnail LOCO segments"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 54
  NAME              = "THUMB_LOCO_PIXEL_SIZE"
  START_BYTE        = 159
  BYTES             = 4
  DATA_TYPE         = MSB_INTEGER
  DESCRIPTION       = "Thumbnail LOCO pixel size"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 55
  NAME              = "ICER_SEGMENTS"
  START_BYTE        = 163
  BYTES             = 1
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "ICER segments"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 56
  NAME              = "ICER_DECOMPS"
  START_BYTE        = 164
  BYTES             = 1
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "ICER decomps"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 57
  NAME              = "ICER_MIN_LOSS"
```

```
    START_BYTE       = 165
    BYTES            = 1
    DATA_TYPE        = MSB_UNSIGNED_INTEGER
    DESCRIPTION      = "ICER Minimum Loss"
  END_OBJECT         = COLUMN

  OBJECT             = COLUMN
    COLUMN_NUMBER    = 58
    NAME             = "ICER_BPP"
    START_BYTE       = 166
    BYTES            = 4
    DATA_TYPE        = MSB_INTEGER
    DESCRIPTION      = "ICER Bits per pixel"
  END_OBJECT         = COLUMN

  OBJECT             = COLUMN
    COLUMN_NUMBER    = 59
    NAME             = "LOCO_SEGMENTS"
    START_BYTE       = 170
    BYTES            = 1
    DATA_TYPE        = MSB_UNSIGNED_INTEGER
    DESCRIPTION      = "LOCO segments"
  END_OBJECT         = COLUMN

  OBJECT             = COLUMN
    COLUMN_NUMBER    = 60
    NAME             = "LOCO_PIXEL_SIZE"
    START_BYTE       = 171
    BYTES            = 4
    DATA_TYPE        = MSB_INTEGER
    DESCRIPTION      = "LOCO Pixel size"
  END_OBJECT         = COLUMN

/* End of format file CCAM_RMI_CMD_ARG_PARMS_V4.FMT */
```

e) "CCAM_LIBS_HEADER" (1st of 2)

```
/* CCAM_LIBS_HEADER_V2.FMT.                                     */

  OBJECT             = COLUMN
    COLUMN_NUMBER    = 1
    NAME             = "BYTECOUNT"
    START_BYTE       = 1
    BYTES            = 4
    DATA_TYPE        = MSB_UNSIGNED_INTEGER
    DESCRIPTION      = "Bytes to follow"
  END_OBJECT         = COLUMN

  OBJECT             = COLUMN
    COLUMN_NUMBER    = 2
    NAME             = "RCECONTROL"
    START_BYTE       = 5
    BYTES            = 4
    DATA_TYPE        = MSB_UNSIGNED_INTEGER
    DESCRIPTION      = "RCEControl:
                           contains: (8bits) Opcode 78 for spectra
```

```
            (2bits) Error Control Type Flag
  00 = reserved
  01 =  The CRC error control algorithm is applied
  10 = The Checksum error control Algorithm applied
  11 = No error Control"
            (1 bit) Data Present
  0 = nodata
  1 = data"
            (21 bits) Status flags (as follows)
bit 1 - command Reply Flag
         (0- not cmd_reply, 1 - cmd_reply)
bit 2 - notSafe
         ( 0 - safe
           1 - Bu doesn't know that MU mortors
             are in sun-safe position)
bit 3 - BootSource
         (Which memory was used for boot
          0 - Init0 PROMO
          1 - Init1 PROM1)
bit 4 - CommSide
         (Talking with which RCE?
          0 - RCE A or none
          1 - RCE B)
bit 5 - CWL_heater_notOn
         (CWL heating loop algorithm off
          0 - on
          1 - off)
bit 6 - Amp_heater_notOn
         ( Amp heating loop algorithm off
          0 - on
          1 - off)
bit 7 - OSC_heater_notOn
         ( Osc heating loop algorithm off
          0 - on
          1 - off)
bit 8 - RMI_notOn
         ( RMI is not Powered
          0 - on
          1 - off)
bit 9 - RMIdata_notOK
         ( RMI off or comm with RMI not OK
          0 - OK
          1 - not OK, =1
              when RMI if off)
bit 10 - Spectrometer_notOK
         ( not used should be 0)
bit 11 - LVPS_notOn
         ( Low Voltage (spectrometer)power supply
           0 - on
           1 - off )
bit 12 - MAST_notOK
         ( Bad communication with the mast unit
           0 - OK
           1 - not OK)
bit 13 - LIBS_HV_notOn
         ( High voltage to LIBS not on
           0 - on
```

```
                                 1 - not on)
                       bit 14 - LIBS_notReady
                                (LIBS not in operating temperature range
                                 0 - ready
                                 1 - not ready)
                       bit 15 - CWL_notReady
                                (CWL not in operating temperature range
                                 0 - ready
                                 1 - not ready)
                       bit 16 - SelfTestFailed
                                 (not used should be 0)
                       bit 17:21 -
                                 0 0000 - reserved
                                 0 0001 - First time command
                                 0 0010 - Retry command (cmd reply frame)
"
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 3
    NAME              = "BYTECOUNT"
    START_BYTE        = 9
    BYTES             = 4
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    DESCRIPTION       = "Number of bytes "
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 4
    NAME              = "DATAID"
    START_BYTE        = 13
    BYTES             = 1
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    DESCRIPTION       = "id byte 0xdd for LIBS spectra"
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 5
    NAME              = "HEAD2"
    START_BYTE        = 14
    BYTES             = 1
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    DESCRIPTION       = "Used internally by DPU"
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 6
    NAME              = "SPECTYPE"
    START_BYTE        = 15
    BYTES             = 1
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    DESCRIPTION       = "Spectra type
                        0 = 1D dark average
                        1 = 1D dark single
                        2 = 1D passive
                        3 = 1D average spectra
                        4 = 1D single spectra
```

```
                      5 = 2D diagnostic"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 7
  NAME             = "HEAD4"
  START_BYTE       = 16
  BYTES            = 1
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Upper nibble clk & tmp enable
                      Lower Nibble spects selected"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 8
  NAME             = "DPBYTECOUNT"
  START_BYTE       = 17
  BYTES            = 4
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Number of bytes following in DataProduct"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 9
  NAME             = "SSTART"
  START_BYTE       = 21
  BYTES            = 4
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Start Time"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 10
  NAME             = "SEND"
  START_BYTE       = 25
  BYTES            = 4
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "End Time"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 11
  NAME             = "SNSHOTS"
  START_BYTE       = 29
  BYTES            = 1
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Total Shots"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 12
  NAME             = "SNAVERGED"
  START_BYTE       = 30
  BYTES            = 1
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Shots Averaged"
END_OBJECT         = COLUMN
```

```
  OBJECT                = COLUMN
    COLUMN_NUMBER       = 13
    NAME                = "SSKIPPED"
    START_BYTE          = 31
    BYTES               = 1
    DATA_TYPE           = MSB_UNSIGNED_INTEGER
    DESCRIPTION         = "Shots Skipped"
  END_OBJECT            = COLUMN

  OBJECT                = COLUMN
    COLUMN_NUMBER       = 14
    NAME                = "SPARE"
    START_BYTE          = 32
    BYTES               = 1
    DATA_TYPE           = MSB_UNSIGNED_INTEGER
    DESCRIPTION         = "Spare"
  END_OBJECT            = COLUMN

/* End of format file CCAM_LIBS_HEADER_V2.FMT. */
```

f) "CCAM_LIBS_HEADER" (2nd of 2)

```
/* CCAM_LIBS_HEADER_V4.FMT.                                        */

  OBJECT                = COLUMN
    COLUMN_NUMBER       = 1
    NAME                = "BYTECOUNT"
    START_BYTE          = 1
    BYTES               = 4
    DATA_TYPE           = MSB_UNSIGNED_INTEGER
    DESCRIPTION         = "Bytes to follow"
  END_OBJECT            = COLUMN

  OBJECT                = COLUMN
    COLUMN_NUMBER       = 2
    NAME                = "RCECONTROL"
    START_BYTE          = 5
    BYTES               = 4
    DATA_TYPE           = MSB_UNSIGNED_INTEGER
    DESCRIPTION         = "RCEControl:
                            contains: (8bits) Opcode 78 for spectra
                                      (2bits) Error Control Type Flag
                            00 = reserved
                            01 =  The CRC error control algorithm is applied
                            10 = The Checksum error control Algorithm applied
                            11 = No error Control
                                      (1 bit) Data Present
                            0 = nodata
                            1 = data
                                      (21 bits) Status flags (as follows)
                          bit 1 - command Reply Flag
                                   (0- not cmd_reply, 1 - cmd_reply)
                          bit 2 - notSafe
                                   ( 0 - safe
                                     1 - Bu doesn't know that MU mortors
```

```
              are in sun-safe position)
bit 3 - BootSource
         (Which memory was used for boot
          0 - Init0 PROMO
          1 - Init1 PROM1)
bit 4 - CommSide
         (Talking with which RCE?
          0 - RCE A or none
          1 - RCE B)
bit 5 - CWL_heater_notOn
         (CWL heating loop algorithm off
          0 - on
          1 - off)
bit 6 - Amp_heater_notOn
         ( Amp heating loop algorithm off
          0 - on
          1 - off)
bit 7 - OSC_heater_notOn
         ( Osc heating loop algorithm off
          0 - on
          1 - off)
bit 8 - RMI_notOn
         ( RMI is not Powered
          0 - on
          1 - off)
bit 9 - RMIdata_notOK
         ( RMI off or comm with RMI not OK
          0 - OK
          1 - not OK, =1
              when RMI if off)
bit 10 - Spectrometer_notOK
         ( not used should be 0)
bit 11 - LVPS_notOn
         ( Low Voltage (spectrometer)power supply
           0 - on
           1 - off )
bit 12 - MAST_notOK
         ( Bad communication with the mast unit
           0 - OK
           1 - not OK)
bit 13 - LIBS_HV_notOn
         ( High voltage to LIBS not on
           0 - on
           1 - not on)
bit 14 - LIBS_notReady
         (LIBS not in operating temperature range
           0 - ready
           1 - not ready)
bit 15 - CWL_notReady
         (CWL not in operating temperature range
           0 - ready
           1 - not ready)
bit 16 - SelfTestFailed
          (not used should be 0)
bit 17:21 -
          0 0000 - reserved
          0 0001 - First time command
```

```
                                        0 0010 - Retry command (cmd reply frame)
"
  END_OBJECT            = COLUMN

  OBJECT                = COLUMN
    COLUMN_NUMBER       = 3
    NAME                = "BYTECOUNT"
    START_BYTE          = 9
    BYTES               = 4
    DATA_TYPE           = MSB_UNSIGNED_INTEGER
    DESCRIPTION         = "Number of bytes "
  END_OBJECT            = COLUMN

  OBJECT                = COLUMN
    COLUMN_NUMBER       = 4
    NAME                = "DATAID"
    START_BYTE          = 13
    BYTES               = 1
    DATA_TYPE           = MSB_UNSIGNED_INTEGER
    DESCRIPTION         = "id byte 0xdd for LIBS spectra"
  END_OBJECT            = COLUMN

  OBJECT                = COLUMN
    COLUMN_NUMBER       = 5
    NAME                = "HEAD2"
    START_BYTE          = 14
    BYTES               = 1
    DATA_TYPE           = MSB_UNSIGNED_INTEGER
    DESCRIPTION         = "Used internally by DPU"
  END_OBJECT            = COLUMN

  OBJECT                = COLUMN
    COLUMN_NUMBER       = 6
    NAME                = "SPECTYPE"
    START_BYTE          = 15
    BYTES               = 1
    DATA_TYPE           = MSB_UNSIGNED_INTEGER
    DESCRIPTION         = "Spectra type
                            0 = 1D dark average
                            1 = 1D dark single
                            2 = 1D passive
                            3 = 1D average spectra
                            4 = 1D single spectra
                            5 = 2D diagnostic"
  END_OBJECT            = COLUMN

  OBJECT                = COLUMN
    COLUMN_NUMBER       = 7
    NAME                = "HEAD4"
    START_BYTE          = 16
    BYTES               = 1
    DATA_TYPE           = MSB_UNSIGNED_INTEGER
    DESCRIPTION         = "Upper nibble clk & tmp enable
                           Lower Nibble spects selected"
  END_OBJECT            = COLUMN

  OBJECT                = COLUMN
```

```
  COLUMN_NUMBER    = 8
  NAME             = "DPBYTECOUNT"
  START_BYTE       = 17
  BYTES            = 4
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Number of bytes following in DataProduct"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 9
  NAME             = "SSTART"
  START_BYTE       = 21
  BYTES            = 4
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Start Time"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 10
  NAME             = "SEND"
  START_BYTE       = 25
  BYTES            = 4
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "End Time"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 11
  NAME             = "SNSHOTS"
  START_BYTE       = 29
  BYTES            = 1
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Total Shots"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 12
  NAME             = "SNAVERGED"
  START_BYTE       = 30
  BYTES            = 1
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Shots Averaged"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 13
  NAME             = "SSKIPPED"
  START_BYTE       = 31
  BYTES            = 1
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Shots Skipped"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 14
  NAME             = "SPARE"
  START_BYTE       = 32
```

```
    BYTES              = 1
    DATA_TYPE          = MSB_UNSIGNED_INTEGER
    DESCRIPTION        = "Spare"
  END_OBJECT           = COLUMN

/* End of format file CCAM_LIBS_HEADER_V4.FMT. */
```

g) “CCAM_LIBS_ANCILLARY” (1st of 2)

```
/* CCAM_LIBS_ANCILLARY_V2.FMT.                                            */

  /* This format file defines the 16 ancillary TEMP data columns.       */
  /* 64bytes                                                        */
  ^STRUCTURE                       = "CCAM_ANCILLARY_TMPS_V2.FMT"

  /* This format file defines the 14 command arguments and    */
  /* 22 command Parameters                                    */
  /* 82 bytes                                                 */
  ^STRUCTURE                       = "CCAM_LIBS_CMD_ARG_PARAMS_V2.FMT"

/* End of format file CCAM_LIBS_ANCILLARY_V2.FMT */
```

h) “CCAM_LIBS_ANCILLARY” (2nd of 2)

```
/* CCAM_LIBS_ANCILLARY_V4.FMT.                                            */

  /* This format file defines the 20 ancillary TEMP data columns.       */
  /* 80bytes                                                        */
  ^STRUCTURE                       = "CCAM_ANCILLARY_TMPS_V4.FMT"

  /* This format file defines the 16 command arguments and    */
  /* 22 command Parameters                                    */
  /* 90 bytes                                                 */
  ^STRUCTURE                       = "CCAM_LIBS_CMD_ARG_PARAMS_V4.FMT"

/* End of format file CCAM_LIBS_ANCILLARY_V4.FMT */
```

i) “CCAM_LIBS_CMD_ARG_PARAMS” (1st of 2)

```
/* CCAM_LIBS_CMD_ARG_PARAMS_V2.FMT                                              */

  OBJECT               = COLUMN
    COLUMN_NUMBER      = 18
    NAME               = "FRAME_ID"
    START_BYTE         = 69
    BYTES              = 4
    DATA_TYPE          = MSB_INTEGER
    DESCRIPTION        = "FRAME in which the RSM pointing coords are specified.
                          This argument is irrelevant if
                          coord_type = JOINTS_*."
  END_OBJECT           = COLUMN

  OBJECT               = COLUMN
    COLUMN_NUMBER      = 19
```

```
  NAME              = "FRAME_INDEX"
  START_BYTE        = 73
  BYTES             = 2
  DATA_TYPE         = MSB_INTEGER
  DESCRIPTION       = "Index of the chosen frame(n/a for many frame_IDs)."
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 20
  NAME              = "COORD_TYPE"
  START_BYTE        = 75
  BYTES             = 4
  DATA_TYPE         = MSB_INTEGER
  DESCRIPTION       = "Types of coordinates specified by coord1, coord2,
                       coord3 args."
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 21
  NAME              = "COORD_1"
  START_BYTE        = 79
  BYTES             = 4
  DATA_TYPE         = IEEE_REAL
  DESCRIPTION       = "X or AZ coordinate for pointing in frame"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 22
  NAME              = "COORD_2"
  START_BYTE        = 83
  BYTES             = 4
  DATA_TYPE         = IEEE_REAL
  DESCRIPTION       = "Y or EL coordinate for pointing in frame"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 23
  NAME              = "COORD_3"
  START_BYTE        = 87
  BYTES             = 4
  DATA_TYPE         = IEEE_REAL
  DESCRIPTION       = "Z or N/A coordinate for pointing in frame"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 24
  NAME              = "FOCUS"
  START_BYTE        = 91
  BYTES             = 4
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Focus algorithm:
                  0 = NO_FOCUS - Dosen't move focus (others mark CCAM unsafe)
                  1 = BASELINE - use CWL to find optimal focus position
                  3 = MANUAL - positions focus based on range argument
                  4 = AF_OFFSET - applies LIBS offset from last autofocus
                      solution"
END_OBJECT          = COLUMN
```

```
OBJECT              = COLUMN
  COLUMN_NUMBER     = 25
  NAME              = "RANGE"
  START_BYTE        = 95
  BYTES             = 2
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Distance to target for MANUAL focus, seed for BASELINE"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 26
  NAME              = "DATA"
  START_BYTE        = 97
  BYTES             = 4
  DATA_TYPE         = MSB_INTEGER
  DESCRIPTION       = "Data Present"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 27
  NAME              = "PRE_POST_DARKSPECTRA"
  START_BYTE        = 101
  BYTES             = 4
  DATA_TYPE         = MSB_INTEGER
  DESCRIPTION       = "pre and post dark spectra
                       0= PRE_ONLY
                       1= POST_ONLY
                       2=PRE_AND_POST
                       3=NONE"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 28
  NAME              = "USE_LASER"
  START_BYTE        = 105
  BYTES             = 4
  DATA_TYPE         = MSB_INTEGER
  DESCRIPTION       = "0= no laser
                       1= laserfired"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 29
  NAME              = "NSHOTS_2_IGNOR"
  START_BYTE        = 109
  BYTES             = 1
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Number of shots to ignor"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 30
  NAME              = "NSHOTS_2_AVG"
  START_BYTE        = 110
  BYTES             = 1
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
```

```
  DESCRIPTION       = "Number of shots to Average"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 31
  NAME              = "NSHOTS"
  START_BYTE        = 111
  BYTES             = 1
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Number of shots"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 32
  NAME              = "SPECT_CONTROL_BYTE"
  START_BYTE        = 112
  BYTES             = 1
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "0x2 - enable spectometer clocks
                       0x4 - enable temperature checks
                       always enable both except when troubleshooting
                       0x6 default"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 33
  NAME              = "SPECT_IMAGE_TYPE"
  START_BYTE        = 113
  BYTES             = 1
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "(imageType <= 6),
                       0 = 1D [dark] average,
                       1 = 1D [dark] single,
                       2 = 1D [passive] single,
                       3 = 1D average,
                       4 = 1D single,
                       5 = 2D [diagnostic],
                       6 reserved.
                       The [dark], [passive] and [diagnostic] are ineffective
                       descriptors, only the Ds and single/average matter.
                       Usually 2."
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 34
  NAME              = "SPECTROMETERSERIALCLOCK"
  START_BYTE        = 114
  BYTES             = 1
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "(spectrometerSerialClock <= 15),
                       0 = 2750kHz
                       1 = 1380kHz
                       2 = 922kHz
                       3 = 692kHz
                       4 = 554kHz
                       5 = 461kHz
                       6 = 396kHz
```

```
                       7 = 346kHz
                       8 = 308kHz
                       9 = 278kHz
                       10 = 252kHz
                       11 = 231kHz
                       12 = 218kHz
                       13 = 198kHz
                       14 = 185kHz
                       15 = 173kHz.  Usually 2 or 3."
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 35
  NAME              = "SPECTROMETERSELECT"
  START_BYTE        = 115
  BYTES             = 1
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "(spectrometerSelect <= 15),
                      Each bit 1 = enable, 0 = disable.
                      0x1 = 2D enable
                      0x2 = UV
                      0x4 = VIS
                      0x8 = VNIR.  Usually 0xE."
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 36
  NAME              = "ICT_DIVIDER"
  START_BYTE        = 116
  BYTES             = 2
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "(ict_divisor != 0),Integration Clock Timer divisor
                       (for time base above) - for msec integration time.
                       Usually 600."
END_OBJECT          = COLUMN


OBJECT              = COLUMN
  COLUMN_NUMBER     = 37
  NAME              = "IPBC_DIVIDER"
  START_BYTE        = 118
  BYTES             = 2
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "(ipbc_divisor != 0),Integration Period Base Clock
                       divisor. Range 0 - 33MHz. Usually 330 = 100KHz"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 38
  NAME              = "STARTROWUV"
  START_BYTE        = 120
  BYTES             = 2
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "(startRowUV <= stopRowUV),Usually 0"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
```

```
    COLUMN_NUMBER    = 39
    NAME             = "STOPROWUV"
    START_BYTE       = 122
    BYTES            = 2
    DATA_TYPE        = MSB_UNSIGNED_INTEGER
    DESCRIPTION      = "(stopRowUV <= 512),Usually 255"
  END_OBJECT         = COLUMN

  OBJECT             = COLUMN
    COLUMN_NUMBER    = 40
    NAME             = "STARTRROWVIS"
    START_BYTE       = 124
    BYTES            = 2
    DATA_TYPE        = MSB_UNSIGNED_INTEGER
    DESCRIPTION      = "(startRowVIS <= stopRowVIS),Usually 0"
  END_OBJECT         = COLUMN

  OBJECT             = COLUMN
    COLUMN_NUMBER    = 41
    NAME             = "STOPROWVIS"
    START_BYTE       = 126
    BYTES            = 2
    DATA_TYPE        = MSB_UNSIGNED_INTEGER
    DESCRIPTION      = "(stopRowVIS <= 512),Usually 255"
  END_OBJECT         = COLUMN

  OBJECT             = COLUMN
    COLUMN_NUMBER    = 42
    NAME             = "STARTROWVNIR"
    START_BYTE       = 128
    BYTES            = 2
    DATA_TYPE        = MSB_UNSIGNED_INTEGER
    DESCRIPTION      = "(startRowVNIR <= stopRowVNIR),Usually 0"
  END_OBJECT         = COLUMN

  OBJECT             = COLUMN
    COLUMN_NUMBER    = 43
    NAME             = "STOPROWVNIR"
    START_BYTE       = 130
    BYTES            = 2
    DATA_TYPE        = MSB_UNSIGNED_INTEGER
    DESCRIPTION      = "(stopRowVNIR <= 512),Usually 255"
  END_OBJECT         = COLUMN

  OBJECT             = COLUMN
    COLUMN_NUMBER    = 44
    NAME             = "OBSFROMLIMITSWITCH"
    START_BYTE       = 132
    BYTES            = 1
    DATA_TYPE        = MSB_UNSIGNED_INTEGER
    DESCRIPTION      = ""
  END_OBJECT         = COLUMN

  OBJECT             = COLUMN
    COLUMN_NUMBER    = 45
    NAME             = "STACK1LEVEL"
    START_BYTE       = 133
```

```
    BYTES           = 4
    DATA_TYPE       = IEEE_REAL
    DESCRIPTION     = "(stack1Level < 4096),
                       0 - 4095 = 0 - 140 Amps.
                       For FM use 2778 = 95 A
                       and do not exceed this value at BOL"
  END_OBJECT        = COLUMN

  OBJECT            = COLUMN
    COLUMN_NUMBER   = 46
    NAME            = "STACK2LEVEL"
    START_BYTE      = 137
    BYTES           = 4
    DATA_TYPE       = IEEE_REAL
    DESCRIPTION     = "(stack2Level < 4096),i
                       0 - 4095 = 0 - 140 Amps.
                       Use 2850.
                       For FM use from 1720 (= 60 A) to 2762 (= 95 A)
                       and to not exceed the latter at BOL."
  END_OBJECT        = COLUMN

  OBJECT            = COLUMN
    COLUMN_NUMBER   = 47
    NAME            = "STACK3LEVEL"
    START_BYTE      = 141
    BYTES           = 4
    DATA_TYPE       = IEEE_REAL
    DESCRIPTION     = "(stack3Level < 4096),0 - 4095 = 0 - 140 Amps.
                       Use 2850.
                       For FM use from 1744 (= 60 A) to 2789 (= 95 A)
                       and do not exceed the latter at BOL"
  END_OBJECT        = COLUMN

  OBJECT            = COLUMN
    COLUMN_NUMBER   = 48
    NAME            = "STACKDURATION"
    START_BYTE      = 145
    BYTES           = 1
    DATA_TYPE       = MSB_UNSIGNED_INTEGER
    DESCRIPTION     = "0 - 255 => 130 - 200 usec (nonlinear). Use 65."
  END_OBJECT        = COLUMN

  OBJECT            = COLUMN
    COLUMN_NUMBER   = 49
    NAME            = "TIMEBETWEENSHOTS"
    START_BYTE      = 146
    BYTES           = 1
    DATA_TYPE       = MSB_UNSIGNED_INTEGER
    DESCRIPTION     = "(14 <= timeBetweenShots),
                      0 = no control.
                      1 - 255 => 48.6 + (N-1)/254 * 950 msec.
                      Use from 14 to 255 (<=10 Hz). 15 => 101 msec."
  END_OBJECT        = COLUMN

  OBJECT            = COLUMN
    COLUMN_NUMBER   = 50
    NAME            = "SPECADCONVERTVNIR"
```

```
    START_BYTE       = 147
    BYTES            = 1
    DATA_TYPE        = MSB_UNSIGNED_INTEGER
    DESCRIPTION      = "(spectTrigDelays < 983040)usually 5"
  END_OBJECT         = COLUMN

  OBJECT             = COLUMN
    COLUMN_NUMBER    = 51
    NAME             = "SPECADCONVERTVIS"
    START_BYTE       = 148
    BYTES            = 1
    DATA_TYPE        = MSB_UNSIGNED_INTEGER
    DESCRIPTION      = "(spectTrigDelays < 983040)usually 5"
  END_OBJECT         = COLUMN

  OBJECT             = COLUMN
    COLUMN_NUMBER    = 52
    NAME             = "SPECADCONVERTUV"
    START_BYTE       = 149
    BYTES            = 1
    DATA_TYPE        = MSB_UNSIGNED_INTEGER
    DESCRIPTION      = "(spectTrigDelays < 983040)usually 5"
  END_OBJECT         = COLUMN

  OBJECT             = COLUMN
    COLUMN_NUMBER    = 53
    NAME             = "SPECVERTCLK"
    START_BYTE       = 150
    BYTES            = 1
    DATA_TYPE        = MSB_UNSIGNED_INTEGER
    DESCRIPTION      = "Vertical clock select usually 2"
  END_OBJECT         = COLUMN

/*NOTE this is not yet complete."
/* End of format file CCAM_LIBS_CMD_ARG_PARAMS_V2.FMT */
```

j) "CCAM_LIBS_CMD_ARG_PARAMS" (2nd of 2)

```
/* CCAM_LIBS_CMD_ARG_PARAMS_V4.FMT                                          */

  OBJECT             = COLUMN
    COLUMN_NUMBER    = 21
    NAME             = "FRAME_ID"
    START_BYTE       = 81
    BYTES            = 4
    DATA_TYPE        = MSB_INTEGER
    DESCRIPTION      = "FRAME in which the RSM pointing coords are specified.
                        This argument is irrelevant if
                        coord_type = JOINTS_*."
  END_OBJECT         = COLUMN

  OBJECT             = COLUMN
    COLUMN_NUMBER    = 22
    NAME             = "FRAME_INDEX"
    START_BYTE       = 85
    BYTES            = 2
```

```
  DATA_TYPE        = MSB_INTEGER
  DESCRIPTION      = "Index of the chosen frame(n/a for many frame_IDs)."
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 23
  NAME             = "COORD_TYPE"
  START_BYTE       = 87
  BYTES            = 4
  DATA_TYPE        = MSB_INTEGER
  DESCRIPTION      = "Types of coordinates specified by coord1, coord2,
                      coord3 args."
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 24
  NAME             = "COORD_1"
  START_BYTE       = 91
  BYTES            = 4
  DATA_TYPE        = IEEE_REAL
  DESCRIPTION      = "X or AZ coordinate for pointing in frame"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 25
  NAME             = "COORD_2"
  START_BYTE       = 95
  BYTES            = 4
  DATA_TYPE        = IEEE_REAL
  DESCRIPTION      = "Y or EL coordinate for pointing in frame"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 26
  NAME             = "COORD_3"
  START_BYTE       = 99
  BYTES            = 4
  DATA_TYPE        = IEEE_REAL
  DESCRIPTION      = "Z or N/A coordinate for pointing in frame"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 27
  NAME             = "FOCUS"
  START_BYTE       = 103
  BYTES            = 4
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Focus algorithm:
                 0 = NO_FOCUS - Dosen't move focus (others mark CCAM unsafe)
                 1 = BASELINE - use CWL to find optimal focus position
                 3 = MANUAL - positions focus based on range argument
                 4 = AF_OFFSET - applies LIBS offset from last autofocus
                     solution"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 28
```

```
  NAME              = "RANGE"
  START_BYTE        = 107
  BYTES             = 2
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Distance to target for MANUAL focus, seed for BASELINE"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 29
  NAME              = "DATA"
  START_BYTE        = 109
  BYTES             = 4
  DATA_TYPE         = MSB_INTEGER
  DESCRIPTION       = "Data Present"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 30
  NAME              = "PRE_POST_DARKSPECTRA"
  START_BYTE        = 113
  BYTES             = 4
  DATA_TYPE         = MSB_INTEGER
  DESCRIPTION       = "pre and post dark spectra
                       0= PRE_ONLY
                       1= POST_ONLY
                       2=PRE_AND_POST
                       3=NONE"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 31
  NAME              = "USE_LASER"
  START_BYTE        = 117
  BYTES             = 4
  DATA_TYPE         = MSB_INTEGER
  DESCRIPTION       = "0= no laser
                       1= laserfired"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 32
  NAME              = "NSHOTS_2_IGNOR"
  START_BYTE        = 121
  BYTES             = 1
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Number of shots to ignor"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 33
  NAME              = "NSHOTS_2_AVG"
  START_BYTE        = 122
  BYTES             = 1
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Number of shots to Average"
END_OBJECT          = COLUMN
```

```
OBJECT               = COLUMN
  COLUMN_NUMBER      = 34
  NAME               = "NSHOTS"
  START_BYTE         = 123
  BYTES              = 1
  DATA_TYPE          = MSB_UNSIGNED_INTEGER
  DESCRIPTION        = "Number of shots"
END_OBJECT           = COLUMN

OBJECT               = COLUMN
  COLUMN_NUMBER      = 35
  NAME               = "STATS"
  START_BYTE         = 124
  BYTES              = 4
  DATA_TYPE          = MSB_INTEGER
  DESCRIPTION        = "Number to indicate whether statistics were
                        requested
                        CCAM_NO_PROCESSING =1
                        CCAM_MEAN_STD_DEV = 2
                        CCAM_ALL_STATS =3"
END_OBJECT           = COLUMN

OBJECT               = COLUMN
  COLUMN_NUMBER      = 36
  NAME               = "GO_TO_WAY"
  START_BYTE         = 128
  BYTES              = 4
  DATA_TYPE          = MSB_INTEGER
  DESCRIPTION        = "Number to indicate way points
                        CCAM_NO_WAYPTS = 0
                        CCAM_WAYPT_1  = 1
                        CCAM_WAYPT_2 = 2
                        CCAM_BOTH_WAYPTS = 3 "
END_OBJECT           = COLUMN

OBJECT               = COLUMN
  COLUMN_NUMBER      = 37
  NAME               = "SPECT_CONTROL_BYTE"
  START_BYTE         = 132
  BYTES              = 1
  DATA_TYPE          = MSB_UNSIGNED_INTEGER
  DESCRIPTION        = "0x2 - enable spectometer clocks
                        0x4 - enable temperature checks
                        always enable both except when troubleshooting
                        0x6 default"
END_OBJECT           = COLUMN

OBJECT               = COLUMN
  COLUMN_NUMBER      = 38
  NAME               = "SPECT_IMAGE_TYPE"
  START_BYTE         = 133
  BYTES              = 1
  DATA_TYPE          = MSB_UNSIGNED_INTEGER
  DESCRIPTION        = "(imageType <= 6),
                        0 = 1D [dark] average,
                        1 = 1D [dark] single,
                        2 = 1D [passive] single,
```

```
                           3 = 1D average,
                           4 = 1D single,
                           5 = 2D [diagnostic],
                           6 reserved.
                           The [dark], [passive] and [diagnostic] are ineffective
                           descriptors, only the Ds and single/average matter.
                           Usually 2."
END_OBJECT             = COLUMN

OBJECT                 = COLUMN
  COLUMN_NUMBER        = 39
  NAME                 = "SPECTROMETERSERIALCLOCK"
  START_BYTE           = 134
  BYTES                = 1
  DATA_TYPE            = MSB_UNSIGNED_INTEGER
  DESCRIPTION          = "(spectrometerSerialClock <= 15),
                           0 = 2750kHz
                           1 = 1380kHz
                           2 = 922kHz
                           3 = 692kHz
                           4 = 554kHz
                           5 = 461kHz
                           6 = 396kHz
                           7 = 346kHz
                           8 = 308kHz
                           9 = 278kHz
                           10 = 252kHz
                           11 = 231kHz
                           12 = 218kHz
                           13 = 198kHz
                           14 = 185kHz
                           15 = 173kHz.  Usually 2 or 3."
END_OBJECT             = COLUMN

OBJECT                 = COLUMN
  COLUMN_NUMBER        = 40
  NAME                 = "SPECTROMETERSELECT"
  START_BYTE           = 135
  BYTES                = 1
  DATA_TYPE            = MSB_UNSIGNED_INTEGER
  DESCRIPTION          = "(spectrometerSelect <= 15),
                          Each bit 1 = enable, 0 = disable.
                          0x1 = 2D enable
                          0x2 = UV
                          0x4 = VIS
                          0x8 = VNIR.  Usually 0xE."
END_OBJECT             = COLUMN

OBJECT                 = COLUMN
  COLUMN_NUMBER        = 41
  NAME                 = "ICT_DIVIDER"
  START_BYTE           = 136
  BYTES                = 2
  DATA_TYPE            = MSB_UNSIGNED_INTEGER
  DESCRIPTION          = "(ict_divisor != 0),Integration Clock Timer divisor
                           (for time base above) - for msec integration time.
                           Usually 600."
```

```
END_OBJECT          = COLUMN


OBJECT              = COLUMN
  COLUMN_NUMBER     = 42
  NAME              = "IPBC_DIVIDER"
  START_BYTE        = 138
  BYTES             = 2
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "(ipbc_divisor != 0),Integration Period Base Clock
                       divisor. Range 0 - 33MHz. Usually 330 = 100KHz"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 43
  NAME              = "STARTROWUV"
  START_BYTE        = 140
  BYTES             = 2
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "(startRowUV <= stopRowUV),Usually 0"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 44
  NAME              = "STOPROWUV"
  START_BYTE        = 142
  BYTES             = 2
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "(stopRowUV <= 512),Usually 255"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 45
  NAME              = "STARTRROWVIS"
  START_BYTE        = 144
  BYTES             = 2
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "(startRowVIS <= stopRowVIS),Usually 0"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 46
  NAME              = "STOPROWVIS"
  START_BYTE        = 146
  BYTES             = 2
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "(stopRowVIS <= 512),Usually 255"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 47
  NAME              = "STARTROWVNIR"
  START_BYTE        = 148
  BYTES             = 2
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "(startRowVNIR <= stopRowVNIR),Usually 0"
END_OBJECT          = COLUMN
```

```
OBJECT                = COLUMN
  COLUMN_NUMBER       = 48
  NAME                = "STOPROWVNIR"
  START_BYTE          = 150
  BYTES               = 2
  DATA_TYPE           = MSB_UNSIGNED_INTEGER
  DESCRIPTION         = "(stopRowVNIR <= 512),Usually 255"
END_OBJECT            = COLUMN

OBJECT                = COLUMN
  COLUMN_NUMBER       = 49
  NAME                = "OBSFROMLIMITSWITCH"
  START_BYTE          = 152
  BYTES               = 1
  DATA_TYPE           = MSB_UNSIGNED_INTEGER
  DESCRIPTION         = ""
END_OBJECT            = COLUMN

OBJECT                = COLUMN
  COLUMN_NUMBER       = 50
  NAME                = "STACK1LEVEL"
  START_BYTE          = 153
  BYTES               = 4
  DATA_TYPE           = IEEE_REAL
  DESCRIPTION         = "(stack1Level < 4096),
                         0 - 4095 = 0 - 140 Amps.
                         For FM use 2778 = 95 A
                         and do not exceed this value at BOL"
END_OBJECT            = COLUMN

OBJECT                = COLUMN
  COLUMN_NUMBER       = 51
  NAME                = "STACK2LEVEL"
  START_BYTE          = 157
  BYTES               = 4
  DATA_TYPE           = IEEE_REAL
  DESCRIPTION         = "(stack2Level < 4096),i
                         0 - 4095 = 0 - 140 Amps.
                         Use 2850.
                         For FM use from 1720 (= 60 A) to 2762 (= 95 A)
                         and to not exceed the latter at BOL."
END_OBJECT            = COLUMN

OBJECT                = COLUMN
  COLUMN_NUMBER       = 52
  NAME                = "STACK3LEVEL"
  START_BYTE          = 161
  BYTES               = 4
  DATA_TYPE           = IEEE_REAL
  DESCRIPTION         = "(stack3Level < 4096),0 - 4095 = 0 - 140 Amps.
                         Use 2850.
                         For FM use from 1744 (= 60 A) to 2789 (= 95 A)
                         and do not exceed the latter at BOL"
END_OBJECT            = COLUMN

OBJECT                = COLUMN
```

```
    COLUMN_NUMBER    = 53
    NAME             = "STACKDURATION"
    START_BYTE       = 165
    BYTES            = 1
    DATA_TYPE        = MSB_UNSIGNED_INTEGER
    DESCRIPTION      = "0 - 255 => 130 - 200 usec (nonlinear). Use 65."
  END_OBJECT         = COLUMN

  OBJECT             = COLUMN
    COLUMN_NUMBER    = 54
    NAME             = "TIMEBETWEENSHOTS"
    START_BYTE       = 166
    BYTES            = 1
    DATA_TYPE        = MSB_UNSIGNED_INTEGER
    DESCRIPTION      = "(14 <= timeBetweenShots),
                       0 = no control.
                       1 - 255 => 48.6 + (N-1)/254 * 950 msec.
                       Use from 14 to 255 (<=10 Hz). 15 => 101 msec."
  END_OBJECT         = COLUMN

  OBJECT             = COLUMN
    COLUMN_NUMBER    = 55
    NAME             = "SPECADCONVERTVNIR"
    START_BYTE       = 167
    BYTES            = 1
    DATA_TYPE        = MSB_UNSIGNED_INTEGER
    DESCRIPTION      = "(spectTrigDelays < 983040)usually 5"
  END_OBJECT         = COLUMN

  OBJECT             = COLUMN
    COLUMN_NUMBER    = 56
    NAME             = "SPECADCONVERTVIS"
    START_BYTE       = 168
    BYTES            = 1
    DATA_TYPE        = MSB_UNSIGNED_INTEGER
    DESCRIPTION      = "(spectTrigDelays < 983040)usually 5"
  END_OBJECT         = COLUMN

  OBJECT             = COLUMN
    COLUMN_NUMBER    = 57
    NAME             = "SPECADCONVERTUV"
    START_BYTE       = 169
    BYTES            = 1
    DATA_TYPE        = MSB_UNSIGNED_INTEGER
    DESCRIPTION      = "(spectTrigDelays < 983040)usually 5"
  END_OBJECT         = COLUMN

  OBJECT             = COLUMN
    COLUMN_NUMBER    = 58
    NAME             = "SPECVERTCLK"
    START_BYTE       = 170
    BYTES            = 1
    DATA_TYPE        = MSB_UNSIGNED_INTEGER
    DESCRIPTION      = "Vertical clock select usually 2"
  END_OBJECT         = COLUMN

/*NOTE this is not yet complete."
```

```
/* End of format file CCAM_LIBS_CMD_ARG_PARAMS_V4.FMT */
```

k) “CCAM_LIBS_MEAN”

```
/* CCAM_LIBS_MEAN_V4.FMT.                                     */

  OBJECT                = COLUMN
    COLUMN_NUMBER       = 1
    NAME                = "PIXEL_COUNT"
    START_BYTE          = 1
    BYTES               = 4
    DATA_TYPE           = MSB_UNSIGNED_INTEGER
    DESCRIPTION         = "Number of Pixels (6444) "
  END_OBJECT            = COLUMN

  OBJECT                = COLUMN
    COLUMN_NUMBER       = 2
    NAME                = "CCAM_MEAN_LIBS_DATA"
    START_BYTE          = 2
    BYTES               = 25776
    DATA_TYPE           = IEEE_REAL
    DESCRIPTION         = "ChemCam MEAN LIBS DATA"
  END_OBJECT            = COLUMN

/* End of format file CCAM_LIBS_MEAN_V4.FMT. */
```

l) “CCAM_LIBS_MEDIAN”

```
/* CCAM_LIBS_MEDIAN_V4.FMT.                                     */

  OBJECT                = COLUMN
    COLUMN_NUMBER       = 1
    NAME                = "PIXEL_COUNT"
    START_BYTE          = 1
    BYTES               = 4
    DATA_TYPE           = MSB_UNSIGNED_INTEGER
    DESCRIPTION         = "Number of Pixels (6444) "
  END_OBJECT            = COLUMN

  OBJECT                = COLUMN
    COLUMN_NUMBER       = 2
    NAME                = "CCAM_MEDIAN_LIBS_DATA"
    START_BYTE          = 2
    BYTES               = 12888
    DATA_TYPE           = MSB_UNSIGNED_INTEGER
    DESCRIPTION         = "ChemCam MEDIAN LIBS DATA"
  END_OBJECT            = COLUMN

/* End of format file CCAM_LIBS_MEDIAN_V4.FMT. */
```

m) “CCAM_LIBS_STDDEV”

```
/* CCAM_LIBS_STDDEV_V4.FMT.                                     */
```

```
  OBJECT              = COLUMN
    COLUMN_NUMBER     = 1
    NAME              = "PIXEL_COUNT"
    START_BYTE        = 1
    BYTES             = 4
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    DESCRIPTION       = "Number of Pixels (6444) "
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 2
    NAME              = "CCAM_STDDEV_LIBS_DATA"
    START_BYTE        = 2
    BYTES             = 25776
    DATA_TYPE         = IEEE_REAL
    DESCRIPTION       = "ChemCam STDDEV LIBS DATA"
  END_OBJECT          = COLUMN

/* End of format file CCAM_LIBS_STDDEV_V4.FMT. */
```

n) “CCAM_LIBS_Q1”

```
/* CCAM_LIBS_Q1_V4.FMT.                                        */

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 1
    NAME              = "PIXEL_COUNT"
    START_BYTE        = 1
    BYTES             = 4
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    DESCRIPTION       = "Number of Pixels (6444) "
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 2
    NAME              = "CCAM_Q1_LIBS_DATA"
    START_BYTE        = 2
    BYTES             = 12888
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    DESCRIPTION       = "ChemCam Q1 LIBS DATA"
  END_OBJECT          = COLUMN

/* End of format file CCAM_LIBS_Q1_V4.FMT. */
```

o) “CCAM_LIBS_Q3”

```
/* CCAM_LIBS_Q3_V4.FMT.                                        */

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 1
    NAME              = "PIXEL_COUNT"
    START_BYTE        = 1
    BYTES             = 4
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    DESCRIPTION       = "Number of Pixels (6444) "
```

```
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 2
    NAME              = "CCAM_Q3_LIBS_DATA"
    START_BYTE        = 2
    BYTES             = 12888
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    DESCRIPTION       = "ChemCam Q3 LIBS DATA"
  END_OBJECT          = COLUMN

/* End of format file CCAM_LIBS_Q3_V4.FMT. */
```

p) "CCAM_SOH_ANCILLARY" (1st of 2)

```
/* CCAM_SOH_ANCILLARY_V2.FMT.                                        */

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 1
    NAME              = "SOH_SCLK"
    START_BYTE        = 1
    BYTES             = 4
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    DESCRIPTION       = "Time collected"
  END_OBJECT          = COLUMN

  /* This format file defines the 20 ancillary TEMP data columns.      */
  /* 84bytes                                                       */
  ^STRUCTURE                        = "CCAM_ANCILLARY_TMPS_V2.FMT"

/* End of format file CCAM_SOH_ANCILLARY_V2.FMT */
```

q) "CCAM_SOH_ANCILLARY" (2nd of 2)

```
/* CCAM_SOH_ANCILLARY_V4.FMT.                                        */

  /* This format file defines the 20 ancillary TEMP data columns.      */
  /* 80bytes                                                       */
  ^STRUCTURE                        = "CCAM_ANCILLARY_TMPS_V4.FMT"

/* End of format file CCAM_SOH_ANCILLARY_V4.FMT */
```

r) "CCAM_SOH_EDR_ANCILLARY"

```
/* CCAM_SOH_EDR_ANCILLARY_V4.FMT.                                        */

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 1
    NAME              = "SOH_FREQUENCY"
    START_BYTE        = 1
    BYTES             = 1
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    DESCRIPTION       = "Soh Frequency"
  END_OBJECT          = COLUMN
```

```
OBJECT              = COLUMN
    COLUMN_NUMBER   = 2
    NAME            = "SOH_DURATION"
    START_BYTE      = 2
    BYTES           = 2
    DATA_TYPE       = MSB_UNSIGNED_INTEGER
    DESCRIPTION     = "Soh Duration"
  END_OBJECT        = COLUMN

  /* This format file defines the 20 ancillary TEMP data columns.       */
  /* 80bytes                                                       */
  ^STRUCTURE                            = "CCAM_ANCILLARY_TMPS_V4.FMT"

/* End of format file CCAM_SOH_EDR_ANCILLARY_V4.FMT */
```

s) "CCAM_SOH_SCIDATA_COLS" (1st of 2)

```
/* CCAM_SOH_SCIDATA_COLS_V2.FMT.                                 */

  OBJECT                        = COLUMN
    COLUMN_NUMBER               = 1
    NAME                        = "BYTE_COUNT"
    START_BYTE                  = 1
    BYTES                       = 4
    DATA_TYPE                   = MSB_UNSIGNED_INTEGER
    DESCRIPTION                 = "Number of bytes in SOH DPO"
  END_OBJECT                    = COLUMN

  OBJECT                        = COLUMN
    COLUMN_NUMBER               = 2
    NAME                        = "RCECONTROL"
    START_BYTE                  = 5
    BYTES                       = 4
    DATA_TYPE                   = MSB_UNSIGNED_INTEGER
    DESCRIPTION                 = "Instrument Transfer frame Header
                                   contains: (8bits) Opcode 52 for soh
                                             (2bits) Error Control Type Flag
                           00 = reserved
                           01 =  The CRC error control algorithm is applied
                           10 = The Checksum error control Algorithm applied
                           11 = No error Control"
                                             (1 bit) Data Present
                           0 = nodata
                           1 = data"
                                             (21 bits) Status as follows
                         bit 1 - command Reply Flag
                                  (0- not cmd_reply, 1 - cmd_reply)
                         bit 2 - notSafe
                                  ( 0 - safe
                                    1 - Bu doesn't know that MU mortors
                                      are in sun-safe position)
                         bit 3 - BootSource
                                  (Which memory was used for boot
                                   0 - Init0 PROMO
                                   1 - Init1 PROM1)
```

```
                                      bit 4 - CommSide
                                               (Talking with which RCE?
                                                0 - RCE A or none
                                                1 - RCE B)
                                      bit 5 - CWL_heater_notOn
                                               (CWL heating loop algorithm off
                                                0 - on
                                                1 - off)
                                      bit 6 - Amp_heater_notOn
                                               ( Amp heating loop algorithm off
                                                0 - on
                                                1 - off)
                                      bit 7 - OSC_heater_notOn
                                               ( Osc heating loop algorithm off
                                                0 - on
                                                1 - off)
                                      bit 8 - RMI_notOn
                                               ( RMI is not Powered
                                                0 - on
                                                1 - off)
                                      bit 9 - RMIdata_notOK
                                               ( RMI off or comm with RMI not OK
                                                0 - OK
                                                1 - not OK, =1
                                                    when RMI if off)
                                      bit 10 - Spectrometer_notOK
                                               ( not used should be 0)
                                      bit 11 - LVPS_notOn
                                               ( Low Voltage (spectrometer)power supply
                                                 0 - on
                                                 1 - off )
                                      bit 12 - MAST_notOK
                                               ( Bad communication with the mast unit
                                                 0 - OK
                                                 1 - not OK)
                                      bit 13 - LIBS_HV_notOn
                                               ( High voltage to LIBS not on
                                                 0 - on
                                                 1 - not on)
                                      bit 14 - LIBS_notReady
                                               (LIBS not in operating temperature range
                                                 0 - ready
                                                 1 - not ready)
                                      bit 15 - CWL_notReady
                                               (CWL not in operating temperature range
                                                 0 - ready
                                                 1 - not ready)
                                      bit 16 - SelfTestFailed
                                                (not used should be 0)
                                      bit 17:21 -
                                                0 0000 - reserved
                                                0 0001 - First time command
                                                0 0010 - Retry command (cmd reply frame)
                                                1 0010 - Retry command (sci frame) "

END_OBJECT                       = COLUMN
```

```
OBJECT                         = COLUMN
  COLUMN_NUMBER                = 3
  NAME                         = "RCEBYTECOUNT"
  START_BYTE                   = 9
  BYTES                        = 4
  DATA_TYPE                    = MSB_UNSIGNED_INTEGER
  DESCRIPTION                  = "rce bytecount, bytes following"
END_OBJECT                     = COLUMN

OBJECT                         = COLUMN
  COLUMN_NUMBER                = 4
  NAME                         = "DATAID"
  START_BYTE                   = 13
  BYTES                        = 2
  DATA_TYPE                    = MSB_UNSIGNED_INTEGER
  DESCRIPTION                  = "id word is 0x55aa for soh"
END_OBJECT                     = COLUMN

OBJECT                         = COLUMN
  COLUMN_NUMBER                = 5
  NAME                         = "HEAD2"
  START_BYTE                   = 15
  BYTES                        = 2
  DATA_TYPE                    = MSB_UNSIGNED_INTEGER
  DESCRIPTION                  = "00 for soh"
END_OBJECT                     = COLUMN

OBJECT                         = COLUMN
  COLUMN_NUMBER                = 6
  NAME                         = "BYTES"
  START_BYTE                   = 17
  BYTES                        = 4
  DATA_TYPE                    = MSB_UNSIGNED_INTEGER
  DESCRIPTION                  = "Number of bytes that follow"
END_OBJECT                     = COLUMN

OBJECT                         = COLUMN
  COLUMN_NUMBER                = 7
  NAME                         = "RCE_TIME"
  START_BYTE                   = 21
  BYTES                        = 4
  DATA_TYPE                    = MSB_UNSIGNED_INTEGER
  DESCRIPTION                  = "Time stamp"
END_OBJECT                     = COLUMN

OBJECT                         = COLUMN
  COLUMN_NUMBER                = 8
  NAME                         = "CUR_MSEC_COUNT"
  START_BYTE                   = 25
  BYTES                        = 4
  DATA_TYPE                    = MSB_UNSIGNED_INTEGER
  DESCRIPTION                  = "Local counter ~1 msec/count"
END_OBJECT                     = COLUMN

OBJECT                         = COLUMN
  COLUMN_NUMBER                = 9
  NAME                         = "GOOD_CMDS"
```

```
  START_BYTE                   = 29
  BYTES                        = 2
  DATA_TYPE                    = MSB_UNSIGNED_INTEGER
  DESCRIPTION                  = "Received from RCE"
END_OBJECT                     = COLUMN

OBJECT                         = COLUMN
  COLUMN_NUMBER                = 10
  NAME                         = "BAD_CMDS"
  START_BYTE                   = 31
  BYTES                        = 2
  DATA_TYPE                    = MSB_UNSIGNED_INTEGER
  DESCRIPTION                  = "Received from RCE"
END_OBJECT                     = COLUMN

OBJECT                         = COLUMN
  COLUMN_NUMBER                = 11
  NAME                         = "MAST_CMDS"
  START_BYTE                   = 33
  BYTES                        = 2
  DATA_TYPE                    = MSB_UNSIGNED_INTEGER
  DESCRIPTION                  = "Cmds sent to mast"
END_OBJECT                     = COLUMN

OBJECT                         = COLUMN
  COLUMN_NUMBER                = 12
  NAME                         = "MAST_ACKS"
  START_BYTE                   = 35
  BYTES                        = 2
  DATA_TYPE                    = MSB_UNSIGNED_INTEGER
  DESCRIPTION                  = "Cmds ACKed by mast"
END_OBJECT                     = COLUMN

OBJECT                         = COLUMN
  COLUMN_NUMBER                = 13
  NAME                         = "MAST_NAKS"
  START_BYTE                   = 37
  BYTES                        = 2
  DATA_TYPE                    = MSB_UNSIGNED_INTEGER
  DESCRIPTION                  = "Cmds NAKed by mast"
END_OBJECT                     = COLUMN

OBJECT                         = COLUMN
  COLUMN_NUMBER                = 14
  NAME                         = "MAST_RESENDS"
  START_BYTE                   = 39
  BYTES                        = 2
  DATA_TYPE                    = MSB_UNSIGNED_INTEGER
  DESCRIPTION                  = "Cmds resent to mast"
END_OBJECT                     = COLUMN

OBJECT                         = COLUMN
  COLUMN_NUMBER                = 15
  NAME                         = "MAST_PKTS_RECD"
  START_BYTE                   = 41
  BYTES                        = 2
  DATA_TYPE                    = MSB_UNSIGNED_INTEGER
```

```
  DESCRIPTION                    = "Cmds received from MAST"
END_OBJECT                       = COLUMN

OBJECT                           = COLUMN
  COLUMN_NUMBER                  = 16
  NAME                           = "MAST_PKTS_BAD"
  START_BYTE                     = 43
  BYTES                          = 2
  DATA_TYPE                      = MSB_UNSIGNED_INTEGER
  DESCRIPTION                    = "Pkts incomplete or with checksum
                                    error"
END_OBJECT                       = COLUMN

OBJECT                           = COLUMN
  COLUMN_NUMBER                  = 17
  NAME                           = "CC_STATE"
  START_BYTE                     = 45
  BYTES                          = 2
  DATA_TYPE                      = MSB_UNSIGNED_INTEGER
  DESCRIPTION                    = "Major and minor state values"
END_OBJECT                       = COLUMN

OBJECT                           = COLUMN
  COLUMN_NUMBER                  = 18
  NAME                           = "STATUS"
  START_BYTE                     = 47
  BYTES                          = 2
  DATA_TYPE                      = MSB_UNSIGNED_INTEGER
  DESCRIPTION                    = "TBD status bits"
END_OBJECT                       = COLUMN

OBJECT                           = COLUMN
  COLUMN_NUMBER                  = 19
  NAME                           = "MEM_FPGA_VERSION"
  START_BYTE                     = 49
  BYTES                          = 2
  DATA_TYPE                      = MSB_UNSIGNED_INTEGER
  DESCRIPTION                    = "Memory FPGA version (33)"
END_OBJECT                       = COLUMN

OBJECT                           = COLUMN
  COLUMN_NUMBER                  = 20
  NAME                           = "MICRO_FPGA_VERSION (34)"
  START_BYTE                     = 51
  BYTES                          = 2
  DATA_TYPE                      = MSB_UNSIGNED_INTEGER
  DESCRIPTION                    = "MICRO FPGA version"
END_OBJECT                       = COLUMN

OBJECT                           = COLUMN
  COLUMN_NUMBER                  = 21
  NAME                           = "SPECT_FPGA_VERSION (20)"
  START_BYTE                     = 53
  BYTES                          = 2
  DATA_TYPE                      = MSB_UNSIGNED_INTEGER
  DESCRIPTION                    = "Spectromitor FPGA Version"
END_OBJECT                       = COLUMN
```

```
OBJECT                          = COLUMN
  COLUMN_NUMBER                 = 22
  NAME                          = "IMAGES_SENT"
  START_BYTE                    = 55
  BYTES                         = 2
  DATA_TYPE                     = MSB_UNSIGNED_INTEGER
  DESCRIPTION                   = "Images sent To RCE"
END_OBJECT                      = COLUMN

OBJECT                          = COLUMN
  COLUMN_NUMBER                 = 23
  NAME                          = "SOH_SENT"
  START_BYTE                    = 57
  BYTES                         = 2
  DATA_TYPE                     = MSB_UNSIGNED_INTEGER
  DESCRIPTION                   = "Soh Sent To RCE"
END_OBJECT                      = COLUMN

OBJECT                          = COLUMN
  COLUMN_NUMBER                 = 24
  NAME                          = "SPECT_SENT"
  START_BYTE                    = 59
  BYTES                         = 2
  DATA_TYPE                     = MSB_UNSIGNED_INTEGER
  DESCRIPTION                   = "Spectra sent To RCE"
END_OBJECT                      = COLUMN

OBJECT                          = COLUMN
  COLUMN_NUMBER                 = 25
  NAME                          = "LASER_DATA_SENT"
  START_BYTE                    = 61
  BYTES                         = 2
  DATA_TYPE                     = MSB_UNSIGNED_INTEGER
  DESCRIPTION                   = "W,V,A per laser shot"
END_OBJECT                      = COLUMN

OBJECT                          = COLUMN
  COLUMN_NUMBER                 = 26
  NAME                          = "SPARE"
  START_BYTE                    = 63
  BYTES                         = 2
  DATA_TYPE                     = MSB_UNSIGNED_INTEGER
  DESCRIPTION                   = "Boot bank plus eepro initialized data
                                   location"
END_OBJECT                      = COLUMN

OBJECT                          = COLUMN
  COLUMN_NUMBER                 = 27
  NAME                          = "SPARE2"
  START_BYTE                    = 65
  BYTES                         = 2
  DATA_TYPE                     = MSB_UNSIGNED_INTEGER
  DESCRIPTION                   = "marker workd to easily id soh"
END_OBJECT                      = COLUMN

OBJECT                          = COLUMN
```

```
    COLUMN_NUMBER                = 28
    NAME                         = "N_FOLLOWING"
    START_BYTE                   = 67
    BYTES                        = 2
    DATA_TYPE                    = MSB_UNSIGNED_INTEGER
    DESCRIPTION                  = "Number of SOH records"
  END_OBJECT                     = COLUMN

/* End of format file CCAM_SOH_SCIDATA_COLS_V2.FMT */
```

t) "CCAM_SOH_SCIDATA_COLS" (2nd of 2)

```
/* CCAM_SOH_SCIDATA_COLS_V4.FMT.                              */

  OBJECT                         = COLUMN
    COLUMN_NUMBER                = 1
    NAME                         = "BYTE_COUNT"
    START_BYTE                   = 1
    BYTES                        = 4
    DATA_TYPE                    = MSB_UNSIGNED_INTEGER
    DESCRIPTION                  = "Number of bytes in SOH DPO"
  END_OBJECT                     = COLUMN

  OBJECT                         = COLUMN
    COLUMN_NUMBER                = 2
    NAME                         = "RCECONTROL"
    START_BYTE                   = 5
    BYTES                        = 4
    DATA_TYPE                    = MSB_UNSIGNED_INTEGER
    DESCRIPTION                  = "Instrument Transfer frame Header
                                    contains: (8bits) Opcode 52 for soh
                                              (2bits) Error Control Type Flag
                         00 = reserved
                         01 =  The CRC error control algorithm is applied
                         10 = The Checksum error control Algorithm applied
                         11 = No error Control
                                              (1 bit) Data Present
                         0 = nodata
                         1 = data
                                              (21 bits) Status as follows
                       bit 1 - command Reply Flag
                                (0- not cmd_reply, 1 - cmd_reply)
                       bit 2 - notSafe
                                ( 0 - safe
                                  1 - Bu doesn't know that MU mortors
                                    are in sun-safe position)
                       bit 3 - BootSource
                                (Which memory was used for boot
                                 0 - Init0 PROMO
                                 1 - Init1 PROM1)
                       bit 4 - CommSide
                                (Talking with which RCE?
                                 0 - RCE A or none
                                 1 - RCE B)
                       bit 5 - CWL_heater_notOn
                                (CWL heating loop algorithm off
```

```
                               0 - on
                               1 - off)
                     bit 6 - Amp_heater_notOn
                              ( Amp heating loop algorithm off
                               0 - on
                               1 - off)
                     bit 7 - OSC_heater_notOn
                              ( Osc heating loop algorithm off
                               0 - on
                               1 - off)
                     bit 8 - RMI_notOn
                              ( RMI is not Powered
                               0 - on
                               1 - off)
                     bit 9 - RMIdata_notOK
                              ( RMI off or comm with RMI not OK
                               0 - OK
                               1 - not OK, =1
                                   when RMI if off)
                     bit 10 - Spectrometer_notOK
                              ( not used should be 0)
                     bit 11 - LVPS_notOn
                              ( Low Voltage (spectrometer)power supply
                                0 - on
                                1 - off )
                     bit 12 - MAST_notOK
                              ( Bad communication with the mast unit
                                0 - OK
                                1 - not OK)
                     bit 13 - LIBS_HV_notOn
                              ( High voltage to LIBS not on
                                0 - on
                                1 - not on)
                     bit 14 - LIBS_notReady
                              (LIBS not in operating temperature range
                                0 - ready
                                1 - not ready)
                     bit 15 - CWL_notReady
                              (CWL not in operating temperature range
                                0 - ready
                                1 - not ready)
                     bit 16 - SelfTestFailed
                               (not used should be 0)
                     bit 17:21 -
                               0 0000 - reserved
                               0 0001 - First time command
                               0 0010 - Retry command (cmd reply frame)
                               1 0010 - Retry command (sci frame) "

END_OBJECT                    = COLUMN

OBJECT                        = COLUMN
  COLUMN_NUMBER               = 3
  NAME                        = "RCEBYTECOUNT"
  START_BYTE                  = 9
  BYTES                       = 4
  DATA_TYPE                   = MSB_UNSIGNED_INTEGER
```

```
  DESCRIPTION                    = "rce bytecount, bytes following"
END_OBJECT                       = COLUMN

OBJECT                           = COLUMN
  COLUMN_NUMBER                  = 4
  NAME                           = "DATAID"
  START_BYTE                     = 13
  BYTES                          = 2
  DATA_TYPE                      = MSB_UNSIGNED_INTEGER
  DESCRIPTION                    = "id word is 0x55aa for soh"
END_OBJECT                       = COLUMN

OBJECT                           = COLUMN
  COLUMN_NUMBER                  = 5
  NAME                           = "HEAD2"
  START_BYTE                     = 15
  BYTES                          = 2
  DATA_TYPE                      = MSB_UNSIGNED_INTEGER
  DESCRIPTION                    = "00 for soh"
END_OBJECT                       = COLUMN

OBJECT                           = COLUMN
  COLUMN_NUMBER                  = 6
  NAME                           = "BYTES"
  START_BYTE                     = 17
  BYTES                          = 4
  DATA_TYPE                      = MSB_UNSIGNED_INTEGER
  DESCRIPTION                    = "Number of bytes that follow"
END_OBJECT                       = COLUMN

OBJECT                           = COLUMN
  COLUMN_NUMBER                  = 7
  NAME                           = "RCE_TIME"
  START_BYTE                     = 21
  BYTES                          = 4
  DATA_TYPE                      = MSB_UNSIGNED_INTEGER
  DESCRIPTION                    = "Time stamp"
END_OBJECT                       = COLUMN

OBJECT                           = COLUMN
  COLUMN_NUMBER                  = 8
  NAME                           = "CUR_MSEC_COUNT"
  START_BYTE                     = 25
  BYTES                          = 4
  DATA_TYPE                      = MSB_UNSIGNED_INTEGER
  DESCRIPTION                    = "Local counter ~1 msec/count"
END_OBJECT                       = COLUMN

OBJECT                           = COLUMN
  COLUMN_NUMBER                  = 9
  NAME                           = "GOOD_CMDS"
  START_BYTE                     = 29
  BYTES                          = 2
  DATA_TYPE                      = MSB_UNSIGNED_INTEGER
  DESCRIPTION                    = "Received from RCE"
END_OBJECT                       = COLUMN
```

```
OBJECT                        = COLUMN
  COLUMN_NUMBER               = 10
  NAME                        = "BAD_CMDS"
  START_BYTE                  = 31
  BYTES                       = 2
  DATA_TYPE                   = MSB_UNSIGNED_INTEGER
  DESCRIPTION                 = "Received from RCE"
END_OBJECT                    = COLUMN

OBJECT                        = COLUMN
  COLUMN_NUMBER               = 11
  NAME                        = "MAST_CMDS"
  START_BYTE                  = 33
  BYTES                       = 2
  DATA_TYPE                   = MSB_UNSIGNED_INTEGER
  DESCRIPTION                 = "Cmds sent to mast"
END_OBJECT                    = COLUMN

OBJECT                        = COLUMN
  COLUMN_NUMBER               = 12
  NAME                        = "MAST_ACKS"
  START_BYTE                  = 35
  BYTES                       = 2
  DATA_TYPE                   = MSB_UNSIGNED_INTEGER
  DESCRIPTION                 = "Cmds ACKed by mast"
END_OBJECT                    = COLUMN

OBJECT                        = COLUMN
  COLUMN_NUMBER               = 13
  NAME                        = "MAST_NAKS"
  START_BYTE                  = 37
  BYTES                       = 2
  DATA_TYPE                   = MSB_UNSIGNED_INTEGER
  DESCRIPTION                 = "Cmds NAKed by mast"
END_OBJECT                    = COLUMN

OBJECT                        = COLUMN
  COLUMN_NUMBER               = 14
  NAME                        = "MAST_RESENDS"
  START_BYTE                  = 39
  BYTES                       = 2
  DATA_TYPE                   = MSB_UNSIGNED_INTEGER
  DESCRIPTION                 = "Cmds resent to mast"
END_OBJECT                    = COLUMN

OBJECT                        = COLUMN
  COLUMN_NUMBER               = 15
  NAME                        = "MAST_PKTS_RECD"
  START_BYTE                  = 41
  BYTES                       = 2
  DATA_TYPE                   = MSB_UNSIGNED_INTEGER
  DESCRIPTION                 = "Cmds received from MAST"
END_OBJECT                    = COLUMN

OBJECT                        = COLUMN
  COLUMN_NUMBER               = 16
  NAME                        = "MAST_PKTS_BAD"
```

```
    START_BYTE                    = 43
    BYTES                         = 2
    DATA_TYPE                     = MSB_UNSIGNED_INTEGER
    DESCRIPTION                   = "Pkts incomplete or with checksum
                                     error"
  END_OBJECT                      = COLUMN

  OBJECT                          = COLUMN
    COLUMN_NUMBER                 = 17
    NAME                          = "CC_STATE"
    START_BYTE                    = 45
    BYTES                         = 2
    DATA_TYPE                     = MSB_UNSIGNED_INTEGER
    DESCRIPTION                   = "Major and minor state values"
  END_OBJECT                      = COLUMN

  OBJECT                          = COLUMN
    COLUMN_NUMBER                 = 18
    NAME                          = "STATUS"
    START_BYTE                    = 47
    BYTES                         = 2
    DATA_TYPE                     = MSB_UNSIGNED_INTEGER
    DESCRIPTION                   = "TBD status bits"
  END_OBJECT                      = COLUMN

  OBJECT                          = COLUMN
    COLUMN_NUMBER                 = 19
    NAME                          = "MEM_FPGA_VERSION"
    START_BYTE                    = 49
    BYTES                         = 2
    DATA_TYPE                     = MSB_UNSIGNED_INTEGER
    DESCRIPTION                   = "Memory FPGA version (33)"
  END_OBJECT                      = COLUMN

  OBJECT                          = COLUMN
    COLUMN_NUMBER                 = 20
    NAME                          = "MICRO_FPGA_VERSION (34)"
    START_BYTE                    = 51
    BYTES                         = 2
    DATA_TYPE                     = MSB_UNSIGNED_INTEGER
    DESCRIPTION                   = "MICRO FPGA version"
  END_OBJECT                      = COLUMN

  OBJECT                          = COLUMN
    COLUMN_NUMBER                 = 21
    NAME                          = "SPECT_FPGA_VERSION (20)"
    START_BYTE                    = 53
    BYTES                         = 2
    DATA_TYPE                     = MSB_UNSIGNED_INTEGER
    DESCRIPTION                   = "Spectromitor FPGA Version"
  END_OBJECT                      = COLUMN

  OBJECT                          = COLUMN
    COLUMN_NUMBER                 = 22
    NAME                          = "IMAGES_SENT"
    START_BYTE                    = 55
    BYTES                         = 2
```

```
  DATA_TYPE                      = MSB_UNSIGNED_INTEGER
  DESCRIPTION                    = "Images sent To RCE"
END_OBJECT                       = COLUMN

OBJECT                           = COLUMN
  COLUMN_NUMBER                  = 23
  NAME                           = "SOH_SENT"
  START_BYTE                     = 57
  BYTES                          = 2
  DATA_TYPE                      = MSB_UNSIGNED_INTEGER
  DESCRIPTION                    = "Soh Sent To RCE"
END_OBJECT                       = COLUMN

OBJECT                           = COLUMN
  COLUMN_NUMBER                  = 24
  NAME                           = "SPECT_SENT"
  START_BYTE                     = 59
  BYTES                          = 2
  DATA_TYPE                      = MSB_UNSIGNED_INTEGER
  DESCRIPTION                    = "Spectra sent To RCE"
END_OBJECT                       = COLUMN

OBJECT                           = COLUMN
  COLUMN_NUMBER                  = 25
  NAME                           = "LASER_DATA_SENT"
  START_BYTE                     = 61
  BYTES                          = 2
  DATA_TYPE                      = MSB_UNSIGNED_INTEGER
  DESCRIPTION                    = "W,V,A per laser shot"
END_OBJECT                       = COLUMN

OBJECT                           = COLUMN
  COLUMN_NUMBER                  = 26
  NAME                           = "SPARE"
  START_BYTE                     = 63
  BYTES                          = 2
  DATA_TYPE                      = MSB_UNSIGNED_INTEGER
  DESCRIPTION                    = "Boot bank plus eepro initialized data
                                    location"
END_OBJECT                       = COLUMN

OBJECT                           = COLUMN
  COLUMN_NUMBER                  = 27
  NAME                           = "SPARE2"
  START_BYTE                     = 65
  BYTES                          = 2
  DATA_TYPE                      = MSB_UNSIGNED_INTEGER
  DESCRIPTION                    = "marker workd to easily id soh"
END_OBJECT                       = COLUMN

OBJECT                           = COLUMN
  COLUMN_NUMBER                  = 28
  NAME                           = "N_FOLLOWING"
  START_BYTE                     = 67
  BYTES                          = 2
  DATA_TYPE                      = MSB_UNSIGNED_INTEGER
  DESCRIPTION                    = "Number of SOH records"
```

```
  END_OBJECT                     = COLUMN

/* End of format file CCAM_SOH_SCIDATA_COLS_V4.FMT */
```

u) "CCAM_SOH_TO_RCE_CONTAINER" (1st of 2)

```
/* Format file CCAM_SOH_TO_RCE_CONTAINER_V2.FMT                          */
/* Defines columns inside the container CCAM_SOH_TO_RCE in a Chemcam */
/* State of Health Science Data table.                                   */
/* The START_BYTE value for each column is relative to the start     */
/* of the container.                                                     */

    OBJECT                         = COLUMN
      COLUMN_NUMBER                = 1
      NAME                         = "TIME"
      START_BYTE                   = 1
      BYTES                        = 4
      DATA_TYPE                    = MSB_UNSIGNED_INTEGER
      DESCRIPTION                  = "Time collected"
    END_OBJECT                     = COLUMN

    OBJECT                         = COLUMN
      COLUMN_NUMBER                = 2
      NAME                         = "CCAM_DPU_SOH_STRUCT"
      START_BYTE                   = 5
      BYTES                        = 18
      DATA_TYPE                    = MSB_UNSIGNED_INTEGER
      ITEMS                        = 9
      ITEM_BYTES                   = 2
      DESCRIPTION         = "Vector of 9 2-byte unsigned integer values:
           Item 0: DPU Analog Ground ( V = counts * (2.5V/255cnts) )
           Item 1: DPU_+5_Digital    ( V = counts * 2.5 * (2.5V/255cnts) )
           Item 2: DPU_+2.5_Digital  ( V = counts * (2.5/2) * (2.5V/255cnts) )
           Item 3: DPU_+5_Analog     ( V = counts * (2.5V/255cnts) )
           Item 4: DPU_-5_Analog     ( V = -counts * (2.5V/255cnts) )
           Item 5: DPU_degC          ( degC = counts * (2.5V/255cnts) *
                                        (150 degC/2V) - 50 degC )
           Item 6: Spectrometer_degC ( degC = counts * (2.5V/255cnts) *
                                        (150 degC/2V) - 50 degC )
           Item 7: LVPS_degC         ( degC = counts * (2.5v/255cnts) *
                                        (150 degC/2V) - 50 degC )
           Item 8: motor_pos         ( counts  )
           "
    END_OBJECT                     = COLUMN

    OBJECT                         = COLUMN
      COLUMN_NUMBER                = 3
      NAME                         = "CCAM_MU_SOH_STRUCT"
      START_BYTE                   = 23
      BYTES                        = 78
      DATA_TYPE                    = MSB_UNSIGNED_INTEGER
      ITEMS                        = 39
      ITEM_BYTES                   = 2
      DESCRIPTION         = "Vector of 39 2-byte unsigned integer values:
           Item 0: Digital HK bit flags, meanings given for values of 0/1:
                     Bit 0 = limit switch open/closed
```

```
              Bit 1 = thermal flag cold/hot
              Bit 2 = 15/30V converter off/on
              Bit 3 = 12V converter off/on
              Bit 4 = limiter osc off/on
              Bit 5 = limiter amp1 off/on
              Bit 6 = limiter amp2 off/on
              Bit 7 = floating 15V off/on
              Bit 8 = pockels off/on
              Bit 9 = 30V motor off/on
              Bit 10 = CWL off/on
              Bit 11 = 12V autofocus off/on
              Bit 12 = camera off/on
              Bit 13 = osc warm-up off/on
              Bit 14 = amp warm-up off/on
              Bit 15 = CWL warm-up off/on
     Item 1: HK_heatsink_degC (-52.1 - 98.5 degC)
                              (V = VMON * 0.04796 - 57.47)
     Item 2: HK_I_+3.3v (0 -1A)(V = VMON * 0.29858)
     Item 3: HK_I_+30v  (0 - 2A)(V = VMON * 0.22229 - 115.7)
     Item 4: HK_I_-5v   (0 - 250mA)(V = VMON * 0.05670 + 31.6)
     Item 5: HK_I_+12v  (0 - 750mA)(V = VMON * 0.05807)
     Item 6: HK_V_+3.3v (0 - 3.5V)(V = VMON * 0.000898)
     Item 7: HK_V_+5v   (0 - 5.2V)(V = VMON * 0.001492)
     Item 8: HK_V_-5v   (0 - -5.2V)(V = VMON * -0.00163)
     Item 9: HK_V_+12v  (0 - 13V)(V = VMON * 0.003458)
    Item 10: HK_V_-12v  (0 - -13V)(V = VMON * -0.00350)
    Item 11: HK_V_+15v  (0 - 16V)(V = VMON * 0.005696)
    Item 12: HK_V_+30v  (0 - 32V)(V = VMON * 0.007977)
    Item 13: Laser diode control current (0 - 150mA)
                              (V = VMON * 0.002935)
    Item 14: CWL_degC          (-52.1 - 98.5degC)
                              (V = VMON * 0.037226 - 57.47)
    Item 15: HK_I_limiterdegC (-52.1 - 98.5degC)
                              (V = VMON * 0.037226 - 57.47)
    Item 16: Autofocus signal output (0 - 2.5V)
                              (V = VMON * 0.610501)
    Item 17: LMD18200_degC  (-52.1 - 98.5degC)
                              (V = VMON * 0.047958 - 57.47)
    Item 18: HK_Laser1_degC (-52.1 - 98.5degC)
                              (V = VMON * 0.037226 - 57.47)
    Item 19: HK_Laser2_degC (-52.1 - 98.5degC)
                              (V = VMON * 0.037226 - 57.47)
    Item 20: HK_Laser3_degC (-52.1 - 98.5degC)
                              (V = VMON * 0.037226 - 57.47)
    Item 21: HK_Laser4_degC (-52.1 - 98.5degC)
                              (V = VMON * 0.037226 - 57.47)
    Item 22: HK_Stack1_V  (0 - 30V)(V = VMON * 0.00763)
    Item 23: HK_Stack1_I  (0 - 136A)(V = VMON * 0.03817 + 1.925)
    Item 24: HK_Stack2_V  (0 - 30V)(V = VMON * 0.00763)
    Item 25: HK_Stack2_I  (0 - 136A)(V = VMON * 0.03971 - 3.994)
    Item 26: HK_Stack3_V  (0 - 30V)(V = VMON * 0.00763)
    Item 27: HK_Stack3_I  (0 - 136A)(V = VMON * 0.03848 - 0.780)
    Item 28: optical_flux_level
                         (power of LIBS shots iff {2000 - 40095})
                        (-14.57 - 30.06)(V = VMON * 0.01369 - 2.96)
    Item 29: HK_pockels_V (0 - 2418V)(V = VMON * 0.61050)
    Item 30: HK_LIMIT_SWITCH(V = VMON * 0.128798 - 7.73)
```

```
        Item 31: HK_Spare2(V = VMON * 0.000611)
        Item 32: HK_RMI_degC     (-60 - 90degC)(V = VMON * 0.04210 - 60.6)
        Item 33: HK_FPGA_degC    (-52.1 - 98.5degC)(V = VMON * 0.04796 -
57.47)
        Item 34: HK_Telescope1_degC  (-52.1 - 98.5degC)
                                            (V = VMON * 0.037226 - 57.47)
        Item 35: HK_Telescope2_degC  (-52.1 - 98.5degC)
                                            (V = VMON * 0.037226 - 57.47)
        Item 36: FPGA_3.3V  (0 - 7.88V)(V = VMON * 0.001906)
        Item 37: displacement_steps
        Item 38: displacement_count
        "
    END_OBJECT                     = COLUMN

/* End of format file CCAM_SOH_TO_RCE_CONTAINER_V2.FMT */
```

v) "CCAM_SOH_TO_RCE_CONTAINER" (2nd of 2)

```
/* Format file CCAM_SOH_TO_RCE_CONTAINER_V4.FMT                        */
/* Defines columns inside the container CCAM_SOH_TO_RCE in a Chemcam */
/* State of Health Science Data table.                                 */
/* The START_BYTE value for each column is relative to the start      */
/* of the container.                                                   */

    OBJECT                         = COLUMN
      COLUMN_NUMBER                = 1
      NAME                         = "TIME"
      START_BYTE                   = 1
      BYTES                        = 4
      DATA_TYPE                    = MSB_UNSIGNED_INTEGER
      DESCRIPTION                  = "Time collected"
    END_OBJECT                     = COLUMN

    OBJECT                         = COLUMN
      COLUMN_NUMBER                = 2
      NAME                         = "CCAM_DPU_SOH_STRUCT"
      START_BYTE                   = 5
      BYTES                        = 18
      DATA_TYPE                    = MSB_UNSIGNED_INTEGER
      ITEMS                        = 9
      ITEM_BYTES                   = 2
      DESCRIPTION         = "Vector of 9 2-byte unsigned integer values:
          Item 0: DPU Analog Ground ( V = counts * (2.5V/255cnts) )
          Item 1: DPU_+5_Digital    ( V = counts * 2.5 * (2.5V/255cnts) )
          Item 2: DPU_+2.5_Digital  ( V = counts * (2.5/2) * (2.5V/255cnts) )
          Item 3: DPU_+5_Analog     ( V = counts * (2.5V/255cnts) )
          Item 4: DPU_-5_Analog     ( V = -counts * (2.5V/255cnts) )
          Item 5: DPU_degC          ( degC = counts * (2.5V/255cnts) *
                                      (150 degC/2V) - 50 degC )
          Item 6: Spectrometer_degC ( degC = counts * (2.5V/255cnts) *
                                      (150 degC/2V) - 50 degC )
          Item 7: LVPS_degC         ( degC = counts * (2.5v/255cnts) *
                                      (150 degC/2V) - 50 degC )
          Item 8: motor_pos         ( counts  )
          "
    END_OBJECT                     = COLUMN
```

```
OBJECT                        = COLUMN
  COLUMN_NUMBER               = 3
  NAME                        = "CCAM_MU_SOH_STRUCT"
  START_BYTE                  = 23
  BYTES                       = 78
  DATA_TYPE                   = MSB_UNSIGNED_INTEGER
  ITEMS                       = 39
  ITEM_BYTES                  = 2
  DESCRIPTION        = "Vector of 39 2-byte unsigned integer values:
       Item 0: Digital HK bit flags, meanings given for values of 0/1:
                 Bit 0 = limit switch open/closed
                 Bit 1 = thermal flag cold/hot
                 Bit 2 = 15/30V converter off/on
                 Bit 3 = 12V converter off/on
                 Bit 4 = limiter osc off/on
                 Bit 5 = limiter amp1 off/on
                 Bit 6 = limiter amp2 off/on
                 Bit 7 = floating 15V off/on
                 Bit 8 = pockels off/on
                 Bit 9 = 30V motor off/on
                 Bit 10 = CWL off/on
                 Bit 11 = 12V autofocus off/on
                 Bit 12 = camera off/on
                 Bit 13 = osc warm-up off/on
                 Bit 14 = amp warm-up off/on
                 Bit 15 = CWL warm-up off/on
       Item 1: HK_heatsink_degC (-52.1 - 98.5 degC)
                                  (V = VMON * 0.04796 - 57.47)
       Item 2: HK_I_+3.3v (0 -1A)(V = VMON * 0.29858)
       Item 3: HK_I_+30v  (0 - 2A)(V = VMON * 0.22229 - 115.7)
       Item 4: HK_I_-5v   (0 - 250mA)(V = VMON * 0.05670 + 31.6)
       Item 5: HK_I_+12v  (0 - 750mA)(V = VMON * 0.05807)
       Item 6: HK_V_+3.3v (0 - 3.5V)(V = VMON * 0.000898)
       Item 7: HK_V_+5v   (0 - 5.2V)(V = VMON * 0.001492)
       Item 8: HK_V_-5v   (0 - -5.2V)(V = VMON * -0.00163)
       Item 9: HK_V_+12v  (0 - 13V)(V = VMON * 0.003458)
      Item 10: HK_V_-12v  (0 - -13V)(V = VMON * -0.00350)
      Item 11: HK_V_+15v  (0 - 16V)(V = VMON * 0.005696)
      Item 12: HK_V_+30v  (0 - 32V)(V = VMON * 0.007977)
      Item 13: Laser diode control current (0 - 150mA)
                                  (V = VMON * 0.002935)
      Item 14: CWL_degC         (-52.1 - 98.5degC)
                                  (V = VMON * 0.037226 - 57.47)
      Item 15: HK_I_limiterdegC (-52.1 - 98.5degC)
                                  (V = VMON * 0.037226 - 57.47)
      Item 16: Autofocus signal output (0 - 2.5V)
                                  (V = VMON * 0.610501)
      Item 17: LMD18200_degC  (-52.1 - 98.5degC)
                                  (V = VMON * 0.047958 - 57.47)
      Item 18: HK_Laser1_degC (-52.1 - 98.5degC)
                                  (V = VMON * 0.037226 - 57.47)
      Item 19: HK_Laser2_degC (-52.1 - 98.5degC)
                                  (V = VMON * 0.037226 - 57.47)
      Item 20: HK_Laser3_degC (-52.1 - 98.5degC)
                                  (V = VMON * 0.037226 - 57.47)
      Item 21: HK_Laser4_degC (-52.1 - 98.5degC)
```

```
                                   (V = VMON * 0.037226 - 57.47)
         Item 22: HK_Stack1_V  (0 - 30V)(V = VMON * 0.00763)
         Item 23: HK_Stack1_I  (0 - 136A)(V = VMON * 0.03817 + 1.925)
         Item 24: HK_Stack2_V  (0 - 30V)(V = VMON * 0.00763)
         Item 25: HK_Stack2_I  (0 - 136A)(V = VMON * 0.03971 - 3.994)
         Item 26: HK_Stack3_V  (0 - 30V)(V = VMON * 0.00763)
         Item 27: HK_Stack3_I  (0 - 136A)(V = VMON * 0.03848 - 0.780)
         Item 28: optical_flux_level
                              (power of LIBS shots iff {2000 - 40095})
                             (-14.57 - 30.06)(V = VMON * 0.01369 - 2.96)
         Item 29: HK_pockels_V (0 - 2418V)(V = VMON * 0.61050)
         Item 30: HK_LIMIT_SWITCH(V = VMON * 0.128798 - 7.73)
         Item 31: HK_Spare2(V = VMON * 0.000611)
         Item 32: HK_RMI_degC     (-60 - 90degC)(V = VMON * 0.04210 - 60.6)
         Item 33: HK_FPGA_degC    (-52.1 - 98.5degC)(V = VMON * 0.04796 -
57.47)
         Item 34: HK_Telescope1_degC  (-52.1 - 98.5degC)
                                         (V = VMON * 0.037226 - 57.47)
         Item 35: HK_Telescope2_degC  (-52.1 - 98.5degC)
                                         (V = VMON * 0.037226 - 57.47)
         Item 36: FPGA_3.3V  (0 - 7.88V)(V = VMON * 0.001906)
         Item 37: displacement_steps
         Item 38: displacement_count
          "
   END_OBJECT                    = COLUMN

/* End of format file CCAM_SOH_TO_RCE_CONTAINER_V4.FMT */
```

w) “CCAM_SOH_CHECKSUM”

```
/* CCAM_SOH_CHECKSUM_V4.FMT.                                             */

  OBJECT                        = COLUMN
      NAME                      = "CMD_REPLY_CHECKSUM"
      DATA_TYPE                 = MSB_UNSIGNED_INTEGER
      START_BYTE                = 1
      BYTES                     = 4
      REPETITIONS               = 1
      DESCRIPTION               = "Second part of CMD_REPLY data: MD5
checksum"
  END_OBJECT                    = COLUMN

/* End of format file CCAM_SOH_CHECKSUM_V4.FMT */
```

x) “CCAM_ANCILLARY_TMPS” (1st of 2)

```
/* CCAM_ANCILLARY_TMPS_V2.FMT.                                          */

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 2
    NAME              = "MU_OBOX_TELESCOPE_TEMP"
    START_BYTE        = 5
    BYTES             = 4
    DATA_TYPE         = IEEE_REAL
    DESCRIPTION       = "Temperature from MAST_UNIT OPTICAL BOX TELESCOPE"
```

```
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 3
    NAME              = "MU_OBOX_TELESCOPE_TEMP_STATUS"
    START_BYTE        = 9
    BYTES             = 4
    DATA_TYPE         = MSB_INTEGER
    DESCRIPTION       = "Temperature status from MAST_UNIT OPTICAL BOX
TELESCOPE"
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 4
    NAME              = "MU_LASER_IF_TEMP"
    START_BYTE        = 13
    BYTES             = 4
    DATA_TYPE         = IEEE_REAL
    DESCRIPTION       = "Temperature from MAST_UNIT LASER INTERFACE"
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 5
    NAME              = "MU_LASER_IF_TEMP_STATUS"
    START_BYTE        = 17
    BYTES             = 4
    DATA_TYPE         = MSB_INTEGER
    DESCRIPTION       = "Temperature status from MAST_UNIT LASER INTERFACE"
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 6
    NAME              = "MU_EBOX_HEATSINK_TEMP"
    START_BYTE        = 21
    BYTES             = 4
    DATA_TYPE         = IEEE_REAL
    DESCRIPTION       = "Temperature from MAST_UNIT EBOX HEATSINK"
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 7
    NAME              = "MU_EBOX_HEATSINK_TEMP_STATUS"
    START_BYTE        = 25
    BYTES             = 4
    DATA_TYPE         = MSB_INTEGER
    DESCRIPTION       = "Temperature status from MAST_UNIT EBOX HEATSINK"
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 8
    NAME              = "MU_EBOX_FPGA_TEMP"
    START_BYTE        = 29
    BYTES             = 4
    DATA_TYPE         = IEEE_REAL
    DESCRIPTION       = "Temperature from MAST_UNIT EBOX FPGA"
  END_OBJECT          = COLUMN
```

```
OBJECT              = COLUMN
  COLUMN_NUMBER     = 9
  NAME              = "MU_EBOX_FPGA_TEMP_STATUS"
  START_BYTE        = 33
  BYTES             = 4
  DATA_TYPE         = MSB_INTEGER
  DESCRIPTION       = "Temperature status from MAST_UNIT EBOX FPGA"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 10
  NAME              = "BU_EBOX_TEMP"
  START_BYTE        = 37
  BYTES             = 4
  DATA_TYPE         = IEEE_REAL
  DESCRIPTION       = "Temperature from BODY_UNIT  cover B-Side"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 11
  NAME              = "BU_EBOX_TEMP_STATUS"
  START_BYTE        = 41
  BYTES             = 4
  DATA_TYPE         = MSB_INTEGER
  DESCRIPTION       = "Temperature status from BODY_UNIT "
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 12
  NAME              = "BU_SPEC_TOP_TEMP"
  START_BYTE        = 45
  BYTES             = 4
  DATA_TYPE         = IEEE_REAL
  DESCRIPTION       = "Temperature from BODY_UNIT SPECTROMETERS TOP"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 13
  NAME              = "BU_SPEC_TOP_TEMP_STATUS"
  START_BYTE        = 49
  BYTES             = 4
  DATA_TYPE         = MSB_INTEGER
  DESCRIPTION       = "Temperature status from BODY_UNIT Spectrometer TOP"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 14
  NAME              = "BU_SPEC_BOTTOM_TEMP"
  START_BYTE        = 53
  BYTES             = 4
  DATA_TYPE         = IEEE_REAL
  DESCRIPTION       = "Temperature from BODY_UNIT BOTTOM"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 15
  NAME              = "BU_SPEC_BOTTOM_TEMP_STATUS"
```

```
    START_BYTE        = 57
    BYTES             = 4
    DATA_TYPE         = MSB_INTEGER
    DESCRIPTION       = "Temperature status from BODY_UNIT BOTTOM"
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 16
    NAME              = "BU_DEMUX_TEMP"
    START_BYTE        = 61
    BYTES             = 4
    DATA_TYPE         = IEEE_REAL
    DESCRIPTION       = "Temperature from BODY_UNIT DEMUX"
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 17
    NAME              = "BU_DEMUX_TEMP_STATUS"
    START_BYTE        = 65
    BYTES             = 4
    DATA_TYPE         = MSB_INTEGER
    DESCRIPTION       = "Temperature status from BODY_UNIT DEMUX"
  END_OBJECT          = COLUMN


/* End of format file CCAM_ANCILLARY_TMPS_V2.FMT. */
```

y) "CCAM_ANCILLARY_TMPS" (2nd of 2)

```
/* CCAM_ANCILLARY_TMPS_V4.FMT.                                             */

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 1
    NAME              = "MU_OBOX_TELESCOPE_TEMP"
    START_BYTE        = 1
    BYTES             = 4
    DATA_TYPE         = IEEE_REAL
    DESCRIPTION       = "Temperature from MAST_UNIT OPTICAL BOX TELESCOPE"
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 2
    NAME              = "MU_OBOX_TELESCOPE_TEMP_STATUS"
    START_BYTE        = 5
    BYTES             = 4
    DATA_TYPE         = MSB_INTEGER
    DESCRIPTION       = "Temperature status from MAST_UNIT OPTICAL BOX
TELESCOPE"
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 3
    NAME              = "MU_LASER_IF_TEMP"
    START_BYTE        = 9
    BYTES             = 4
    DATA_TYPE         = IEEE_REAL
```

```
  DESCRIPTION      = "Temperature from MAST_UNIT LASER INTERFACE"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 4
  NAME             = "MU_LASER_IF_TEMP_STATUS"
  START_BYTE       = 13
  BYTES            = 4
  DATA_TYPE        = MSB_INTEGER
  DESCRIPTION      = "Temperature status from MAST_UNIT LASER INTERFACE"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 5
  NAME             = "MU_EBOX_HEATSINK_TEMP"
  START_BYTE       = 17
  BYTES            = 4
  DATA_TYPE        = IEEE_REAL
  DESCRIPTION      = "Temperature from MAST_UNIT EBOX HEATSINK"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 6
  NAME             = "MU_EBOX_HEATSINK_TEMP_STATUS"
  START_BYTE       = 21
  BYTES            = 4
  DATA_TYPE        = MSB_INTEGER
  DESCRIPTION      = "Temperature status from MAST_UNIT EBOX HEATSINK"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 7
  NAME             = "MU_EBOX_FPGA_TEMP"
  START_BYTE       = 25
  BYTES            = 4
  DATA_TYPE        = IEEE_REAL
  DESCRIPTION      = "Temperature from MAST_UNIT EBOX FPGA"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 8
  NAME             = "MU_EBOX_FPGA_TEMP_STATUS"
  START_BYTE       = 29
  BYTES            = 4
  DATA_TYPE        = MSB_INTEGER
  DESCRIPTION      = "Temperature status from MAST_UNIT EBOX FPGA"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 9
  NAME             = "BU_CCD_VNIR_B_TEMP"
  START_BYTE       = 33
  BYTES            = 4
  DATA_TYPE        = IEEE_REAL
  DESCRIPTION      = "Temperature from BODY_UNIT VNIR CCD cover B-Side"
END_OBJECT         = COLUMN
```

```
OBJECT              = COLUMN
  COLUMN_NUMBER     = 10
  NAME              = "BU_CCD_VNIR_B_TEMP_STATUS"
  START_BYTE        = 37
  BYTES             = 4
  DATA_TYPE         = MSB_INTEGER
  DESCRIPTION       = "Temperature status from BODY_UNIT VNIR CCD cover"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 11
  NAME              = "BU_SPEC_B_TEMP"
  START_BYTE        = 41
  BYTES             = 4
  DATA_TYPE         = IEEE_REAL
  DESCRIPTION       = "Temperature from BODY_UNIT SPECTROMETERS B-side"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 12
  NAME              = "BU_SPEC_B_TEMP_STATUS"
  START_BYTE        = 45
  BYTES             = 4
  DATA_TYPE         = MSB_INTEGER
  DESCRIPTION       = "Temperature status from BODY_UNIT Spectrometer B-side"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 13
  NAME              = "BU_CCD_UV_A_TEMP"
  START_BYTE        = 49
  BYTES             = 4
  DATA_TYPE         = IEEE_REAL
  DESCRIPTION       = "Temperature from BODY_UNIT UV CCD A-side"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 14
  NAME              = "BU_CCD_UV_A_TEMP_STATUS"
  START_BYTE        = 53
  BYTES             = 4
  DATA_TYPE         = MSB_INTEGER
  DESCRIPTION       = "Temperature status from BODY_UNIT UV CCD A-Side"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 15
  NAME              = "BU_SPEC_A_TEMP"
  START_BYTE        = 57
  BYTES             = 4
  DATA_TYPE         = IEEE_REAL
  DESCRIPTION       = "Temperature from BODY_UNIT SPECTROMETERS B-side"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 16
  NAME              = "BU_SPEC_A_TEMP_STATUS"
```

```
    START_BYTE      = 61
    BYTES           = 4
    DATA_TYPE       = MSB_INTEGER
    DESCRIPTION     = "Temperature status from BODY_UNIT SPECTROMETERS B-side"
  END_OBJECT        = COLUMN

  OBJECT            = COLUMN
    COLUMN_NUMBER   = 17
    NAME            = "BU_DEMUX_A_TEMP"
    START_BYTE      = 65
    BYTES           = 4
    DATA_TYPE       = IEEE_REAL
    DESCRIPTION     = "Temperature from BODY_UNIT Demux A-side"
  END_OBJECT        = COLUMN

  OBJECT            = COLUMN
    COLUMN_NUMBER   = 18
    NAME            = "BU_DEMUX_A_TEMP_STATUS"
    START_BYTE      = 69
    BYTES           = 4
    DATA_TYPE       = MSB_INTEGER
    DESCRIPTION     = "Temperature status from BODY_UNIT Demux A-side"
  END_OBJECT        = COLUMN

  OBJECT            = COLUMN
    COLUMN_NUMBER   = 19
    NAME            = "BU_DEMUX_B_TEMP"
    START_BYTE      = 73
    BYTES           = 4
    DATA_TYPE       = IEEE_REAL
    DESCRIPTION     = "Temperature from BODY_UNIT Demux B-side"
  END_OBJECT        = COLUMN

  OBJECT            = COLUMN
    COLUMN_NUMBER   = 20
    NAME            = "BU_DEMUX_B_TEMP_STATUS"
    START_BYTE      = 77
    BYTES           = 4
    DATA_TYPE       = MSB_INTEGER
    DESCRIPTION     = "Temperature status from BODY_UNIT Demux B-side"
  END_OBJECT        = COLUMN

/* End of format file CCAM_ANCILLARY_TMPS_V4.FMT. */
```

z) “CCAM_CMD_REPLY”

```
/* CCAM_CMD_REPLY_V4.FMT.                                        */


  /* This format file Does not include checksum.       */

  OBJECT                        = COLUMN
      NAME                       = "CMD_REPLY_BYTESIZE"
      DATA_TYPE                  = MSB_UNSIGNED_INTEGER
      START_BYTE                 = 1
      BYTES                      = 4
```

```
    DESCRIPTION                   = "Size of cmd_reply packet"
  END_OBJECT                      = COLUMN

  OBJECT                          = CONTAINER
    NAME                          = "CCAM_CMD_REPLY"
    ^STRUCTURE                    = "CCAM_CMD_REPLY_CONTAINER_V4.FMT"
    START_BYTE                    = 5
    BYTES                         = 4
    REPETITIONS                   = 1
    DESCRIPTION                   = "ChemCam CMD_REPLY frame"
  END_OBJECT                      = CONTAINER


/* End of format file CCAM_CMD_REPLY_V4.FMT */
```

aa) "CCAM_CMD_REPLY_CONTAINER" (1st of 2)

```
/* CCAM_CMD_REPLY_CONTAINER_V2.FMT                                               */
 OBJECT                 = COLUMN
  OBJECT                = BIT_COLUMN
   NAME                 = CMD_REPLY_OPCODE
   BIT_DATA_TYPE        = MSB_UNSIGNED_INTEGER
   START_BIT            = 1
   BITS                 = 8
   DESCRIPTION          = "0xdd - spectrometer data
                           0xc0 - laser data
                           0x77 - rmi image data
                           0xc2 - autofocus data
                           0x55aa - soh data"
  END_OBJECT            = BIT_COLUMN

  OBJECT                = BIT_COLUMN
   NAME                 = CMD_REPLY_ERROR_CONTROL
   BIT_DATA_TYPE        = MSB_UNSIGNED_INTEGER
   START_BIT            = 9
   BITS                 = 2
   DESCRIPTION          = "Error control :
                           0 = reserved
                           1 =  The CRC error control algorithm is applied
                           2 = The Checksum error control Algorithm was applied
                           3 = No error Control"
  END_OBJECT            = BIT_COLUMN

  OBJECT                = BIT_COLUMN
   NAME                 = CMD_REPLY_DATA_PRESENT
   BIT_DATA_TYPE        = BOOLEAN
   START_BIT            = 11
   BITS                 = 1
   DESCRIPTION          = "Is data present
                           0 = nodata
                           1 = data"
  END_OBJECT            = BIT_COLUMN

  OBJECT                = BIT_COLUMN
   NAME                 = CMD_REPLY_STATUS_FLAGS
   BIT_DATA_TYPE        = MSB_UNSIGNED_INTEGER
```

```
    START_BIT         = 12
    BITS              = 21
    DESCRIPTION       = "Status flags:
                       bit 1 - command Reply Flag
                               (0- not cmd_reply, 1 - cmd_reply)
                       bit 2:16 - described in attached document under
                               flags tab
                       bit 17:21 -
                                0 0000 - reserved
                                0 0001 - First time command
                                0 0010 - Retry command (cmd reply frame)
                                1 0010 - Retry command (sci frame) "
  END_OBJECT          = BIT_COLUMN
END_OBJECT            = COLUMN

/* End of format file CCAM_CMD_REPLY_CONTAINER_V2.FMT */
```

bb) "CCAM_CMD_REPLY_CONTAINER" (2nd of 2)

```
/* CCAM_CMD_REPLY_CONTAINER_V4.FMT                                         */
 OBJECT               = COLUMN
   NAME               = "CMD_REPLY BIT STRINGS"
   DATA_TYPE          = MSB_BIT_STRING
   START_BYTE         = 1
   BYTES              = 4

   OBJECT             = BIT_COLUMN
     NAME             = CMD_REPLY_OPCODE
     BIT_DATA_TYPE    = MSB_UNSIGNED_INTEGER
     START_BIT        = 1
     BITS             = 8
     DESCRIPTION      = "0xdd - spectrometer data
                         0xc0 - laser data
                         0x77 - rmi image data
                         0xc2 - autofocus data
                         0x55aa - soh data"
   END_OBJECT         = BIT_COLUMN

   OBJECT             = BIT_COLUMN
     NAME             = CMD_REPLY_ERROR_CONTROL
     BIT_DATA_TYPE    = MSB_UNSIGNED_INTEGER
     START_BIT        = 9
     BITS             = 2
     DESCRIPTION      = "Error control :
                         0 = reserved
                         1 =  The CRC error control algorithm is applied
                         2 = The Checksum error control Algorithm was applied
                         3 = No error Control"
   END_OBJECT         = BIT_COLUMN

   OBJECT             = BIT_COLUMN
     NAME             = CMD_REPLY_DATA_PRESENT
     BIT_DATA_TYPE    = BOOLEAN
     START_BIT        = 11
     BITS             = 1
     DESCRIPTION      = "Is data present
```

```
                          0 = nodata
                          1 = data"
   END_OBJECT           = BIT_COLUMN

   OBJECT               = BIT_COLUMN
     NAME               = CMD_REPLY_STATUS_FLAGS
     BIT_DATA_TYPE      = MSB_UNSIGNED_INTEGER
     START_BIT          = 12
     BITS               = 21
     DESCRIPTION        = "Status flags:
                        bit 1 - command Reply Flag
                                 (0- not cmd_reply, 1 - cmd_reply)
                        bit 2:16 - described in attached document under
                                 flags tab
                        bit 17:21 -
                                  0 0000 - reserved
                                  0 0001 - First time command
                                  0 0010 - Retry command (cmd reply frame)
                                  1 0010 - Retry command (sci frame) "
   END_OBJECT           = BIT_COLUMN
 END_OBJECT             = COLUMN

/* End of format file CCAM_CMD_REPLY_CONTAINER_V4.FMT */
```

cc) "CCAM_CMD_REPLY_CHECKSUM"

```
/* CCAM_CMD_REPLY_CHECKSUM_V4.FMT.                                          */


  /* This format file includes checksum.                                    */
  /* if CMD_REPLY_ERROR_CONTROL from CCAM_CMD_REPLY_CONTAINER_V4.FMT        */
  /* eq 1 or 2                                                              */

  OBJECT                          = COLUMN
      NAME                         = "CMD_REPLY_BYTESIZE"
      DATA_TYPE                    = MSB_UNSIGNED_INTEGER
      START_BYTE                   = 1
      BYTES                        = 4
      DESCRIPTION                  = "Size of cmd_reply packet"
   END_OBJECT                      = COLUMN

   OBJECT                          = CONTAINER
      NAME                         = "CCAM_CMD_REPLY"
      ^STRUCTURE                   = "CCAM_CMD_REPLY_CONTAINER_V4.FMT"
      START_BYTE                   = 5
      BYTES                        = 4
      REPETITIONS                  = 1
      DESCRIPTION                  = "ChemCam CMD_REPLY frame"
   END_OBJECT                      = CONTAINER

  OBJECT                          = COLUMN
      NAME                         = "CMD_REPLY_CHECKSUM"
      DATA_TYPE                    = MSB_UNSIGNED_INTEGER
      START_BYTE                   = 9
      BYTES                        = 4
```

```
      DESCRIPTION                = "Second part of CMD_REPLY data: MD5
checksum"
  END_OBJECT                    = COLUMN


/* End of format file CCAM_CMD_REPLY_CHECKSUM_V4.FMT */
```

dd) "CCAM_LASER_HEADER_COLS" (1st of 2)

```
/* CCAM_LASER_HEADER_COLS_V2.FMT.                                       */

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 1
    NAME              = "TOTAL_BYTECOUNT"
    START_BYTE        = 1
    BYTES             = 4
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    DESCRIPTION       = "Bytes to follow"
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 2
    NAME              = "RCECONTROL"
    START_BYTE        = 5
    BYTES             = 4
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    DESCRIPTION       = "RCEControl
                        contains: (8bits) Opcode 80 for laserdata
                                  (2bits) Error Control Type Flag
                          00 = reserved
                          01 =  The CRC error control algorithm is applied
                          10 = The Checksum error control Algorithm was
applied
                          11 = No error Control"
                                  (1 bit) Data Present
                          0 = nodata
                          1 = data"
                                  (21 bits) Status flags (as follows)
                        bit 1 - command Reply Flag
                                 (0- not cmd_reply, 1 - cmd_reply)
                        bit 2 - notSafe
                                 ( 0 - safe
                                   1 - Bu doesn't know that MU mortors
                                     are in sun-safe position)
                        bit 3 - BootSource
                                 (Which memory was used for boot
                                  0 - Init0 PROMO
                                  1 - Init1 PROM1)
                        bit 4 - CommSide
                                 (Talking with which RCE?
                                  0 - RCE A or none
                                  1 - RCE B)
                        bit 5 - CWL_heater_notOn
                                 (CWL heating loop algorithm off
                                  0 - on
                                  1 - off)
```

```
                        bit 6 - Amp_heater_notOn
                                 ( Amp heating loop algorithm off
                                  0 - on
                                  1 - off)
                        bit 7 - OSC_heater_notOn
                                 ( Osc heating loop algorithm off
                                  0 - on
                                  1 - off)
                        bit 8 - RMI_notOn
                                 ( RMI is not Powered
                                  0 - on
                                  1 - off)
                        bit 9 - RMIdata_notOK
                                 ( RMI off or comm with RMI not OK
                                  0 - OK
                                  1 - not OK, =1
                                      when RMI if off)
                        bit 10 - Spectrometer_notOK
                                 ( not used should be 0)
                        bit 11 - LVPS_notOn
                                 ( Low Voltage (spectrometer)power supply
                                   0 - on
                                   1 - off )
                        bit 12 - MAST_notOK
                                 ( Bad communication with the mast unit
                                   0 - OK
                                   1 - not OK)
                        bit 13 - LIBS_HV_notOn
                                 ( High voltage to LIBS not on
                                   0 - on
                                   1 - not on)
                        bit 14 - LIBS_notReady
                                 (LIBS not in operating temperature range
                                   0 - ready
                                   1 - not ready)
                        bit 15 - CWL_notReady
                                 (CWL not in operating temperature range
                                   0 - ready
                                   1 - not ready)
                        bit 16 - SelfTestFailed
                                  (not used should be 0)
                        bit 17:21 -
                                  0 0000 - reserved
                                  0 0001 - First time command
                                  0 0010 - Retry command (cmd reply frame)
                                  1 0010 - Retry command (sci frame) "
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 3
  NAME              = "DATA_BYTECOUNT"
  START_BYTE        = 9
  BYTES             = 4
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Number of bytes "
END_OBJECT          = COLUMN
```

```
  OBJECT               = COLUMN
    COLUMN_NUMBER      = 4
    NAME               = "DATAID"
    START_BYTE         = 13
    BYTES              = 1
    DATA_TYPE          = BYTE
    DESCRIPTION        = "id byte 0xc0 for LASERDATA"
  END_OBJECT           = COLUMN

  OBJECT               = COLUMN
    COLUMN_NUMBER      = 5
    NAME               = "HEAD2"
    START_BYTE         = 14
    BYTES              = 1
    DATA_TYPE          = BYTE
    DESCRIPTION        = "Not Used - 0"
  END_OBJECT           = COLUMN

  OBJECT               = COLUMN
    COLUMN_NUMBER      = 6
    NAME               = "HEAD3"
    START_BYTE         = 15
    BYTES              = 1
    DATA_TYPE          = BYTE
    DESCRIPTION        = "Not Used - 0"
  END_OBJECT           = COLUMN

  OBJECT               = COLUMN
    COLUMN_NUMBER      = 7
    NAME               = "HEAD4"
    START_BYTE         = 16
    BYTES              = 1
    DATA_TYPE          = BYTE
    DESCRIPTION        = "Not Used - 0"
  END_OBJECT           = COLUMN

  OBJECT               = COLUMN
    COLUMN_NUMBER      = 8
    NAME               = "DPBYTECOUNT"
    START_BYTE         = 17
    BYTES              = 4
    DATA_TYPE          = MSB_UNSIGNED_INTEGER
    DESCRIPTION        = "Number of bytes following in DataProduct"
  END_OBJECT           = COLUMN

  OBJECT               = COLUMN
    COLUMN_NUMBER      = 9
    NAME               = "MILLITIME"
    START_BYTE         = 21
    BYTES              = 4
    DATA_TYPE          = MSB_UNSIGNED_INTEGER
    DESCRIPTION        = "msecs since last rce time set "
  END_OBJECT           = COLUMN

/* End of format file CCAM_LASER_HEADER_COLS_V2.FMT. */
```

ee) "CCAM_LASER_HEADER_COLS" (2nd of 2)

```
/* CCAM_LASER_HEADER_COLS_V4.FMT.                                        */

  OBJECT                = COLUMN
    COLUMN_NUMBER       = 1
    NAME                = "TOTAL_BYTECOUNT"
    START_BYTE          = 1
    BYTES               = 4
    DATA_TYPE           = MSB_UNSIGNED_INTEGER
    DESCRIPTION         = "Bytes to follow"
  END_OBJECT            = COLUMN

  OBJECT                = COLUMN
    COLUMN_NUMBER       = 2
    NAME                = "RCECONTROL"
    START_BYTE          = 5
    BYTES               = 4
    DATA_TYPE           = MSB_UNSIGNED_INTEGER
    DESCRIPTION         = "RCEControl
                            contains: (8bits) Opcode 80 for laserdata
                                      (2bits) Error Control Type Flag
                              00 = reserved
                              01 =  The CRC error control algorithm is applied
                              10 = The Checksum error control Algorithm was
applied
                              11 = No error Control
                                      (1 bit) Data Present
                              0 = nodata
                              1 = data
                                      (21 bits) Status flags (as follows)
                            bit 1 - command Reply Flag
                                     (0- not cmd_reply, 1 - cmd_reply)
                            bit 2 - notSafe
                                     ( 0 - safe
                                       1 - Bu doesn't know that MU mortors
                                         are in sun-safe position)
                            bit 3 - BootSource
                                     (Which memory was used for boot
                                      0 - Init0 PROMO
                                      1 - Init1 PROM1)
                            bit 4 - CommSide
                                     (Talking with which RCE?
                                      0 - RCE A or none
                                      1 - RCE B)
                            bit 5 - CWL_heater_notOn
                                     (CWL heating loop algorithm off
                                      0 - on
                                      1 - off)
                            bit 6 - Amp_heater_notOn
                                     ( Amp heating loop algorithm off
                                      0 - on
                                      1 - off)
                            bit 7 - OSC_heater_notOn
                                     ( Osc heating loop algorithm off
                                      0 - on
```

```
                                     1 - off)
                          bit 8 - RMI_notOn
                                    ( RMI is not Powered
                                     0 - on
                                     1 - off)
                          bit 9 - RMIdata_notOK
                                    ( RMI off or comm with RMI not OK
                                     0 - OK
                                     1 - not OK, =1
                                         when RMI if off)
                          bit 10 - Spectrometer_notOK
                                    ( not used should be 0)
                          bit 11 - LVPS_notOn
                                    ( Low Voltage (spectrometer)power supply
                                      0 - on
                                      1 - off )
                          bit 12 - MAST_notOK
                                    ( Bad communication with the mast unit
                                      0 - OK
                                      1 - not OK)
                          bit 13 - LIBS_HV_notOn
                                    ( High voltage to LIBS not on
                                      0 - on
                                      1 - not on)
                          bit 14 - LIBS_notReady
                                    (LIBS not in operating temperature range
                                      0 - ready
                                      1 - not ready)
                          bit 15 - CWL_notReady
                                    (CWL not in operating temperature range
                                      0 - ready
                                      1 - not ready)
                          bit 16 - SelfTestFailed
                                     (not used should be 0)
                          bit 17:21 -
                                     0 0000 - reserved
                                     0 0001 - First time command
                                     0 0010 - Retry command (cmd reply frame)
                                     1 0010 - Retry command (sci frame) "
END_OBJECT            = COLUMN

OBJECT                = COLUMN
  COLUMN_NUMBER       = 3
  NAME                = "DATA_BYTECOUNT"
  START_BYTE          = 9
  BYTES               = 4
  DATA_TYPE           = MSB_UNSIGNED_INTEGER
  DESCRIPTION         = "Number of bytes "
END_OBJECT            = COLUMN

OBJECT                = COLUMN
  COLUMN_NUMBER       = 4
  NAME                = "DATAID"
  START_BYTE          = 13
  BYTES               = 1
  DATA_TYPE           = MSB_UNSIGNED_INTEGER
  DESCRIPTION         = "id byte 0xc0 for LASERDATA"
```

```
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 5
    NAME              = "HEAD2"
    START_BYTE        = 14
    BYTES             = 1
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    DESCRIPTION       = "Not Used - 0"
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 6
    NAME              = "HEAD3"
    START_BYTE        = 15
    BYTES             = 1
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    DESCRIPTION       = "Not Used - 0"
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 7
    NAME              = "HEAD4"
    START_BYTE        = 16
    BYTES             = 1
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    DESCRIPTION       = "Not Used - 0"
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 8
    NAME              = "DPBYTECOUNT"
    START_BYTE        = 17
    BYTES             = 4
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    DESCRIPTION       = "Number of bytes following in DataProduct"
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 9
    NAME              = "MILLITIME"
    START_BYTE        = 21
    BYTES             = 4
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    DESCRIPTION       = "msecs since last rce time set "
  END_OBJECT          = COLUMN

/* End of format file CCAM_LASER_HEADER_COLS_V4.FMT. */
```

ff) “CCAM_LASER_SCIDATA_COLS” (1st of 2)

```
/* CCAM_LASER_SCIDATA_COLS_V2.FMT.                                        */

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 1
    NAME              = "HK_V_STACK_1"
```

```
    START_BYTE       = 1
    BYTES            = 2
    DATA_TYPE        = MSB_UNSIGNED_INTEGER
    DESCRIPTION      = "Voltage oscillator (stack 1)"
  END_OBJECT         = COLUMN

  OBJECT             = COLUMN
    COLUMN_NUMBER    = 2
    NAME             = "HK_I_STACK_1"
    START_BYTE       = 3
    BYTES            = 2
    DATA_TYPE        = MSB_UNSIGNED_INTEGER
    DESCRIPTION      = "Current oscillator (stack 1)"
  END_OBJECT         = COLUMN

  OBJECT             = COLUMN
    COLUMN_NUMBER    = 3
    NAME             = "HK_V_STACK_2"
    START_BYTE       = 5
    BYTES            = 2
    DATA_TYPE        = MSB_UNSIGNED_INTEGER
    DESCRIPTION      = "Voltage amplifier 1 (stack 2) "
  END_OBJECT         = COLUMN

  OBJECT             = COLUMN
    COLUMN_NUMBER    = 4
    NAME             = "HK_I_STACK_2"
    START_BYTE       = 7
    BYTES            = 2
    DATA_TYPE        = MSB_UNSIGNED_INTEGER
    DESCRIPTION      = "Current amplifier 1 (stack 2)"
  END_OBJECT         = COLUMN

  OBJECT             = COLUMN
    COLUMN_NUMBER    = 5
    NAME             = "HK_V_STACK_3"
    START_BYTE       = 9
    BYTES            = 2
    DATA_TYPE        = MSB_UNSIGNED_INTEGER
    DESCRIPTION      = "Voltage amplifier 2 (stack 3)"
  END_OBJECT         = COLUMN

  OBJECT             = COLUMN
    COLUMN_NUMBER    = 6
    NAME             = "HK_I_STACK_3"
    START_BYTE       = 11
    BYTES            = 2
    DATA_TYPE        = MSB_UNSIGNED_INTEGER
    DESCRIPTION      = "Current amplifier 2 (stack 3)"
  END_OBJECT         = COLUMN

  OBJECT             = COLUMN
    COLUMN_NUMBER    = 7
    NAME             = "OPTICAL_FLUX_LEVEL"
    START_BYTE       = 13
    BYTES            = 2
    DATA_TYPE        = MSB_UNSIGNED_INTEGER
```

```
    DESCRIPTION       = "optical_flux_level,Power of laser shots"
  END_OBJECT          = COLUMN

/* End of format file CCAM_LASER_SCIDATA_COLS_V2.FMT. */
```

gg) "CCAM_LASER_SCIDATA_COLS" (2nd of 2)

```
/* CCAM_LASER_SCIDATA_COLS_V4.FMT.                                       */

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 1
    NAME              = "HK_V_STACK_1"
    START_BYTE        = 1
    BYTES             = 2
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    DESCRIPTION       = "Voltage oscillator (stack 1)"
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 2
    NAME              = "HK_I_STACK_1"
    START_BYTE        = 3
    BYTES             = 2
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    DESCRIPTION       = "Current oscillator (stack 1)"
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 3
    NAME              = "HK_V_STACK_2"
    START_BYTE        = 5
    BYTES             = 2
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    DESCRIPTION       = "Voltage amplifier 1 (stack 2) "
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 4
    NAME              = "HK_I_STACK_2"
    START_BYTE        = 7
    BYTES             = 2
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    DESCRIPTION       = "Current amplifier 1 (stack 2)"
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 5
    NAME              = "HK_V_STACK_3"
    START_BYTE        = 9
    BYTES             = 2
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    DESCRIPTION       = "Voltage amplifier 2 (stack 3)"
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 6
```

```
    NAME              = "HK_I_STACK_3"
    START_BYTE        = 11
    BYTES             = 2
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    DESCRIPTION       = "Current amplifier 2 (stack 3)"
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 7
    NAME              = "OPTICAL_FLUX_LEVEL"
    START_BYTE        = 13
    BYTES             = 2
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    DESCRIPTION       = "optical_flux_level,Power of laser shots"
  END_OBJECT          = COLUMN

/* End of format file CCAM_LASER_SCIDATA_COLS_V4.FMT. */
```

hh) “CCAM_RPIX_CMD_ARG_PARAMS”

```
/* CCAM_RPIX_CMD_ARG_PARAMS_V4.FMT                                        */

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 1
    NAME              = "FRAME_ID"
    START_BYTE        = 1
    BYTES             = 4
    DATA_TYPE         = MSB_INTEGER
    DESCRIPTION       = "FRAME in which the RSM pointing coords are specified.
                         This argument is irrelevant if coord_type = JOINTS_*."
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 2
    NAME              = "FRAME_INDEX"
    START_BYTE        = 5
    BYTES             = 2
    DATA_TYPE         = MSB_INTEGER
    DESCRIPTION       = "Index of the chosen frame(n/a for many frame_IDs)."
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 3
    NAME              = "COORD_TYPE"
    START_BYTE        = 7
    BYTES             = 4
    DATA_TYPE         = MSB_INTEGER
    DESCRIPTION       = "Types of coordinates specified by coord1, coord2,
                         coord3 args."
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 4
    NAME              = "COORD_1"
    START_BYTE        = 11
    BYTES             = 4
```

```
  DATA_TYPE        = IEEE_REAL
  DESCRIPTION      = "X or AZ coordinate for pointing in frame"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 5
  NAME             = "COORD_2"
  START_BYTE       = 15
  BYTES            = 4
  DATA_TYPE        = IEEE_REAL
  DESCRIPTION      = "Y or EL coordinate for pointing in frame"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 6
  NAME             = "COORD_3"
  START_BYTE       = 19
  BYTES            = 4
  DATA_TYPE        = IEEE_REAL
  DESCRIPTION      = "Z or N/A coordinate for pointing in frame"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 7
  NAME             = "FOCUS"
  START_BYTE       = 23
  BYTES            = 4
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Focus algorithm:
                 0 = NO_FOCUS - Dosen't move focus (others mark CCAM unsafe)
                 1 = BASELINE - use CWL to find optimal focus position
                 3 = MANUAL - positions focus based on range argument
                 4 = AF_OFFSET - applies RMI offset
                     from last autofocus solution"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 8
  NAME             = "RANGE"
  START_BYTE       = 27
  BYTES            = 2
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Distance to target for MANUAL focus, seed for BASELINE"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 9
  NAME             = "EXPOSURE_TYPE"
  START_BYTE       = 29
  BYTES            = 4
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "AUTO; MANUAL MANUAL uses exposure time argument"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 10
  NAME             = "EXPOSURE_TIME"
```

```
  START_BYTE       = 33
  BYTES            = 2
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Exposure time for MANUAL exposure,
                      seed time for BASELINE"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 11
  NAME             = "START_C_PIXEL"
  START_BYTE       = 35
  BYTES            = 2
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Starting coord for sub-framing (ROI)"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 12
  NAME             = "START_R_PIXEL"
  START_BYTE       = 37
  BYTES            = 2
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Starting coord for sub-framing (ROI)"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 13
  NAME             = "C_HEIGHT"
  START_BYTE       = 39
  BYTES            = 2
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Vertical size for sub-framing (ROI)"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 14
  NAME             = "R_HEIGHT"
  START_BYTE       = 41
  BYTES            = 2
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Horizontal size for sub-framing (ROI)"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 15
  NAME             = "COMPRESSION"
  START_BYTE       = 43
  BYTES            = 4
  DATA_TYPE        = MSB_INTEGER
  DESCRIPTION      = "Compression to apply to the sub-framed image:
                      0 = No RMI compression
                      1 = LOCO
                      2 = ICER_1BPP"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 16
```

```
  NAME              = "LINK_TO_USE"
  START_BYTE        = 47
  BYTES             = 1
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "CCMU-CCBU link:
                       0 = synchronous
                       1 = asynchronous
                       currently unavailable for RMI image data transfer"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 17
  NAME              = "UPPER_THRESHOLD"
  START_BYTE        = 48
  BYTES             = 2
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Pixel values outside this range are not included in
                       good pixel count"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 18
  NAME              = "LOWER_THRESHOLD"
  START_BYTE        = 50
  BYTES             = 2
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Pixel values outside this range are not included in
                       good pixel count"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 19
  NAME              = "START_IMAGE_ID"
  START_BYTE        = 52
  BYTES             = 1
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Selectable in case the SRAM in MU goes bad."
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 20
  NAME              = "AD_OFFSET"
  START_BYTE        = 53
  BYTES             = 1
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Controls an analog offset in CCMU"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 21
  NAME              = "AD_GAIN"
  START_BYTE        = 54
  BYTES             = 1
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Controls gain value in CCMU"
END_OBJECT          = COLUMN
```

```
OBJECT              = COLUMN
  COLUMN_NUMBER     = 22
  NAME              = "CCD_CLEAN_COUNT"
  START_BYTE        = 55
  BYTES             = 1
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "N images before transferring image to FPGA"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 23
  NAME              = "OBS_FROM_LIMIT_SWITCH"
  START_BYTE        = 56
  BYTES             = 1
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "0 = default
                       1 = brings focus stage to limit switch 1st to
                            initiliaze position"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 24
  NAME              = "THUMBNAIL_SIZE"
  START_BYTE        = 57
  BYTES             = 4
  DATA_TYPE         = MSB_INTEGER
  DESCRIPTION       = "0 = NO_THUMB turns off thumbnail creation
                       1 = THUMB_64 creates 64x64 down sample"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 25
  NAME              = "THUMBNAIL_COMPRESSION"
  START_BYTE        = 61
  BYTES             = 4
  DATA_TYPE         = MSB_INTEGER
  DESCRIPTION       = "0 = NO Compression
                       1 = LOCO
                       2 = ICER_1BPP"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 26
  NAME              = "THUMBNAIl_PRIORITY"
  START_BYTE        = 65
  BYTES             = 1
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Data priority of the thumbnail DP if any 0 = default"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 27
  NAME              = "RMI_REF_PIX_PRIORITY"
  START_BYTE        = 66
  BYTES             = 1
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Priority for the reference pixels data product"
```

```
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 28
  NAME              = "RMI_REF_PIX_DP"
  START_BYTE        = 67
  BYTES             = 4
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Indicates if reference pixels should be collected and
                       packaged into a Data Product:
                       0 = FALSE - (default)
                       1 = TRUE"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 29
  NAME              = "THUMB_ICER_SEGMENTS"
  START_BYTE        = 71
  BYTES             = 1
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Thumbnail icer segments"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 30
  NAME              = "THUMB_ICER_DECOMPS"
  START_BYTE        = 72
  BYTES             = 1
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Thumbnail icer decomps"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 31
  NAME              = "THUMB_ICER_MIN_LOSS"
  START_BYTE        = 73
  BYTES             = 1
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Thumbnail icer minimum loss"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 32
  NAME              = "THUMB_ICER_BPP"
  START_BYTE        = 74
  BYTES             = 4
  DATA_TYPE         = MSB_INTEGER
  DESCRIPTION       = "Thumbnail icer bits per pixel"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 33
  NAME              = "THUMB_LOCO_SEGMENTS"
  START_BYTE        = 78
  BYTES             = 1
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Thumbnail LOCO segments"
```

```
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 34
  NAME              = "THUMB_LOCO_PIXEL_SIZE"
  START_BYTE        = 79
  BYTES             = 4
  DATA_TYPE         = MSB_INTEGER
  DESCRIPTION       = "Thumbnail LOCO pixel size"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 35
  NAME              = "ICER_SEGMENTS"
  START_BYTE        = 83
  BYTES             = 1
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "ICER segments"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 36
  NAME              = "ICER_DECOMPS"
  START_BYTE        = 84
  BYTES             = 1
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "ICER decomps"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 37
  NAME              = "ICER_MIN_LOSS"
  START_BYTE        = 85
  BYTES             = 1
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "ICER Minimum Loss"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 38
  NAME              = "ICER_BPP"
  START_BYTE        = 86
  BYTES             = 4
  DATA_TYPE         = MSB_INTEGER
  DESCRIPTION       = "ICER Bits per pixel"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 39
  NAME              = "LOCO_SEGMENTS"
  START_BYTE        = 90
  BYTES             = 1
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "LOCO segments"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
```

```
    COLUMN_NUMBER    = 40
    NAME             = "LOCO_PIXEL_SIZE"
    START_BYTE       = 91
    BYTES            = 4
    DATA_TYPE        = MSB_INTEGER
    DESCRIPTION      = "LOCO Pixel size"
  END_OBJECT         = COLUMN

/* End of format file CCAM_RPIX_CMD_ARG_PARAMS_V4.FMT */
```

ii) "CCAM_TAKE_IMAGE_TIME"

```
/* CCAM_TAKE_IMAGE_TIME_V4.FMT.                                   */

  OBJECT             = COLUMN
    COLUMN_NUMBER    = 1
    NAME             = "TIME_BEFORE"
    START_BYTE       = 1
    BYTES            = 4
    DATA_TYPE        = IEEE_REAL
    DESCRIPTION      = "Time before image taken"
  END_OBJECT         = COLUMN

  OBJECT             = COLUMN
    COLUMN_NUMBER    = 2
    NAME             = "TIME_AFTER"
    START_BYTE       = 5
    BYTES            = 4
    DATA_TYPE        = IEEE_REAL
    DESCRIPTION      = "Time after image taken"
  END_OBJECT         = COLUMN

/* End of format file CCAM_TAKE_IMAGE_TIME_V4.FMT. */
```

jj) "CCAM_AF_SCIDATA_COLS"

```
/* CCAM_AF_SCIDATA_COLS_V4.FMT.                                   */

  OBJECT             = COLUMN
    COLUMN_NUMBER    = 1
    NAME             = "TOTAL_BYTECOUNT"
    START_BYTE       = 1
    BYTES            = 4
    DATA_TYPE        = MSB_UNSIGNED_INTEGER
    DESCRIPTION      = "Bytes to follow"
  END_OBJECT         = COLUMN

  OBJECT             = COLUMN
    COLUMN_NUMBER    = 2
    NAME             = "RCECONTROL"
    START_BYTE       = 5
    BYTES            = 4
    DATA_TYPE        = MSB_UNSIGNED_INTEGER
    DESCRIPTION      = "RCEControl:
                          contains: (8bits) Opcode 81 for autofocus data
```

```
           (2bits) Error Control Type Flag
  00 = reserved
  01 =  The CRC error control algorithm is applied
  10 = The Checksum error control Algorithm  applied
  11 = No error Control
           (1 bit) Data Present
  0 = nodata
  1 = data
           (21 bits) Status flags (as follows)
bit 1 - command Reply Flag
         (0- not cmd_reply, 1 - cmd_reply)
bit 2 - notSafe
         ( 0 - safe
           1 - Bu doesn't know that MU mortors
             are in sun-safe position)
bit 3 - BootSource
         (Which memory was used for boot
          0 - Init0 PROMO
          1 - Init1 PROM1)
bit 4 - CommSide
         (Talking with which RCE?
          0 - RCE A or none
          1 - RCE B)
bit 5 - CWL_heater_notOn
         (CWL heating loop algorithm off
          0 - on
          1 - off)
bit 6 - Amp_heater_notOn
         ( Amp heating loop algorithm off
          0 - on
          1 - off)
bit 7 - OSC_heater_notOn
         ( Osc heating loop algorithm off
          0 - on
          1 - off)
bit 8 - RMI_notOn
         ( RMI is not Powered
          0 - on
          1 - off)
bit 9 - RMIdata_notOK
         ( RMI off or comm with RMI not OK
          0 - OK
          1 - not OK, =1
              when RMI if off)
bit 10 - Spectrometer_notOK
         ( not used should be 0)
bit 11 - LVPS_notOn
         ( Low Voltage (spectrometer)power supply
           0 - on
           1 - off )
bit 12 - MAST_notOK
         ( Bad communication with the mast unit
           0 - OK
           1 - not OK)
bit 13 - LIBS_HV_notOn
         ( High voltage to LIBS not on
           0 - on
```

```
                                 1 - not on)
                      bit 14 - LIBS_notReady
                               (LIBS not in operating temperature range
                                 0 - ready
                                 1 - not ready)
                      bit 15 - CWL_notReady
                               (CWL not in operating temperature range
                                 0 - ready
                                 1 - not ready)
                      bit 16 - SelfTestFailed
                                (not used should be 0)
                      bit 17:21 -
                                0 0000 - reserved
                                0 0001 - First time command
                                0 0010 - Retry command (cmd reply frame)
                                1 0010 - Retry command (sci frame) "

END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 3
  NAME              = "DATA_BYTECOUNT"
  START_BYTE        = 9
  BYTES             = 4
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Number of bytes "
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 4
  NAME              = "DATAID"
  START_BYTE        = 13
  BYTES             = 1
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "id byte 0xc2 for autofocus"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 5
  NAME              = "AFGAIN"
  START_BYTE        = 14
  BYTES             = 1
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "gain setting for this data"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 6
  NAME              = "P1"
  START_BYTE        = 15
  BYTES             = 2
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "motor step position at start of scan"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 7
```

```
    NAME              = "DPBYTECOUNT"
    START_BYTE        = 17
    BYTES             = 4
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    DESCRIPTION       = "Number of bytes following in DataProduct"
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 8
    NAME              = "MILLITIME"
    START_BYTE        = 21
    BYTES             = 4
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    DESCRIPTION       = "msecs since last rce time set "
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 9
    NAME              = "ADDATA"
    START_BYTE        = 25
    ITEMS             = 639
    ITEM_BYTES        = 2
    BYTES             = 1278
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    DESCRIPTION       = "diode reading form the MU unsigned short [639]"
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 10
    NAME              = "MTAIL"
    START_BYTE        = 1303
    BYTES             = 2
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    DESCRIPTION       = "steps in last displacement"
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 11
    NAME              = "MTAIL2"
    START_BYTE        = 1305
    BYTES             = 2
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    DESCRIPTION       = "N displacements thermal and switch flags"
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 12
    NAME              = "RCECHECKSUM"
    START_BYTE        = 1307
    BYTES             = 4
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    DESCRIPTION       = "checksum"
  END_OBJECT          = COLUMN

/* End of format file CCAM_AF_SCIDATA_COLS_V4.FMT. */
```

kk) "CCAM_NAF_SCIDATA_COLS"

Note that the effective SCLK range for this FMT is SCLK > 484590825.

```
/* CCAM_NAF_SCIDATA_COLS_V4.FMT.                                         */

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 1
    NAME              = "TOTAL_BYTECOUNT"
    START_BYTE        = 1
    BYTES             = 4
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    DESCRIPTION       = "Bytes to follow"
  END_OBJECT          = COLUMN

  OBJECT              = COLUMN
    COLUMN_NUMBER     = 2
    NAME              = "RCECONTROL"
    START_BYTE        = 5
    BYTES             = 4
    DATA_TYPE         = MSB_UNSIGNED_INTEGER
    DESCRIPTION       = "RCEControl:
                           contains: (8bits) Opcode 81 for autofocus data
                                     (2bits) Error Control Type Flag
                          00 = reserved
                          01 =  The CRC error control algorithm is applied
                          10 = The Checksum error control Algorithm  applied
                          11 = No error Control
                                     (1 bit) Data Present
                          0 = nodata
                          1 = data
                                     (21 bits) Status flags (as follows)
                        bit 1 - command Reply Flag
                                 (0- not cmd_reply, 1 - cmd_reply)
                        bit 2 - notSafe
                                 ( 0 - safe
                                   1 - Bu doesn't know that MU mortors
                                     are in sun-safe position)
                        bit 3 - BootSource
                                 (Which memory was used for boot
                                  0 - Init0 PROMO
                                  1 - Init1 PROM1)
                        bit 4 - CommSide
                                 (Talking with which RCE?
                                  0 - RCE A or none
                                  1 - RCE B)
                        bit 5 - CWL_heater_notOn
                                 (CWL heating loop algorithm off
                                  0 - on
                                  1 - off)
                        bit 6 - Amp_heater_notOn
                                 ( Amp heating loop algorithm off
                                  0 - on
                                  1 - off)
                        bit 7 - OSC_heater_notOn
                                 ( Osc heating loop algorithm off
```

```
                                0 - on
                                1 - off)
                      bit 8 - RMI_notOn
                               ( RMI is not Powered
                                0 - on
                                1 - off)
                      bit 9 - RMIdata_notOK
                               ( RMI off or comm with RMI not OK
                                0 - OK
                                1 - not OK, =1
                                    when RMI if off)
                      bit 10 - Spectrometer_notOK
                               ( not used should be 0)
                      bit 11 - LVPS_notOn
                               ( Low Voltage (spectrometer)power supply
                                 0 - on
                                 1 - off )
                      bit 12 - MAST_notOK
                               ( Bad communication with the mast unit
                                 0 - OK
                                 1 - not OK)
                      bit 13 - LIBS_HV_notOn
                               ( High voltage to LIBS not on
                                 0 - on
                                 1 - not on)
                      bit 14 - LIBS_notReady
                               (LIBS not in operating temperature range
                                 0 - ready
                                 1 - not ready)
                      bit 15 - CWL_notReady
                               (CWL not in operating temperature range
                                 0 - ready
                                 1 - not ready)
                      bit 16 - SelfTestFailed
                                (not used should be 0)
                      bit 17:21 -
                                0 0000 - reserved
                                0 0001 - First time command
                                0 0010 - Retry command (cmd reply frame)
                                1 0010 - Retry command (sci frame) "

END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 3
  NAME              = "DATA_BYTECOUNT"
  START_BYTE        = 9
  BYTES             = 4
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Number of bytes "
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 4
  NAME              = "DATAID"
  START_BYTE        = 13
  BYTES             = 1
```

```
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "id byte 0xc2 for autofocus"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 5
  NAME             = "SEASON"
  START_BYTE       = 14
  BYTES            = 1
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Season 03"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 6
  NAME             = "UNUSED1"
  START_BYTE       = 15
  BYTES            = 2
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Unused - should be 0"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 7
  NAME             = "DPBYTESCOUNT"
  START_BYTE       = 17
  BYTES            = 4
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Number of bytes to follow - should be 1286"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 8
  NAME             = "ROISIZE"
  START_BYTE       = 21
  BYTES            = 1
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "CCS3 region of interest"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 9
  NAME             = "NUMFRAMES"
  START_BYTE       = 22
  BYTES            = 1
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "CCS3 number of frames"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 10
  NAME             = "NUMSTEPS"
  START_BYTE       = 23
  BYTES            = 1
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "CCS3 number of steps between frames"
END_OBJECT         = COLUMN
```

```
OBJECT              = COLUMN
  COLUMN_NUMBER     = 11
  NAME              = "FOCUSAEENABLE"
  START_BYTE        = 24
  BYTES             = 1
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "CCS3 autoexposure enable during autofocus"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 12
  NAME              = "UNUSED2"
  START_BYTE        = 25
  BYTES             = 1
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Unused should be 0"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 13
  NAME              = "RMIOFFSET"
  START_BYTE        = 26
  BYTES             = 1
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Offset applied to distance to target based on temp"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 14
  NAME              = "UNUSED3"
  START_BYTE        = 27
  BYTES             = 2
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Unused should be 0"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 15
  NAME              = "STARTSCAN"
  START_BYTE        = 29
  BYTES             = 2
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "AutoFocus - step position for 1st rmi image"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 16
  NAME              = "OPTOUTPUT"
  START_BYTE        = 31
  BYTES             = 2
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Results from the OPTIMUM routine"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 17
```

```
  NAME            = "CALCFOCUS"
  START_BYTE      = 33
  BYTES           = 2
  DATA_TYPE       = MSB_UNSIGNED_INTEGER
  DESCRIPTION     = "Calculated focus if cwl was used"
END_OBJECT        = COLUMN

OBJECT            = COLUMN
  COLUMN_NUMBER   = 18
  NAME            = "CALCLIBS"
  START_BYTE      = 35
  BYTES           = 2
  DATA_TYPE       = MSB_UNSIGNED_INTEGER
  DESCRIPTION     = "Calculated best libs focus"
END_OBJECT        = COLUMN

OBJECT            = COLUMN
  COLUMN_NUMBER   = 19
  NAME            = "CALCRMI"
  START_BYTE      = 37
  BYTES           = 2
  DATA_TYPE       = MSB_UNSIGNED_INTEGER
  DESCRIPTION     = "Calculated best rmi focus"
END_OBJECT        = COLUMN

OBJECT            = COLUMN
  COLUMN_NUMBER   = 20
  NAME            = "AUTOEXPSEED"
  START_BYTE      = 39
  BYTES           = 2
  DATA_TYPE       = MSB_UNSIGNED_INTEGER
  DESCRIPTION     = "CCS3 autoexposure seed value"
END_OBJECT        = COLUMN

OBJECT            = COLUMN
  COLUMN_NUMBER   = 21
  NAME            = "OFFSETTOMAX"
  START_BYTE      = 41
  BYTES           = 2
  DATA_TYPE       = MSB_UNSIGNED_INTEGER
  DESCRIPTION     = "CCS3 optional adjustment"
END_OBJECT        = COLUMN

OBJECT            = COLUMN
  COLUMN_NUMBER   = 22
  NAME            = "DTOT"
  START_BYTE      = 43
  BYTES           = 2
  DATA_TYPE       = MSB_UNSIGNED_INTEGER
  DESCRIPTION     = "Distance to target as supplied by RCE MM"
END_OBJECT        = COLUMN

OBJECT            = COLUMN
  COLUMN_NUMBER   = 23
  NAME            = "UNUSED4"
  START_BYTE      = 45
  BYTES           = 8
```

```
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Unused should be 0"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 24
  NAME              = "LAPLACIANF1"
  START_BYTE        = 53
  BYTES             = 4
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Laplacian Output Frame 1"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 25
  NAME              = "LAPLACIANF2"
  START_BYTE        = 57
  BYTES             = 4
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Laplacian Output Frame 2"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 26
  NAME              = "LAPLACIANF3"
  START_BYTE        = 61
  BYTES             = 4
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Laplacian Output Frame 3"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 27
  NAME              = "LAPLACIANF4"
  START_BYTE        = 65
  BYTES             = 4
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Laplacian Output Frame 4"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 28
  NAME              = "LAPLACIANF5"
  START_BYTE        = 69
  BYTES             = 4
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Laplacian Output Frame 5"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 29
  NAME              = "LAPLACIANF6"
  START_BYTE        = 73
  BYTES             = 4
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Laplacian Output Frame 6"
END_OBJECT          = COLUMN
```

```
OBJECT              = COLUMN
  COLUMN_NUMBER     = 30
  NAME              = "LAPLACIANF7"
  START_BYTE        = 77
  BYTES             = 4
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Laplacian Output Frame 7"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 31
  NAME              = "LAPLACIANF8"
  START_BYTE        = 81
  BYTES             = 4
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Laplacian Output Frame 8"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 32
  NAME              = "LAPLACIANF9"
  START_BYTE        = 85
  BYTES             = 4
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Laplacian Output Frame 9"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 33
  NAME              = "LAPLACIANF10"
  START_BYTE        = 89
  BYTES             = 4
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Laplacian Output Frame 10"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 34
  NAME              = "LAPLACIANF11"
  START_BYTE        = 93
  BYTES             = 4
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Laplacian Output Frame 11"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 35
  NAME              = "LAPLACIANF12"
  START_BYTE        = 97
  BYTES             = 4
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Laplacian Output Frame 12"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 36
```

```
  NAME             = "LAPLACIANF13"
  START_BYTE       = 101
  BYTES            = 4
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Laplacian Output Frame 13"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 37
  NAME             = "LAPLACIANF14"
  START_BYTE       = 105
  BYTES            = 4
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Laplacian Output Frame 14"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 38
  NAME             = "LAPLACIANF15"
  START_BYTE       = 109
  BYTES            = 4
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Laplacian Output Frame 15"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 39
  NAME             = "LAPLACIANF16"
  START_BYTE       = 113
  BYTES            = 4
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Laplacian Output Frame 16"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 40
  NAME             = "LAPLACIANF17"
  START_BYTE       = 117
  BYTES            = 4
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Laplacian Output Frame 17"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 41
  NAME             = "TIMEF1"
  START_BYTE       = 121
  BYTES            = 4
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Collect time frame 1"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 42
  NAME             = "TIMEF2"
  START_BYTE       = 125
  BYTES            = 4
```

```
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Collect time frame 2"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 43
  NAME             = "TIMEF3"
  START_BYTE       = 129
  BYTES            = 4
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Collect time frame 3"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 44
  NAME             = "TIMEF4"
  START_BYTE       = 133
  BYTES            = 4
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Collect time frame 4"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 45
  NAME             = "TIMEF5"
  START_BYTE       = 137
  BYTES            = 4
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Collect time frame 5"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 46
  NAME             = "TIMEF6"
  START_BYTE       = 141
  BYTES            = 4
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Collect time frame 6"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 47
  NAME             = "TIMEF7"
  START_BYTE       = 145
  BYTES            = 4
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Collect time frame 7"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 48
  NAME             = "TIMEF8"
  START_BYTE       = 149
  BYTES            = 4
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Collect time frame 8"
END_OBJECT         = COLUMN
```

```
OBJECT              = COLUMN
  COLUMN_NUMBER     = 49
  NAME              = "TIMEF9"
  START_BYTE        = 153
  BYTES             = 4
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Collect time frame 9"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 50
  NAME              = "TIMEF10"
  START_BYTE        = 157
  BYTES             = 4
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Collect time frame 10"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 51
  NAME              = "TIMEF11"
  START_BYTE        = 161
  BYTES             = 4
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Collect time frame 11"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 52
  NAME              = "TIMEF12"
  START_BYTE        = 165
  BYTES             = 4
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Collect time frame 12"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 53
  NAME              = "TIMEF13"
  START_BYTE        = 169
  BYTES             = 4
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Collect time frame 13"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 54
  NAME              = "TIMEF14"
  START_BYTE        = 173
  BYTES             = 4
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Collect time frame 14"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 55
```

```
  NAME             = "TIMEF15"
  START_BYTE       = 177
  BYTES            = 4
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Collect time frame 15"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 56
  NAME             = "TIMEF16"
  START_BYTE       = 181
  BYTES            = 4
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Collect time frame 16"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 57
  NAME             = "TIMEF17"
  START_BYTE       = 185
  BYTES            = 4
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Collect time frame 17"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 58
  NAME             = "UNUSED5"
  START_BYTE       = 189
  BYTES            = 26
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Unused should be 0"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 59
  NAME             = "MPOSF1"
  START_BYTE       = 215
  BYTES            = 2
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Motor Position Frame 1"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 60
  NAME             = "MPOSF2"
  START_BYTE       = 217
  BYTES            = 2
  DATA_TYPE        = MSB_UNSIGNED_INTEGER
  DESCRIPTION      = "Motor Position Frame 2"
END_OBJECT         = COLUMN

OBJECT             = COLUMN
  COLUMN_NUMBER    = 61
  NAME             = "MPOSF3"
  START_BYTE       = 219
  BYTES            = 2
```

```
    DATA_TYPE          = MSB_UNSIGNED_INTEGER
    DESCRIPTION        = "Motor Position Frame 3"
  END_OBJECT           = COLUMN

  OBJECT               = COLUMN
    COLUMN_NUMBER      = 62
    NAME               = "MPOSF4"
    START_BYTE         = 221
    BYTES              = 2
    DATA_TYPE          = MSB_UNSIGNED_INTEGER
    DESCRIPTION        = "Motor Position Frame 4"
  END_OBJECT           = COLUMN

  OBJECT               = COLUMN
    COLUMN_NUMBER      = 63
    NAME               = "MPOSF5"
    START_BYTE         = 223
    BYTES              = 2
    DATA_TYPE          = MSB_UNSIGNED_INTEGER
    DESCRIPTION        = "Motor Position Frame 5"
  END_OBJECT           = COLUMN

  OBJECT               = COLUMN
    COLUMN_NUMBER      = 64
    NAME               = "MPOSF6"
    START_BYTE         = 225
    BYTES              = 2
    DATA_TYPE          = MSB_UNSIGNED_INTEGER
    DESCRIPTION        = "Motor Position Frame 6"
  END_OBJECT           = COLUMN

  OBJECT               = COLUMN
    COLUMN_NUMBER      = 65
    NAME               = "MPOSF7"
    START_BYTE         = 227
    BYTES              = 2
    DATA_TYPE          = MSB_UNSIGNED_INTEGER
    DESCRIPTION        = "Motor Position Frame 7"
  END_OBJECT           = COLUMN

  OBJECT               = COLUMN
    COLUMN_NUMBER      = 66
    NAME               = "MPOSF8"
    START_BYTE         = 229
    BYTES              = 2
    DATA_TYPE          = MSB_UNSIGNED_INTEGER
    DESCRIPTION        = "Motor Position Frame 8"
  END_OBJECT           = COLUMN

  OBJECT               = COLUMN
    COLUMN_NUMBER      = 67
    NAME               = "MPOSF9"
    START_BYTE         = 231
    BYTES              = 2
    DATA_TYPE          = MSB_UNSIGNED_INTEGER
    DESCRIPTION        = "Motor Position Frame 9"
  END_OBJECT           = COLUMN
```

```
OBJECT              = COLUMN
  COLUMN_NUMBER     = 68
  NAME              = "MPOSF10"
  START_BYTE        = 233
  BYTES             = 2
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Motor Position Frame 10"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 69
  NAME              = "MPOSF11"
  START_BYTE        = 235
  BYTES             = 2
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Motor Position Frame 11"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 70
  NAME              = "MPOSF12"
  START_BYTE        = 237
  BYTES             = 2
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Motor Position Frame 12"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 71
  NAME              = "MPOSF13"
  START_BYTE        = 239
  BYTES             = 2
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Motor Position Frame 13"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 72
  NAME              = "MPOSF14"
  START_BYTE        = 241
  BYTES             = 2
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Motor Position Frame 14"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 73
  NAME              = "MPOSF15"
  START_BYTE        = 243
  BYTES             = 2
  DATA_TYPE         = MSB_UNSIGNED_INTEGER
  DESCRIPTION       = "Motor Position Frame 15"
END_OBJECT          = COLUMN

OBJECT              = COLUMN
  COLUMN_NUMBER     = 74
```

```
    NAME            = "MPOSF16"
    START_BYTE      = 245
    BYTES           = 2
    DATA_TYPE       = MSB_UNSIGNED_INTEGER
    DESCRIPTION     = "Motor Position Frame 16"
  END_OBJECT        = COLUMN

  OBJECT            = COLUMN
    COLUMN_NUMBER   = 75
    NAME            = "MPOSF17"
    START_BYTE      = 247
    BYTES           = 2
    DATA_TYPE       = MSB_UNSIGNED_INTEGER
    DESCRIPTION     = "Motor Position Frame 17"
  END_OBJECT        = COLUMN

  OBJECT            = COLUMN
    COLUMN_NUMBER   = 76
    NAME            = "UNUSED6"
    START_BYTE      = 249
    BYTES           = 1058
    DATA_TYPE       = MSB_UNSIGNED_INTEGER
    DESCRIPTION     = "Unused should be 0"
  END_OBJECT        = COLUMN

  OBJECT            = COLUMN
    COLUMN_NUMBER   = 77
    NAME            = "RCECHECKSUM"
    START_BYTE      = 1307
    BYTES           = 4
    DATA_TYPE       = MSB_UNSIGNED_INTEGER
    DESCRIPTION     = "checksum"
  END_OBJECT        = COLUMN

/* End of format file CCAM_NAF_SCIDATA_COLS_V4.FMT. */
```

APPENDIX E – Data Product Names (same as APID Names)

Instrument	APID Names	Names of DPOs Used as Sources for Metadata		
		IDPH	Ancillary	Supplementary
Hazcams	**Fullframe, Subframe, Downsampled Images:** ImgImageFhl, ImgImageFhr, ImgImageRhl, ImgImageRhr **Fullframe, Subframe, Downsampled Images (ICER compressed):** ImgImageIcerFhl, ImgImageIcerFhr, ImgImageIcerRhl, ImgImageIcerRhr **Fullframe, Subframe, Downsampled Images (LOCO compressed):** ImgImageLocoFhl, ImgImageLocoFhr, ImgImageLocoRhl, ImgImageLocoRhr	ImgIdph	n/a	n/a
	Thumbnail Images: ImgThumbFhl, ImgThumbFhr, ImgThumbRhl, ImgThumbRhr **Thumbnail Images (ICER compressed):** ImgThumbIcerFhl, ImgThumbIcerFhr, ImgThumbIcerRhl, ImgThumbIcerRhr **Thumbnail Images (LOCO compressed):** ImgThumbLocoFhl, ImgThumbLocoFhr, ImgThumbLocoRhl, ImgThumbLocoRhr	ImgIdph	n/a	n/a
	Reference Pixel Products: ImgRefFhl, ImgRefFhr, ImgRefRhl, ImgRefRhr **Reference Pixel Products (ICER compressed):** ImgRefIcerFhl, ImgRefIcerFhr, ImgRefIcerRhl, ImgRefIcerRhr **Reference Pixel Products (LOCO compressed):** ImgRefLocoFhl, ImgRefLocoFhr, ImgRefLocoRhl, ImgRefLocoRhr	ImgIdph	n/a	n/a
	Histogram Products: ImgHistogramFhl, ImgHistogramFhr, ImgHistogramRhl, ImgHistogramRhr	ImgIdph	n/a	n/a
	Row-summed Products: ImgRowsumsFhl, ImgRowsumsFhr, ImgRowsumsRhl, ImgRowsumsRhr	ImgIdph	n/a	n/a
	Column-summed Products: ImgColsumsFhl, ImgColsumsFhr, ImgColsumsRhl, ImgColsumsRhr	ImgIdph	n/a	n/a
Navcam	**Fullframe, Subframe, Downsampled Images:** ImgImageNl, ImgImageNr **Fullframe, Subframe, Downsampled Images (ICER compressed):**	ImgIdph	n/a	n/a

Instrument	APID Names	Names of DPOs Used as Sources for Metadata		
		IDPH	Ancillary	Supplementary
	ImgImageIcerNl, ImgImageIcerNr **Fullframe, Subframe, Downsampled Images (LOCO compressed):** ImgImageLocoRhr, ImgImageLocoNl, ImgImageLocoNr			
	Thumbnail Images: ImgThumbNl, ImgThumbNr **Thumbnail Images (ICER compressed):** ImgThumbIcerNl, ImgThumbIcerNr **Thumbnail Images (LOCO compressed):** ImgThumbLocoNl, ImgThumbLocoNr	ImgIdph	n/a	n/a
	Reference Pixel Products: ImgRefNl, ImgRefNr **Reference Pixel Products (ICER compressed):** ImgRefIcerNl, ImgRefIcerNr **Reference Pixel Products (LOCO compressed):** ImgRefLocoNl, ImgRefLocoNr	ImgIdph	n/a	n/a
	Histogram Products: ImgHistogramNl, ImgHistogramNr	ImgIdph	n/a	n/a
	Row-summed Products: ImgRowsumsNl, ImgRowsumsNr	ImgIdph	n/a	n/a
	Column-summed Products: ImgColsumsNl, ImgColsumsNr	ImgIdph	n/a	n/a
ChemCam RMI	**Images:** CcamRmiImage	Idph	CcamRmiImageAncillaryData	n/a
	Images (ICER compressed): CcamRmiImageIcer	Idph	CcamRmiImageIcerAncillaryData	n/a
	Images (LOCO compressed): CcamRmiImageLoco	Idph	CcamRmiImageLocoAncillaryData	n/a
	Thumbnail Images: CcamRmiThumb	Idph	CcamRmiThmbAncillaryData	n/a
	Thumbnail Images (ICER compressed): CcamRmiThumbIcer	Idph	CcamRmiThmbIcerAncillaryData	n/a
	Thumbnail Images (LOCO compressed): CcamRmiThumbLoco	Idph	CcamRmiThmbLocoAncillaryData	n/a
	Reference Pixel Products: CcamRmiRefPix	Idph	CcamRmiRefPixAncillaryData	n/a
ChemCam LIBS	**Spectra:** CcamSpectra	Idph	CcamSpectraAncillaryData	n/a
ChemCam SOH	**State of Health:** CcamSoh	n/a	CcamSohAncillaryData	n/a
	State of Health Initialize:	n/a	CcamSohInitAncillaryData	n/a

Instrument	APID Names	Names of DPOs Used as Sources for Metadata		
		IDPH	Ancillary	Supplementary
	CcamSohInit			
	State of Health Power On: CcamSohPowerOn	n/a	CcamSohPowerOnAncillaryData	n/a
	State of Health Power Off: CcamSohPowerOff	n/a	CcamSohPowerOffAncillaryData	n/a
	State of Health Warmup: CcamSohWarmUp	n/a	CcamSohWarmUpAncillaryData	n/a
	State of Health Sun Protection: CcamSohSunSafe	n/a	CcamSohSunSafeAncillaryData	n/a
ChemCam Generic (Cruise only)	**Command Parameters:** CcamParms	n/a	CcamSohAncillaryData	n/a
	Memory Dump: CcamMemoryDump	n/a	CcamSohInitAncillaryData	n/a
	Debug Dump: CcamDebugDump	n/a	CcamSohPowerOnAncillaryData	n/a
	Move Focus: CcamMoveFocus	n/a	CcamSohPowerOffAncillaryData	n/a
	Util Test: CcamUtilTest	n/a	CcamSohWarmUpAncillaryData	n/a
Mastcam	**Images:** McamLImage, McamRImage	DpoCidph	MmmImageAncillaryData	McamSupplementaryCommandArguments
	Thumbnail Images: McamLThumbnail, McamRThumbnail	DpoCidph	MmmImageAncillaryData	McamSupplementaryCommandArguments
	Video: McamLVideo, McamRVideo	DpoCidph	MmmVideoAncillaryData	McamSupplementaryCommandArguments
	Recovered Product: McamLRecoveredProduct, McamRRecoveredProduct	DpoCidph	MmmZstackAncillaryData	McamSupplementaryCommandArguments
	Recovered Thumbnail: McamLRecoveredThumbnail, McamRRecoveredThumbnail	DpoCidph	MmmZstackAncillaryData	McamSupplementaryCommandArguments
	Zstack: McamLZstack, McamRZstack, McamLZstackList, McamRZstackList	DpoCidph	MmmZstackAncillaryData	McamSupplementaryCommandArguments
	Range Map: McamLRangemap, McamRRangemap	DpoCidph	MmmZstackAncillaryData	McamSupplementaryCommandArguments
MAHLI	**Images:** MhliImage	DpoCidph	MmmImageAncillaryData	MhliSupplementaryCommandArguments
	Thumbnail Images: MhliThumbnail	DpoCidph	MmmImageAncillaryData	MhliSupplementaryCommandArguments
	Video: MhliVideo	DpoCidph	MmmVideoAncillaryData	MhliSupplementaryCommandArguments

Instrument	APID Names	Names of DPOs Used as Sources for Metadata		
		IDPH	Ancillary	Supplementary
	Recovered Product: MhliRecoveredProduct	DpoCidph	MmmVideoAncillaryData	MhliSupplementaryCommandArguments
	Recovered Thumbnail: MhliRecoveredThumbnail	DpoCidph	MmmZstackAncillaryData	MhliSupplementaryCommandArguments
	Zstack: MhliZstack, MhliZstackList	DpoCidph	MmmZstackAncillaryData	MhliSupplementaryCommandArguments
	Range Map: MhliRangemap	DpoCidph	MmmZstackAncillaryData	MhliSupplementaryCommandArguments
MARDI	**Images:** MrdiImage	DpoCidph	MmmImageAncillaryData	MrdiSupplementaryCommandArguments
	Thumbnail Images: MrdiThumbnail	DpoCidph	MmmImageAncillaryData	MrdiSupplementaryCommandArguments
	Video: MrdiVideo	DpoCidph	MmmVideoAncillaryData	MrdiSupplementaryCommandArguments
	Recovered Product: MrdiRecoveredProduct	DpoCidph	MmmVideoAncillaryData	MrdiSupplementaryCommandArguments
	Recovered Thumbnail: MrdiRecoveredThumbnail	DpoCidph	MmmZstackAncillaryData	MrdiSupplementaryCommandArguments
	Zstack: MrdiZstack, MrdiZstackList	DpoCidph	MmmZstackAncillaryData	MrdiSupplementaryCommandArguments
	Range Map: MrdiRangemap	DpoCidph	MmmZstackAncillaryData	MrdiSupplementaryCommandArguments

APPENDIX F – Product Label Keyword Definitions, Values, Sources

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
Ops Keyword ACTIVE_FLIGHT_STRING_ID **PDS Keyword** MSL:ACTIVE_FLIGHT_STRING_ID **Definition** Indicates which flight computer "string" (separate sets of electronics) was active when this product was acquired. For MSL there are two redundant flight computers (called "strings"), also known as Rover Compute Elements (RCE's). Either string (A or B) may be active at any given time.	**Valid Values** "A", "B" **Type** string **Units** n/a **Location** IDENTIFICATION (Class)	**Mode** EMD in XML format **Field as "<xml name>:[<element>]:[<element>]:<field>"** "MslEarthProductMetadata:MslProductMetadata:CreationStringId" **Type** U16
Ops Keyword APPLICATION_PROCESS_ID **PDS Keyword** same **Definition** Specifies the name associated with the source or process that created the data. For MSL, the Application Process Identifier (APID) identifies the data type encapsulated in the packet, including whether the packet is a data product packet or a non-data product packet. MSL shall implement the following APID assignments (all numbers below are in decimal, all ranges are inclusive): a) APID 2047 is used for Idle Packets b) APID 2040-2046 will not be produced by MSL. (Rationale: these APIDs are reserved by the CCSDS standard) c) APID 0 will not be produced by MSL d) APID 1 is reserved for X-band time correlation packets e) APIDs 2-99 are assigned to non-product packets. Individual values will be assigned to particular packet types as they are identified. Definitions of these packet formats are defined in this document. Definition of APID assignments is included in the Rover Flight Software APID XML. f) APIDs 100-2039 are assigned to RCE Flight Software data product packets. g) APIDs 1500-2039 are reserved for SSE (Simulation and Support Equipment) product types. (Above text from MSL FGICD v2.2.1) For MSL, only APID Names uniquely identify Data Product	**Valid Values** n/a **Type** integer **Units** n/a **Location** TELEMETRY (Class)	**Mode** EMD in XML format **Field as "<xml name>:[<element>]:[<element>]:<field>"** "MslEarthProductMetadata:MslProductMetadata:Apid" **Type** n/a

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
types across all FSW versions. For this reason, the integer APIDs are not documented here. See also APPLICATION_PROCESS_NAME and Appendix E.		
Ops Keyword APPLICATION_PROCESS_NAME **PDS Keyword** same **Definition** Specifies the name associated with the source or process that created the data. For MSL, only APID Names uniquely identify Data Product types across all FSW versions. See also APPLICATION_PROCESS_ID and Appendix E.	**Valid Values** • Hazcams Fullframe, Subframe, Downsampled: "ImgImageFhl", "ImgImageFhr", "ImgImageRhl", "ImgImageRhr" Fullframe, Subframe, Downsampled (ICER comp): "ImgImageIcerFhl", "ImgImageIcerFhr", "ImgImageIcerRhl", "ImgImageIcerRhr" Fullframe, Subframe, Downsampled (LOCO comp): "ImgImageLocoFhl", "ImgImageLocoFhr", "ImgImageLocoRhl", "ImgImageLocoRhr" Thumbnail: "ImgThumbFhl", "ImgThumbFhr", "ImgThumbRhl", "ImgThumbRhr" Thumbnail (ICER comp): "ImgThumbIcerFhl", "ImgThumbIcerFhr", "ImgThumbIcerRhl", "ImgThumbIcerRhr" Thumbnail (LOCO comp): "ImgThumbLocoFhl", "ImgThumbLocoFhr", "ImgThumbLocoRhl", "ImgThumbLocoRhr" Reference Pixel: "ImgRefFhl", "ImgRefFhr", "ImgRefRhl", "ImgRefRhr" Reference Pixel (ICER comp): "ImgRefIcerFhl", "ImgRefIcerFhr", "ImgRefIcerRhl", "ImgRefIcerRhr" Reference Pixel (LOCO comp): "ImgRefLocoFhl", "ImgRefLocoFhr", "ImgRefLocoRhl", "ImgRefLocoRhr" Row-summed: "ImgRowsumsFhl", "ImgRowsumsFhr", "ImgRowsumsRhl", "ImgRowsumsRhr" Column-summed: "ImgColsumsFhl", "ImgColsumsFhr", "ImgColsumsRhl", "ImgColsumsRhr" Histogram: "ImgHistogramFhl", "ImgHistogramFhr", "ImgHistogramRhl", "ImgHistogramRhr" • Navcams Fullframe, Subframe, Downsampled: "ImgImageNl", "ImgImageNr"	**Mode** EMD in XML format **Field as "<xml name>:[<element>]:[<element>]:<field>"** "MslEarthProductMetadata:MslProductMetadata:ProductName" **Type** n/a

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
	Fullframe, Subframe, Downsampled (ICER comp): “ImgImageIcerNl”, “ImgImageIcerNr” Fullframe, Subframe, Downsampled (LOCO comp): “ImgImageLocoNl”, “ImgImageLocoNr” Thumbnail: “ImgThumbNl”, “ImgThumbNr” Thumbnail (ICER comp): “ImgThumbIcerNl”, “ImgThumbIcerNr” Thumbnail (LOCO comp): “ImgThumbLocoNl”, “ImgThumbLocoNr” Reference Pixel: “ImgRefNl”, “ImgRefNr” Reference Pixel (ICER comp) “ImgRefIcerNl”, “ImgRefIcerNr” Reference Pixel (LOCO comp): “ImgRefLocoNl”, “ImgRefLocoNr” Row-summed: “ImgRowsumsNl”, “ImgRowsumsNr” Column-summed: “ImgColsumsNl”, “ImgColsumsNr” Histogram: “ImgHistogramNl”, “ImgHistogramNr” • ChemCam RMI Images: “CcamRmiImage” Images (ICER comp): “CcamRmiImageIcer” Images (LOCO comp): “CcamRmiImageLoco” Thumbnail Images: “CcamRmiThumb” Thumbnail Images (ICER comp): “CcamRmiThumbIcer” Thumbnail Images (LOCO comp): “CcamRmiThumbLoco • ChemCam LIBS Spectra: “CcamSpectra” • ChemCam SOH State of Health: “CcamSoh”	

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
	State of Health Power: “CcamSohPowerOn” State of Health Warmup: “CcamSohPowerOff” State of Health Sun Protection: “CcamSohSunSafe” State of Health Initialize: “CcamSohInit” • Mastcam Images: “McamLImage”, “McamRImage” Thumbnail Images: “McamLThumbnail”, “McamRThumbnail” Video: “McamLVideo”, “McamRVideo” Recovered Product: “McamLRecoveredProduct”, “McamRRecoveredProduct” Recovered Thumbnail: “McamLRecoveredThumbnail”, “McamRRecoveredThumbnail” Zstack: “McamLZstack”, “McamRZstack”, “McamLZstackList”, “McamRZstackList” Range Map: “McamLRangemap”, “McamRRangemap” • MAHLI Images: “MhliImage” Thumbnail Images: “MhliThumbnail” Video: “MhliVideo” Recovered Product: “MhliRecoveredProduct” Recovered Thumbnail: “MhliRecoveredThumbnail” Zstack: “MhliZstack”, “MhliZstackList” Range Map: “MhliRangemap”	

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
	• MARDI Images: "MrdiImage" Thumbnail Images: "MrdiThumbnail" Video: "MrdiVideo" Recovered Product: "MrdiRecoveredProduct" Recovered Thumbnail: "MrdiRecoveredThumbnail" Zstack: "MrdiZstack", "MrdiZstackList" Range Map: "MrdiRangemap" **Type** string(256) **Units** n/a **Location** TELEMETRY (Class)	
Ops Keyword ARTICULATION_DEVICE_ANGLE **PDS Keyword** same **Definition** Specifies the value of an angle between two parts or segments of an articulated device. NOTE: MSL uses radians. The PDS default unit for this keyword is degrees, so the <rad> tag is required. For the RSM, the "MEASURED" values (see ARTICULATION_DEVICE_ANGLE_NAME) represent the value of the resolver (attached to the output side of the joint), while the "FINAL" values represent the encoder (attached to the motor). The resolver should be preferentially used if available, as it measures the angle after joint backlash. A value of 1e+30 indicates the angle is not available, in which case the encoder should be used instead. Note that the "INITIAL" and "REQUESTED" values are also encoder measurements, and could be used to determine the joint's direction of motion for backlash determination.	**Valid Values** n/a **Type** 1) float array[8] 2) float array[2] 3) float array[6] 4) float array[10] **Units** radians (<rad> unit tag required) **Location** 1) CHASSIS_ARTICULATION_STATE (Group) 2) HGA_ARTICULATION_STATE (Group) 3) RSM_ARTICULATION_STATE (Group) 4) ARM_ARTICULATION_STATE (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** • Eng. Cameras 1) "<IDPH DPO>:idph:steer_fl", "<IDPH DPO>:idph:steer_fr", "<IDPH DPO>:idph:steer_rl", "<IDPH DPO>:idph:steer_rr", "<IDPH DPO>:idph:resolver[0]", "<IDPH DPO>:idph:resolver[1]", "<IDPH DPO>:idph:resolver[2]", "<IDPH DPO>:idph:resolver[3]" 2) "<IDPH DPO>:idph:hga_azimuth", "<IDPH DPO>:idph:hga_elevation" 3) "<IDPH DPO>:idph:rsm_res_azimuth", "<IDPH DPO>:idph:rsm_res_elevation", "<IDPH DPO>:idph:rsm_target_azimuth", "<IDPH DPO>:idph:rsm_target_elevation", "<IDPH DPO>:idph:rsm_initial_azimuth", "<IDPH DPO>:idph:rsm_initial_elevation" "<IDPH DPO>:idph:rsm_final_azimuth", "<IDPH DPO>:idph:rsm_final_elevation" 4) a. "<IDPH DPO>:idph:arm_qenc[5]",

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
		"<IDPH DPO>:idph:arm_qres[5]" • MMM Cameras 1) "<IDPH DPO>:cidph:steering:front_left", "<IDPH DPO>:cidph:steering:front_right", "<IDPH DPO>:cidph:steering:rear_left", "<IDPH DPO>:cidph:steering:rear_right", "<IDPH DPO>:cidph:bogie_left", "<IDPH DPO>:cidph:bogie_right", "<IDPH DPO>:cidph:differential_left", "<IDPH DPO>:cidph:differential_right" 2) "<IDPH DPO>:cidph:hga:azimuth", "<IDPH DPO>:cidph:hga:elevation" 3) "<IDPH DPO>:cidph:rsm_state:res:azimuth", "<IDPH DPO>:cidph:rsm_state:res:elevation", "<IDPH DPO>:cidph:rsm_state:target:azimuth", "<IDPH DPO>:cidph:rsm_state:target:elevation", "<IDPH DPO>:cidph:rsm_state:initial:azimuth", "<IDPH DPO>:cidph:rsm_state:initial:elevation", "<IDPH DPO>:cidph:rsm_state:final:azimuth", "<IDPH DPO>:cidph:rsm_state:final:elevation" 4) b. "<IDPH DPO>:cidph:arm_state:qenc[0]", "<IDPH DPO>:cidph:arm_state:qenc[1]", "<IDPH DPO>:cidph:arm_state:qenc[2]", "<IDPH DPO>:cidph:arm_state:qenc[3]", "<IDPH DPO>:cidph:arm_state:qenc[4]", "<IDPH DPO>:cidph:arm_state:qres[0]", "<IDPH DPO>:cidph:arm_state:qres[1]", "<IDPH DPO>:cidph:arm_state:qres[2]", "<IDPH DPO>:cidph:arm_state:qres[3]", "<IDPH DPO>:cidph:arm_state:qres[4]" **Type** 1) F32 2) F32 3) F32 4) a. F32[5] b. F32
Ops Keyword ARTICULATION_DEVICE_ANGLE_NAME **PDS Keyword** same **Definition** Specifies the formal name that identifies each of the values used in ARTICULATION_DEVICE_ANGLE.	**Valid Values** 1) CHASSIS ("LEFT FRONT WHEEL", "RIGHT FRONT WHEEL", "LEFT REAR WHEEL", "RIGHT REAR WHEEL", "LEFT BOGIE", "RIGHT BOGIE", "LEFT DIFFERENTIAL", "RIGHT DIFFERENTIAL") 2) HGA ("AZIMUTH", "ELEVATION") 3) RSM ("AZIMUTH-MEASURED", "ELEVATION-MEASURED", "AZIMUTH-REQUESTED", "ELEVATION-REQUESTED", "AZIMUTH-INITIAL",	**Mode** Static values **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
	"ELEVATION-INITIAL", "AZIMUTH-FINAL", "ELEVATION-FINAL") 4) ARM ("JOINT 1 AZIMUTH-ENCODER", "JOINT 2 ELEVATION-ENCODER", "JOINT 3 ELBOW-ENCODER", "JOINT 4 WRIST-ENCODER", "JOINT 5 TURRET-ENCODER", "JOINT 1 AZIMUTH-RESOLVER", "JOINT 2 ELEVATION-RESOLVER ", "JOINT 3 ELBOW-RESOLVER ", "JOINT 4 WRIST-RESOLVER ", "JOINT 5 TURRET-RESOLVER ") **Type** 1) string array[8] 2) string array[2] 3) string array[8] 4) string array[10] **Units** n/a **Location** 1) CHASSIS_ARTICULATION_STATE (Group) 2) HGA_ARTICULATION_STATE (Group) 3) RSM_ARTICULATION_STATE (Group) 4) ARM_ARTICULATION_STATE (Group)	
Ops Keyword ARTICULATION_DEVICE_ID **PDS Keyword** same **Definition** Specifies the unique abbreviated identification of an articulation device. An articulation device is anything that can move independently of the spacecraft to which it is attached (e.g., mast heads, wheel bogies, arms, etc.). NOTE: The ARTICULATION_DEVICE_ID is not a unique identifier for a given articulated device. Note also that the associated ARTICULATION_DEVICE_NAME element provides the full name of the articulated device.	**Valid Values** 1) "CHASSIS" 2) "HGA" 3) "RSM" 4) "ARM" **Type** string **Units** n/a **Location** 1) CHASSIS_ARTICULATION_STATE (Group) 2) HGA_ARTICULATION_STATE (Group) 3) RSM_ARTICULATION_STATE (Group) 4) ARM_ARTICULATION_STATE (Group)	**Mode** Static values **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword ARTICULATION_DEVICE_MODE **PDS Keyword** same	**Valid Values** 1) CHASSIS "DEPLOYED" 2) HGA "DEPLOYED"	**Mode** • Static value: 1) for Chassis 2) for HGA • DPO in XML format (referenced to APID Name in Appendix E)

<table>
<tr><th colspan="2">OUTPUT METADATA (PRODUCT LABEL)</th><th>INPUT METADATA (SOURCE)</th></tr>
<tr><th>• Ops Keyword
• PDS-Compliant Keyword
• Definition</th><th>• Valid Values (quoted)
• Type
• Units
• Keyword Location in Label</th><th>• Mode
• Metadata Field
• Type</th></tr>
<tr><td><u>Definition</u>
Specifies the deployment state (i.e., physical configuration) of an articulation device at the time of data acquisition. This includes the mode of the last move of the Arm.

Valid values for the Arm are defined as:
a) “FREE SPACE” - Arm was moved where there was no contact with a target expected.
b) “GUARDED” - Arm was moved where contact with the target was expected.
c) “RETRACTING” - Arm was moved where an instrument is removed from a target.
d) “PRELOAD” - Arm stays in contact with the target and applies force or overtravel on an instrument.
e) “ARM VIBE” - Arm movement ignores all turret-mounted contact switches.</td><td>3) <u>RSM</u>
0 = "STOWED"
1 = "DEPLOYED"
4) <u>ARM</u>
0 = "FREE SPACE"
1 = "GUARDED"
2 = "RETRACTING"
3 = "PRELOAD"
4 = "ARM VIBE"

<u>Type</u>
string

<u>Units</u>
n/a

<u>Location</u>
1) CHASSIS_ARTICULATION_STATE (Group)
2) HGA_ARTICULATION_STATE (Group)
3) RSM_ARTICULATION_STATE (Group)
4) ARM_ARTICULATION_STATE (Group)</td><td><u>Field as “<xml name>:[<element>]:[<element>]:<field>”</u>
• <u>Eng. Cameras</u>
3) a. “<IDPH DPO>:idph:rsm_deployed”
4) “<IDPH DPO>:idph:arm_mode”
• <u>MMM Cameras</u>
3) b. “<IDPH DPO>:cidph:rsm_state:deployed”
4) “<IDPH DPO>:cidph:rsm_state:mode”

<u>Type</u>
3) a. boolean
b. U8
4) U8</td></tr>
<tr><td><u>Ops Keyword</u>
ARTICULATION_DEVICE_NAME

<u>PDS Keyword</u>
same

<u>Definition</u>
Specifies the common name of an articulation device. An articulation device is anything that can move independently of the spacecraft to which it is attached, (e.g. mast heads, wheel bogies, arms, etc.)

NOTE: The associated ARTICULATION_DEVICE_ID element provides an abbreviated name or acronym for the articulated device.</td><td><u>Valid Values</u>
1) “MOBILITY CHASSIS"
2) “HIGH GAIN ANTENNA”
3) “REMOTE SENSING MAST”
4) “SAMPLE ARM”

<u>Type</u>
string

<u>Units</u>
n/a

<u>Location</u>
1) CHASSIS_ARTICULATION_STATE (Group)
2) HGA_ARTICULATION_STATE (Group)
3) RSM_ARTICULATION_STATE (Group)
4) ARM_ARTICULATION_STATE (Group)</td><td><u>Mode</u>
Static values

<u>Field as “<xml name>:[<element>]:[<element>]:<field>”</u>
n/a

<u>Type</u>
n/a</td></tr>
<tr><td><u>Ops Keyword</u>
ARTICULATION_DEVICE_TEMP

<u>PDS Keyword</u>
same

<u>Definition</u>
Specifies the temperature, in degrees Celsius, of an articulated device or some part of an articulated device.</td><td><u>Valid Values</u>
“-3.4e38” to “3.4e38”

<u>Type</u>
float array[5]

<u>Units</u>
deg C (<degC> unit tag required)

<u>Location</u>
ARM_ARTICULATION_STATE (Group)</td><td><u>Mode</u>
DPO in XML format (referenced to APID Name in Appendix E)

<u>Field as “<xml name>:[<element>]:[<element>]:<field>”</u>
1) <u>Eng. Cameras</u>
“<IDPH DPO>:idph:arm_temp[5]”
2) <u>MMM Cameras</u>
“<IDPH DPO>:cidph:arm_state:temp[0]”,
“<IDPH DPO>:cidph:arm_state:temp[1]”,
“<IDPH DPO>:cidph:arm_state:temp[2]”,
“<IDPH DPO>:cidph:arm_state:temp[3]”,</td></tr>
</table>

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
		"<IDPH DPO>:cidph:arm_state:temp[4]" **Type** 1) F32[5] 2) F32
Ops Keyword ARTICULATION_DEVICE_TEMP_NAME **PDS Keyword** same **Definition** Specifies the array of formal names identifying each of the values used in ARTICULATION_DEVICE_TEMP.	**Valid Values** ("AZIMUTH JOINT", "ELEVATION JOINT", "ELBOW JOINT", "WRIST JOINT", "TURRET JOINT") **Type** string array[2] **Units** n/a **Location** ARM_ARTICULATION_STATE (Group)	**Mode** Static value **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword ARTICULATION_DEV_INSTRUMENT_ID **PDS Keyword** same **Definition** Specifies an abbreviated name or acronym which identifies the instrument mounted on the articulation device.	**Valid Values** 0 = "TURRET" 1 = "DRILL" 2 = "DRT" 3 = "MAHLI" 4 = "APXS" 5 = "PORTIONER TUBE" 6 = "SCOOP TIP" 7 = "SCOOP TCP" **Type** string **Units** n/a **Location** ARM_ARTICULATION_STATE (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** • Eng. Cameras "<IDPH DPO>:idph:arm_instrument" • MMM Cameras "<IDPH DPO>:cidph:arm_state:instrument" **Type** U8
Ops Keyword ARTICULATION_DEV_POSITION **PDS Keyword** none **Definition** Specifies the set of indices for articulation devices that contain moving parts with discrete positions. The associated ARCTICULATION_DEV_POSITION_NAME names each moving device. In order to get the actual filter used for this specific image, the FILTER_NAME/FILTER_NUMBER keywords in the INSTRUMENT_DATA group should be used.	**Valid Values** n/a **Type** int array[2] **Units** n/a **Location** MINI_HEADER (Group)	**Mode** TBD **Field as "<xml name>:[<element>]:[<element>]:<field>"** TBD MMM_Image_Mini_Header **Type** TBD

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
See also ARTICULATION_DEV_POSITION_NAME		
Ops Keyword ARTICULATION_DEV_POSITION_NAME **PDS Keyword** none **Definition** Specifies an array of values that provides the formal names for each entry in ARTICULATION_DEV_POSITION. This element names the actual device doing the moving, (e.g., a filter wheel), not the name of a position (e.g., the filter itself).	**Valid Values** n/a **Type** string array[2] **Units** n/a **Location** MINI_HEADER (Group)	**Mode** TBD **Field as "<xml name>:[<element>]:[<element>]:<field>"** TBD MMM_Image_Mini_Header **Type** TBD
Ops Keyword ARTICULATION_DEV_VECTOR **PDS Keyword** same **Definition** Specifies the direction and magnitude of an external force acting on the articulation device, in the rover's coordinate system, at the time the pose was computed.	**Valid Values** "-1.0" to "1.0" **Type** float array[3] **Units** n/a **Location** ARM_ARTICULATION_STATE (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** 1) Eng. Cameras "<IDPH DPO>:idph:arm_tilt[3]" 2) MMM Cameras "<IDPH DPO>:cidph:arm_state:tilt[0]", "<IDPH DPO>:cidph:arm_state:tilt[1]", "<IDPH DPO>:cidph:arm_state:tilt[2]" **Type** 1) F32[3] 2) F32
Ops Keyword ARTICULATION_DEV_VECTOR_NAME **PDS Keyword** same **Definition** Specifies the formal name of the vector type acting on the articulation device.	**Valid Values** "GRAVITY" **Type** string **Units** n/a **Location** ARM_ARTICULATION_STATE (Group)	**Mode** Static value **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword AUTO_DELETE_FLAG **PDS Keyword** MSL:AUTO_DELETE_FLAG **Definition** Indicates if the product will be automatically deleted onboard after it is transmitted.	**Valid Values** 0 = "FALSE" 1 = "TRUE" **Type** string **Units** n/a **Location** TELEMETRY (Class)	**Mode** EMD in XML format **Field as "<xml name>:[<element>]:[<element>]:<field>"** "MslEarthProductMetadata:MslProductMetadata:DeleteOnSend" **Type** n/a

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
Ops Keyword AUTO_EXPOSURE_DATA_CUT **PDS Keyword** same **Definition** Specifies the DN value at which a specified fraction of pixels is permitted to exceed. The fraction is specified using the keyword AUTO_EXPOSURE_DATA_FRACTION.	**Valid Values** "0" to n **Type** integer **Units** n/a **Location** 1) OBSERVATION_REQUEST_PARMS (Group) 2) a. OBSERVATION_REQUEST_PARMS (Group) b. MINI_HEADER (Group) ** ** Except MAHLI Z-Stack & Range Map	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** 1) Eng. Cameras "<IDPH DPO>:idph:params:exp_auto_dn" 2) MMM Cameras a. "<Ancillary DPO>:cmd_arguments_image:exp_target_DN" b. "MMM_Image_Mini_Header[**TBD**]" **Type** 1) U16 2) U8
Ops Keyword AUTO_EXPOSURE_PERCENT **PDS Keyword** same **Definition** Specifies the auto-exposure early-termination percent. If the calculated exposure time has written this value, then terminate auto exposure early.	**Valid Values** "0.0" to "100.0" **Type** float **Units** n/a **Location** 1) OBSERVATION_REQUEST_PARMS (Group) 2) a. OBSERVATION_REQUEST_PARMS (Group) b. MINI_HEADER (Group) ** ** Except MAHLI Z-Stack & Range Map	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** 1) Eng. Cameras "<IDPH DPO>:idph:params:exp_auto_percent" 2) MMM Cameras a. "<AncillaryDPO>:cmd_arguments_image:exp_early _termination" b. "MMM_Image_Mini_Header[**TBD**]" **Type** U8
Ops Keyword AUTO_EXPOSURE_PIXEL_FRACTION **PDS Keyword** same **Definition** Specifies the percentage of pixels whose targeted value is higher than the AUTO_EXPOSURE_DATA_CUT keyword.	**Valid Values** "0.0" to "100.0" **Type** float **Units** n/a **Location** 1) OBSERVATION_REQUEST_PARMS (Group) 2) a. OBSERVATION_REQUEST_PARMS (Group) b. MINI_HEADER (Group) ** ** Except MAHLI Z-Stack & Range Map	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** 1) Eng. Cameras "<IDPH DPO>:idph:params:exp_auto_frac" 2) MMM Cameras a. "<Ancillary DPO>:cmd_arguments_image:exp_pixel_fraction" b. "MMM_Image_Mini_Header[**TBD**]" **Type** 1) F32 2) U8
Ops Keyword AZIMUTH_FOV **PDS Keyword** same **Definition** Specifies the angular measure of the horizontal field of view of	**Valid Values** "0.0" to "360.0" **Type** float **Units** deg (<deg> unit tag required)	**Mode** • 1) DPO in XML format (referenced to APID Name in Appendix E) 2) DPO in XML format (referenced to APID Name in Appendix E) or Image DPO mini-header • Calculation: - Line/sample is translated into an origin and "look direction" vectors (left and right). The vectors are then used to

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
an imaged scene. NOTE: For MER, this was computed as IFOV times the number of horizontal pixels. For MSL, it is computed by projecting rays from the left and right edges of the image at the center through the camera model, and computing the angle subtended by those rays.	**Location** INSTRUMENT_STATE_PARMS (Group)	calculate Azimuth field of view. **Field as "<xml name>:[<element>]:[<element>]:<field>"** 1) Eng. Cameras a. "<IDPH DPO>:idph:cols" b. "<IDPH DPO>:idph:rows" c. "<IDPH DPO>:idph:res_cols" d. "<IDPH DPO>:idph:res_rows" 2) MMM Cameras a. "<Ancillary DPO>:cmd_arguments_image:window_width" b. "<Ancillary DPO>:cmd_arguments_image:window_height" c. "MMM_Image_Mini_Header[22]" d. "MMM_Image_Mini_Header[23]" NOTES: • For MMM non-recovered data products (cases "a" and "b"), float values come from the Ancillary DPO. • For MMM recovered data products (cases "c" and "d"), values analogous to cases "a" (window_width) and "b" (window_height) are each comprised of one byte coming from Image DPO mini-header at byte offsets 22 and 23, respectively. • For MMM, parm "window_width" is number of image samples. • For MMM, parm "window_height" is number of image lines. • For MMM non-Thumbnail data products, multiply by factor of 8 to convert to correct line/sample value. **Type** 1) U16 2) U8
Ops Keyword BAD_PIXEL_REPLACEMENT_FLAG **PDS Keyword** same **Definition** Specifies whether or not bad pixel replacement processing was requested or completed. If set to TRUE, certain pixels in the image were replaced based on a bad pixel table. See BAD_PIXEL_REPLACEMENT_ID.	**Valid Values** • Eng. Cameras 0 = "FALSE" 1 = "TRUE" • MMM Cameras "UNK" **Type** string(5) **Units** n/a **Location** OBSERVATION_REQUEST_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** Eng. Cameras "<IDPH DPO>:idph:params:bad" **Type** boolean
Ops Keyword BAD_PIXEL_REPLACEMENT_ID **PDS Keyword**	**Valid Values** • Eng. Cameras 0 = "N/A" "1" to "65535"	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"**

OUTPUT METADATA (PRODUCT LABEL)		**INPUT METADATA (SOURCE)**
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
same **Definition** Specifies the ID of the bad pixel table used in the bad pixel replacement process. The bad pixel table ID is incremented every time an update to the bad pixel table is made. See BAD_PIXEL_REPLACEMENT_FLAG.	• MMM Cameras "UNK" **Type** string **Units** n/a **Location** INSTRUMENT_STATE_PARMS (Group)	Eng. Cameras "<IDPH DPO>:idph:bad" **Type** U16
Ops Keyword BANDS **PDS Keyword** same **Definition** Specifies the number of spectral bands in image or other object.	**Valid Values** "1" to "16" **Type** integer **Units** n/a **Location** IMAGE (Object)	**Mode** Static Value **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword BAND_STORAGE_TYPE **PDS Keyword** same **Definition** Specifies the number of spectral bands in image or other object.	**Valid Values** "BAND_SEQUENTIAL", "SAMPLE_INTERLEAVED", "LINE_INTERLEAVED" **Type** string(20) **Units** n/a **Location** IMAGE (Object)	**Mode** VICAR label **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword BAYER_MODE **PDS Keyword** same **Definition** Specifies whether or not the data are encoded in a Bayer pattern, and if not, how the pattern was removed. For MSL, this applies only to the MMM cameras; others will not have this keyword. The list of valid values below starting with MIPL_ correspond to MIPL/OPGS algorithms; other teams may add their own	**Valid Values** • For EDRs … - 1 band: "RAW_BAYER" "ONBOARD_LUMINANCE" - 3 bands: "ONBOARD_COLOR" • For RDRs … "RAW_BAYER", "ONBOARD_COLOR", "MIPL_MALVAR_INTERPOLATED", "MIPL_INTERPOLATED", "MIPL_REPLICATED", "MIPL_AVERAGED_COLOR", "MIPL_UPPER", "MIPL_LOWER", "MIPL_TALL_INTERPOLATED", "MIPL_TALL_REPLICATED", "MIPL_AVERAGED_MONO", "MIPL_CORRECTED", "MSSS_<to-be-determined>" **Type**	**Mode** • For EDRs: Static values conditional on number of bands • For RDRs: RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a

OUTPUT METADATA (PRODUCT LABEL)		**INPUT METADATA (SOURCE)**
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
algorithm names as needed: a) “RAW_BAYER” - Raw Bayer-encoded data from the CCD. b) “ONBOARD_COLOR” - Bayer pattern removed onboard to produce full-resolution color. c) “MIPL_INTERPOLATED” - Interpolated, full-resolution color. d) “MIPL_REPLICATED” - Pixel-replicated, full-resolution color. e) “MIPL_AVERAGED_COLOR” - Averaged, half-resolution color. f) “MIPL_UPPER” - Half-resolution color using upper Green cells only. g) “MIPL_LOWER” - Half-resolution color using lower Green cells only. h) “MIPL_TALL_INTERPOLATED” - Full-resolution interpolated in line direction, half-resolution in sample direction. i) “MIPL_TALL_REPLICATED - Full-resolution replicated in line direction, half-resolution in sample direction. j) “MIPL_AVERAGED_MONO” - 2x2 averaging to remove Bayer pattern, single band (not color). k) “MIPL_CORRECTED” - Full resolution, single band (not color), corrected for filter responsivity. l) “MSSS_*” - Names reserved for MSSS algorithms. See Section 4.3.3.1 for further discussion.	string **Units** n/a **Location** INSTRUMENT_STATE_PARMS (Group)	
Ops Keyword BIAS_COEFFS_FILE_NAME **PDS Keyword** MSL:BIAS_COEFFS_FILE_NAME **Definition** Specifies the name of bias coefficients file used in generating the RDR.	**Valid Values** n/a **Type** string **Units** n/a **Location** DERIVED_IMAGE_PARMS (Group)	**Mode** RDR-generating software **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a
Ops Keyword BIAS_COEFFS_FILE_DESC **PDS Keyword** MSL:BIAS_COEFFS_FILE_DESC **Definition** Specifies a description of bias coefficients file named in BIAS_COEFFS_FILE_NAME.	**Valid Values** n/a **Type** string **Units** n/a **Location** DERIVED_IMAGE_PARMS (Group)	**Mode** RDR-generating software **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a
Ops Keyword	**Valid Values**	**Mode**

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
BRIGHTNESS_CORRECTION_FILE **PDS Keyword** MSL:BRIGHTNESS_CORRECTION_FILE **Definition** Specifies the name of the file containing brightness correction parameters for each input file of a mosaic. The file is in XML format.	n/a **Type** string **Units** n/a **Location** DERIVED_IMAGE_PARMS (Group)	RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword BRIGHTNESS_CORRECTION_TYPE **PDS Keyword** MSL:BRIGHTNESS_CORRECTION_TYPE **Definition** Identifies the type of brightness correction that was applied to a mosaic. Brightness correction is defined as a mosaic radiometric seam matching process that is done on top of ordinary radiometric correction. Its intent is to reduce visual seams at the expense of radiometric accuracy. Applies to RDRs only, not EDRs. Types may be added over time but currently they consist of: a) "NONE" - No correction made. b) "LINEAR" - Each input image gets a single additive and a single multiplicative factor, which are applied to every pixel in the image as follows: (new_image = image * mult + add). These factors are defined via the IMAGE_RADIANCE_FACTOR and IMAGE_RADIANCE_OFFSET keywords. c) "MIXED" - Multiple types of correction were applied to different input images. This should be avoided but is included as a placeholder.	**Valid Values** "NONE", "LINEAR", "MIXED" **Type** string **Units** n/a **Location** DERIVED_IMAGE_PARMS (Group)	**Mode** RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword BYTES **PDS Keyword** same **Definition** Specifies the number of bytes allocated for a particular data representation.	**Valid Values** "0" to n **Type** integer **Units** n/a **Location** IMAGE_HEADER (Object)	**Mode** Calculation based on size of VICAR label **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword CALIBRATION_SOURCE_ID	**Valid Values** n/a	**Mode** DPO in XML format (referenced to APID Name in Appendix E)

<table>
<tr><th colspan="2">OUTPUT METADATA (PRODUCT LABEL)</th><th>INPUT METADATA (SOURCE)</th></tr>
<tr><th>• Ops Keyword
• PDS-Compliant Keyword
• Definition</th><th>• Valid Values (quoted)
• Type
• Units
• Keyword Location in Label</th><th>• Mode
• Metadata Field
• Type</th></tr>
<tr><td>PDS Keyword
same

Definition
Specifies a unique identifier (within a data set) indicating the source of the calibration data used in generating the entity described by the enclosing group (often, a camera model).

The construction of this identifier is mission-specific, but should indicate which specific calibration data set was used (via date or other means) and may also indicate the calibration method.</td><td>Type
string(47)

Units
n/a

Location
GEOMETRIC_CAMERA_MODEL (Group)</td><td>Field as “<xml name>:[<element>]:[<element>]:<field>”
• Eng. Cameras
“<IDPH DPO>:idph:cmod_model_id”
• MMM Cameras
“<IDPH DPO>:idph:camera_model:mid”

Type
U16</td></tr>
<tr><td>Ops Keyword
CAMERA_ROTATION_AXIS_VECTOR

PDS Keyword
MSL:CAMERA_ROTATION_AXIS_VECTOR

Definition
Specifies the axis around which the camera rotates.

For the Cylindrical-Perspective projection, this is the axis around which one rotates the camera model in azimuth, before (optionally) correcting for rover tilt using PROJECTION_Z_AXIS_VECTOR. It corresponds to the physical azimuth rotation axis for mast-mounted cameras.</td><td>Valid Values
n/a

Type
float array[3]

Units
n/a

Location
SURFACE_PROJECTION_PARMS (Group)</td><td>Mode
RDR-generating software

Field as “<xml name>:[<element>]:[<element>]:<field>”
n/a

Type
n/a</td></tr>
<tr><td>Ops Keyword
CAMERA_SERIAL_NUMBER

PDS Keyword
same

Definition
Specifies the manufacturer's serial number assigned to a camera instrument. This number may be used to uniquely identify a particular camera instrument for tracing its components or determining its calibration history, for example.

For MSL, the value might not be identical to that of INSTRUMENT_SERIAL_NUMBER.</td><td>Valid Values
• Eng. Cameras …
EM Strings FM Strings
Instrument A B A B
Front Left Hazcam …… “057” “056” “205” “208”
Front Right Hazcam … “058” “055” “213” “209”
Rear Left Hazcam …… “204” “202” “211” “212”
Rear Right Hazcam …. “028” “027” “217” “207”
Left Navcam …….…… “053” “054” “216” “215”
Right Navcam ……..…. “051” “052” “206” “218”

• ChemCam …
Instrument FM
RMI ……….................. “0001”

• MMM …
Instrument EM FM
MastCam Left …… “1001” “3003”
MastCam Right … “1005” “3004”
MARDI …………… “1002” “3001”
MAHLI ……………. “1003” “3002”

Type
string</td><td>Mode
DPO in XML format (referenced to APID Name in Appendix E)

Field as “<xml name>:[<element>]:[<element>]:<field>”
“<IDPH DPO>:idph:cmod_serial_no”

Type
U8</td></tr>
</table>

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
	Units n/a **Location** GEOMETRIC_CAMERA_MODEL (Group)	
Ops Keyword CLASSIFIER_BAND_HUE **PDS Keyword** None **Definition** Applies to TEN/TENL products only. Hexadecimal color value corresponding to each band listed in CLASSIFIER_BAND_INDEX. A hexadecimal color is specified with: #RRGGBB. RR (red), GG (green) and BB (blue) are hexadecimal integers between 00 and FF specifying the intensity of the color. For example, #0000FF is displayed as blue, because the blue component is set to its highest value (FF) and the others are set to 00. See also CLASSIFIER_LABEL_HUE.	**Valid Values** #000000 to #FFFFFF **Type** String array **Units** n/a **Location** DERIVED_IMAGE_PARMS (Group)	**Mode** RDR-generating software **Field as “<xml name>:[<element>]:[<element>]:<field>”** N/A **Type** N/A
Ops Keyword CLASSIFIER_BAND_INDEX **PDS Keyword** None **Definition** Applies to TEN/TENL and TER/TERL products. For TENL products, this keyword contains numbered list of bands, e.g. (0, 1, 2, 3, 4, 5, 6), where each band number maps to a terrain type (CLASSIFIER_BAND_INDEX_NAME) and HEX color code (CLASSIFIER_BAND_HUE). New bands may be added periodically, corresponding to newly defined terrain classes. Updates or additions to the list of terrain classes will always be appended to the existing list, and result in an increment of the decimal portion of CLASSIFIER_VERSION. For TERL products, this keyword is always a 2-element string array describing the contents of TERL band 1 and band 2, which are fixed to these values: (“CLASSIFIER_LABEL_INDEX”, “CONFIDENCE”) See also CLASSIFIER_LABEL_INDEX.	**Valid Values** TENL: Unsigned integer starting from 0 TERL: (“CLASSIFIER_LABEL_INDEX”, “CONFIDENCE”) **Type** TENL: int array TERL: string array **Units** n/a **Location** DERIVED_IMAGE_PARMS (Group)	**Mode** RDR-generating software **Field as “<xml name>:[<element>]:[<element>]:<field>”** N/A **Type** N/A

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
Ops Keyword CLASSIFIER_BAND_INDEX_NAME **PDS Keyword** None **Definition** Applies to TEN/TENL products only. A list of names, e.g. (“Ripples”, “Smooth”, “Rover Tracks”, ...and so on), describing the terrain class. The position of a string in this array corresponds to the band number listed in CLASSIFIER_BAND_INDEX. New classes may be added periodically. Updates or additions to the list of terrain classes will always be appended to the existing list, and result in an increment of the decimal portion of CLASSIFIER_VERSION. See also CLASSIFIER_LABEL_INDEX_NAME.	**Valid Values** n/a **Type** string array **Units** n/a **Location** DERIVED_IMAGE_PARMS (Group)	**Mode** RDR-generating software **Field as “<xml name>:[<element>]:[<element>]:<field>”** N/A **Type** N/A
Ops Keyword CLASSIFIER_LABEL_HUE **PDS Keyword** None **Definition** Applies to TER/TERL products only. Hexadecimal color value corresponding to CLASSIFIER_LABEL_INDEX. A hexadecimal color is specified with: #RRGGBB. RR (red), GG (green) and BB (blue) are hexadecimal integers between 00 and FF specifying the intensity of the color. For example, #0000FF is displayed as blue, because the blue component is set to its highest value (FF) and the others are set to 00. See also CLASSIFIER_BAND_HUE.	**Valid Values** #000000 to #FFFFFF **Type** String array **Units** n/a **Location** DERIVED_IMAGE_PARMS (Group)	**Mode** RDR-generating software **Field as “<xml name>:[<element>]:[<element>]:<field>”** N/A **Type** N/A
Ops Keyword CLASSIFIER_LABEL_INDEX **PDS Keyword** None **Definition** Applies to TER/TERL products only. A numbered list of terrain class indices, e.g. (0, 1, 2, 3, 4, 5, 6, 255), where each class index maps to a class name (CLASSIFIER_LABEL_INDEX_NAME) and HEX color code (CLASSIFIER_LABEL_HUE).	**Valid Values** Unsigned integer from 0-255 **Type** int array **Units** n/a **Location** DERIVED_IMAGE_PARMS (Group)	**Mode** RDR-generating software **Field as “<xml name>:[<element>]:[<element>]:<field>”** N/A **Type** N/A

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
The class index "255" is reserved for pixels with no assigned terrain class ("None"), and is always the last index on the list. New class indices may be added periodically, corresponding to newly defined terrain classes. Updates or additions to the list of terrain classes will always be appended to the existing list, and result in an increment of the decimal portion of CLASSIFIER_VERSION. See also CLASSIFIER_BAND_INDEX.		
Ops Keyword CLASSIFIER_LABEL_INDEX_NAME **PDS Keyword** None **Definition** Applies to TER/TERL products only. A list of names, e.g. ("Ripples", "Smooth", "Rover Tracks", "None"), describing the terrain class. The position of a string in this array corresponds to the 0-based DN value in CLASSIFIER_LABEL_INDEX. New class indices may be added periodically. Updates or additions to the list of terrain classes will always be appended to the existing list, and will result in an increment of the decimal portion of CLASSIFIER_VERSION. See also CLASSIFIER_BAND_INDEX_NAME.	**Valid Values** n/a **Type** string array **Units** n/a **Location** DERIVED_IMAGE_PARMS (Group)	**Mode** RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** N/A **Type** N/A
Ops Keyword CLASSIFIER_MAX_RANGE **PDS Keyword** None **Definition** Applies to TEN/TENL and TER/TERL products. The maximum distance from the camera for which the classifier will produce terrain class probabilities. Pixels beyond this distance are assigned to the "None" class in TERL products. Value applies to the entire line of pixels. Note that this may not exactly correspond to the range in the RNG product due to approximations made by the classifier.	**Valid Values** n/a **Type** integer **Units** meters **Location** DERIVED_IMAGE_PARMS (Group)	**Mode** RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** N/A **Type** N/A
Ops Keyword CLASSIFIER_PROJECTION_SCALE **PDS Keyword** None	**Valid Values** n/a **Type** float	**Mode** RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** N/A

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
Definition Applies to TEN/TENL and TER/TERL products. A scale factor in units of meters/pixel. This is only used internally by the terrain classifier, which produces an intermediate image such that distances are a constant scale in meters/pixel across the entire image	**Units** meters/pixel **Location** DERIVED_IMAGE_PARMS (Group)	**Type** N/A
Ops Keyword CLASSIFIER_VERSION **PDS Keyword** None **Definition** Applies to TEN/TENL and TER/TERL products. The version of the terrain classifier code. Integer increments represent updates to the terrain classification algorithm, while decimal increments represent changes or additions to the list of terrain classes.	**Valid Values** n/a **Type** float **Units** n/a **Location** DERIVED_IMAGE_PARMS (Group)	**Mode** RDR-generating software **Field as “<xml name>:[<element>]:[<element>]:<field>”** N/A **Type** N/A
Ops Keyword COMMAND_INSTRUMENT_ID **PDS Keyword** same **Definition** Specifies an abbreviated name or acronym which identifies the instrument that was commanded. NOTE: INSTRUMENT_ID is not a unique identifier for a given instrument. Note also that the associated INSTRUMENT_NAME element provides the full name of the instrument. Example values: IRTM (for Viking Infrared Thermal Mapper), PWS (for plasma wave spectrometer).	**Valid Values** 0 = “NONE” 1 = “FRONT_HAZCAM_LEFT” 2 = “FRONT_HAZCAM_RIGHT” 3 = “REAR_HAZCAM_LEFT” 4 = “REAR_HAZCAM_RIGHT” 5 = “NAVCAM_LEFT” 6 = “NAVCAM_RIGHT” 7 = “CAMERA_IDS” **Type** string(20) **Units** n/a **Location** OBSERVATION_REQUEST_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as “<xml name>:[<element>]:[<element>]:<field>”** • Eng. Cameras “<IDPH DPO>:idph:params:camera” • MMM Cameras “<Ancillary DPO>:cmd_arguments_image:camera” **Type** enum
Ops Keyword COMMAND_SEQUENCE_NUMBER **PDS Keyword** same **Definition** Specifies a numeric identifier for a sequence of commands sent to a spacecraft or instrument. NOTE: For MSL, this is the command number which identifies the specific generating command within the specified	**Valid Values** n/a **Type** integer **Units** n/a **Location** IDENTIFICATION (Class)	**Mode** EMD in XML format **Field as “<xml name>:[<element>]:[<element>]:<field>”** “MslEarthProductMetadata:MslProductMetadata:CommandNumber” **Type** n/a

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
sequence.		
Ops Keyword COMMUNICATION_SESSION_ID **PDS Keyword** MSL:COMMUNICATION_SESSION_ID **Definition** Identifies the communication session that was used to downlink this product. Sessions are identified in a mission-specific manner. For MSL, this is the UHF window ID used to downlink the product. JPL MSL Operations personnel can find details for specific UHF window IDs on MSL-Reports and MaROS (login required). See also EXPECTED_TRANSMISSION_PATH.	**Valid Values** n/a **Type** string **Units** n/a **Location** TELEMETRY (Class)	**Mode** EMD in XML format **Field as “<xml name>:[<element>]:[<element>]:<field>”** “MslEarthProductMetadata:MslProductMetadata:CommSessionId” **Type** n/a
Ops Keyword CONFIGURATION_BIT_ID **PDS Keyword** MSL:CONFIGURATION_BIT_ID **Definition** Specifies an array of strings identifying the configuration of the arm represented by the bits in the reachability product. Each configuration requires 2 bits to describe. The first entry in the array contains the configuration for the 2 most significant bits, while the last entry contains the configuration for the 2 least significant bits. For MSL, the instrument/tools that are useful in a reachability map are DRILL, DRT, MAHLI, APXS and SCOOP_TIP. For MSL, the 8 configuration values are interpreted as follows: a) “ARM_SO_EU_WU” - Shoulder out, Elbow up, Wrist up. b) “ARM_SO_EU_WD” - Shoulder out, Elbow up, Wrist down. c) “ARM_SO_ED_WU” - Shoulder out, Elbow down, Wrist up. d) “ARM_SO_ED_WD” - Shoulder out, Elbow down, Wrist down. e) “ARM_SI_EU_WU” - Shoulder in, Elbow up, Wrist up. f) “ARM_SI_EU_WD” - Shoulder in, Elbow up, Wrist down. g) “ARM_SI_ED_WU” - Shoulder in, Elbow down, Wrist up. h) “ARM_SI_ED_WD” - Shoulder in, Elbow down, Wrist down.	**Valid Values** “ARM_SO_EU_WU”, “ARM_SO_EU_WD”, “ARM_SO_ED_WU”, “ARM_SO_ED_WD”, “ARM_SI_EU_WU”, “ARM_SI_EU_WD”, “ARM_SI_ED_WU”, “ARM_SI_ED_WD” **Type** string array[8] **Units** n/a **Location** DERIVED_IMAGE_PARMS (Group)	**Mode** RDR-generating software, Static value **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
See also INSTRUMENT_BAND_ID.		
Ops Keyword CONTACT_SENSOR_STATE **PDS Keyword** same **Definition** Specifies an array of identifiers for the state of an instrument or an instrument host's contact sensors at a specified time.	**Valid Values** • Bit 0: ARM-APXS Contact Switch 1 … 0 = "NO CONTACT" 1 = "CONTACT" • Bit 1: ARM-APXS Contact Switch 2 … 0 = "NO CONTACT" 1 = "CONTACT" • Bit 2: ARM-DRILL Switch 1 … 0 = "NO CONTACT" 1 = "CONTACT" • Bit 3: ARM-DRILL Switch 2 … 0 = "NO CONTACT" 1 = "CONTACT" • Bit 4: ARM-MAHLI Switch 1A … 0 = "NO CONTACT" 1 = "CONTACT" • Bit 5: ARM-MAHLI Switch 1B … 0 = "NO CONTACT" 1 = "CONTACT" • Bit 6: ARM-MAHLI Switch 2A … 0 = "NO CONTACT" 1 = "CONTACT" • Bit 7: ARM-MAHLI Switch 2B … 0 = "NO CONTACT" 1 = "CONTACT" **Type** string array[8] **Units** n/a **Location** ARM_ARTICULATION_STATE (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** • Eng. Cameras "<IDPH DPO>:idph:arm_contact" • MMM Cameras "<IDPH DPO>:cidph:arm_state:contact" **Type** U8
Ops Keyword CONTACT_SENSOR_STATE_NAME **PDS Keyword** same **Definition** Specifies the possible value that can be contained in the CONTACT_SENSOR_STATE array.	**Valid Values** ("APXS CONTACT SWITCH 1", "APXS CONTACT SWITCH 2", "DRILL SWITCH 1", "DRILL SWITCH 2", "MAHLI SWITCH 1A", "MAHLI SWITCH 1B", "MAHLI SWITCH 2A", "MAHLI SWITCH 2B") **Type** string array[8] **Units** n/a	**Mode** Static value: - Single value representing array of Bits 0 thru 7 from CONTACT_SENSOR_STATE. **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a

<table>
<tr><th colspan="2">OUTPUT METADATA (PRODUCT LABEL)</th><th>INPUT METADATA (SOURCE)</th></tr>
<tr><th>• Ops Keyword
• PDS-Compliant Keyword
• Definition</th><th>• Valid Values (quoted)
• Type
• Units
• Keyword Location in Label</th><th>• Mode
• Metadata Field
• Type</th></tr>
<tr><td></td><td>Location
ARM_ARTICULATION_STATE (Group)</td><td></td></tr>
<tr><td>Ops Keyword
COORDINATE_SYSTEM_INDEX

PDS Keyword
same

Definition
Specifies an integer array used to record and track the movement of a rover or lander during surface operations. When in a COORDINATE_SYSTEM_STATE group, this keyword identifies which instance of the coordinate frame, named by COORDINATE_SYSTEM_NAME, is being defined by the group.

For MSL, these indices are based on the ROVER_MOTION_COUNTER and are in the same order as specified by ROVER_MOTION_COUNTER_NAME.
• EDRs will always contain all 10 elements for this keyword.
• For RDRs, the number of indices can be anything from 1 (used for SITE_FRAME) up to 10; however only 1, 2, 3, and 10 indices are common in RDRs.

Example:
COORDINATE_SYSTEM_INDEX=(1,10)</td><td>Valid Values
n/a

Type
integer array[10]

Units
n/a

Location
• For EDRs …
- ROVER_COORDINATE_SYSTEM (Group)
- RSM_COORDINATE_SYSTEM (Group)
- ARM_COORDINATE_SYSTEM (Group)
• For RDRs …
- ROVER_COORDINATE_SYSTEM (Group)
- RSM_COORDINATE_SYSTEM (Group)
- ARM_COORDINATE_SYSTEM (Group)
- SITE_COORDINATE_SYSTEM (Group)
- LOCAL_LEVEL_COORDINATE_SYSTEM (Group)</td><td>Mode
DPO in XML format (referenced to APID Name in Appendix E), Calculation

Field as “<xml name>:[<element>]:[<element>]:<field>”
“<IDPH DPO>:idph:rmc_site”,
“<IDPH DPO>:idph:rmc_drive”,
“<IDPH DPO>:idph:rmc_pose”,
“<IDPH DPO>:idph:rmc_arm”,
“<IDPH DPO>:idph:rmc_chimra”,
“<IDPH DPO>:idph:rmc_drill”,
“<IDPH DPO>:idph:rmc_rsm”,
“<IDPH DPO>:idph:rmc_hga”,
“<IDPH DPO>:idph:rmc_drt”,
“<IDPH DPO>:idph:rmc_ic”

Type
U16</td></tr>
<tr><td>Ops Keyword
COORDINATE_SYSTEM_INDEX_NAME

PDS Keyword
same

Definition
Specifies an array of the formal names identifying each integer specified in COORDINATE_SYSTEM_INDEX.

NOTE: Should match the number of values in COORDINATE_SYSTEM_INDEX.</td><td>Valid Values
1) “SITE”
2) “DRIVE”
3) “POSE”
4) “ARM”
5) “CHIMRA”
6) “DRILL”
7) “RSM”
8) “HGA”
9) “DRT”
10) “IC”

Type
string array[10]

Units
n/a

Location
• For EDRs …
- ROVER_COORDINATE_SYSTEM (Group)
- RSM_COORDINATE_SYSTEM (Group)
- ARM_COORDINATE_SYSTEM (Group)</td><td>Mode
Static values:
1) for RMC element Site
2) for RMC element Drive
3) for RMC element Pose
4) for RMC element Arm
5) for RMC element CHIMRA
6) for RMC element Drill
7) for RMC element RSM
8) for RMC element HGA
9) for RMC element DRT
10) for RMC element IC

Field as “<xml name>:[<element>]:[<element>]:<field>”
n/a

Type
n/a</td></tr>
</table>

OUTPUT METADATA (PRODUCT LABEL)		**INPUT METADATA (SOURCE)**
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
	• For RDRs … - ROVER_COORDINATE_SYSTEM (Group) - RSM_COORDINATE_SYSTEM (Group) - ARM_COORDINATE_SYSTEM (Group) - SITE_COORDINATE_SYSTEM (Group) - LOCAL_LEVEL_COORDINATE_SYSTEM (Group)	
Ops Keyword COORDINATE_SYSTEM_NAME **PDS Keyword** same **Definition** Specifies the full name of the coordinate system to which the state vectors are referenced. When in a COORDINATE_SYSTEM group, this keyword provides the full name of the coordinate system being defined by the group. The rest of the keywords in the group describe how this coordinate system is related to some other (the "reference"). Non-unique coordinate systems (such as "SITE" for rover or lander missions), which have multiple instances using the same name, also require COORDINATE_SYSTEM_INDEX to completely identify the coordinate system. NOTE: A CS is named by three things: 1) the CS name (e.g. site, rover), 2) the set of indices, and 3) the solution ID (see SOLUTION_ID). A set of index-names is sometimes included for documentation purposes only. The solution ID is often omitted to indicate the default, but is logically part of the name. Any time there's a location (XYZ) or an orientation (in quaternion or angle form), the label group containing that location/orientation should contain the name of the CS in which it is expressed. This means it should contain the three "REFERENCE_*" keywords (or two if the solution ID is omitted). This includes CS definition groups, which have to define in what frame the origin/rotation numbers are expressed (i.e. what is the reference frame).	**Valid Values** • For EDRs … 1) "ROVER_NAV_FRAME" 3) "RSM_HEAD_FRAME" 4) a. "ARM_TURRET_FRAME" b. "ARM_DRILL_FRAME" c. "ARM_DRT_FRAME" d. "ARM_MAHLI_FRAME" e. "ARM_APXS_FRAME" f. "ARM_PORTION_FRAME" g. "ARM_SCOOP_TIP_FRAME" h. "ARM_SCOOP_TCP_FRAME" • For RDRs … 1) "ROVER_NAV_FRAME" 2) "ROVER_MECH_FRAME" 3) "RSM_HEAD_FRAME" 4) a. "ARM_TURRET_FRAME" b. "ARM_DRILL_FRAME" c. "ARM_DRT_FRAME" d. "ARM_MAHLI_FRAME" e. "ARM_APXS_FRAME" f. "ARM_PORTION_FRAME" g. "ARM_SCOOP_TIP_FRAME" h. "ARM_SCOOP_TCP_FRAME" 5) "SITE_FRAME" 6) "LOCAL_LEVEL_FRAME" **Type** string **Units** n/a **Location** • For EDRs … 1) ROVER_COORDINATE_SYSTEM (Group) 3) RSM_COORDINATE_SYSTEM (Group) 4) ARM_COORDINATE_SYSTEM (Group) • For RDRs … 1) ROVER_COORDINATE_SYSTEM (Group) 2) RMECH_COORDINATE_SYSTEM (Group) 3) RSM_COORDINATE_SYSTEM (Group)	**Mode** • Static value: 1) for Rover Navigation Frame 2) for Rover Mechanical Frame 3) for RSM Head Frame 4) a. for Arm Turret Frame b. for Arm Drill Frame c. for Arm DRT Frame d. for Arm MAHLI Frame e. for Arm APXS Frame f. for Arm Portion Frame g. for Arm Scoop TIP Frame h. for Arm Scoop TCP Frame 5) for Site Frame 6) for Local Level Frame • DPO in XML format (referenced to APID Name in Appendix E), **Field as "<xml name>:[<element>]:[<element>]:<field>"** 4) "<IDPH DPO>:idph:arm_instrument" **Type** n/a

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
	4) ARM_COORDINATE_SYSTEM (Group) 5) SITE_COORDINATE_SYSTEM (Group) 6) LOCAL_LEVEL_COORDINATE_SYSTEM (Group)	
Ops Keyword DARK_CURRENT_FILE_NAME **PDS Keyword** same **Definition** Specifies the name, or array of names, of the dark current image file(s) (image taken without opening the camera shutter) which should be used to perform radiometric calibration of the image. The dark current image provides a reference label of the build-up of any charges on the sensor that need to be subtracted from a shuttered image during calibration. Selection of the appropriate dark current image may be based on time, camera, temperature, readout conditions, light flood, gain and offset.	**Valid Values** n/a **Type** string array **Units** n/a **Location** DERIVED_IMAGE_PARMS (Group)	**Mode** RDR-generating software **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a
Ops Keyword DARK_CURRENT_FILE_DESC **PDS Keyword** same **Definition** Specifies a description of the corresponding dark current files listed in DARK_CURRENT_FILE_NAME.	**Valid Values** n/a **Type** string **Units** n/a **Location** DERIVED_IMAGE_PARMS (Group)	**Mode** RDR-generating software **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a
Ops Keyword DARK_SPECTRA_MODE **PDS Keyword** MSL:DARK_SPECTRA_MODE **Definition** For ChemCam on MSL, specifies the command to acquire “pre-LIBS” and/or “post-LIBS” dark spectra. It is an argument for command CCAM_ACTV_SPECTRAL_OBS (see ChemCam FDD).	**Valid Values** 0 = “PRE_ONLY” 1 = “POST_ONLY” 2 = “PRE_AND_POST" 3 = “NONE” **Type** string **Units** n/a **Location** OBSERVATION_REQUEST_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as “<xml name>:[<element>]:[<element>]:<field>”** ChemCam “<Ancillary DPO>:cmd_arguments:pre_post_dark_spectra” **Type** enum
Ops Keyword DATA_SET_ID **PDS Keyword** same	**Valid Values** • “Operations” EDRs “MSL-M-HAZCAM-2-EDR-V1.0”, “MSL-M-NAVCAM-2-EDR-V1.0”, “MSL-M-CHEMCAM-RMI-2-EDR-V1.0”,	**Mode** PDS, Table Lookup **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a

<table>
<tr><th colspan="2">OUTPUT METADATA (PRODUCT LABEL)</th><th>INPUT METADATA (SOURCE)</th></tr>
<tr><th>• Ops Keyword
• PDS-Compliant Keyword
• Definition</th><th>• Valid Values (quoted)
• Type
• Units
• Keyword Location in Label</th><th>• Mode
• Metadata Field
• Type</th></tr>
<tr><td>Definition
Specifies a unique alphanumeric identifier for a data set or a data product.

The DATA_SET_ID value for a given data set or product is constructed according to flight project naming conventions. In most cases the DATA_SET_ID is an abbreviation of the DATA_SET_NAME.

In the PDS, the values for DATA_SET_ID are constructed according to standards outlined in the Standards Reference.</td><td>“MSL-M-CHEMCAM-LIBS-2-EDR-V1.0”,
“MSL-M-CHEMCAM-SOH-2-EDR-V1.0”,
“MSL-M-MASTCAM-2-EDR-OPS-V1.0”,
“MSL-M-MAHLI-2-EDR-OPS-V1.0”,
“MSL-M-MARDI-2-EDR-OPS-V1.0”

• <u>“Operations” Image Single-frame RDRs</u>
“MSL-M-HAZCAM-5-RDR-V1.0”,
“MSL-M-NAVCAM-5-RDR-V1.0”,
“MSL-M-CHEMCAM-RMI-5-RDR-V1.0”,
“MSL-M-MASTCAM-5-RDR-V1.0”,
“MSL-M-MAHLI-5-RDR-V1.0”,
“MSL-M-MARDI-5-RDR-V1.0”

• <u>“Operations” Terrain Mesh RDRs</u>
“MSL-M-HAZCAM-5-RDR-MESH-V1.0”,
“MSL-M-NAVCAM-5-RDR-MESH-V1.0”,
“MSL-M-MASTCAM-5-RDR-MESH-V1.0”

• <u>“Operations” Image Mosaic RDRs</u>
“MSL-M-NAVCAM-5-RDR-MOSAIC-V1.0”,
“MSL-M-CHEMCAM-RMI-5-RDR-MOSAIC-V1.0”,
“MSL-M-MASTCAM-5-RDR-MOSAIC-V1.0”,
“MSL-M-MAHLI-5-RDR-MOSAIC-V1.0”

• <u>“Operations” Spectroscopy RDR</u>
“MSL-M-CHEMCAM-LIBS-5-RDR-V1.0”

• <u>“Operations” RDRs (other)</u>
“MSL-M-ROVER-6-RDR-RMC-V1.0”

Type
string(40)

Units
n/a

Location
IDENTIFICATION (Class)</td><td>Type
n/a</td></tr>
<tr><td>Ops Keyword
DATA_SET_NAME

PDS Keyword
same

Definition
Specifies the full name given to a data set or a data product.

The DATA_SET_NAME typically identifies the instrument that acquired the data, the target of that instrument, and the</td><td>Valid Values
• <u>“Operations” EDRs</u>
"MSL MARS HAZARD AVOIDANCE CAMERA 2 EDR V1.0",
“MSL MARS NAVIGATION CAMERA 2 EDR V1.0",
"MSL MARS CHEMCAM REMOTE MICRO-IMAGER CAMERA 2 EDR V1.0",
"MSL MARS CHEMCAM LASER-INDUCED BREAKDOWN SPECTRA 2 EDR V1.0",
"MSL MARS CHEMCAM STATE OF HEALTH 2 EDR V1.0",
"MSL MARS MAST CAMERA 2 EDR OPS V1.0",</td><td>Mode
PDS, Table Lookup

Field as “<xml name>:[<element>]:[<element>]:<field>”
n/a

Type
n/a</td></tr>
</table>

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
processing level of the data. In the PDS, values for DATA_SET_NAME are constructed according to standards outlined in the Standards Reference.	"MSL MARS HAND LENS IMAGER CAMERA 2 EDR OPS V1.0", "MSL MARS DESCENT IMAGER CAMERA 2 EDR OPS V1.0" • “Operations” Image Single-frame RDRs “MSL MARS HAZARD AVOIDANCE CAMERA 5 RDR V1.0", "MSL MARS NAVIGATION CAMERA 5 RDR V1.0", "MSL MARS CHEMCAM REMOTE MICRO-IMAGER CAMERA 5 RDR V1.0" • “Operations” Terrain Mesh RDRs "MSL MARS HAZARD AVOIDANCE CAMERA 5 RDR TERRAIN MESH V1.0", "MSL MARS NAVIGATION CAMERA 5 RDR TERRAIN MESH V1.0", "MSL MARS MAST CAMERA 5 RDR TERRAIN MESH OPS V1.0" • “Operations” Image Mosaic RDRs "MSL MARS NAVIGATION CAMERA 5 RDR MOSAIC V1.0", "MSL MARS CHEMCAM RMI 5 RDR MOSAIC V1.0", "MSL MARS MAST CAMERA 5 RDR MOSAIC OPS V1.0" • “Operations” Spectroscopy RDR "MSL MARS CHEMCAM LASER-INDUCED BREAKDOWN SPECTRA 5 RDR V1.0" • “Operations” RDRs (other) “MSL MARS ROVER 6 RDR ROVER MOTION COUNTER V1.0" **Type** string **Units** n/a **Location** IDENTIFICATION (Class)	
Ops Keyword DERIVED_IMAGE_TYPE **PDS Keyword** same **Definition** Specifies how to interpret the pixel values in a derived image RDR (or colloquially, the type of the derived image itself).	**Valid Values** "IMAGE", "DISPARITY_MAP", "DISPARITY_LINE_MAP", "DISPARITY_SAMPLE_MAP", “DISPARITY_ERROR_MAP”, “DELTA_DISPARITY_MAP”, “DELTA_DISPARITY_LINE_MAP”, “DELTA_DISPARITY_SAMPLE_MAP”, “MASK”, "XYZ_MAP", “XYZ_ERROR_MAP”, "X_MAP", "Y_MAP", "Z_MAP", “ANGLE_MAP”, "RANGE_MAP", “RANGE_ERROR_MAP”, "UVW_MAP", "U_MAP", "V_MAP", "W_MAP", “ROUGHNESS_MAP”, “REACHABILITY_MAP”,	**Mode** RDR-generating software **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
Values are defined as: • “IMAGE” - Standard image, where intensity. Note that this implies nothing about radiometric, geometric, or other corrections that mayhave been applied. • “DISPARITY_MAP” - Line and sample disparity coordinates with respect to another image (2 bands). • “DISPARITY_LINE_MAP” - Line disparity only. • “DISPARITY_SAMPLE_MAP” - Sample disparity only. • “DISPARITY_ERROR_MAP” - Disparity error metric. • “DELTA_DISPARITY_MAP” - Line and sample as an offset with respect to another image (2 bands). • “DELTA_DISPARITY_LINE_MAP” - Line delta disparity only. • “DELTA_DISPARITY_SAMPLE_MAP” - Sample delta disparity only. • “MASK” - Mask to apply to another image. • “XYZ_MAP” - XYZ values (3 bands). • “XYZ_ERROR_MAP” - XYZ error metric. • “X_MAP” - X component of an XYZ image. • “Y_MAP” - Y component of an XYZ image. • “Z_MAP” - Z component of an XYZ image. • “ANGLE_MAP” - Angle between 2 vectors. • “RANGE_MAP” - Distance from the camera center. • “RANGE_ERROR_MAP” - Range error metric. • “UVW_MAP” - Surface Normal values (3 bands associating to X,Y,Z). • “U_MAP” - U (X) component of a Surface Normal image. • “V_MAP” - V (Y) component of a Surface Normal image. • “W_MAP” - W (Z) component of a Surface Normal image. • “ROUGHNESS_MAP” - Measure of surface roughness. • “REACHABILITY_MAP” - Flags pixels reachable by the robotic arm instruments (5 bands) • “PRELOAD_MAP” - Arm preload value. • “SLOPE_MAP” - Slope angle. • “RADIAL_SLOPE_MAP” - Component of slope radial to the rover. • “SLOPE_HEADING_MAP” - Slope heading. • “SLOPE_MAGNITUDE” - Slope magnitude. • “NORTHERLY_TILT_MAP” - North-facing component of slope. • “SOLAR_ENERGY_MAP” - Available fraction of maximum solar energy. • “IEP_MAP” - Incidence, Emission and Phase angles. • "TERRAIN_CLASSIFICATION_MAP" - Terrain Class Map or confidence score map. • "AEGIS_MAP" - Regions of Interest map as identified onboard by AEGIS software. • "ICM_MAP" - Source image pixel location map • "IDX_MAP" - Source image list index map	“PRELOAD_MAP”, “SLOPE_MAP”, “RADIAL_SLOPE_MAP”, “SLOPE_HEADING_MAP”, “SLOPE_MAGNITUDE”, “NORTHERLY_TILT_MAP”, “SOLAR_ENERGY_MAP”, “IEP_MAP”, “TERRAIN_CLASSIFICATION_MAP”, “AEGIS_MAP”, “ICM_MAP”, “IDX_MAP” **Type** string **Units** n/a **Location** DERIVED_IMAGE_PARMS (Group)	

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
Ops Keyword ^DESCRIPTION **PDS Keyword** same **Definition** Specifies a pointer that provides a free-form, unlimited-length character string that represents or gives an account of something.	**Valid Values** "VICAR2.TXT" **Type** string **Units** n/a **Location** IMAGE_HEADER (Object)	**Mode** Static Value **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword DETECTOR_ERASE_COUNT **PDS Keyword** same **Definition** Specifies the number of times a detector has been flushed of data in raw counts. For ChemCam on MSL, specifies the commanded number of images before transferring the current image to the FPGA. For MMM cameras on MSL, this reports the value of the camera head vertical register (vflush) parameter.	**Valid Values** "0" to "15" **Type** integer **Units** n/a **Location** 1) OBSERVATION_REQUEST_PARMS (Group) 2) OBSERVATION_REQUEST_PARMS (Group) 3) a. OBSERVATION_REQUEST_PARMS (Group) b. MINI_HEADER (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** 1) Eng. Cameras "<IDPH DPO>:idph:params:flush" 2) Chemcam "<Ancillary DPO>:cmd_parameters:CCD_cleanCount" 3) MMM Cameras a. "MMM_Image_Mini_Header[**TBD**]", b. "MMM_Image_Mini_Header[**TBD**]" **Type** U8
Ops Keyword DETECTOR_FIRST_LINE **PDS Keyword** same **Definition** Specifies the starting row from the hardware, such as a charge-coupled device (CCD), that contains data.	**Valid Values** • Eng. Cameras "1" to "1024" • MMM Cameras "UNK" **Type** integer **Units** n/a **Location** 1) INSTRUMENT_STATE_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** Eng. Cameras "<IDPH DPO>:idph:hw_minrow" **Type** U16
Ops Keyword DETECTOR_LINES **PDS Keyword** same **Definition** Specifies the number of rows extracted from the hardware, such as a charge-coupled device (CCD), that contain data.	**Valid Values** • Eng. Cameras "0" to "1024" • MMM Cameras "UNK" **Type** integer	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** Eng. Cameras "<IDPH DPO>:idph:hw_numrows" **Type** U16

<table>
<tr><th colspan="2">OUTPUT METADATA (PRODUCT LABEL)</th><th>INPUT METADATA (SOURCE)</th></tr>
<tr><th>• Ops Keyword
• PDS-Compliant Keyword
• Definition</th><th>• Valid Values (quoted)
• Type
• Units
• Keyword Location in Label</th><th>• Mode
• Metadata Field
• Type</th></tr>
<tr><td></td><td>Units
n/a

Location
INSTRUMENT_STATE_PARMS (Group)</td><td></td></tr>
<tr><td>Ops Keyword
DETECTOR_TO_IMAGE_ROTATION

PDS Keyword
same

Definition
Specifies the clockwise rotation, in degrees, that was applied to an image along its optical path through an instrument, from detector to final image orientation.</td><td>Valid Values
• Eng. Cameras

90.0 : All Eng. Cams Except NLB
270.0 : NLB (Navcam L, B-side)

• MMM Cameras
“UNK”

Type
float

Units
Deg

Location
INSTRUMENT_STATE_PARMS (Group)</td><td>Mode
DPO in XML format (referenced to APID Name in Appendix E), Table Lookup

Field as “<xml name>:[<element>]:[<element>]:<field>”
Eng. Cameras
“<IDPH DPO>:idph:rotation”

Type
enum</td></tr>
<tr><td>Ops Keyword
DOWNLOAD_PRIORITY

PDS Keyword
same

Definition
Specifies which data to downlink/transmit, based on order of importance. The lower numerical priority (higher-ranked number) data products are transmitted before higher numerical priority (lower-ranked number) data products.

For example, an image with a downlink priority of 25 will be transmitted from the rover before an image with a downlink priority of 50.

Value of “0” specifies use of on-board default.</td><td>Valid Values
“0” to “101”

Type
integer

Units
n/a

Location
1) TELEMETRY (Class)
2) IMAGE_REQUEST_PARMS (Group)
3) THUMBNAIL_REQUEST_PARMS (Group)
4) COLUMN_SUM_REQUEST_PARMS (Group)
5) ROW_SUM_REQUEST_PARMS (Group)
6) HISTOGRAM_REQUEST_PARMS (Group)
7) REFERENCE_PIXEL_REQUEST_PARMS (Group)
8) CHEMCAM_SOH_REQUEST_PARMS (Group)</td><td>Mode
DPO in XML format (referenced to APID Name in Appendix E)

Field as “<xml name>:[<element>]:[<element>]:<field>”
• Eng. Cameras & ChemCam
1) “MslEarthProductMetadata:MslProductMetadata: Product_Priority”
2) “<IDPH DPO>:idph:params:img_prio”
3) “<IDPH DPO>:idph:params:thumb_prio”
4) “<IDPH DPO>:idph:params:col_prio”
5) “<IDPH DPO>:idph:params:row_prio”
6) “<IDPH DPO>:idph:params:hist_prio”
7) “<IDPH DPO>:idph:params:ref_prio”
• ChemCam only
3) “<Ancillary DPO>:cmd_parameters:Thumbnail_DP_Priority”
7) “<Ancillary DPO>:cmd_parameters: Reference_Pix_DP_Priority”
8) “MslEarthProductMetadata:MslProductMetadata: Product_Priority”
• MMM Cameras only
1) “<Ancillary DPO>:cmd_arguments_image: image_data_priority”

Type
U8</td></tr>
</table>

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
Ops Keyword DOWNSAMPLE_METHOD **PDS Keyword** same **Definition** Specifies whether or not hardware downsampling was applied to an image.	**Valid Values** • Eng. Cameras "HARDWARE", "SOFTWARE", "BOTH", "NONE" • MMM Cameras "HARDWARE", "NONE" **Type** string **Units** n/a **Location** INSTRUMENT_STATE_PARMS (Group)	**Mode** 1) DPO in XML format (referenced to APID Name in Appendix E), Calculation 2) EMD in XML format **Field as "<xml name>:[<element>]:[<element>]:<field>"** • Eng. Cameras 1) a. "<IDPH DPO>:idph:hw_binning" b. "<IDPH DPO>:idph:rotation" c. "<IDPH DPO>:idph:res_rows" d. "<IDPH DPO>:idph:res_cols" • MMM Cameras 2) "MslEarthProductMetadata:MslProductMetadata: DataFileName" NOTES: • For Eng. Cameras, if res_rows=1 and res_cols=1, then "NONE". • For Eng. Cameras, if hw_binning is false, then "SOFTWARE". • For Eng. Cameras, if hw_binning is TRUE and (rotation=0 or 2 and res_rows=4 and res_cols=1) or (rotation=1 or 3 and res_rows=1 and res_cols=4), then "HARDWARE". • For Eng. Cameras, if hw_binning is TRUE and you get res_rows and res_cols as something else, then "BOTH". • For MMM Cameras, if product filename is Thumbnail, then "HARDWARE". • For MMM Cameras, if product filename is not Thumbnail, then "NONE". **Type** 1) a. boolean b. enum c. U16 d U16 2) n/a
Ops Keyword EARLY_IMAGE_RETURN_FLAG **PDS Keyword** same **Definition** Specifies the deferral of on-board post processing of the image and the returns the image early to an onboard client.	**Valid Values** • Eng. Cameras 0 = "FALSE" 1 = "TRUE" • MMM Cameras "N/A" **Type** string **Units** n/a **Location**	**Mode** DPO in XML format (referenced to APID Name in Appendix E), Table Lookup **Field as "<xml name>:[<element>]:[<element>]:<field>"** Eng. Cameras "<IDPH DPO>:idph:params:img_early" **Type** boolean

<table>
<tr><th colspan="2">OUTPUT METADATA (PRODUCT LABEL)</th><th>INPUT METADATA (SOURCE)</th></tr>
<tr><th>• Ops Keyword
• PDS-Compliant Keyword
• Definition</th><th>• Valid Values (quoted)
• Type
• Units
• Keyword Location in Label</th><th>• Mode
• Metadata Field
• Type</th></tr>
<tr><td></td><td>OBSERVATION_REQUEST_PARMS (Group)</td><td></td></tr>
<tr><td>Ops Keyword
EARLY_PIXEL_SCALE_FLAG

PDS Keyword
same

Definition
Specifies the scaling of pixels. If TRUE, pixels are scaled early (from 12 to 8 bits).</td><td>Valid Values
• Eng. Cameras
0 = “FALSE”
1 = “TRUE”

Type
string

Units
n/a

Location
OBSERVATION_REQUEST_PARMS (Group)</td><td>Mode
DPO in XML format (referenced to APID Name in Appendix E), Table Lookup

Field as “<xml name>:[<element>]:[<element>]:<field>”
Eng. Cameras
“<IDPH DPO>:idph:params:scale_early”

Type
boolean</td></tr>
<tr><td>Ops Keyword
EARTH_RECEIVED_START_TIME

PDS Keyword
same

Definition
Specifies the beginning time at which telemetry was received during a time period of interest. This should be represented in UTC system format.</td><td>Valid Values
<YYYY>-<MM>-<DD>T<hh>:<mm>:<ss>[.<fff>]

Type
time

Units
n/a

Location
TELEMETRY (Class)</td><td>Mode
EMD in XML format, Calculation

Field as “<xml name>:[<element>]:[<element>]:<field>”
“MslEarthProductMetadata:MslProductMetadata:PartList:Part:Ert”

NOTES:
• Value from field “Ert” is extracted for all Parts in PartList. The resulting list is sorted. The lowest value is used as Start time.

Type
n/a</td></tr>
<tr><td>Ops Keyword
EARTH_RECEIVED_STOP_TIME

PDS Keyword
same

Definition
Specifies the ending time for receiving telemetry during a time period of interest. This should be represented in UTC system format.</td><td>Valid Values
<YYYY>-<MM>-<DD>T<hh>:<mm>:<ss>[.<fff>]

Type
time

Units
n/a

Location
TELEMETRY (Class)</td><td>Mode
EMD in XML format, Calculation

Field as “<xml name>:[<element>]:[<element>]:<field>”
“MslEarthProductMetadata:MslProductMetadata:PartList:Part:Ert”

NOTES:
• Value from field “Ert” is extracted for all Parts in PartList. The resulting list is sorted. The highest value is used as Stop time.

Type
n/a</td></tr>
<tr><td>Ops Keyword
ELEVATION_FOV

PDS Keyword
same

Definition
Specifies the angular measure of the vertical field of view of an imaged scene.

NOTE: For MER, this was computed as IFOV times the</td><td>Valid Values
n/a

Type
float

Units
deg

Location
INSTRUMENT_STATE_PARMS (Group)</td><td>Mode
• 1) DPO in XML format (referenced to APID Name in Appendix E)
2) DPO in XML format (referenced to APID Name in Appendix E) or Image DPO mini-header
• Calculation:
- Line/sample is translated into an origin and “look direction” vectors (left and right). The vectors are then used to calculate Elevation field of view.

Field as “<xml name>:[<element>]:[<element>]:<field>”</td></tr>
</table>

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
number of vertical pixels. For MSL, it is computed by projecting rays from the top and bottom edges of the image at the center through the camera model, and computing the angle subtended by those rays.		• Eng. Cameras 1) a. "<IDPH DPO>:idph:cols" b. "<IDPH DPO>:idph:rows" c. "<IDPH DPO>:idph:res_cols" d. "<IDPH DPO>:idph:res_rows" • MMM Cameras 2) a. "<Ancillary DPO>:cmd_arguments_image:window_width" b. "<Ancillary DPO>:cmd_arguments_image:window_height" c. "MMM_Image_Mini_Header[22]" d. "MMM_Image_Mini_Header[23]" NOTES: • For MMM non-recovered data products (cases "a" and "b"), float values come from the Ancillary DPO. • For MMM recovered data products (cases "c" and "d"), values analogous to cases "a" (window_width) and "b" (window_height) are each comprised of one byte coming from Image DPO mini-header at byte offsets 22 and 23, respectively. • For MMM, parm "window_width" is number of image samples. • For MMM, parm "window_height" is number of image lines. • For MMM non-Thumbnail data products, multiply by factor of 8 to convert to correct line/sample value. **Type** 1) U16 2) U8
Ops Keyword ERROR_PIXELS **PDS Keyword** same **Definition** Specifies the number of pixels that are outside a valid DN range, after all decompression and post decompression processing has been completed.	**Valid Values** n/a **Type** integer **Units** n/a **Location** COMPRESSION_PARMS (Group)	**Mode** Calculation based on telemetry **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword EXPECTED_PACKETS **PDS Keyword** same **Definition** Specifies the total number of telemetry packets which constitute a complete data product, i.e., a data product without missing data. NOTE: For MSL, telemetry data processing does not track	**Valid Values** n/a **Type** integer **Units** n/a **Location** TELEMETRY (Class)	**Mode** EMD in XML format **Field as "<xml name>:[<element>]:[<element>]:<field>"** "MslEarthProductMetadata:MslProductMetadata:PartList: TotalExpected" **Type** n/a

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
"packets", but instead data product "parts".		
Ops Keyword EXPECTED_TRANSMISSION_PATH **PDS Keyword** MSL:EXPECTED_TRANSMISSION_PATH **Definition** Indicates the planned transmission paths for the Data Product. Specific interpretations are mission-specific. For MSL, this is a bitmask that indicates the types of comm windows in which a data product may be transmitted. Most science data products use 3851 (default, as defined in flight software). Ongoing test campaigns with the MAVEN and TGO orbiters have used path 4096 ("MEX") until ~March of 2018, and currently use path 128 ("SPARE_5"). See also COMMUNICATION_SESSION_ID.	**Valid Values** "3851" = DEFAULT "65535" = ALL [-1] "128" = SPARE_5 "4096" = MEX "0" = NONE "1" = XBD "2" = BG "4" = EDL "8" = SPARE_1 "16" = SPARE_2 "32" = SPARE_3 "64" = SPARE_4 "256" = PM_MRO "512" = PM_ODY "768" = CRIT_UHF "1024" = AM_MRO "2048" = AM_ODY "3072" = SCI_UHF "3840" = UHF "8192" = ANOMALY **Type** string **Units** n/a **Location** TELEMETRY (Class)	**Mode** EMD in XML format **Field as "<xml name>:[<element>]:[<element>]:<field>"** "MslEarthProductMetadata:MslProductMetadata: TransmissionControlCriterion" **Type** n/a
Ops Keyword EXPOSURE_COUNT **PDS Keyword** same **Definition** Specifies the maximum number of exposures taken during a specified interval. The value is dependent on exposure type.	**Valid Values** "0" to "255" **Type** integer **Units** n/a **Location** 1) INSTRUMENT_STATE_PARMS (Group) 2) INSTRUMENT_STATE_PARMS (Group) MINI_HEADER (Group) ** ** Only MAHLI Z-Stack & Range Map	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** • Eng. Cameras 1) "<IDPH DPO>:idph:exp_count" • MMM Cameras 2) "MMM_Image_Mini_Header[TBD]" **Type** U16
Ops Keyword EXPOSURE_DURATION **PDS Keyword**	**Valid Values** n/a **Type**	**Mode** • 1) DPO in XML format (referenced to APID Name in Appendix E) 2) Image DPO mini-header • Calculation:

OUTPUT METADATA (PRODUCT LABEL)		**INPUT METADATA (SOURCE)**
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
same **Definition** Specifies the value of the time between the opening and closing of an instrument aperture (such as a camera shutter). For MSL, there are no mechanical shutters. Instead, an "electronic shutter" concept was adopted whereby the detectors accumulate charge for EXPOSURE_DURATION amount of time and then that charge is flushed to a masked frame transfer area for readout and digitization.	float **Units** ms (<ms> unit tag required) **Location** 1) INSTRUMENT_STATE_PARMS (Group)	- Value = exp_time * <factor> **Field as "<xml name>:[<element>]:[<element>]:<field>"** • Eng. Cameras 1) "<IDPH DPO>:idph:exp_time" • MMM Cameras 2) a. "MMM_Image_Mini_Header[17|", "MMM_Image_Mini_Header[18|", "MMM_Image_Mini_Header[19|" b. "MMM_Image_Mini_Header[28|" NOTES: • Value of "exp_time" is in raw counts. • For MMM non-Z-stack data products (case "a"), a value analogous to "exp_time" is comprised of three bytes coming from Image DPO mini-header at offsets 17, 18 and 19. • For MMM Z-stack data products (case "b"), a value analogous to "stack depth count" is comprised of one byte coming from Image DPO mini-header at offset 28. • For Eng. Cameras, each raw count translates to 5.12 ms. So, calculation is: value = exp_time * 5.12 • For MMM Cameras, each raw count translates to 0.1 ms. So, calculation is: value = exp_time * 0.1 **Type** U16
Ops Keyword EXPOSURE_DURATION_COUNT **PDS Keyword** same **Definition** Specifies the value, in raw counts, of the time interval between the opening and closing of an instrument aperture (such as a camera shutter). This is a raw value taken directly from telemetry, as opposed to EXPOSURE_DURATION, which has been converted to engineering units. For ChemCam on MSL, it is the exposure time for "manual" exposure and seed time for "auto" exposure of the RMI.	**Valid Values** "0" to "65535" **Type** integer **Units** ms (<ms> unit tag required) **Location** 1) a. OBSERVATION_REQUEST_PARMS (Group) b. INSTRUMENT_STATE_PARMS (Group) 2) OBSERVATION_REQUEST_PARMS (Group) 3) a. OBSERVATION_REQUEST_PARMS (Group) b. MINI_HEADER (Group) ** ** Except MAHLI Z-Stack & Range Map	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** 1) Eng. Cameras a. "<IDPH DPO>:idph:params:exp_time" b. "<IDPH DPO>:idph:exp_time" 2) Chemcam "<Ancillary DPO>:cmd_arguments:exposure_time" 3) MMM Cameras a. "<Ancillary DPO>:cmd_arguments_image:exp_time" b. "MMM_Image_Mini_Header[TBD]" **Type** U16
Ops Keyword EXPOSURE_SCALE_FACTOR **PDS Keyword** same	**Valid Values** • MMM Cameras "N/A" **Type**	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** Eng. Cameras "<IDPH DPO>:idph:params:exp_time_scale"

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
Definition Specifies a multiplier to the base exposure time. The base exposure time is either user-commanded or is read from the onboard exposure time table. The resulting number is used by the cameras as the actual commanded exposure time. This scale factor is commonly used during multi-spectral imaging, when the base exposure time is known for one filter and EXPOSURE_SCALE_FACTOR is used to scale the exposure time to levels appropriate for the other filters.	float **Units** n/a **Location** OBSERVATION_REQUEST_PARMS (Group)	 **Type** F32
Ops Keyword EXPOSURE_TABLE_ID **PDS Keyword** same **Definition** Specifies a description for the exposure count value.	**Valid Values** • Eng. Cameras 0 = “NONE” 1 = “FHAZCAM_L” 2 = “FHAZCAM_R” 3 = “RHAZCAM_L” 4 = “RHAZCAM_R” 5 = “NAVCAM_L” 6 = “NAVCAM_R” 7 = “CAMCONFIG_IDS” • MMM Cameras “N/A” **Type** string **Units** n/a **Location** OBSERVATION_REQUEST_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E), Table Lookup **Field as “<xml name>:[<element>]:[<element>]:<field>”** Eng. Cameras “<IDPH DPO>:idph:params:exp_table” **Type** enum
Ops Keyword EXPOSURE_TBL_UPDATE_FLAG **PDS Keyword** same **Definition** Specifies whether or not an exposure table entry was updated.	**Valid Values** 0 = “FALSE” 1 = “TRUE” **Type** string **Units** n/a **Location** OBSERVATION_REQUEST_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as “<xml name>:[<element>]:[<element>]:<field>”** “<IDPH DPO>:idph:params:exp_update” **Type** boolean
Ops Keyword EXPOSURE_TYPE **PDS Keyword** same **Definition**	**Valid Values** 1) Eng. Cameras 0 = “NONE” 1 = “MANUAL” 2 = “AUTO” 3 = “TEST”	**Mode** 1) DPO in XML format (referenced to APID Name in Appendix E), Table Lookup 2) DPO in XML format (referenced to APID Name in Appendix E), Table Lookup 3) Image DPO mini-header

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
Specifies the exposure mode used for image acquisition.	2) Chemcam 0 = "MANUAL" 1 = "AUTO" 3) MMM Cameras 0 = "MANUAL" non-0 = "AUTO" **Type** string(15) **Units** n/a **Location** 1) OBSERVATION_REQUEST_PARMS (Group) 2) OBSERVATION_REQUEST_PARMS (Group) 3) OBSERVATION_REQUEST_PARMS (Group) MINI_HEADER (Group) ** ** Except MAHLI Z-Stack & Range Map	**Field as "<xml name>:[<element>]:[<element>]:<field>"** 1) Eng. Cameras "<IDPH DPO>:idph:params:exposure" 2) Chemcam "<Ancillary DPO>:cmd_arguments:exposure_type" 3) MMM Cameras "MMM_Image_Mini_Header[28]", "MMM_Image_Mini_Header[29]", "MMM_Image_Mini_Header[30]", "MMM_Image_Mini_Header[31]" NOTES: • For MMM Cameras, a value analogous to "exposure_type" is comprised of four bytes coming from Image DPO mini-header at offsets 28, 29, 30 and 31. **Type** Enum
Ops Keyword FILE_RECORDS **PDS Keyword** same **Definition** Specifies the number of physical file records, including both label records and data records. NOTE: In the PDS the use of FILE_RECORDS along with other file-related data elements is fully described in the Standards Reference.	**Valid Values** "0" to n **Type** integer **Units** n/a **Location** FILE_DATA_ELEMENT (Class)	**Mode** DPO in XML format (referenced to APID Name in Appendix E), Calculation: - Value = <DPO value> + size of PDS and VICAR labels **Field as "<xml name>:[<element>]:[<element>]:<field>"** "<IDPH DPO>:idph:rows" **Type** U16
Ops Keyword FILTER_NAME **PDS Keyword** same **Definition** Specifies the commonly-used name of the instrument filter through which an image or measurement was acquired or which is associated with a given instrument mode. See also FILTER_NUMBER. NOTE: FILTER_NAME is unique, while the FILTER_NUMBER is not.	**Valid Values** • Eng. Cameras: *N/A* • Mastcam: Filter # Filter Name (Left Eye) 0 "MASTCAM_L1_BROADBAND" 1 "MASTCAM_L2_525NM" 2 "MASTCAM_L3_440NM" 3 "MASTCAM_L4_750NM" 4 "MASTCAM_L5_675NM" 5 "MASTCAM_L6_865NM" 6 "MASTCAM_L7_1035NM" 7 "MASTCAM_L8_880NM_ND5" 8 "NO_POS_CHANGE" Filter # Filter Name (Right Eye) 0 "MASTCAM_R1_BROADBAND"	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** • Eng. Cameras N/A • MMM Cameras "<Supplementary DPO>:left_filter", "<Supplementary DPO>:right_filter" **Type** I8

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
	1 “MASTCAM_R2_525NM” 2 “MASTCAM_R3_440NM” 3 “MASTCAM_R4_750NM” 4 “MASTCAM_R5_905NM” 5 “MASTCAM_R6_935NM” 6 “MASTCAM_R7_1035NM” 7 “MASTCAM_R8_440NM_ND5” 8 “NO_POS_CHANGE” • MARDI & MAHLI: <u>Filter #</u> <u>Filter Name (Right Eye)</u> “N/A” “N/A” *[OBSERVATION_REQUEST_PARMS]* “UNK” “UNK” *[INSTRUMENT_STATE_PARMS]* **<u>Type</u>** string array **<u>Units</u>** n/a **<u>Location</u>** • OBSERVATION_REQUEST_PARMS (Group) • INSTRUMENT_STATE_PARMS (Group)	
<u>Ops Keyword</u> FILTER_NUMBER **<u>PDS Keyword</u>** same **<u>Definition</u>** Specifies the number of an instrument filter through which an image or measurement was acquired or which is associated with a given instrument mode. See also FILTER_NAME. Note: FILTER_NAME is unique, while the FILTER_NUMBER is not.	**<u>Valid Values</u>** See “Filter #” column in “FILTER_NAME” keyword. For MINI_HEADER, MMM Cameras only: Mastcam: *Same as FILTER_NUMBER in OBSERVATION_REQUEST_PARMS and INSTRUMENT_STATE_PARMS.* MARDI & MAHLI**: 0 **<u>Type</u>** integer **<u>Units</u>** n/a **<u>Location</u>** a. OBSERVATION_REQUEST_PARMS (Group) INSTRUMENT_STATE_PARMS (Group) b. MINI_HEADER (Group) ** ** Except MAHLI Z-Stack & Range Map	**<u>Mode</u>** DPO in XML format (referenced to APID Name in Appendix E) **<u>Field as “<xml name>:[<element>]:[<element>]:<field>”</u>** <u>MMM Cameras</u> a. “<Supplementary DPO>:left_filter”, “<Supplementary DPO>:right_filter” b. “MMM_Image_Mini_Header[**TBD**]” **<u>Type</u>** I8
<u>Ops Keyword</u> FIRST_LINE **<u>PDS Keyword</u>** same	**<u>Valid Values</u>** “1” to “1024” **<u>Type</u>** integer	**<u>Mode</u>** 1) DPO in XML format (referenced to APID Name in Appendix E) 2) DPO in XML format (referenced to APID Name in Appendix E) 3) DPO in XML format (referenced to APID Name in Appendix E) or Image DPO mini-header

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
Definition Specifies the line within a source image that corresponds to the first line in a sub-image.	**Units** n/a **Location** 1) a. IMAGE (Object) b. SUBFRAME_REQUEST_PARMS (Group) 2) SUBFRAME_REQUEST_PARMS (Group) 3) a. SUBFRAME_REQUEST_PARMS (Group) b. MINI_HEADER (Group) ** ** Except MAHLI Z-Stack & Range Map	**Field as "<xml name>:[<element>]:[<element>]:<field>"** 1) Eng. Cameras a. "<IDPH DPO>:idph:row0" b. "<IDPH DPO>:idph:params:sub_row0" 2) Chemcam "<Ancillary DPO>:cmd_arguments:start_c_pixel" 3) MMM Cameras a. "<Ancillary DPO>:cmd_arguments_image:window_x" b. "MMM_Image_Mini_Header[20]" NOTES: • Add "1" to the source value so that image lines begin at Line 1 (see Valid Value range). • For MMM non-recovered non-Z-stack data products (case "a"), value comes from the Ancillary DPO. • For MMM recovered data products (case "b"), value analogous to case "a" (window_x) is comprised of one byte coming from Image DPO mini-header at byte offset 20. • For MMM Z-stack data products, the value is unknown. • For MMM, parm "window_x" is starting line of the window. **Type** U16
Ops Keyword FIRST_LINE_SAMPLE **PDS Keyword** same **Definition** Specifies the sample within a source image that corresponds to the first sample in a sub-image.	**Valid Values** "1" to "1024" **Type** integer **Units** n/a **Location** 1) a. IMAGE (Object) b. SUBFRAME_REQUEST_PARMS (Group) 2) SUBFRAME_REQUEST_PARMS (Group) 3) a. SUBFRAME_REQUEST_PARMS (Group) b. MINI_HEADER (Group) ** ** Except MAHLI Z-Stack & Range Map	**Mode** 1) DPO in XML format (referenced to APID Name in Appendix E) 2) DPO in XML format (referenced to APID Name in Appendix E) 3) DPO in XML format (referenced to APID Name in Appendix E) or Image DPO mini-header **Field as "<xml name>:[<element>]:[<element>]:<field>"** 1) Eng. Cameras a. "<IDPH DPO>:idph:col0" b. "<IDPH DPO>:idph:params:sub_col0" 2) Chemcam "<Ancillary DPO>:cmd_arguments:start_r_pixel" 3) MMM Cameras a. "<Ancillary DPO>:cmd_arguments_image:window_y" b. "MMM_Image_Mini_Header[21]" NOTES: • Add "1" to the source value so that image lines begin at Line 1 (see Valid Value range). • For MMM non-recovered non-Z-stack data products (case "a"), value comes from the Ancillary DPO. • For MMM recovered data products (case "b"), value analogous to case "a" (window_y) is comprised of one byte coming from Image DPO mini-header at byte offset 21. • For MMM Z-stack data products, the value is unknown.

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
		• For MMM, parm "window_y" is starting sample of the window. **Type** U16
Ops Keyword FLAT_FIELD_CORRECTION_FLAG **PDS Keyword** same **Definition** Specifies whether or not a flat field correction was applied to an image.	**Valid Values** • Eng. Cameras 1) 0 = "FALSE" 1 = "TRUE" 2) 0 = "FALSE" non-0 = "TRUE" • MMM Cameras "UNK" **Type** string(13) **Units** n/a **Location** 1) OBSERVATION_REQUEST_PARMS (Group) 2) INSTRUMENT_STATE_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** Eng. Cameras 1) "<IDPH DPO>:idph:params:flat" 2) "<IDPH DPO>:idph:flat_params[5]" **Type** 1) boolean 2) F32[5]
Ops Keyword FLAT_FIELD_CORRECTION_PARM **PDS Keyword** same **Definition** Specifies the onboard flat-field coefficients/parameters used in the algorithm to remove the flat field signature. The FLAT_FIELD_CORRECTION_FLAG will indicate if the signature was removed. NOTE: The algorithm used by MSL follows: $new(x,y) = orig(x,y) * ff(x,y)$ where, $r = (x-a)^2 + (y-b)^2$ $ff(x,y) = 1 + c*r + d*r^2 + e*r^3$ and, a = array element 1 b = array element 2 c = array element 3 d = array element 4 e = array element 5	**Valid Values** • Eng. Cameras n/a • MMM Cameras "UNK" **Type** float array[5] **Units** n/a **Location** INSTRUMENT_STATE_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** Eng. Cameras "<IDPH DPO>:idph:flat_params[5]" **Type** F32[5]
Ops Keyword FLAT_FIELD_FILE_NAME **PDS Keyword**	**Valid Values** n/a **Type**	**Mode** RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"**

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
same **Definition** Specifies the name, or array of names, of the flat field image(s) file (an image taken in an optical laboratory of a white background, or an image taken in the dawn with the intention to have an equally illuminated background for the whole image) which should be used to perform radiometric calibration of the image. The flat field image provides a reference label of the sensitivity of the used optics across the field of view. The shuttered image needs to be divided by the flat field image during calibration. Selection of the appropriate flat field image may be based on time, camera, temperature, readout conditions, light flood, gain and offset. For MSL Engineering cameras, the flat field file may be derived from pre-launch ground calibration images (Navcam A-side), algorithmically-generated flats (Hazcams), or in-flight imagery of the martian sky in the anti-sun direction (Navcam B-side).	string array **Units** n/a **Location** DERIVED_IMAGE_PARMS (Group)	n/a **Type** n/a
Ops Keyword FLAT_FIELD_FILE_DESC **PDS Keyword** MSL:FLAT_FIELD_FILE_DESC **Definition** Specifies a description of the corresponding flat field files listed in FLAT_FIELD_FILE_NAME.	**Valid Values** “Flat field image.”, “Flat field standard deviation image.”, "Flat field derived from sky flat sequence NCAM00565 produced by Mark Lemmon at Texas A and M" **Type** string array **Units** n/a **Location** DERIVED_IMAGE_PARMS (Group)	**Mode** RDR-generating software **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a
Ops Keyword FLIGHT_SOFTWARE_MODE **PDS Keyword** MSL:FLIGHT_SOFTWARE_MODE **Definition** Active Flight Software mode at Data Product creation.	**Valid Values** Keyword Value — FSW Field Value “0” UNKNOWN “1” TEST “2” PRELAUNCH “3” LAUNCH “4” ECLIPSE “5” CRUISE “6” EDL_APPROAH “7” EDL_MAIN “8” SURFACE_NOMINAL “9” SURFACE_STANDBY “10” NONPRIME_TEST “11” NONPRIME_PRELAUNCH “12” NONPRIME_LAUNCH “13” NONPRIME_ECLIPSE “14” NONPRIME_CRUISE “15” NONPRIME_EDL_APPROACH	**Mode** EMD in XML format **Field as “<xml name>:[<element>]:[<element>]:<field>”** “MslEarthProductMetadata:MslProductMetadata:FswMode” **Type** unsigned integer

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
	"16" NONPRIME_EDL_MAIN "17" NONPRIME_SURFACE_NOMINAL "18" NONPRIME_SURFACE_STANDBY **Type** string **Units** n/a **Location** TELEMETRY (Class)	
Ops Keyword FLIGHT_SOFTWARE_VERSION_ID **PDS Keyword** same **Definition** Active Flight Software version at Data Product creation. The Flight Software version is an opaque token – there is no arithmetic value associated with the token. The version is defined as the time of the FSW build, in seconds since 12:00:00, Jan. 1, 2000.	**Valid Values** n/a **Type** string **Units** n/a **Location** TELEMETRY (Class)	**Mode** EMD in XML format **Field as "<xml name>:[<element>]:[<element>]:<field>"** "MslEarthProductMetadata:MslProductMetadata:FswVersion" **Type** unsigned integer
Ops Keyword FRAME_ID **PDS Keyword** same **Definition** Specifies an identification for a particular instrument measurement frame. A frame consists of a sequence of measurements made over a specified time interval, and may include measurements from different instrument modes. These sequences repeat from cycle to cycle and sometimes within a cycle. Note that mosaics may contain more than one value in an array.	**Valid Values** • ChemCam RMI 1) "MONO" • ChemCam other 2) "N/A" • MARDI & MAHLI 3) "MONO" • Eng. Cameras & MastCam 4) ("LEFT", "RIGHT", "MONO", "N/A") **Type** string array **Units** n/a **Location** IDENTIFICATION (Class)	**Mode** • Static value: 1) for ChemCam RMI, specify "MONO" 2) for ChemCam other, specify "N/A" 3) for MARDI & MAHLI, specify "MONO" • EMD in XML format **Field as "<xml name>:[<element>]:[<element>]:<field>"** Eng. Cameras & MastCam 4) "MslEarthProductMetadata:MslProductMetadata:ProductName" NOTES: • If ending character in field "ProductName" is "l", valid value is "LEFT". If ending character is "r", valid value is "RIGHT". **Type** n/a
Ops Keyword FRAME_TYPE **PDS Keyword** same	**Valid Values** • ChemCam RMI 1) "MONO" • ChemCam other 2) "N/A"	**Mode** • Static value: 1) for ChemCam RMI, specify "MONO" 2) for ChemCam other, specify "N/A" 3) for MARDI & MAHLI, specify "MONO" 4) for MastCam, specify "STEREO"

<table>
<tr><th colspan="2">OUTPUT METADATA (PRODUCT LABEL)</th><th>INPUT METADATA (SOURCE)</th></tr>
<tr><th>• Ops Keyword
• PDS-Compliant Keyword
• Definition</th><th>• Valid Values (quoted)
• Type
• Units
• Keyword Location in Label</th><th>• Mode
• Metadata Field
• Type</th></tr>
<tr><td>Definition
Specifies whether the image was commanded as part of a stereo pair or as a single left or right monoscopic image.

If FRAME_TYPE=STEREO, a left and a right image should be present for the same IMAGE_ID</td><td>• MARDI & MAHLI
3) "MONO"
• MastCam
4) "STEREO"
• Eng. Cameras
5) 0 = "MONO"
1 = "STEREO"

Type
string(10)

Units
n/a

Location
IDENTIFICATION (Class)</td><td>• DPO in XML format (referenced to APID Name in Appendix E)

Field as "<xml name>:[<element>]:[<element>]:<field>"
Eng. Cameras
5) "IDPH DPO>:idph:stereo"

Type
5) boolean</td></tr>
<tr><td>Ops Keyword
GAIN_NUMBER

PDS Keyword
same

Definition
Specifies the gain value used in the analog to digital conversion. The gain value is a multiplicative factor used in the analog to digital conversion.

For ChemCam on MSL, specifies the commanded gain value in the ChemCam Mast Unit (CCMU).

See also OFFSET_NUMBER.</td><td>Valid Values
"64" to "79"

Type
integer

Units
n/a

Location
OBSERVATION_REQUEST_PARMS (Group)</td><td>Mode
DPO in XML format (referenced to APID Name in Appendix E)

Field as "<xml name>:[<element>]:[<element>]:<field>"
ChemCam
"<Ancillary DPO>:cmd_parameters:AD_gain"

Type
U8</td></tr>
<tr><td>Ops Keyword
GEOMETRY_PROJECTION_TYPE

PDS Keyword
same

Definition
Specifies the state of the pixels in an image before a re-projection has been applied. Describes if or how the pixels have been reprojected. RAW indicates reprojection has not been done; the pixels are as they came from the camera.

For MSL, this means the image uses a CAHVOR or one of the CAHVORE camera models. LINEARIZED means that reprojection has been performed to linearize the camera model (thus removing things like lens distortion). This means the image uses a CAHV camera model.</td><td>Valid Values
"RAW", "LINEARIZED"

Type
string

Units
n/a

Location
IDENTIFICATION (Class)</td><td>Mode
Dependent on EDR/RDR state:
- For EDRs, static (default = "RAW")
- For RDRs, RDR-generating software

Field as "<xml name>:[<element>]:[<element>]:<field>"
n/a

Type
n/a</td></tr>
<tr><td>Ops Keyword</td><td>Valid Values</td><td>Mode</td></tr>
</table>

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
GROUP_APPLICABILITY_FLAG **PDS Keyword** same **Definition** Specifies whether a group of keywords are valid values. It is present in a Group only when information is received from telemetry. For MSL, when in a REQUEST_PARMS group, it specifies whether or not the activity represented by the group was commanded. If TRUE, the rest of the contents of the group specify the commanded arguments or parameters for that activity. For example, a value of TRUE in a HISTOGRAM_REQUEST_GROUP means a histogram was requested.	• Eng. Cameras & ChemCam 0 = "FALSE" non-0 = "TRUE" • MMM Cameras "UNK" **Type** string **Units** n/a **Location** 1) IMAGE_REQUEST_PARMS (Group) 2) SUBFRAME_REQUEST_PARMS (Group) 3) THUMBNAIL_REQUEST_PARMS (Group) 4) COLUMN_SUM_REQUEST_PARMS (Group) 5) ROW_SUM_REQUEST_PARMS (Group) 6) HISTOGRAM_REQUEST_PARMS (Group) 7) REFERENCE_PIXEL_REQUEST_PARMS (Group) 8) REFERENCE_PIXEL_REQUEST_PARMS (Group)	DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** • All Cameras 1) "<IDPH DPO>:idph:params:image" 2) "<IDPH DPO>:idph:params:subframe" 3) "<IDPH DPO>:idph:params:thumbnail" 4) "<IDPH DPO>:idph:params:colsums" 5) "<IDPH DPO>:idph:params:rowsums" 6) "<IDPH DPO>:idph:params:histogram" 7) "<IDPH DPO>:idph:params:ref" • ChemCam only 8) "<Ancillary DPO>:cmd_parameters:Reference_Pix_DP" **Type** 1) boolean 2) enum 3) boolean 4) boolean 5) boolean 6) boolean 7) boolean 8) I32
Ops Keyword HEADER_TYPE **PDS Keyword** same **Definition** Specifies a specific type of header data structure. For example: FITS, VICAR. NOTE: In the PDS, HEADER_TYPE is used to indicate non-PDS headers.	**Valid Values** "VICAR2", "ODL" **Type** string(12) **Units** n/a **Location** IMAGE_HEADER (Object)	**Mode** Static Value **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword HORIZON_MASK_ELEVATION **PDS Keyword** MSL:HORIZON_MASK_ELEVATION **Definition** Specifies the elevation (in degrees) used as the horizontal cutoff in a mask file that, when applied to another image file, prevents the horizon and sky features in the image from being processed. If this keyword is not present in the product label, no horizon mask was used.	**Valid Values** n/a **Type** float **Units** n/a **Location** DERIVED_IMAGE_PARMS (Group)	**Mode** RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
For MSL, this is in degrees as measured in the Local Level (or, equivalently, Site) frame. See also MASK_DESC_FILE_NAME.		
Ops Keyword ICT_DIVIDER **PDS Keyword** MSL:ICT_DIVIDER **Definition** For ChemCam on MSL, specifies the Integration Clock Timer divisor for msec integration time. It is usually 600.	**Valid Values** n/a **Type** integer **Units** msec **Location** OBSERVATION_REQUEST_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as “<xml name>:[<element>]:[<element>]:<field>”** “<Ancillary DPO>:cmd_parameters:ICT_Divider” **Type** U16
Ops Keyword IPBC_DIVIDER **PDS Keyword** MSL:IPBC_DIVIDER **Definition** For ChemCam on MSL, specifies the Integration Period Base Clock divisor. The range is 0 to 33 MHz. It is usually 330 MHz (equal to 100 KHz).	**Valid Values** n/a **Type** integer **Units** MHz **Location** OBSERVATION_REQUEST_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as “<xml name>:[<element>]:[<element>]:<field>”** “<Ancillary DPO>:cmd_parameters:IPBC_Divider” **Type** U16
Ops Keyword ^IMAGE **PDS Keyword** same **Definition** Specifies a pointer to the IMAGE object. See chapter 14 of the PDS Standards Reference for more information on pointer usage.	**Valid Values** n/a **Type** NULL **Units** n/a **Location** POINTERS	**Mode** Calculation **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a
Ops Keyword IMAGE_ACQUIRE_MODE **PDS Keyword** MSL:IMAGE_ACQUIRE_MODE **Definition** This keyword describes the mode of image acquisition. Valid values are defined as: a) “NONE” - No image acquired b) “SERNO” - No image acquired, camera serial number	**Valid Values** • Eng. Cameras 0 = "NONE" 1 = "SERNO" 2 = "IMAGE" • MMM Cameras “IMAGE" **Type** string	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as “<xml name>:[<element>]:[<element>]:<field>”** “<IDPH DPO>:idph:params:acquire” **Type** enum

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
returned only c) "IMAGE" - The image was acquired For MSL, there are two acquisition modes pertaining to camera data: "IMAGE" and "SERNO". The normal mode ("IMAGE") acquires and transmits the image according to all the command parameter settings. The "SERNO" case is a mode of acquisition that does just enough to get the camera's hardware serial number. Normally, such a mode should result in a data product with a header only and no image. However, the IMG onboard software supersedes during acquisition and acquires data in hardware windowed mode, resulting in a single row of an image. So, ground processing of the data product will yield a value of 1 for "hw_numrows", but the actual image is still set as 0 rows and 0 columns.	**Units** n/a **Location** IDENTIFICATION (Class)	
Ops Keyword IMAGE_BLENDING_FLAG **PDS Keyword** MSL:IMAGE_BLENDING_FLAG **Definition** Indicates whether intra-stack image blending has been performed during the focus merge operation. FALSE means images were merged without blending.	**Valid Values** 0 = "FALSE" 1 = "TRUE" **Type** string **Units** n/a **Location** MINI_HEADER (Group) ** ** Only MAHLI Z-Stack & Range Map	**Mode** TBD **Field as "<xml name>:[<element>]:[<element>]:<field>"** MAHLI Range Map, Z-Stack "MMM_Image_Mini_Header" **Type** TBD
Ops Keyword IMAGE_DATA_SIZE **PDS Keyword** same **Definition** Specifies the size of the image data product on the MMM camera DEA.	**Valid Values** n/a **Type** integer **Units** n/a **Location** TELEMETRY (Class)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** MMM Cameras "<Ancillary DPO>:image_id_data:image_size" **Type** U32
Ops Keyword ^IMAGE_HEADER **PDS Keyword** same **Definition** Specifies a pointer to the IMAGE_HEADER object. See	**Valid Values** n/a **Type** NULL **Units** n/a	**Mode** Calculation **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
chapter 14 of the PDS Standards Reference for more information on pointer usage.	**Location** POINTERS	
Ops Keyword IMAGE_ID **PDS Keyword** same **Definition** Specifies an image and typically consists of a sequence of characters representing 1) a routinely occurring measure, such as revolution number, 2) a letter identifying the spacecraft, target, or camera, and 3) a representation of a count within the measure, such as picture number within a given revolution. NOTE: See the Imaging FDD for more detailed definition. Examples: Mariner 9 - LevanthalIdentifier - (orbit, camera, pic #, total # of pics in orbit) Viking Orbiter - (orbit #, sc, pic # (FSC/16)) Viking Lander - (sc, camera, mars doy, diode (filter), pic # for that day) Voyager - (pic # for encounter, FDS for cruise)	**Valid Values** n/a **Type** string(30) **Units** n/a **Location** 1) IDENTIFICATION (Class) 2) a. IDENTIFICATION (Class) b. IDENTIFICATION (Class) c. MINI_HEADER (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as “<xml name>:[<element>]:[<element>]:<field>”** 1) Eng. Cameras “<IDPH DPO>:idph:params:imgid” 2) MMM Cameras a. “<Ancillary DPO>:cmd_arguments_image:image_ID” b. “<Ancillary DPO>:cmd_arguments_zstack:starting_ID” c. “MMM_Image_Mini_Header[**TBD**]” NOTES: • For MMM non-Z-stack data products, case “a”. • For MMM Z-stack data products, case “b”. **Type** U32
Ops Keyword IMAGE_RADIANCE_FACTOR **PDS Keyword** MSL:IMAGE_RADIANCE_FACTOR **Definition** For a multi-input image (such as a mosaic), specifies the overall multiplicative factor that was applied to each image as the mosaic was being assembled. Together with IMAGE_RADIANCE_OFFSET, this specifies a simple linear adjustment of the form “dn * factor + offset”, that can be used for (limited) radiometric seam correction. Each pixel within a given input image receives the same correction. The order of images is as defined in the input list file (generally delivered along with the mosaic) or INPUT_PRODUCT_ID. Applies to RDRs only, not EDRs. NOTE: This correction is applied after radiometric correction, if any, performed on the inputs.	**Valid Values** n/a **Type** float array **Units** n/a **Location** DERIVED_IMAGE_PARMS (Group)	**Mode** RDR-generating software **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a
Ops Keyword IMAGE_RADIANCE_OFFSET **PDS Keyword**	**Valid Values** n/a **Type** float array	**Mode** RDR-generating software **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
MSL:IMAGE_RADIANCE_OFFSET **Definition** For a multi-input image (such as a mosaic), specifies the overall additive offset that was applied to each image as the mosaic was being assembled. Together with IMAGE_RADIANCE_FACTOR, this specifies a simple linear adjustment of the form "dn * factor + offset", that can be used for (limited) radiometric seam correction. Each pixel within a given input image receives the same correction. Input values of 0 are not adjusted. The order of images is as defined in the input list file (generally delivered along with the mosaic) or INPUT_PRODUCT_ID. Applies to RDRs only, not EDRs. NOTE: This correction is applied after radiometric correction, if any, performed on the inputs.	**Units** n/a **Location** DERIVED_IMAGE_PARMS (Group)	**Type** n/a
Ops Keyword IMAGE_REGISTRATION_FLAG **PDS Keyword** MSL:IMAGE_REGISTRATION_FLAG **Definition** Indicates whether intra-stack image registration has been performed during the focus merge operation. TRUE indicates that intra-stack image registration has been performed during the focus merge operation. FALSE indicates that images have been merged without translation.	**Valid Values** 0 = "FALSE" 1 = "TRUE" **Type** string **Units** n/a **Location** MINI_HEADER (Group) ** ** Only MAHLI Z-Stack & Range Map	**Mode** TBD **Field as "<xml name>:[<element>]:[<element>]:<field>"** MAHLI Range Map, Z-Stack "MMM_Image_Mini_Header" **Type** TBD
Ops Keyword IMAGE_TYPE **PDS Keyword** same **Definition** Specifies the type of image acquired. This may be used to describe characteristics that differentiate one group of images from another such as the nature of the data in the image file, the purpose for which the image was acquired, or the way in which it was acquired. This element is very similar to the older image_observation_type element, but is designed to resolve ambiguities in cases where missions utilize a naming convention for both specific images and more general observations, which consist of multiple images. In those cases, the latter may be described by the observation_type element.	**Valid Values** • Eng. Cameras … Keyword Value — APID Names "REGULAR" — Any string value starting with "ImgImage" "THUMBNAIL" — Any string value starting with "ImgThumb" "REF_PIXELS" — Any string value starting with "ImgRef" "HISTOGRAM" — Any string value starting with "ImgHistogram" "ROW_SUM" — Any string value starting with "ImgRow" "COL_SUM" — Any string value starting with "ImgCol" • ChemCam … Keyword Value — APID Names "REGULAR" — Any string value starting with "CcamRmiImage"	**Mode** EMD in XML format **Field as "<xml name>:[<element>]:[<element>]:<field>"** "MslEarthProductMetadata:MslProductMetadata:ProductName" **Type** n/a

OUTPUT METADATA (PRODUCT LABEL)		**INPUT METADATA (SOURCE)**
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
	"THUMBNAIL" Any string value starting with "CcamRmiThumb" "REF_PIXELS" Any string value starting with "CcamRmiRef" "SPECTRA" Any string value starting with "CcamSpectra" "HEALTH" Any string value starting with "CcamSoh" • MMM Cameras … Keyword Value APID Names "REGULAR" Any string value that contains "Image" "THUMBNAIL" Any string value that contains "Thumbnail" **Type** string(15) **Units** n/a **Location** IDENTIFICATION (Class)	
Ops Keyword INPUT_PRODUCT_ID **PDS Keyword** MSL:INPUT_PRODUCT_ID **Definition** Specifies the product(s) directly used as input to create this product. It may contain either the PRODUCT_ID or the filename of the input products. For MSL, this RDR-only keyword specifies the PRODUCT_IDs of the inputs most directly used to create this product, which may themselves be RDRs. For example, a linearized XYZ image would list the left and right linearized input images as well as the disparity file. Contrast this with SOURCE_PRODUCT_ID, which lists only the root EDR source; INPUT_PRODUCT_ID lists the proximate inputs (EDR or RDR) for the process used to create this product.	**Valid Values** n/a **Type** string array **Units** n/a **Location** DERIVED_IMAGE_PARMS (Group)	**Mode** RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword INSTRUMENT_AZIMUTH **PDS Keyword** same	**Valid Values** n/a **Type** float	**Mode** Calculation: - The image center is translated into an origin and and "look direction" vectors (left and right). The vectors are then used to calculate instrument azimuth.

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
Definition Specifies the value for an instrument's rotation in the horizontal direction. It is usually measured from some kind of low hard stop. Although it may be used for any instrument where it makes sense, it is primarily intended for use in surface-based instruments that measure pointing in terms of azimuth and elevation. When in a DERIVED_GEOMETRY group, defines the azimuth (horizontal rotation) at which the instrument is pointed. This value is expressed using the cooridinate system referred to by REFERENCE_COORD_SYSTEM_NAME and REFERENCE_COORD_SYSTEM_INDEX contained within the same group. The interpretation of exactly what part of the instrument is being pointed is mission-specific. It could be the boresight, the camera head direction, the CAHV camera model A vector direction, or any of a number of other things. As such, for multimission use this value should be used mostly as an approximation, e.g. identifying scenes which might contain a given object. For MSL, the interpretation is the boresight of the camera, defined as projecting the center of the nominal image (before downsampling or subframing) through the camera model.	**Units** deg (<deg> unit tag required) **Location** • ROVER_DERIVED_GEOMETRY_PARMS (Group) • SITE_DERIVED_GEOMETRY_PARMS (Group)	**Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword INSTRUMENT_BAND_ID **PDS Keyword** same **Definition** Specifies an array of strings identifying the instrument represented by the corresponding band in the image. The first entry in the array identifies the instrument for the first band, the second entry for the second band, etc. For MSL Preload products, the string also indicates whether the value is a minimum or maximum preload. See also CONFIGURATION_BIT_ID.	**Valid Values** • Reachability RDRs ("DRILL", "DRT", "MAHLI", "APXS", "SCOOP_TIP") • Preload RDRs ("DRILL_MINIMUM", "DRILL_MAXIMUM") **Type** string array[5] **Units** n/a **Location** DERIVED_IMAGE_PARMS (Group)	**Mode** Static Value **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword INSTRUMENT_COORD_FRAME_ID **PDS Keyword** MSL:INSTRUMENT_COORD_FRAME_ID	**Valid Values** n/a (Enum mapping not listed until FSW is static) **Type** string	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** • Eng. Cameras 1) "<IDPH DPO>:idph:params:rsm_frame"

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
Definition Along with INSTRUMENT_COORD_FRAME_INDEX, specifies the name of the coordinate frame in which the pointing for this instrument is expressed. The values of INSTRUMENT_COORDINATE should be interpreted using this coordinate frame.	**Units** n/a **Location** OBSERVATION_REQUEST_PARMS (Group)	• Chemcam 2) "<IDPH DPO>:idph:params:rsm_frame" 3) "<Ancillary DPO>:cmd_arguments:frame_id" • MMM Cameras 4) "<Supplementary DPO>:frame_id" **Type** enum
Ops Keyword INSTRUMENT_COORD_FRAME_INDEX **PDS Keyword** MSL:INSTRUMENT_COORD_FRAME_INDEX **Definition** Along with INSTRUMENT_COORD_FRAME_ID, specifies the index of the coordinate frame in which the pointing for this instrument is expressed. Only the SITE, LL, RSM_JOINTS and FIDUCIAL frames are indexed. A positive value means to use specifically that index; a negative value means an index relative to the current.	**Valid Values** "N/A" **Type** integer **Units** n/a **Location** OBSERVATION_REQUEST_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** • Eng. Cameras 1) "<IDPH DPO>:idph:params:rsm_findex" • Chemcam 2) "<Ancillary DPO>:cmd_arguments:frame_index" • MMM Cameras 3) "<Supplementary DPO>:frame_index" **Type** I16
Ops Keyword INSTRUMENT_COORDINATE **PDS Keyword** same **Definition** Specifies an array of coordinate parameters. These values, typically either XYZ or azimuth/elevation values (see INSTRUMENT_COORDINATE_NAME), specify where the instrument is actually pointed in the frame defined by INSTRUMENT_COORD_FRAME_ID/INDEX.	**Valid Values** Also, 0 = "N/A" **Type** double array[3] **Units** • radians for following Frames (<rad> unit tag required): "AZIMUTH", "ELEVATION", "AZIMUTH_RELATIVE", "ELEVATION_RELATIVE", "RSM_AZIMUTH", "RSM_ELEVATION", "RSM_AZIMUTH_RELATIVE", "RSM_ELEVATION_RELATIVE" • meters for following Frames (<m> unit tag required): "X", "Y", "Z" **Location** OBSERVATION_REQUEST_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** • Eng. Cameras 1) "<IDPH DPO>:idph:params:rsm_coord[3]" • Chemcam 2) "<Ancillary DPO>:cmd_arguments:coord1", "<Ancillary DPO>:cmd_arguments:coord2", "<Ancillary DPO>:cmd_arguments:coord3" • MMM Cameras 3) "<Supplementary DPO>:coord1", "<Supplementary DPO>:coord2", "<Supplementary DPO>:coord3" **Type** 1) F64[3] 2) U32 3) F32
Ops Keyword INSTRUMENT_COORDINATE_NAME **PDS Keyword** same **Definition**	**Valid Values** • Eng. Cameras 1) 0 = "N/A" 1 = ("X", "Y", "Z") 2 = ("AZIMUTH", "ELEVATION") 3 = ("AZIMUTH_RELATIVE", "ELEVATION_RELATIVE")	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** • Eng. Cameras 1) "<IDPH DPO>:idph:params:rsm_ctype"

<table>
<tr><th colspan="2">OUTPUT METADATA (PRODUCT LABEL)</th><th>INPUT METADATA (SOURCE)</th></tr>
<tr><th>• Ops Keyword
• PDS-Compliant Keyword
• Definition</th><th>• Valid Values (quoted)
• Type
• Units
• Keyword Location in Label</th><th>• Mode
• Metadata Field
• Type</th></tr>
<tr><td>Specifies the name of each element in INSTRUMENT_COORDINATE, indicating how to interpret the value. The "X","Y","Z" values are in meters; all others are in radians. The values should appear in sets as specified in Valid Values.</td><td>4 = (“RSM_AZIMUTH”, “RSM_ELEVATION”)
5 = (“RSM_AZIMUTH_RELATIVE”, “RSM_ELEVATION_RELATIVE”)

• <u>Chemcam</u>
2) 0 = “N/A”
1 = (“X”, “Y”, “Z”)
2 = (“AZIMUTH”, “ELEVATION”)
3 = (“AZIMUTH_RELATIVE”, “ELEVATION_RELATIVE”)

<u>Type</u>
string

<u>Units</u>
n/a

<u>Location</u>
OBSERVATION_REQUEST_PARMS (Group)</td><td>• <u>Chemcam</u>
2) “<Ancillary DPO>:cmd_arguments:coord_type”

<u>Type</u>
enum</td></tr>
<tr><td><u>Ops Keyword</u>
INSTRUMENT_COORDINATE_TYPE

<u>PDS Keyword</u>
MSL:INSTRUMENT_COORDINATE_TYPE

<u>Definition</u>
Specifies what kind of coordinates are in the associated INSTRUMENT_COORDINATE keyword. These could be XYZ coordinates, azimuth/elevation values, etc.</td><td><u>Valid Values</u>
“AZEL_ABS”, “AZEL_REL”, “JOINTS_ AZEL_ABS”, “NO_MOTION”, “XYZ”

<u>Type</u>
string

<u>Units</u>
n/a

<u>Location</u>
OBSERVATION_REQUEST_PARMS (Group)</td><td><u>Mode</u>
DPO in XML format (referenced to APID Name in Appendix E)

<u>Field as “<xml name>:[<element>]:[<element>]:<field>”</u>
• <u>Eng. Cameras and MMM Cameras</u>
“<IDPH DPO>:idph:params:rsm_ctype”
• <u>Chemcam</u>
“<Ancillary DPO>:cmd_arguments:coord_type”

<u>Type</u>
enum</td></tr>
<tr><td><u>Ops Keyword</u>
INSTRUMENT_ELEVATION

<u>PDS Keyword</u>
same

<u>Definition</u>
Specifies a value for an instrument's rotation in the vertical direction. It is usually measured from some kind of low hard stop. Although it may be used for any instrument where it makes sense, it is primarily intended for use in surface-based instruments that measure pointing in terms of azimuth and elevation.

When in a DERIVED_GEOMETRY group, defines the elevation (vertical rotation) at which the instrument is pointed. This value is expressed using the cooridinate system referred to by REFERENCE_COORD_SYSTEM_NAME and REFERENCE_COORD_SYSTEM_INDEX contained within the same group.</td><td><u>Valid Values</u>
“-91.0” to “91.0”

<u>Type</u>
float

<u>Units</u>
deg (<deg> unit tag required)

<u>Location</u>
• ROVER_DERIVED_GEOMETRY_PARMS (Group)
• SITE_DERIVED_GEOMETRY_PARMS (Group)</td><td><u>Mode</u>
Calculation:
- The image center is translated into an origin and and “look direction” vectors (left and right). The vectors are then used to calculate instrument elevation.

<u>Field as “<xml name>:[<element>]:[<element>]:<field>”</u>
n/a

<u>Type</u>
n/a</td></tr>
</table>

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
The interpretation of exactly what part of the instrument is being pointed is mission-specific. It could be the boresight, the camera head direction, the CAHV camera model A vector direction, or any of a number of other things. As such, for multimission use this value should be used mostly as an approximation, e.g. identifying scenes which might contain a given object. For MSL, the interpretation is the boresight of the camera, defined as projecting the center of the nominal image (before downsampling or subframing) through the camera model.		
Ops Keyword INSTRUMENT_FOCUS_DISTANCE **PDS Keyword** MSL:INSTRUMENT_FOCUS_DISTANCE **Definition** Specifies the distance to the target to use for focusing the instrument. See also INSTRUMENT_FOCUS_MODE. For MSL ChemCam, specifies the distance to target for "MANUAL" focus, or the seed for "BASELINE" focus.	**Valid Values** • ChemCam, Mastcam, MAHLI "0" to "65535" • MARDI "UNK" **Type** integer **Units** millimeters (<mm> unit tag required) **Location** OBSERVATION_REQUEST_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** • ChemCam RMI "<Ancillary DPO>:cmd_arguments:range" • ChemCam LIBS "<Ancillary DPO>:cmd_arguments:range" • MMM Cameras a. "<Supplementary DPO>:left_focus_initial_position" b. "<Supplementary DPO>:right_focus_initial_position" c. "<Supplementary DPO>:focus_initial_position" NOTES: • For MMM Cameras, Mastcam left is case "a". • For MMM Cameras, Mastcam right is case "b". • For MMM Cameras, MAHLI is case "c". • For MMM Cameras, MARDI is "UNK". **Type** I16
Ops Keyword INSTRUMENT_FOCUS_INIT_FLAG **PDS Keyword** MSL:INSTRUMENT_FOCUS_INIT_FLAG **Definition** Specifies whether the instrument focus mechanism should be (or was) initialized before use. For MSL ChemCam, this pertains to RMI observations only (not LIBS). Value "TRUE" brings the focus stage to the limit switch to initialize its position.	**Valid Values** 0 = "FALSE" 1 = "TRUE" **Type** string **Units** n/a **Location** OBSERVATION_REQUEST_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** "<Ancillary DPO>:cmd_parameters:fromLimitSwitch" **Type** U8
Ops Keyword INSTRUMENT_FOCUS_MODE	**Valid Values** ChemCam	**Mode** DPO in XML format (referenced to APID Name in Appendix E)

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
PDS Keyword MSL:INSTRUMENT_FOCUS_MODE **Definition** Specifies how an instrument was focused. The values and their interpretation are mission-specific. For MSL ChemCam, it is the commanded mode of focus during both RMI and LIBS observations. Valid values for ChemCam are defined as: a) "NO_FOCUS" - Doesn't move focus (others mark CCAM unsafe). b) "BASELINE" - Uses CWL to find optimal focus position. c) "MANUAL" - Positions focus based on keyword INSTRUMENT_FOCUS_PARAM value. d) "AF_OFFSET" - Applies RMI offset from last autofocus solution.	"NO_FOCUS", "BASELINE", "MANUAL", "AF_OFFSET" **Type** string **Units** n/a **Location** 1) OBSERVATION_REQUEST_PARMS (Group) 2) MINI_HEADER (Group) ** ** Except MAHLI Z-Stack & Range Map	**Field as "<xml name>:[<element>]:[<element>]:<field>"** 1) Chemcam "<Ancillary DPO>:cmd_arguments:focus" 2) MMM Cameras "MMM_Image_Mini_Header[**TBD**|" **Type** U8
Ops Keyword INSTRUMENT_FOCUS_POSITION **PDS Keyword** MSL:INSTRUMENT_FOCUS_POSITION **Definition** Specifies the position in motor counts of the focus motor on a camera. For MSL Chemcam and MMM, it specifies the actual focus position that was used by the image.	**Valid Values** • MMM Cameras … Mode — Value Z-stack — n/a Non-Z-stack — Comprised of four bytes coming from Image DPO mini-header at offsets 44, 45, 46 and 47 **Type** integer **Units** n/a **Location** INSTRUMENT_STATE_PARMS (Group) MINI_HEADER (Group) ** ** Except MAHLI Z-Stack & Range Map	**Mode** Image DPO mini-header **Field as "<xml name>:[<element>]:[<element>]:<field>"** MMM Cameras "MMM_Image_Mini_Header[44|", "MMM_Image_Mini_Header[45|", "MMM_Image_Mini_Header[46|", "MMM_Image_Mini_Header[47|" **Type** U32
Ops Keyword INSTRUMENT_FOCUS_STEP_SIZE **PDS Keyword** MSL:INSTRUMENT_FOCUS_STEP_SIZE **Definition** For MMM cameras on MSL, specifies the size in motor counts of each (or the initial) step taken by the focus adjustment mechanism in an autofocus algorithm.	**Valid Values** n/a **Type** integer **Units** n/a **Location** a. OBSERVATION_REQUEST_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** MMM Cameras a. "<IDPH DPO>:idph:step_size" b. "MMM_Image_Mini_Header[**TBD**|" **Type** U16

<table>
<tr><th colspan="2">OUTPUT METADATA (PRODUCT LABEL)</th><th>INPUT METADATA (SOURCE)</th></tr>
<tr><th>• Ops Keyword
• PDS-Compliant Keyword
• Definition</th><th>• Valid Values (quoted)
• Type
• Units
• Keyword Location in Label</th><th>• Mode
• Metadata Field
• Type</th></tr>
<tr><td></td><td>b. MINI_HEADER (Group) **
** Except MAHLI Z-Stack & Range Map</td><td></td></tr>
<tr><td>Ops Keyword
INSTRUMENT_FOCUS_STEPS

PDS Keyword
MSL:INSTRUMENT_FOCUS_STEPS

Definition
For MMM cameras on MSL, specifies the number of steps (images) to be taken by an autofocus algorithm.</td><td>Valid Values
n/a

Type
integer

Units
n/a

Location
OBSERVATION_REQUEST_PARMS (Group)</td><td>Mode
DPO in XML format (referenced to APID Name in Appendix E)

Field as “<xml name>:[<element>]:[<element>]:<field>”
MMM Cameras
“<IDPH DPO>:idph:n_steps”

Type
U8</td></tr>
<tr><td>Ops Keyword
INSTRUMENT_HOST_ID

PDS Keyword
same

Definition
Specifies a unique identifier for the host where an instrument is located. This host can be either a spacecraft or an earth base (e.g., and observatory or laboratory on the earth). Thus, INSTRUMENT_HOST_ID can contain values which are either SPACECRAFT_ID values or EARTH_BASE_ID values.</td><td>Valid Values
• Eng. Cameras …
SCID Value
158 “SIM”
76 “MSL”
• MMM Cameras …
Value
“MSL”

Type
string array

Units
n/a

Location
IDENTIFICATION (Class)</td><td>Mode
EMD in XML format

Field as “<xml name>:[<element>]:[<element>]:<field>”
“MslEarthProductMetadata:MslProductMetadata:Scid”

Type
n/a</td></tr>
<tr><td>Ops Keyword
INSTRUMENT_HOST_NAME

PDS Keyword
same

Definition
Specifies the full name of the host on which an instrument is based. This host can be either a spacecraft or an earth base. Thus, the INSTRUMENT_HOST_NAME element can contain values which are either SPACECRAFT_NAME values or EARTH_BASE_NAME values.

Note that mosaics may contain more than one value in an array.</td><td>Valid Values
• Eng. Cameras …
SCID Value
158 “SIMULATED MARS SCIENCE LABORATORY”
76 “MARS SCIENCE LABORATORY”
• MMM Cameras …
Value
“MARS SCIENCE LABORATORY”

Type
string array

Units
n/a

Location
IDENTIFICATION (Class)</td><td>Mode
EMD in XML format

Field as “<xml name>:[<element>]:[<element>]:<field>”
“MslEarthProductMetadata:MslProductMetadata:Scid”

Type
n/a</td></tr>
</table>

<table>
<tr><th colspan="2">OUTPUT METADATA (PRODUCT LABEL)</th><th>INPUT METADATA (SOURCE)</th></tr>
<tr><th>• Ops Keyword
• PDS-Compliant Keyword
• Definition</th><th>• Valid Values (quoted)
• Type
• Units
• Keyword Location in Label</th><th>• Mode
• Metadata Field
• Type</th></tr>
<tr>
<td><u>Ops Keyword</u>
INSTRUMENT_ID

<u>PDS Keyword</u>
same

<u>Definition</u>
Specifies an abbreviated name or acronym which identifies an instrument.

NOTE: INSTRUMENT_ID is not a unique identifier for a given instrument. Note also that the associated INSTRUMENT_NAME element provides the full name of the instrument.

Example values: IRTM (for Viking Infrared Thermal Mapper), PWS (for plasma wave spectrometer).</td>
<td><u>Valid Values</u>
<table>
<tr><th><u>Value (some denote String A or B)</u></th><th><u>APID Names</u></th></tr>
<tr><td>“FHAZ_LEFT_<A|B>”</td><td>• Any string value starting with “Img” and ending with “Fhl”.</td></tr>
<tr><td>“FHAZ_RIGHT_<A|B>”</td><td>• Any string value starting with “Img” and ending with “Fhr”.</td></tr>
<tr><td>“RHAZ_LEFT_<A|B>”</td><td>• Any string value starting with “Img” and ending with “Rhl”.</td></tr>
<tr><td>“RHAZ_RIGHT_<A|B>”</td><td>• Any string value starting with “Img” and ending with “Rhr”.</td></tr>
<tr><td>“NAV_LEFT_<A|B>”</td><td>• Any string value starting with “Img” and ending with “Nl”.</td></tr>
<tr><td>“NAV_RIGHT_<A|B>”</td><td>• Any string value starting with “Img” and ending with “Nr”.</td></tr>
<tr><td>“CHEMCAM_RMI”</td><td>• Any string value starting with “CcamRmi”.</td></tr>
<tr><td>“CHEMCAM_LIBS”</td><td>• Any string value starting with “CcamSpectra”.</td></tr>
<tr><td>“CHEMCAM_SOH”</td><td>• Any string value starting with “CcamSoh”.</td></tr>
<tr><td>“MAST_LEFT”</td><td>• Any string value starting with “McamL”.</td></tr>
<tr><td>“MAST_RIGHT”</td><td>• Any string value starting with “McamR”.</td></tr>
<tr><td>“MAHLI”</td><td>• Any string value starting with “Mhli”.</td></tr>
<tr><td>“MARDI”</td><td>• Any string value starting with “Mrdi”.</td></tr>
</table>

<u>Type</u>
string(12)

<u>Units</u>
n/a

<u>Location</u>
IDENTIFICATION (Class)</td>
<td><u>Mode</u>
EMD in XML format

<u>Field as “<xml name>:[<element>]:[<element>]:<field>”</u>
“MslEarthProductMetadata:MslProductMetadata:ProductName”,
“MslEarthProductMetadata:MslProductMetadata:CreationStringId”

<u>Type</u>
n/a</td>
</tr>
</table>

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
Ops Keyword INSTRUMENT_IDLE_TIMEOUT **PDS Keyword** same **Definition** Specifies the amount of time in seconds the instrument may be idle before powering off the instrument.	**Valid Values** • Eng. Cameras "0" to "4294967295" • MMM Cameras "N/A" **Type** integer **Units** seconds (<s> unit tag required) **Location** OBSERVATION_REQUEST_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** "<IDPH DPO>:idph:params:power_timeout" **Type** U32
Ops Keyword INSTRUMENT_MODE_ID **PDS Keyword** same **Definition** Specifies an instrument-dependent designation of operating mode. This may be simply a number, letter or code, or a word such as 'normal', 'full resolution', 'near encounter', or 'fixed grating'. For ChemCam on MSL … a) for RMI instrument, specifies the link between the ChemCam Mast Unit (CCMU) and ChemCam Body Unit (CCBU) to use during RMI image commanding. Value of "1" denotes async (currently unavailable). b) for LIBS instrument, specifies the commanded mode of observation. If value is "ABLATION_ONLY", LIBS only fires the laser in ablation mode, with no spectra collected. If value is "SPECTRAL_DATA", LIBS fires the laser and captures spectral data. Argument for command CCAM_ACTV_SPECTRAL_OBS (see ChemCam FDD). For MMM cameras on MSL, this specifies the image or video readout mode, which determines the rate the image is read out.	**Valid Values** 1) Eng. Cameras "FULL_FRAME", "WINDOWED_FRAME", "4X1SUMMATION_FRAME", "FIXED_PATTERN_FRAME" 2) ChemCam RMI "UNK", "ASYNC" 3) ChemCam LIBS "ABLATION_ONLY", "SPECTRAL_DATA" **Type** string(20) **Units** n/a **Location** 1) INSTRUMENT_STATE_PARMS (Group) 2) OBSERVATION_REQUEST_PARMS (Group) 3) OBSERVATION_REQUEST_PARMS (Group) 4) MINI_HEADER (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** 1) Eng. Cameras "<IDPH DPO>:idph:params:resolution", "<IDPH DPO>:idph:params:subframe", "<IDPH DPO>:idph:params:exposure" NOTES: - If resolution=2 or 3 (contains HW), then "4X1SUMMATION_FRAME". - If subframe > 0 (TRUE), then "WINDOWED_FRAME". - If exposure=3 (IMG_EXPOSURE_TEST), then "FIXED_PATTERN_FRAME". - All else, "FULL_FRAME". 2) ChemCam RMI "<Ancillary DPO>:cmd_parameters:linkToUse" NOTES: - If linkToUse = 0, then "UNK" - If linkToUse = 1, then "ASYNC" 3) ChemCam LIBS "<Ancillary DPO>:cmd_arguments:data" NOTES: - If data = no, then "ABLATION_ONLY" - If data = yes, then "SPECTRAL_DATA" 4) MMM Cameras "MMM_Image_Mini_Header[TBD]" **Type** 1) enum 2) U8 3) I32
Ops Keyword	**Valid Values**	**Mode**

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
INSTRUMENT_NAME **PDS Keyword** same **Definition** Specifies the full name of an instrument. Note that the associated INSTRUMENT_ID element provides an abbreviated name or acronym for the instrument. Example values: FLUXGATE MAGNETOMETER, NEAR_INFRARED MAPPING SPECTROMETER.	Value (some denote String A or B) / APID Names "FRONT HAZARD AVOIDANCE CAMERA LEFT STRING <A|B>" • Any string value starting with "Img" and ending with "Fhl" "FRONT HAZARD AVOIDANCE CAMERA RIGHT STRING <A|B>" • Any string value starting with "Img" and ending with "Fhr" "REAR HAZARD AVOIDANCE CAMERA LEFT STRING <A|B>" • Any string value starting with "Img" and ending with "Rhl" "REAR HAZARD AVOIDANCE CAMERA RIGHT STRING <A|B>" • Any string value starting with "Img" and ending with "Rhr" "NAVIGATION CAMERA LEFT STRING <A|B>" • Any string value starting with "Img" and ending with "Nl" "NAVIGATION CAMERA RIGHT STRING <A|B>" • Any string value starting with "Img" and ending with "Nr" "CHEMISTRY CAMERA REMOTE MICRO-IMAGER" • Any string value starting with "CcamRmi" "CHEMISTRY CAMERA LASER INDUCED BREAKDOWN SPECTROMETER" • Any string value starting with "CcamSpectra" "CHEMISTRY CAMERA STATE OF HEALTH" • Any string value starting with "CcamSoh" "MASTCAM IMAGER" • Any string value starting with "Mcam" "MAHLI IMAGER" • Any string value starting with "Mhli" "MARDI IMAGER" • Any string value starting with "Mrdi" **Type** string array **Units** n/a **Location** IDENTIFICATION (Class)	EMD in XML format **Field as "<xml name>:[<element>]:[<element>]:<field>"** "MslEarthProductMetadata:MslProductMetadata:ProductName", "MslEarthProductMetadata:MslProductMetadata:CreationStringId" **Type** n/a
Ops Keyword INSTRUMENT_SERIAL_NUMBER	**Valid Values** • Eng. Cameras … EM Strings FM Strings	**Mode** 1) DPO in XML format (referenced to APID Name in Appendix E) 2) DPO in XML format (referenced to APID Name in Appendix E)

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
PDS Keyword same **Definition** Specifies the manufacturer's serial number assigned to an instrument. This number may be used to uniquely identify a particular instrument for tracing its components or determining its calibration history, for example.	Instrument — A — B — A — B Front Left Hazcam …… "057" "056" "205" "208" Front Right Hazcam … "058" "055" "213" "209" Rear Left Hazcam …… "204" "202" "211" "212" Rear Right Hazcam …. "028" "027" "217" "207" Left Navcam ……..….. "053" "054" "216" "215" Right Navcam ……..…. "051" "052" "206" "218" • ChemCam … Instrument — FM RMI ……….................. "0001" LIBS ………………….. "001" • MMM … Instrument — EM — FM MastCam Left …… "1001" "3003" MastCam Right … "1005" "3004" MARDI …………… "1002" "3001" MAHLI ……………. "1003" "3002" **Type** string **Units** n/a **Location** 1) IDENTIFICATION (Class) 2) IDENTIFICATION (Class) MINI_HEADER (Group)	or Image DPO mini-header **Field as "<xml name>:[<element>]:[<element>]:<field>"** • Eng. Cameras & ChemCam 1) "<IDPH DPO>:idph:serial_no" • MMM Cameras 2) "MMM_Image_Mini_Header[41]", "MMM_Image_Mini_Header[42]", "MMM_Image_Mini_Header[43]" NOTES: • For MMM Cameras, value analogous to "serial_no" is comprised of three bytes coming from Image DPO mini-header at byte offsets 41, 42 and 43. **Type** U8
Ops Keyword INSTRUMENT_STATE **PDS Keyword** n/a **Definition** Specifies an array of identifiers for the state of an instrument at a specified time.	**Valid Values** • Bit 0: **TBD** … 0 = "**TBD**" 1 = "**TBD**" • Bit 1: **TBD** … 0 = "**TBD**" 1 = "**TBD**" • Bit 2: **TBD** … 0 = "**TBD**" 1 = "**TBD**" • Bit 3: **TBD** … 0 = "**TBD**" 1 = "**TBD**" • Bit 4: **TBD** … 0 = "**TBD**" 1 = "**TBD**" • Bit 5: **TBD** … 0 = "**TBD**"	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** MMM Cameras "MMM_Image_Mini_Header[**TBD**]" **Type** U8

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
	1 = "**TBD**" • Bit 6: **TBD** … 0 = "**TBD**" 1 = "**TBD**" • Bit 7: **TBD** … 0 = "**TBD**" 1 = "**TBD**" **Type** string array[8] **Units** n/a **Location** MINI_HEADER (Group)	
Ops Keyword INSTRUMENT_STATE_NAME **PDS Keyword** n/a **Definition** Specifies the possible value that can be contained in the INSTRUMENT_STATE array.	**Valid Values** ("**TBD**", "**TBD**", "**TBD**", "**TBD**", "**TBD**", "**TBD**", "**TBD**", "**TBD**") **Type** string array[8] **Units** n/a **Location** MINI_HEADER (Group)	**Mode** Static value: - Single value representing array of Bits 0 thru 7 from INSTRUMENT_STATE. **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword INSTRUMENT_TEMPERATURE **PDS Keyword** same **Definition** Specifies the temperature, in degrees Celsius, of an instrument or some part of an instrument. Note that this may be an array of multiple values for temperatures on different parts of the instrument. (Example: CCD array and sensor head)	**Valid Values** n/a **Type** float array **Units** deg C (<degC> unit tag required) **Location** INSTRUMENT_STATE_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** • Eng. Cameras 1) "<IDPH DPO>:idph:temp[24]" • Chemcam 2) "<Ancillary DPO>:instrument_temperatures: mu_obox_telescope_temp", "<Ancillary DPO>:instrument_temperatures: mu_laser_if_temp", "<Ancillary DPO>:instrument_temperatures: mu_ebox_heatsink_temp", "<Ancillary DPO>:instrument_temperatures: mu_ebox_fpga_temp", "<Ancillary DPO>:instrument_temperatures: bu_ccd_vnir_b_temp", "<Ancillary DPO>:instrument_temperatures:bu_spec_b_temp", "<Ancillary DPO>:instrument_temperatures: bu_ccd_uv_a_temp", "<Ancillary DPO>:instrument_temperatures:bu_spec_a_temp", "<Ancillary DPO>:instrument_temperatures:

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
		bu_demux_a_temp", "<Ancillary DPO>:instrument_temperatures: bu_demux_b_temp" • MMM Cameras 3) "<Ancillary DPO>:instrument_temperatures:dea_temp", "<Ancillary DPO>:instrument_temperatures:head_fpa_temp", "<Ancillary DPO>:instrument_temperatures:head_htr_temp_1", "<Ancillary DPO>:instrument_temperatures:head_htr_temp_2" **Type** 1) F32[24] 2) F32 3) F32
Ops Keyword INSTRUMENT_TEMPERATURE_NAME **PDS Keyword** same **Definition** Specifies an array of the formal names identifying each of the values used in INSTRUMENT_TEMPERATURE.	**Valid Values** • Eng. Cameras ("A_FRONT_LEFT_HAZ_CCD", "A_FRONT_LEFT_HAZ_ELECTRONICS", "A_FRONT_RIGHT_HAZ_CCD", "A_FRONT_RIGHT_HAZ_ELECTRONICS", "A_REAR_LEFT_HAZ_CCD", "A_REAR_LEFT_HAZ_ELECTRONICS", "A_REAR_RIGHT_HAZ_CCD", "A_REAR_RIGHT_HAZ_ELECTRONICS", "A_LEFT_NAV_CCD", "A_LEFT_NAV_ELECTRONICS", "A_RIGHT_NAV_CCD", "A_RIGHT_NAV_ELECTRONICS", "B_FRONT_LEFT_HAZ_CCD", "B_FRONT_LEFT_HAZ_ELECTRONICS", "B_FRONT_RIGHT_HAZ_CCD", "B_FRONT_RIGHT_HAZ_ELECTRONICS", "B_REAR_LEFT_HAZ_CCD", "B_REAR_LEFT_HAZ_ELECTRONICS", "B_REAR_RIGHT_HAZ_CCD", "B_REAR_RIGHT_HAZ_ELECTRONICS", "B_LEFT_NAV_CCD", "B_LEFT_NAV_ELECTRONICS", "B_RIGHT_NAV_CCD", "B_RIGHT_NAV_ELECTRONICS") • Chemcam ("MU_OBOX_TELESCOPE", "MU_LASER_IF", "MU_EBOX_HEATSINK", "MU_EBOX_FPGA", "BU_CCD_VNIR_B", "BU_SPEC_B", "BU_CCD_UV_A", "BU_SPEC_A", "BU_DEMUX_A", "BU_DEMUX_B") • MMM Cameras ("DEA", "HEAD_FPA","HEAD_HTR_1", "HEAD_HTR_2") **Type** string array[24] **Units** n/a	**Mode** Static values **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
	Location INSTRUMENT_STATE_PARMS (Group)	
Ops Keyword INSTRUMENT_TEMPERATURE_STATUS **PDS Keyword** MSL:INSTRUMENT_TEMPERATURE_STATUS **Definition** For ChemCam on MSL, specifies the status of temperature read from Thermal module. Note that INSTRUMENT_TEMPERATURE_NAME specifies the name associated with each value for this keyword.	**Valid Values** n/a **Type** integer array **Units** n/a **Location** INSTRUMENT_STATE_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** • Chemcam 1) "<Ancillary DPO>:instrument_temperatures: mu_obox_telescope_temp_status", "<Ancillary DPO>:instrument_temperatures: mu_laser_if_temp_status", "<Ancillary DPO>:instrument_temperatures: mu_ebox_heatsink_temp_status", "<Ancillary DPO>:instrument_temperatures: mu_ebox_fpga_temp_status", "<Ancillary DPO>:instrument_temperatures: bu_ccd_vnir_b_temp_status", "<Ancillary DPO>:instrument_temperatures: bu_spec_b_temp_status", "<Ancillary DPO>:instrument_temperatures: bu_ccd_uv_a_temp_status", "<Ancillary DPO>:instrument_temperatures: bu_spec_a_temp_status", "<Ancillary DPO>:instrument_temperatures: bu_demux_a_temp_status", "<Ancillary DPO>:instrument_temperatures: bu_demux_b_temp_status" • MMM Cameras 2) "<Ancillary DPO>:instrument_temperatures:dea_temp", "<Ancillary DPO>:instrument_temperatures:head_fpa_temp", "<Ancillary DPO>:instrument_temperatures:head_htr_temp_1", "<Ancillary DPO>:instrument_temperatures:head_htr_temp_2" **Type** 1) I32 2) F32
Ops Keyword INSTRUMENT_TYPE **PDS Keyword** same **Definition** Specifies the type of an instrument. Example values: POLARIMETER, RADIOMETER, REFLECTANCE SPECTROMETER, VIDICON CAMERA.	**Valid Values** • Eng. Cameras … Value — APID Names "IMAGING CAMERA" — • For Eng Cameras, any string value starting with "Img". • For ChemCam, any string value starting with "CcamRmi". "SPECTROMETER" — • For ChemCam, any string value starting with "CcamSpectra". "N/A" — • For ChemCam, any string	**Mode** EMD in XML format **Field as "<xml name>:[<element>]:[<element>]:<field>"** "MslEarthProductMetadata:MslProductMetadata:ProductName" **Type** n/a

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
Note that mosaics may contain more than one value in an array.	value NOT starting with "CcamRmi" or "CcamSpectra". • MMM Cameras … Value "IMAGING CAMERA" **Type** string(15) **Units** n/a **Location** IDENTIFICATION (Class)	
Ops Keyword INSTRUMENT_VERSION_ID **PDS Keyword** same **Definition** Specifies the model of an instrument used to obtain data. For example, this keyword could be used to distinguish between an engineering model of a camera used to acquire test data, and a flight model of a camera used to acquire science data during a mission. Identifiers for use in MSL are: a) "BB" - Breadboard b) "EM" - Engineering Model c) "FM" - Flight Model	**Valid Values** • Eng. Cameras & ChemCam … Serial Number Range — Value 001 - 009 — "BB" 010 - 034 — "EM" 035 - 050 — n/a 051 - 058 — "EM" 059 - 099 — n/a 100 - 125 — "FM" 126 - 199 — n/a 200 - 218 — "FM" Serial Number > 218 — n/a • MMM Cameras … Serial Number Range — Value Serial Number < 3000 — "EM" Serial Number >= 3000 — "FM" **Type** string(8) **Units** n/a **Location** IDENTIFICATION (Class)	**Mode** 1) DPO in XML format (referenced to APID Name in Appendix E) 2) DPO in XML format (referenced to APID Name in Appendix E) or Image DPO mini-header **Field as "<xml name>:[<element>]:[<element>]:<field>"** • Eng. Cameras & ChemCam 1) "<IDPH DPO>:idph:serial_no" • MMM Cameras 2) "MMM_Image_Mini_Header[41]", "MMM_Image_Mini_Header[42]", "MMM_Image_Mini_Header[43]" NOTES: • For MMM Cameras, value analogous to "serial_no" is comprised of three bytes coming from Image DPO mini-header at byte offsets 41, 42 and 43. **Type** U8
Ops Keyword INST_CMPRS_COLOR_MODE **PDS Keyword** none **Definition** Specifies the color mode used during compression.	**Valid Values** 0 or 1 = "N/A" 2 = "COLOR_MODE_GRAY" 3 = "COLOR_MODE_422" 4 = "COLOR_MODE_444" **Type** string	**Mode** Image DPO mini-header, Table Lookup **Field as "<xml name>:[<element>]:[<element>]:<field>"** MMM Cameras "MMM_Image_Mini_Header[32]", "MMM_Image_Mini_Header[33]", "MMM_Image_Mini_Header[34]",

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
For MSL, this is the JPEG color subsampling mode. COLOR_MODE_GRAY indicates a grayscale image, COLOR_MODE_422 indicates 4:2:2 chroma subsampling (the typical case), and COLOR_MODE_444 indicates 4:4:4 chroma sampling (no subsampling). The keyword is not applicable if the INST_CMPRS_NAME is not JPEG and may be omitted.	**Units** n/a **Location** COMPRESSION_PARMS (Group)	"MMM_Image_Mini_Header[35]", "MMM_Image_Mini_Header[36]", "MMM_Image_Mini_Header[37]", "MMM_Image_Mini_Header[38]", "MMM_Image_Mini_Header[39]" NOTES: • For MMM Cameras, value is comprised of eight bytes coming from Image DPO mini-header at byte offsets 32, 33, 34, 35, 36, 37, 38 and 39. **Type** enum
Ops Keyword INST_CMPRS_DEFERRED_FLAG **PDS Keyword** none **Definition** Specifies a flag to indicate whether compression was done at the time of image acquisition, or was deferred until later (typically at downlink time). For MSL, deferred compression is commonly used with MMM to downlink differently-compressed versions of the same image; the image is stored onboard in raw form. Deferred compression is specified but not currently implemented for the engineering cameras.	**Valid Values** 0 = "FALSE" non-0 = "TRUE" **Type** string **Units** n/a **Location** 1) COMPRESSION_PARMS (Group) 2) a. IMAGE_REQUEST_PARMS (Group) b. REFERENCE_PIXEL_REQUEST_PARMS (Group) c. THUMBNAIL_REQUEST_PARMS (Group) 3) a. IMAGE_REQUEST_PARMS (Group) b. THUMBNAIL_REQUEST_PARMS (Group) 4) COMPRESSION_PARMS (Group)	**Mode** 1) EMD in XML format, calculation 2) DPO in XML format (referenced to APID Name in Appendix E), Table Lookup 3) DPO in XML format (referenced to APID Name in Appendix E), Table Lookup 4) DPO in XML format (referenced to APID Name in Appendix E) or Image DPO mini-header, Table Lookup **Field as "<xml name>:[<element>]:[<element>]:<field>"** • All Cameras 1) "MslEarthProductMetadata:MslProductMetadata: ProductName" • Eng. Cameras 2) a. "<IDPH DPO>:idph:params:comp:compress" b. "<IDPH DPO>:idph:params:ref_comp:compress" c. "<IDPH DPO>:idph:params:thumb_comp:compress" • Chemcam RMI 3) a. "<Ancillary DPO>:cmd_arguments:compression" b. "<Ancillary DPO>:cmd_arguments: THUMBNAIL_COMPRESSION" • MMM Cameras 4) "MMM_Image_Mini_Header[32]", "MMM_Image_Mini_Header[33]", "MMM_Image_Mini_Header[34]", "MMM_Image_Mini_Header[35]", "MMM_Image_Mini_Header[36]", "MMM_Image_Mini_Header[37]", "MMM_Image_Mini_Header[38]", "MMM_Image_Mini_Header[39]" NOTES: • For MMM Cameras, value analogous to "compress" is comprised of eight bytes coming from Image DPO mini-header at byte offsets 32, 33, 34, 35, 36, 37, 38 and 39.

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
		Type enum
Ops Keyword INST_CMPRS_FILTER **PDS Keyword** same **Definition** Specifies the wavelet filter used in the ICER or LOCO compression and decompression algorithm.	**Valid Values** • Eng. Cameras "A" = 0 = WAVELET_A "B" = 1 = WAVELET_B "C" = 2 = WAVELET_C "D" = 3 = WAVELET_D "E" = 4 = WAVELET_E "F" = 5 = WAVELET_F "Q" = 6 = WAVELET_Q • MMM Cameras lossless = "N/A" not lossless = "UNK" **Type** string **Units** n/a **Location** a. COMPRESSION_PARMS (Group) b. IMAGE_REQUEST_PARMS (Group) c. REFERENCE_PIXEL_REQUEST_PARMS (Group) d. THUMBNAIL_REQUEST_PARMS (Group)	**Mode** 1) Compression algorithm, calculation a. Value returned by compression as variable 2) DPO in XML format (referenced to APID Name in Appendix E), Table Lookup **Field as "<xml name>:[<element>]:[<element>]:<field>"** 2) b. "<IDPH DPO>:idph:params:comp:wfilter" c. "<IDPH DPO>:idph:params:ref_comp:wfilter" d. "<IDPH DPO>:idph:params:thumb_comp:wfilter" **Type** a. n/a b. enum c. enum d. enum
Ops Keyword INST_CMPRS_MODE **PDS Keyword** same **Definition** Specifies the method used for on-board compression of data. For the MSL mission, a value of 0 indicates some form of lossless (or no) compression, while non-0 values indicate modes the lossy compressors may use. See INST_CMPRS_NAME for the actual compression type used. See also INST_CMPRS_COLOR_MODE for JPEG.	**Valid Values** 1) b. "0" = None or Lossless "1" = Lossy 2) "0" = None "0" = 3 = Lossless (LOCO) "0" = 4 = Lossless (LOCO_DEFERRED) "1" = 1 = Lossy (ICER) "1" = 2 = Lossy (ICER_DEFERRED) 3) "0" = None "0" = 1 = Lossless (LOCO) "1" = 2 = Lossy (ICER_1BPP) "2" = 3 = Lossy (ICER_2BPP) "3" = 4 = Lossy (ICER_3BPP) "4" = 5 = Lossy (ICER_4BPP) "5" = 6 = Lossy (ICER_5BPP) "6" = 7 = Lossy (ICER_6BPP) 4) "0" = None "0" = 1 = MSSS Lossless "1" = 2 = JPEG color mode GRAY "2" = 3 = JPEG color mode 422 "3" = 4 = JPEG color mode 444	**Mode** 1) a. Compression algorithm (where value returned by compression as variable), calculation b. EMD in XML format, calculation 2) DPO in XML format (referenced to APID Name in Appendix E), Table Lookup 3) DPO in XML format (referenced to APID Name in Appendix E), Table Lookup 4) DPO in XML format (referenced to APID Name in Appendix E) or Image DPO mini-header, Table Lookup **Field as "<xml name>:[<element>]:[<element>]:<field>"** • All Cameras 1) b. "MslEarthProductMetadata:MslProductMetadata: ProductName" • Eng. Cameras 2) a. "<IDPH DPO>:idph:params:comp:compress" b. "<IDPH DPO>:idph:params:ref_comp:compress" c. "<IDPH DPO>:idph:params:thumb_comp:compress" • Chemcam RMI 3) a. "<Ancillary DPO>:cmd_arguments:compression" b. "<Ancillary DPO>:cmd_arguments:

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
	Type integer **Units** n/a **Location** 1) COMPRESSION_PARMS (Group) 2) a. IMAGE_REQUEST_PARMS (Group) b. REFERENCE_PIXEL_REQUEST_PARMS (Group) c. THUMBNAIL_REQUEST_PARMS (Group) 3) a. IMAGE_REQUEST_PARMS (Group) b. THUMBNAIL_REQUEST_PARMS (Group) 4) - IMAGE_REQUEST_PARMS (Group) - THUMBNAIL_REQUEST_PARMS (Group) - COMPRESSION_PARMS (Group) - MINI_HEADER (Group) ** ** Except MAHLI Z-Stack & Range Map	THUMBNAIL_COMPRESSION” • MMM Cameras 4) a. “<Ancillary DPO>:cmd_arguments_image:comp_quality” b. “<Ancillary DPO>:cmd_arguments_zstack:comp_quality” c. “MMM_Image_Mini_Header[32]”, “MMM_Image_Mini_Header[33]”, “MMM_Image_Mini_Header[34]”, “MMM_Image_Mini_Header[35]”, “MMM_Image_Mini_Header[36]”, “MMM_Image_Mini_Header[37]”, “MMM_Image_Mini_Header[38]”, “MMM_Image_Mini_Header[39]” NOTES: • For MMM non-recovered (rangemap or zstack) data products, value comes from Ancillary DPO per case “4a”. • For MMM recovered (rangemap or zstack) data products, value comes from Ancillary DPO per case “4b”. • For MMM Cameras, value analogous to “compress” is comprised of eight bytes coming from Image DPO mini-header at byte offsets 32, 33, 34, 35, 36, 37, 38 and 39 per case “4c”. **Type** 1) a. n/a b. enum 2) enum 3) enum 4) enum
Ops Keyword INST_CMPRS_NAME **PDS Keyword** same **Definition** Specifies the type of on-board compression used for data storage and transmission. Valid values for compression types supported in MSL are: a) “NONE” - No compression. b) “ICER” - Adaptive Variable-Length Coding, a lossy or lossless compression algorithm developed at JPL. c) “LOCO” - Low-Complexity Lossless Compression, a lossless compression algorithm developed at JPL. d) “JPEG” - Joint Photographic Experts Group, an industry standard lossy compression algrorithm. e) “MSSS_LOSSLESS” - Lossless compression algorithm developed by Malin Space Science Systems. NOTE: For the MSL mission, LOCO is lossless, JPEG is	**Valid Values** 1) b. 0 = “None or Lossless” 1 = “Lossy” 2) 0 = “None” 1 = “ICER” 2 = “ICER” 3 = “LOCO” 4 = “LOCO” 3) 0 = “None” 1 = “LOCO” 2 = “ICER” 3 = “ICER” 4 = “ICER” 5 = “ICER” 6 = “ICER” 7 = “ICER” 4) 0 = “None” 1 = “MSSS_LOSSLESS” 2 = “JPEG” 3 = “JPEG” 4 = “JPEG”	**Mode** 1) a. Compression algorithm (where value returned by compression as variable), calculation b. EMD in XML format, calculation 2) DPO in XML format (referenced to APID Name in Appendix E), Table Lookup 3) DPO in XML format (referenced to APID Name in Appendix E), Table Lookup 4) Image DPO mini-header, Table Lookup 5) DPO in XML format (referenced to APID Name in Appendix E), Table Lookup **Field as “<xml name>:[<element>]:[<element>]:<field>”** • All Cameras 1) b. “MslEarthProductMetadata:MslProductMetadata: ProductName” • Eng. Cameras 2) a. “<IDPH DPO>:idph:params:comp:compress” b. “<IDPH DPO>:idph:params:ref_comp:compress” c. “<IDPH DPO>:idph:params:thumb_comp:compress”

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
lossy, and ICER can be either (generally lossy). See INST_CMPRS_MODE for lossless/lossy status.	**Type** integer **Units** n/a **Location** 1) COMPRESSION_PARMS (Group) 2) a. IMAGE_REQUEST_PARMS (Group) b. REFERENCE_PIXEL_REQUEST_PARMS (Group) c. THUMBNAIL_REQUEST_PARMS (Group) 3) a. IMAGE_REQUEST_PARMS (Group) b. THUMBNAIL_REQUEST_PARMS (Group) 4) - IMAGE_REQUEST_PARMS (Group) - THUMBNAIL_REQUEST_PARMS (Group) - COMPRESSION_PARMS (Group)	• Chemcam RMI 3) a. “<Ancillary DPO>:cmd_arguments:compression” b. “<Ancillary DPO>:cmd_arguments: THUMBNAIL_COMPRESSION” • MMM Cameras 4) a. “<Ancillary DPO>:cmd_arguments_image:comp_quality” b. “<Ancillary DPO>:cmd_arguments_zstack:comp_quality” c. “MMM_Image_Mini_Header[32]”, “MMM_Image_Mini_Header[33]”, “MMM_Image_Mini_Header[34]”, “MMM_Image_Mini_Header[35]”, “MMM_Image_Mini_Header[36]”, “MMM_Image_Mini_Header[37]”, “MMM_Image_Mini_Header[38]”, “MMM_Image_Mini_Header[39]” NOTES: • For MMM non-recovered (rangemap or zstack) data products, value comes from Ancillary DPO per case “4a”. • For MMM recovered (rangemap or zstack) data products, value comes from Ancillary DPO per case “4b”. • For MMM Cameras, value analogous to “compress” is comprised of eight bytes coming from Image DPO mini-header at byte offsets 32, 33, 34, 35, 36, 37, 38 and 39 per case “4c”. **Type** 1) a. n/a b. enum 2) enum 3) enum 4) enum
Ops Keyword INST_CMPRS_QUALITY **PDS Keyword** same **Definition** Specifies a JPEG- or ICER-specific variable which identifies the resultant or targeted image quality index for on-board data compression. NOTE: For MMM, a value of 0 indicates lossless or no Compression; 1-100 represent JPEG quality levels. Specifies companding, spatial sampling, colorspace processing and compression algorithms (meaning described below and related to compression quality factor and/or output bit rate and algorithms used). If	**Valid Values** • Eng. Cameras “0” to “18” • MMM Cameras “0” to “100” **Type** integer **Units** n/a **Location** 1) a. IMAGE_REQUEST_PARMS (Group) b. REFERENCE_PIXEL_REQUEST_PARMS (Group) c. THUMBNAIL_REQUEST_PARMS (Group) 2) a. COMPRESSION _PARMS (Group)	**Mode** 1) DPO in XML format (referenced to APID Name in Appendix E) 2) DPO in XML format (referenced to APID Name in Appendix E) or Image DPO mini-header **Field as “<xml name>:[<element>]:[<element>]:<field>”** • Eng. Cameras 1) a. “<IDPH DPO>:idph:params:comp:min_loss” b. “<IDPH DPO>:idph:params:ref_comp:min_loss” c. “<IDPH DPO>:idph:params:thumb_comp:min_loss” • MMM Cameras 2) a. “<Ancillary DPO>:image_id_data:comp_quality” b. “MMM_Image_Mini_Header[32]”, “MMM_Image_Mini_Header[33]”, “MMM_Image_Mini_Header[34]”, “MMM_Image_Mini_Header[35]”, “MMM_Image_Mini_Header[36]”,

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
a filter for a given image is not set to clear (filter 0), then compression is performed in "narrowband science" mode; otherwise it is performed in "RGB" mode. If multiple images are acquired by this command, they increment in the order i, i+1, i+2, … i+(n_images-1). Viewing the compression parameters as 8 bytes ABCDEFGH from MSB to LSB, the bytes mean the following: • A, B, and EFG are unused and should be commanded as zero. • C is the color mode used for compression, where 0 is gray scale, 1 is 422 color subsampling mode, and 2 is 444 color mode (no subsampling). Most broadband color images should use C=1. • For D, if zero and C is 0xff, then lossless compression is used. If D is non-zero, then the image is JPEG-compressed by the quality factor D which ranges from 1 (lowest quality) to 100 (highest quality). Quality 75 is "usually nearly indistinguishable from the source image" in the ITU T.81 JPEG specification. • H specifies the 12-to-8 bit companding table to use. 0 is the default, nominally lossless square-root table. 1-16 encode the pixles linearly by dividing by N with saturation at 255; 17-32 encode the pixels linearly without saturation (the low-order 8 bits are simply transmitted). 0xff selects 16-bit calibration mode, which has restrictions on image dimension, may not be compressed, and is not intended to be used in flight. Other values are not yet defined.	b. MINI_HEADER (Group) ** ** Except MAHLI Z-Stack & Range Map	"MMM_Image_Mini_Header[37]", "MMM_Image_Mini_Header[38]", "MMM_Image_Mini_Header[39]" NOTES: • For MMM Cameras, if user-specified processing is not configured to use Mini-header compression bytes, then value comes from the Ancillary DPO (case "2a"). • For MMM Cameras, if user-specified processing is configured to use Mini-header compression bytes, then value is comprised of eight bytes coming from Image DPO mini-header at byte offsets 32, 33, 34, 35, 36, 37, 38 and 39 (case "2b"). **Type** U8
Ops Keyword INST_CMPRS_RATE **PDS Keyword** same **Definition** Specifies the average number of bits needed to represent a pixel for an on-board compressed image.	**Valid Values** • Eng. Cameras 1) n/a 2) " 1.0" to "12.0" • Chemcam 3) "1.0" = 2 = CCAM_ICER_1BPP "2.0" = 3 = CCAM_ICER_2BPP "3.0" = 4 = CCAM_ICER_3BPP "4.0" = 5 = CCAM_ICER_4BPP "5.0" = 6 = CCAM_ICER_5BPP "6.0" = 7 = CCAM_ICER_6BPP • MMM Cameras 4) n/a • Additional value for Location "1"	**Mode** 1) Calculation: - Value = bits_in/pixels_out 2) DPO in XML format (referenced to APID Name in Appendix E), Table Lookup 3) DPO in XML format (referenced to APID Name in Appendix E), Table Lookup 4) Calculation: - Value = bits_in * 8/pixels **Field as "<xml name>:[<element>]:[<element>]:<field>"** • Eng. Cameras 2) a. "<IDPH DPO>:idph:params:comp:bpp" b. "<IDPH DPO>:idph:params:ref_comp:bpp" c. "<IDPH DPO>:idph:params:thumb_comp:bpp"

<table>
<tr><th colspan="2">OUTPUT METADATA (PRODUCT LABEL)</th><th>INPUT METADATA (SOURCE)</th></tr>
<tr><th>• Ops Keyword
• PDS-Compliant Keyword
• Definition</th><th>• Valid Values (quoted)
• Type
• Units
• Keyword Location in Label</th><th>• Mode
• Metadata Field
• Type</th></tr>
<tr><td></td><td>"N/A" = 0

<u>**Type**</u>
float

<u>**Units**</u>
n/a

<u>**Location**</u>
1) COMPRESSION_PARMS (Group)
2) a. IMAGE_REQUEST_PARMS (Group)
b. REFERENCE_PIXEL_REQUEST_PARMS (Group)
c. THUMBNAIL_REQUEST_PARMS (Group)
3) a. IMAGE_REQUEST_PARMS (Group)
b. THUMBNAIL_REQUEST_PARMS (Group)
4) - IMAGE_REQUEST_PARMS (Group)
- THUMBNAIL_REQUEST_PARMS (Group)
- COMPRESSION_PARMS (Group)</td><td>• <u>Chemcam</u>
3) a. "<Ancillary DPO>:cmd_arguments:compression"
b. "<Ancillary DPO>:cmd_parameters: THUMBNAIL_COMPRESSION"

<u>**Type**</u>
1) n/a
2) F32
3) enum
4) n/a</td></tr>
<tr><td><u>**Ops Keyword**</u>
INST_CMPRS_RATIO

<u>**PDS Keyword**</u>
same

<u>**Definition**</u>
Specifies the ratio of the size, in bytes, of the original uncompressed data file to its compressed form.</td><td><u>**Valid Values**</u>
Uncompressed = "N/A"

<u>**Type**</u>
float

<u>**Units**</u>
n/a

<u>**Location**</u>
COMPRESSION_PARMS (Group)</td><td><u>**Mode**</u>
1) For Eng. Cameras, Calculation:
- Sum of the size of ICER uncompressed image over the compressed segments area (bits_out/bits_in)
2) For MMM Cameras, Calculation:
- Value = pixels/bits_in

<u>**Field as "<xml name>:[<element>]:[<element>]:<field>"**</u>
n/a

<u>**Type**</u>
n/a</td></tr>
<tr><td><u>**Ops Keyword**</u>
INST_CMPRS_SEGMENTS

<u>**PDS Keyword**</u>
same

<u>**Definition**</u>
Specifies the number of segments into which the image was partitioned for the error containment purposes. For ICER compression, the data within each segment is compressed independently, so that data loss across segments is compartmentalized or contained across segments.</td><td><u>**Valid Values**</u>
• <u>Eng. Cameras</u>
1) "1" to "32"
• <u>MMM Cameras</u>
2) "N/A"
3) "UNK"
• <u>Additional value for Location "1a"</u>
"N/A"

<u>**Type**</u>
integer

<u>**Units**</u>
n/a

<u>**Location**</u>
1) a. COMPRESSION_PARMS (Group)
b. IMAGE_REQUEST_PARMS (Group)
c. REFERENCE_PIXEL_REQUEST_PARMS (Group)</td><td><u>**Mode**</u>
• ICER segment
1) a. Extracted from ICER segment
• DPO in XML format (referenced to APID Name in Appendix E), Table Lookup

<u>**Field as "<xml name>:[<element>]:[<element>]:<field>"**</u>
1) b. "<IDPH DPO>:idph:params:comp:n_segs"
c. "<IDPH DPO>:idph:params:ref_comp:n_segs"
d. "<IDPH DPO>:idph:params:thumb_comp:n_segs"

<u>**Type**</u>
1) a. n/a
b. U8
c. U8
d. U8</td></tr>
</table>

<table>
<tr><th colspan="2">OUTPUT METADATA (PRODUCT LABEL)</th><th>INPUT METADATA (SOURCE)</th></tr>
<tr><th>• Ops Keyword
• PDS-Compliant Keyword
• Definition</th><th>• Valid Values (quoted)
• Type
• Units
• Keyword Location in Label</th><th>• Mode
• Metadata Field
• Type</th></tr>
<tr><td></td><td>d. THUMBNAIL_REQUEST_PARMS (Group)
2) COMPRESSION_PARMS (Group)
3) - IMAGE_REQUEST_PARMS (Group)
- REFERENCE_PIXEL_REQUEST_PARMS (Group)
- THUMBNAIL_REQUEST_PARMS (Group)</td><td></td></tr>
<tr><td><u>Ops Keyword</u>
INST_CMPRS_SEGMENT_QUALITY

<u>PDS Keyword</u>
same

<u>Definition</u>
Specifies the quality level for each ICER segment.

For MSL, upon return by the ICER decompress function “deicer_decompress” [Ref 31] for data segment “seg”, the output quantity “seg_quality[seg]” provides an indication of the quality of the reconstructed segment. Specifically, the value returned is a double for which the integer values correspond to attained min loss values, but in general is an interpolation between these values. Thus lower values of seg_quality[seg] correspond to higher reconstructed qualities, and a value of indicates lossless compression. Note that the compressed stream does not directly contain the value of min loss that was given to the compressor, but the decompressor does know how far along in the decompression process it got before it ran out of bits; this information is used to determine seg quality[seg].

In rare circumstances the decompressor may not be able to determine seg quality[seg] for a segment that it decompresses. In this case it sets seg quality[seg] to −1.0. The reconstructed segment might be either lossy or lossless when this occurs. The technical condition under which a quality value is not determined is that the decompressor runs out of the data for the segment before decoding any bit plane information. (The only way such a reconstructed segment could be lossless is if every bit plane of this segment contains all zeros, so there is no bit plane information to decode. This could only happen if every pixel of the segment and all pixels sufficiently close to the segment have exactly the same value.)

Note that if segment seg cannot be reconstructed, then nothing is written to real seg num[seg], int seg bound first line[seg], int seg bound first sample[seg], int seg bound n lines[seg], int seg bound n samples[seg], and seg quality[seg].

The decompressor also returns a status word which may have one of the following flags set:
• BADNUMDATASEG FLAG (bit 0): If this flag is set, then n data seg, the number of data segments, is outside the</td><td><u>Valid Values</u>
• <u>Eng. Cameras</u>
Uncompressed or LOCO = “N/A”
• <u>MMM Cameras</u>
“N/A”

<u>Type</u>
float array

<u>Units</u>
n/a

<u>Location</u>
COMPRESSION_PARMS (Group)</td><td><u>Mode</u>
Extracted from ICER segment

<u>Field as “<xml name>:[<element>]:[<element>]:<field>”</u>
n/a

<u>Type</u>
n/a</td></tr>
</table>

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
allowed range [1, ICER MAX N SEGS]. Decompression could not be performed. • NOGOODSEGMENTS FLAG (bit 1): If this flag is set, then decompression could not be performed because none of the segments contained usable data.		
Ops Keyword INST_CMPRS_SEGMENT_STATUS **PDS Keyword** MSL:INST_CMPRS_SEGMENT_STATUS **Definition** Specifies a bit mask which provides the status of decoding the nth segment. For MSL, upon return by the ICER decompress function “deicer_decompress” [Ref 31] for data segment “seg”, the output quantity “data_seg_status[seg]” contains a number indicating the decode status. The decode status may have one or more of the following flags set: • SHORTDATASEG FLAG (bit 0): If this flag is set, then the segment contained so little data that nothing could be reconstructed in the segment. • INCONSISTENTDATA FLAG (bit 1): If this flag is set, then one or more pieces of information in the segment header (specifically, image width, image height, n segs, wavelet filter, n decomps) are inconsistent with the value(s) in the first (valid) segment. ICER will ignore the data in this segment. • DUPLICATESEG FLAG (bit 2): If this flag is set, then the segment index given in the header equals that given by a previous segment. The decompressor will ignore the data in this segment. • BADBITPLANENUMBER FLAG (bit 3): If this flag is set, then an ICER internal parameter in the header for this segment has probably been corrupted. The decompressor will ignore the data in this segment. • BADBITPLANECOUNT FLAG (bit 4): If this flag is set, then an ICER internal parameter in the header for this segment has probably been corrupted. The decompressor will ignore the data in this segment. • BADDATA FLAG (bit 5): If this flag is set, then either the parameter combination given in the header for this segment are not allowed by ICER, or the segment number is bad. This probably indicates corrupted data. The decompressor will ignore the data in this segment.	**Valid Values** • Eng. Cameras Uncompressed or LOCO = “N/A” • MMM Cameras “N/A” **Type** integer array **Units** n/a **Location** COMPRESSION_PARMS (Group)	**Mode** Returned from ICER decompression routine **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a
Ops Keyword INST_CMPRS_SEG_FIRST_LINE	**Valid Values** • Eng. Cameras	**Mode** Extracted from ICER segment

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
PDS Keyword same **Definition** Specifies an array of values in which each nth element identifies the line within a source image that corresponds to the first line where the nth compression segment applies. Value of “-1” denotes the indeterminate case when the decompressor cannot process the segment, or cannot determine seq_quality for a segment that it could decompress.	a. “-1” to “1024” b. Uncompressed or LOCO = “N/A” • MMM Cameras “N/A” **Type** integer array **Units** n/a **Location** COMPRESSION_PARMS (Group)	**Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a
Ops Keyword INST_CMPRS_SEG_FIRST_LINE_SAMP **PDS Keyword** same **Definition** Specifies an array of values in which each nth element identifies the line sample within a source image that corresponds to the first line sample where the nth compression segment applies. Value of “-1” denotes the indeterminate case when the decompressor cannot process the segment, or cannot determine seq_quality for a segment that it could decompress.	**Valid Values** • Eng. Cameras a. “-1” to “1024” b. Uncompressed or LOCO = “N/A” • MMM Cameras “N/A” **Type** integer array **Units** n/a **Location** COMPRESSION_PARMS (Group)	**Mode** Extracted from ICER segment **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a
Ops Keyword INST_CMPRS_SEG_LINES **PDS Keyword** same **Definition** Specifies an array of elements in which the nth element identifies the total number of data instances along the vertical axis that the nth compression segment defines. Value of “-1” denotes the indeterminate case when the decompressor cannot process the segment, or cannot determine seq_quality for a segment that it could decompress.	**Valid Values** • Eng. Cameras a. “-1” to “1024” b. Uncompressed or LOCO = “N/A” • MMM Cameras “N/A” **Type** integer array **Units** n/a **Location** COMPRESSION_PARMS (Group)	**Mode** Extracted from ICER segment **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a
Ops Keyword INST_CMPRS_SEG_MISSING_PIXELS **PDS Keyword** same	**Valid Values** • Eng. Cameras Uncompressed or ICER = “N/A” • MMM Cameras	**Mode** Extracted from LOCO segment **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
Definition Specifies an array of elements in which the nth element identifies the total number of missing pixels that the nth compression segment defines.	"N/A" **Type** integer array **Units** n/a **Location** COMPRESSION_PARMS (Group)	**Type** n/a
Ops Keyword INST_CMPRS_SEG_SAMPLES **PDS Keyword** same **Definition** Specifies an array of elements in which the nth element identifies the total number of data instances along the horizontal axis that the nth compression segment defines. Value of "-1" denotes the indeterminate case when the decompressor cannot process the segment, or cannot determine seq_quality for a segment that it could decompress.	**Valid Values** • Eng. Cameras a. "-1" to "1024" b. Uncompressed or LOCO = "N/A" • MMM Cameras "N/A" **Type** integer array **Units** n/a **Location** COMPRESSION_PARMS (Group)	**Mode** Extracted from ICER segment **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword INST_DECOMP_STAGES **PDS Keyword** same **Definition** Specifies the number of stages of wavelet decompositions.	**Valid Values** • Eng. Cameras "1" to "6" • MMM Cameras "N/A" • Additional value for Location "1" Uncompressed or LOCO = "N/A" **Type** integer **Units** n/a **Location** 1) COMPRESSION_PARMS (Group) 2) IMAGE_REQUEST_PARMS (Group) 3) REFERENCE_PIXEL_REQUEST_PARMS (Group) 4) THUMBNAIL_REQUEST_PARMS (Group)	**Mode** • ICER segment 1) Extracted from ICER segment • DPO in XML format (referenced to APID Name in Appendix E), Table Lookup **Field as "<xml name>:[<element>]:[<element>]:<field>"** 2) "<IDPH DPO>:idph:params:comp:n_decomps" 3) "<IDPH DPO>:idph:params:ref_comp:n_decomps" 4) "<IDPH DPO>:idph:params:thumb_comp:n_decomps" **Type** 1) n/a 2) U8 3) U8 4) U8
Ops Keyword INTERCHANGE_FORMAT **PDS Keyword**	**Valid Values** "ASCII", "BINARY" **Type**	**Mode** Static Value **Field as "<xml name>:[<element>]:[<element>]:<field>"**

<table>
<tr><th colspan="2">OUTPUT METADATA (PRODUCT LABEL)</th><th>INPUT METADATA (SOURCE)</th></tr>
<tr><th>• Ops Keyword
• PDS-Compliant Keyword
• Definition</th><th>• Valid Values (quoted)
• Type
• Units
• Keyword Location in Label</th><th>• Mode
• Metadata Field
• Type</th></tr>
<tr><td>same

<u>Definition</u>
Specifies the manner in which data items are stored.</td><td>string(6)

<u>Units</u>
n/a

<u>Location</u>
• IMAGE_HEADER (Object)
• IMAGE (Object)</td><td>n/a

<u>Type</u>
n/a</td></tr>
<tr><td><u>Ops Keyword</u>
INTERPOLATION_METHOD

<u>PDS Keyword</u>
MSL:INTERPOLATION_METHOD

<u>Definition</u>
Some camera models exhibit continuous changes in geometric behavior along a dimension(s). For example, the B-side Navcams vary geometrically based on temperature, and the Mastcams vary based on focus setting. In order to model this continuous variation, camera models may be interpolated, which creates camera models "in between" a set of rigorously calibrated models using piecewise linear interpolation. INTERPOLATION_METHOD specifies the dimension(s) along which interpolation has been performed, and/or where it was performed.

For MSL, the valid values for EDR's are "ONBOARD" (temperature interpolation done onboard for Navcam), "NONE" (no interpolation), or "OTHER". For RDR's, the above are still valid, additionally the values may be "TEMPERATURE" or "FOCUS" to indicate the dimension over which interpolation was done. Multiple dimensions of interpolation would be indicated by multiple values in INTERPOLATION_METHOD.</td><td><u>Valid Values</u>
See Definition

<u>Type</u>
string

<u>Units</u>
n/a

<u>Location</u>
GEOMETRIC_CAMERA_MODEL_PARMS (Group)</td><td><u>Mode</u>
RDR-generating software

<u>Field as “<xml name>:[<element>]:[<element>]:<field>”</u>
n/a

<u>Type</u>
n/a</td></tr>
<tr><td><u>Ops Keyword</u>
INTERPOLATION_VALUE

<u>PDS Keyword</u>
MSL:INTERPOLATION_VALUE

<u>Definition</u>
Specifies the value to which the camera models were interpolated (usually, the specific focus or temperature value used).
See INTERPOLATION_METHOD. Multiple dimensions of interpolation would lead to multiple values in this label.</td><td><u>Valid Values</u>
n/a

<u>Type</u>
float

<u>Units</u>
n/a

<u>Location</u>
GEOMETRIC_CAMERA_MODEL_PARMS (Group)</td><td><u>Mode</u>
RDR-generating software

<u>Field as “<xml name>:[<element>]:[<element>]:<field>”</u>
n/a

<u>Type</u>
n/a</td></tr>
<tr><td><u>Ops Keyword</u>
INVALID_CONSTANT

<u>PDS Keyword</u></td><td><u>Valid Values</u>
<u>Product</u> <u>Value</u>
most OPGS-gen’d products ….. “0.0”
XYZ & Surface Normal ………. “(0.0,0.0,0.0)”</td><td><u>Mode</u>
Static Value

<u>Field as “<xml name>:[<element>]:[<element>]:<field>”</u></td></tr>
</table>

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
same **Definition** Specifies the value used when the received data are out of the legitimate range of values. For MSL, the value should be 0.0 for most OPGS-generated products, with the exception of Surface Roughness RDRs. For SOAS-generated products, the value may be different.	Surface Roughness ………….. param value (default is “0.1”) **Type** float or float array **Units** n/a **Location** IMAGE (Object)	n/a **Type** n/a
Ops Keyword INVERSE_LUT_FILE_NAME **PDS Keyword** MSL:INVERSE_LUT_FILE_NAME **Definition** Specifies the name of the inverse-lookup-table file used in generating the RDR. NOTE: This keyword only applies if SAMPLE_BIT_MODE_ID in Group INSTRUMENT_STATE_PARMS starts with "LUT", i.e. if an inverse lookup table file was used.	**Valid Values** “N/A” **Type** string **Units** n/a **Location** DERIVED_IMAGE_PARMS (Group)	**Mode** RDR-generating software **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a
Ops Keyword LABEL_RECORDS **PDS Keyword** same **Definition** Specifies the number of physical file records that contain only (ODL or PDS) label information. The number of data records in a file is determined by subtracting the value of label_records from the value of file_records. NOTE: In the PDS, the use of label_records along with other file-related data elements is fully described in the Standards Reference.	**Valid Values** “0” to n **Type** string array **Units** n/a **Location** FILE (Class)	**Mode** Calculation: - Based on size of PDS label **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a
Ops Keyword LASER_MODE **PDS Keyword** MSL:LASER_MODE **Definition** For ChemCam on MSL, specifies whether to fire or not fire the laser. It is an argument for command CCAM_ACTV_SPECTRAL_OBS (see ChemCam FDD).	**Valid Values** 0 = “NO” 1 = “YES” **Type** string **Units** n/a **Location** OBSERVATION_REQUEST_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as “<xml name>:[<element>]:[<element>]:<field>”** Chemcam “<Ancillary DPO>:cmd_arguments:use_laser” **Type** enum

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
Ops Keyword LINEARIZATION_MODE **PDS Keyword** MSL:LINEARIZATION_MODE **Definition** Specifies the way in which an image has been linearized (see GEOMETRY_PROJECTION_TYPE). It can contain one or two values. The first value specifies what kind of stereo partner was used to linearize the image (the process requires two camera models). A value of "NOMINAL" means that it was linearized using the nominal stereo partner at the same pointing (whether or not the partner image was actually acquired). "ACTUAL" means it was linearized using an actual stereo partner image, which may be at a different pointing or rover location (e.g. long-baseline or re-pointed stereo). The product ID of the actual partner used will be in LINEARIZATION_PRODUCT_ID. A value of NONE, means that linearization has not been performed. If the value is not present and linearization is on, "NOMINAL" should be assumed. The second value specifies how the linearized camera model's field of view (FOV) was constructed (corresponding to the "cahv_fov" parameter in MIPL software; see [Ref 27]). "MIN" indicates the FOV is the intersection of the two cameras, which means some data are cut off and typically stretched horizontally, but there are no black areas. "MAX" indicates the FOV is the union of the two cameras, which preserves the edges of the images but typically compresses the image horizontally, and creates black areas. "LINEAR" uses only the CAHV vectors, which tends to preserve the aspect ratio and scale of the original but both cuts off data and has black areas. "NONE" means no linearization. If the value is not present and linearization is on, "MIN" should be assumed.	**Valid Values** • 1st Element in Array "NOMINAL", "ACTUAL", "NONE" • 2nd Element in Array "MIN", "MAX", "LINEAR", "NONE" **Type** string array(2) **Units** n/a **Location** DERIVED_IMAGE_PARMS (Group)	**Mode** RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword LINEARIZATION_PRODUCT_ID **PDS Keyword** none **Definition** Specifies the PRODUCT_ID for the stereo partner used to linearize the image when LINEARIZATION_MODE is "ACTUAL". This keyword may be absent if the mode is not "ACTUAL".	**Valid Values** n/a **Type** string **Units** n/a **Location** DERIVED_IMAGE_PARMS (Group)	**Mode** RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
Ops Keyword LINE_CAMERA_MODEL_OFFSET **PDS Keyword** same **Definition** Specifies the location of the image origin with respect to the camera model's origin. For CAHV/CAHVOR models, this origin is not the center of the camera, but is the upper-left corner of the "standard"-size image, which is encoded in the CAHV vectors. (MIPL Projection - Perspective)	**Valid Values** n/a **Type** float **Units** pixel (<pixel> unit tag required) **Location** SURFACE_PROJECTION_PARMS (Group)	**Mode** RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword LINE_PROJECTION_OFFSET **PDS Keyword** same **Definition** Specifies the line coordinate of the location in the image of the "special" point of the mosaic. For Polar projections, this is the nadir of the polar projection. For Vertical, Orthographic and Orthorectified projections, this is the origin of the projected coordinate system (corresponding to PROJECTION_ORIGIN_VECTOR), and may be off the image.	**Valid Values** n/a **Type** float **Units** pixel (<pixel> unit tag required) **Location** SURFACE_PROJECTION_PARMS (Group)	**Mode** RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword LINE_SAMPLES **PDS Keyword** same **Definition** Specifies the total number of data instances along the horizontal axis of an image.	**Valid Values** "0" to "1024" **Type** integer **Units** n/a **Location** 1) a. IMAGE (Object) b. SUBFRAME_REQUEST_PARMS (Group) c. THUMBNAIL_REQUEST_PARMS (Group) 2) SUBFRAME_REQUEST_PARMS (Group) THUMBNAIL_REQUEST_PARMS (Group) 3) a. SUBFRAME_REQUEST_PARMS (Group) b. MINI_HEADER (Group) ** ** Except MAHLI Z-Stack & Range Map	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** 1) Eng. Cameras a. "<IDPH DPO>:idph:cols" b. "<IDPH DPO>:idph:params:sub_cols" c. "<IDPH DPO>:idph:params:thumb_cols" 2) Chemcam "<Ancillary DPO>:cmd_arguments:r_height" 3) MMM Cameras a. "<Ancillary DPO>:cmd_arguments_image:window_width" b. "MMM_Image_Mini_Header[TBD]" **Type** U16
Ops Keyword LINES **PDS Keyword** same **Definition**	**Valid Values** "0" to "1024" **Type** integer **Units**	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** 1) Eng. Cameras a. "<IDPH DPO>:idph:rows"

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
Specifies the total number of data instances along the vertical axis of an image. NOTE: In PDS label convention, the number of lines is stored in a 32-bit integer field. The minimum value of 0 indicates no data received.	n/a **Location** 1) a. IMAGE (Object) b. SUBFRAME_REQUEST_PARMS (Group) c. THUMBNAIL_REQUEST_PARMS (Group) 2) SUBFRAME_REQUEST_PARMS (Group) THUMBNAIL_REQUEST_PARMS (Group) 3) a. SUBFRAME_REQUEST_PARMS (Group) b. MINI_HEADER (Group) ** ** Except MAHLI Z-Stack & Range Map	b. "<IDPH DPO>:idph:params:sub_rows" c. "<IDPH DPO>:idph:params:thumb_rows" 2) Chemcam "<Ancillary DPO>:cmd_arguments:c_height" 3) MMM Cameras a. "<Ancillary DPO>:cmd_arguments_image:window_height" b. "MMM_Image_Mini_Header[**TBD**]" **Type** U16
Ops Keyword LOCAL_MEAN_SOLAR_TIME **PDS Keyword** MSL:LOCAL_MEAN_SOLAR_TIME **Definition** Specifies the Local Mean Solar Time, or LMST. It is one of two types of solar time used to express the time of day at a point on the surface of a planetary body. The desire to work with solar days, hours, minutes, and seconds of uniform length led to the concept of the fictitious mean Sun or FMS. The FMS is defined as a point that moves on the celestial equator of a planetary body at a constant rate that represents the average mean motion of the Sun over a planetary year. Local mean solar time is defined, by analogy with Local True Solar Time (LTST), as the difference between the areocentric right ascensions of a point on the surface and of the FMS. The difference between LTST and LMST varies over time. The length of a mean solar day is constant and can be computed from the mean motion of the FMS and the rotation rate of a planet. The mean solar day is also called a 'Sol'. Mean solar hours, minutes, and seconds are defined in the same way as the true solar units. For MSL, the valid value is embedded with a Sol value that can be different than the Sol (see PLANET_DAY_NUMBER) associated with LTST (see LOCAL_TRUE_SOLAR_TIME). The time portion of the valid value is expressed in terms of a 24-hour clock. So, in an example using Sol 27, the valid value range for the 24-hour clock would be represented as "Sol-00027M00:00:00.000" to "Sol-00027M23:59:59.999".	**Valid Values** Sol-<nnnnn>M<hh>:<mm>:<ss>[.<fff>] NOTE: Value will be uncalibrated if SPICE kernels are unavailable. **Type** string(30) **Units** n/a **Location** IDENTIFICATION (Class)	**Mode** 1) a. Calculation: - SCLK Kernel - Landing Site Kernel - P Kernel b. DPO in XML format (referenced to APID Name in Appendix E) 2) a. Image DPO mini-header, Calculation: - SCLK Kernel - Landing Site Kernel - P Kernel b. DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** 1) Eng. Cameras b. "<IDPH DPO>:idph:sclk_seconds", "<IDPH DPO>:idph:sclk_subseconds" 2) MMM Cameras a. "MMM_Image_Mini_Header[8]", "MMM_Image_Mini_Header[9]", "MMM_Image_Mini_Header[10]", "MMM_Image_Mini_Header[11]" b. "<IDPH DPO>:cidph:sclk:seconds" c. "<Ancillary DPO>:sclk:seconds" NOTES: • For MMM recovered data products, value is comprised of four bytes coming from Image DPO mini-header at byte offsets 8, 9, 10 and 11 per case "2a". • For MMM non-recovered data products, if IDPH is present, value comes from IDPH DPO per case "2b". • For MMM non-recovered data products, if IDPH is not present, value comes from Ancillary DPO per case "2c". **Type** U32
Ops Keyword LOCAL_TRUE_SOLAR_TIME	**Valid Values** <hh>:<mm>:<ss>[.<fff>]	**Mode** 1) a. Calculation: - SCLK Kernel

<table>
<tr><th colspan="2">OUTPUT METADATA (PRODUCT LABEL)</th><th>INPUT METADATA (SOURCE)</th></tr>
<tr><th>• Ops Keyword
• PDS-Compliant Keyword
• Definition</th><th>• Valid Values (quoted)
• Type
• Units
• Keyword Location in Label</th><th>• Mode
• Metadata Field
• Type</th></tr>
<tr><td><u>PDS Keyword</u>
same

<u>Definition</u>
Specifies the local true solar time, or LTST. It is one of two types of solar time used to express the time of day at a point on the surface of a planetary body. LTST is measured relative to the true position of the Sun as seen from a point on the planet's surface.

The coordinate system used to define LTST has its origin at the center of the planet. Its Z-axis is the north pole vector (or spin axis) of the planet. The X-axis is chosen to point in the direction of the vernal equinox of the planet's orbit. (The vernal or autumnal equinox vectors are found by searching the planetary ephemeris for those times when the vector from the planet's center to the Sun is perpendicular to the planet's north pole vector. The vernal equinox is the time when the Sun appears to rise above the planet's equator.)

Positions of points in this frame can be expressed as a radius and areocentric 'right ascension' and 'declination' angles. The areocentric right ascension angle, or ARA, is measured positive eastward in the equatorial plane from the vernal equinox vector to the intersection of the meridian containing the point with the equator. Similarly, the areocentric declination is the angle between the equatorial plane and the vector to the point. LTST is a function of the difference between the ARAs of the vectors to the Sun and to the point on the planet's surface. Specifically,
LTST = (a(P) - a(TS)) * (24 / 360) + 12
where,
LTST = the local true solar time in true solar hours
a(P) = ARA of the point on the planet's surface in deg
a(TS) = ARA of the true sun in deg

The conversion factor of 24/360 is applied to transform the angular measure in decimal degrees into hours-minutes-seconds of arc. This standard representation divides 360 degrees into 24 hours, each hour into 60 minutes, and each minute into 60 seconds of arc. The hours, minutes, and seconds of arc are called 'true solar' hours, minutes, and seconds when used to measure LTST. The constant offset of 12 hours is added to the difference in ARAs to place local noon (12:00:00 in hours, minutes, seconds) at the point where the Sun is directly overhead; at this time, the ARA of the true sun is the same as that of the surface point so that
a(P) - a(TS) = 0.</td><td>NOTE: Value will be uncalibrated if SPICE kernels unavailable.

<u>Type</u>
string(12)

<u>Units</u>
n/a

<u>Location</u>
IDENTIFICATION (Class)</td><td>- Landing Site Kernel
- P Kernel
b. DPO in XML format (referenced to APID Name in Appendix E)
2) a. Image DPO mini-header, Calculation:
- SCLK Kernel
- Landing Site Kernel
- P Kernel
b. DPO in XML format (referenced to APID Name in Appendix E)

<u>Field as “<xml name>:[<element>]:[<element>]:<field>”</u>
1) <u>Eng. Cameras</u>
b. “<IDPH DPO>:idph:sclk_seconds”, “<IDPH DPO>:idph:sclk_subseconds”
2) <u>MMM Cameras</u>
a. “MMM_Image_Mini_Header[8]”, “MMM_Image_Mini_Header[9]”, “MMM_Image_Mini_Header[10]”, “MMM_Image_Mini_Header[11]”
b. “<IDPH DPO>:cidph:sclk:seconds”
c. “<Ancillary DPO>:sclk:seconds”

NOTES:
• For MMM recovered data products, value is comprised of four bytes coming from Image DPO mini-header at byte offsets 8, 9, 10 and 11 per case “2a”.
• For MMM non-recovered data products, if IDPH is present, value comes from IDPH DPO per case “2b”.
• For MMM non-recovered data products, if IDPH is not present, value comes from Ancillary DPO per case “2c”.

<u>Type</u>
U32</td></tr>
</table>

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
The use of 'true solar' time units can be extended to define a true solar day as 24 true solar hours. Due to the eccentricity of planetary orbits and the inclination of orbital planes to equatorial planes (obliquity), the Sun does not move at a uniform rate over the course of a planetary year. Consequently, the number of SI seconds in a true solar day, hour, minute or second is not constant. For MSL, the valid value is expressed in terms of a 24-hour clock, so the acceptable range is “00:00:00.000” to “23:59:59.999”. See also LOCAL_MEAN_SOLAR_TIME and PLANET_DAY_NUMBER.		
Ops Keyword LOCAL_TRUE_SOLAR_TIME_SOL **PDS Keyword** MSL:LOCAL_TRUE_SOLAR_TIME_SOL **Definition** Specifies the number of solar days elapsed since a reference day (e.g., the day on which a landing vehicle set down) for local true solar time (LTST). Days are measured in rotations of the planet in question from midnight to midnight. For MSL, the reference day is “0”, as Landing day is Sol 0. If before Landing day, then value will be less than or equal to “0” and can be negetive. NOTE: Value will be uncalibrated if SPICE kernels are unavailable. See also LOCAL_TRUE_SOLAR_TIME and PLANET_DAY_NUMBER.	**Valid Values** Mission Phase — Values Cruise — less than or equal to 0 Surface — “0” to n **Type** integer **Units** n/a **Location** IDENTIFICATION (Class)	**Mode** • Calculation: - SCLK Kernel - Landing Site Kernel - P Kernel • DPO in XML format (referenced to APID Name in Appendix E) **Field as “<xml name>:[<element>]:[<element>]:<field>”** “<IDPH DPO>:idph:sclk_seconds”, “<IDPH DPO>:idph:sclk_subseconds” **Type** U32
Ops Keyword MAGIC_NUMBERS **PDS Keyword** n/a **Definition** Specifies constant numbers that should be in a data product to confirm the product is being decoded properly. For MMM, this is the “magic0” and “magic1” fields of the Mini-header, in Hex format.	**Valid Values** n/a **Type** string **Units** n/a **Location** MINI_HEADER (Group)	**Mode** Image DPO mini-header **Field as “<xml name>:[<element>]:[<element>]:<field>”** MMM Cameras “MMM_Image_Mini_Header[**TBD**]”, “MMM_Image_Mini_Header[**TBD**]” **Type** n/a
Ops Keyword MAP_PROJECTION_TYPE	**Valid Values** “CYLINDRICAL”, “VERTICAL”, “PERSPECTIVE”, “POLAR”,	**Mode** RDR-generating software

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
PDS Keyword same **Definition** Specifies the type of projection characteristic of a given map. When in a SURFACE_PROJECTION group, defines the surface-based map projection used in the image.	"ORTHOGRAPHIC", "ORTHORECTIFIED", "CYLINDRICAL-PERSPECTIVE" **Type** string **Units** n/a **Location** SURFACE_PROJECTION_PARMS (Group)	**Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword MAP_RESOLUTION **PDS Keyword** same **Definition** Specifies the scale of a given map. Please refer to the definition for MAP_SCALE for a more complete definition. When in a SURFACE_PROJECTION group, defines the resolution of the map in pixels/degree. For Cylindrical projections, this is constant throughout. For Polar, this is for the Elevation (radial) direction only. For Perspective and Cylindrical-Perspective, this is at the center of the output camera model. Not applicable to Orthographic, Orthorectified and Vertical projections. NOTE: MAP_RESOLUTION and MAP_SCALE both define The scale of a map except that they are expressed in different units: MAP_RESOLUTION is in pixels/deg and MAP_SCALE is in meters/pixel. If two values are present, the first measures in the line direction while the second measures in the sample direction.	**Valid Values** n/a **Type** float array[2] **Units** pixels/deg (<pix/deg> unit tag required) **Location** SURFACE_PROJECTION_PARMS (Group)	**Mode** RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword MAP_SCALE **PDS Keyword** same **Definition** Specifies the scale of a given map. The scale is defined as the ratio of the actual distance between two points on the surface of the target body to the distance between the corresponding points on the map. MAP_SCALE references the scale of a map at a certain	**Valid Values** n/a **Type** float array[2] **Units** m/pixel (<m/pixel> unit tag required) **Location** SURFACE_PROJECTION_PARMS (Group)	**Mode** RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
reference point or line. Certain map projections vary in scale throughout the map. When in a SURFACE_PROJECTION group, defines the scale of the map in meters/pixel. Applicable to Vertical, Orthographic and Orthorectified projections only. NOTE: MAP_RESOLUTION and MAP_SCALE both define the scale of a map except that they are expressed in different units: MAP_RESOLUTION is in pixels/deg and MAP_SCALE is in meters/pixel. If two values are present, the first measures in the line direction while the second measures in the sample direction.		
Ops Keyword MASK_DESC_FILE_NAME **PDS Keyword** MSL:MASK_DESC_FILE_NAME **Definition** Specifies the name of a file or files containing the parameters used to create a binary mask file that, when applied to another image file, prevents specific areas in the image from being processed. For products generated by MIPL, these files are XML formatted and describe each of the mask components. They are serviceable to program “marsfilter”. For products generated by other teams, different formats and methods may apply. See also HORIZON_MASK_ELEVATION.	**Valid Values** n/a **Type** string **Units** n/a **Location** DERIVED_IMAGE_PARMS (Group)	**Mode** RDR-generating software **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a
Ops Keyword MAXIMUM **PDS Keyword** same **Definition** Specifies the largest value occurring in a given instance of the data object. Note that for PDS applications -- because of the unconventional data type of this data element, the element should appear in labels only within an explicit object, i.e., anywhere between an 'OBJECT =' and an 'END_OBJECT'.	**Valid Values** n/a **Type** float **Units** n/a **Location** IMAGE (Object)	**Mode** Calculation **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a
Ops Keyword MAXIMUM_ELEVATION	**Valid Values** n/a	**Mode** RDR-generating software

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
PDS Keyword MSL:MAXIMUM_ELEVATION **Definition** Specifies the elevation (as defined by the coordinate system) of the first line of the image. For the Polar projection, specifies the highest elevation used, i.e. the elevation of the outermost circle of pixels. Applies to MIPL projections Cylindrical, Polar, Sinusoidal, Perspective and Cylindrical-Perspective.	**Type** float **Units** deg (<deg> unit tag required) **Location** SURFACE_PROJECTION_PARMS (Group)	**Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword MAX_AUTO_EXPOS_ITERATION_COUNT **PDS Keyword** same **Definition** Specifies the maximum number of exposure iterations the instrument will perform in order to obtain the requested exposure when operating in an autonomous mode.	**Valid Values** "0" to "10" **Type** integer **Units** n/a **Location** 1) OBSERVATION_REQUEST_PARMS (Group) 2) a. OBSERVATION_REQUEST_PARMS (Group) b. MINI_HEADER (Group) ** ** Except MAHLI Z-Stack & Range Map	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** 1) Eng. Cameras "<IDPH DPO>:idph:params:exp_auto_iter" 2) MMM Cameras a. "<IDPH DPO>:idph:params:exp_auto_iter b. "MMM_Image_Mini_Header[**TBD**]" **Type** U8
Ops Keyword MEAN **PDS Keyword** same **Definition** Specifies the average of the DN values in the image array.	**Valid Values** n/a **Type** float **Units** n/a **Location** IMAGE (Object)	**Mode** Calculation **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword MEDIAN **PDS Keyword** same **Definition** Specifies the median value (middle value) occurring in a given instance of the data object. Because of the unconventional data type of this data element, the element should appear in labels only within an explicit object, i.e., anywhere between an 'OBJECT =' and an 'END OBJECT'.	**Valid Values** n/a **Type** float **Units** n/a **Location** IMAGE (Object)	**Mode** Calculation **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword	**Valid Values**	**Mode**

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
MINIMUM **PDS Keyword** same **Definition** Specifies the smallest value occurring in a given instance of the data object. Note that for PDS applications -- because of the unconventional data type of this data element, the element should appear in labels only within an explicit object, i.e., anywhere between an 'OBJECT =' and an 'END_OBJECT'.	n/a **Type** float **Units** n/a **Location** IMAGE (Object)	Calculation **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword MINIMUM_ELEVATION **PDS Keyword** MSL:MINIMUM_ELEVATION **Definition** Specifies the elevation (as defined by the coordinate system) of the last line of the image for Cylindrical map projections. Applies to Cylindrical, Perspective and Cylindrical-Perspective projections.	**Valid Values** n/a **Type** float **Units** deg (<deg> unit tag required) **Location** SURFACE_PROJECTION_PARMS (Group)	**Mode** Calculation **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword MISSING_CONSTANT **PDS Keyword** same **Definition** Specifies the flag value used in the image to indicate that no science data are available for any given pixel. See the specific product definitions for standard values used for each product.	**Valid Values** • Nominal "0.0" • XYZ "(0.0, 0.0, 0.0)" • Surface Normal (UVW) "(0.0, 0.0, 0.0)" **Type** float or float array **Units** n/a **Location** IMAGE (Object)	**Mode** Static Value **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword MISSION_NAME **PDS Keyword** same **Definition**	**Valid Values** "MARS SCIENCE LABORATORY" **Type** string array **Units**	**Mode** Static Value **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type**

<table>
<tr><th colspan="2">OUTPUT METADATA (PRODUCT LABEL)</th><th>INPUT METADATA (SOURCE)</th></tr>
<tr><th>• Ops Keyword
• PDS-Compliant Keyword
• Definition</th><th>• Valid Values (quoted)
• Type
• Units
• Keyword Location in Label</th><th>• Mode
• Metadata Field
• Type</th></tr>
<tr><td>Specifies a major planetary mission or project. A given planetary mission may be associated with one or more spacecraft.

Note that mosaics may contain more than one value in an array.</td><td>n/a

<u>Location</u>
IDENTIFICATION (Class</td><td>n/a</td></tr>
<tr><td><u>Ops Keyword</u>
MISSION_PHASE_NAME

<u>PDS Keyword</u>
same

<u>Definition</u>
Specifies the commonly-used identifier of a mission phase.</td><td><u>Valid Values</u>
“DEVELOPMENT”, “LAUNCH”, “CRUISE AND APPROACH”, “ENTRY DESCENT AND LANDING”, “PRIMARY SURFACE MISSION”, “EXTENDED SURFACE MISSION”, “TEST”

<u>Type</u>
string(30)

<u>Units</u>
n/a

<u>Location</u>
IDENTIFICATION (Class)</td><td><u>Mode</u>
User specified parameter value

<u>Field as “<xml name>:[<element>]:[<element>]:<field>”</u>
n/a

<u>Type</u>
n/a</td></tr>
<tr><td><u>Ops Keyword</u>
MODEL_COMPONENT_1

<u>PDS Keyword</u>
same

<u>Definition</u>
Specifies a set of values representing the first component of a model. The significance (or meaning) of this array of values is indicated by the first value of the MODEL_COMPONENT_ID and/or MODEL_COMPONENT_NAME elements. The interpretation of the values themselves depends on the model but they commonly represent a vector, a set of polynomial coefficients, or a simple numeric parameter.

For example, for a geometric camera model with a value of “CAHV” for MODEL_TYPE, the first value of the MODEL_COMPONENT_NAME data element is CENTER, meaning that the MODEL_COMPONENT_1 is a focal center vector. The three items in this vector provide X, Y, and Z coordinates of the focal point of the camera.

The exact details about each model component vector are provided in MODEL_DESC.</td><td><u>Valid Values</u>
n/a

<u>Type</u>
float array

<u>Units</u>
n/a

<u>Location</u>
GEOMETRIC_CAMERA_MODEL (Group)</td><td><u>Mode</u>
DPO in XML format (referenced to APID Name in Appendix E)

<u>Field as “<xml name>:[<element>]:[<element>]:<field>”</u>
• <u>Eng. Cameras</u>
“<IDPH DPO>:idph:cmod_c[3]”
• <u>MMM Cameras</u>
“<IDPH DPO>:cidph:camera_model:c[0]”,
“<IDPH DPO>:cidph:camera_model:c[1]”,
“<IDPH DPO>:cidph:camera_model:c[2]”

<u>Type</u>
F64[3]</td></tr>
<tr><td><u>Ops Keyword</u>
MODEL_COMPONENT_2

<u>PDS Keyword</u>
same</td><td><u>Valid Values</u>
n/a

<u>Type</u>
float array</td><td><u>Mode</u>
DPO in XML format (referenced to APID Name in Appendix E)

<u>Field as “<xml name>:[<element>]:[<element>]:<field>”</u>
• <u>Eng. Cameras</u></td></tr>
</table>

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
Definition Specifies the value of the component of the MODEL_COMPONENT_ID for the second element.	**Units** n/a **Location** GEOMETRIC_CAMERA_MODEL (Group)	"<IDPH DPO>:idph:cmod_a[3]" • MMM Cameras "<IDPH DPO>:cidph:camera_model:a[0]", "<IDPH DPO>:cidph:camera_model:a[1]", "<IDPH DPO>:cidph:camera_model:a[2]" **Type** F64[3]
Ops Keyword MODEL_COMPONENT_3 **PDS Keyword** same **Definition** Specifies the value of the component of the MODEL_COMPONENT_ID for the third element.	**Valid Values** n/a **Type** float array **Units** n/a **Location** GEOMETRIC_CAMERA_MODEL (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** • Eng. Cameras "<IDPH DPO>:idph:cmod_h[3]" • MMM Cameras "<IDPH DPO>:cidph:camera_model:h[0]", "<IDPH DPO>:cidph:camera_model:h[1]", "<IDPH DPO>:cidph:camera_model:h[2]" **Type** F64[3]
Ops Keyword MODEL_COMPONENT_4 **PDS Keyword** same **Definition** Specifies the value of the component of the MODEL_COMPONENT_ID for the fourth element.	**Valid Values** n/a **Type** float array **Units** n/a **Location** GEOMETRIC_CAMERA_MODEL (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** • Eng. Cameras "<IDPH DPO>:idph:cmod_v[3]" • MMM Cameras "<IDPH DPO>:cidph:camera_model:v[0]", "<IDPH DPO>:cidph:camera_model:v[1]", "<IDPH DPO>:cidph:camera_model:v[2]" **Type** F64[3]
Ops Keyword MODEL_COMPONENT_5 **PDS Keyword** same **Definition** Specifies the value of the component of the MODEL_COMPONENT_ID for the fifth element.	**Valid Values** n/a **Type** float array **Units** n/a **Location** GEOMETRIC_CAMERA_MODEL (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** • Eng. Cameras "<IDPH DPO>:idph:cmod_o[3]" • MMM Cameras "<IDPH DPO>:cidph:camera_model:o[0]", "<IDPH DPO>:cidph:camera_model:o[1]", "<IDPH DPO>:cidph:camera_model:o[2]" **Type** F64[3]

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
Ops Keyword MODEL_COMPONENT_6 **PDS Keyword** same **Definition** Specifies the value of the component of the MODEL_COMPONENT_ID for the sixth element.	**Valid Values** n/a **Type** float array **Units** n/a **Location** GEOMETRIC_CAMERA_MODEL (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** • Eng. Cameras "<IDPH DPO>:idph:cmod_r[3]" • MMM Cameras "<IDPH DPO>:cidph:camera_model:r[0]", "<IDPH DPO>:cidph:camera_model:r[1]", "<IDPH DPO>:cidph:camera_model:r[2]" **Type** F64[3]
Ops Keyword MODEL_COMPONENT_7 **PDS Keyword** same **Definition** Specifies the value of the component of the MODEL_COMPONENT_ID for the seventh element.	**Valid Values** n/a **Type** float array **Units** n/a **Location** GEOMETRIC_CAMERA_MODEL (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** • Eng. Cameras "<IDPH DPO>:idph:cmod_e[3]" • MMM Cameras "<IDPH DPO>:cidph:camera_model:e[0]", "<IDPH DPO>:cidph:camera_model:e[1]", "<IDPH DPO>:cidph:camera_model:e[2]" **Type** F64[3]
Ops Keyword MODEL_COMPONENT_8 **PDS Keyword** same **Definition** Specifies the value of the component of the MODEL_COMPONENT_ID for the eighth element.	**Valid Values** n/a **Type** float array **Units** n/a **Location** GEOMETRIC_CAMERA_MODEL (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** • Eng. Cameras "<IDPH DPO>:idph:cmod_mtype" • MMM Cameras "<IDPH DPO>:cidph:camera_model:mtype" **Type** U8
Ops Keyword MODEL_COMPONENT_9 **PDS Keyword** same **Definition** Specifies the value of the component of the MODEL_COMPONENT_ID for the ninth element.	**Valid Values** n/a **Type** float array **Units** n/a **Location**	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** • Eng. Cameras "<IDPH DPO>:idph:cmod_mparm" • MMM Cameras "<IDPH DPO>:cidph:camera_model:mparm" **Type**

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
	GEOMETRIC_CAMERA_MODEL (Group)	F64
Ops Keyword MODEL_COMPONENT_ID **PDS Keyword** same **Definition** Specifies a sequence of identifiers (usually 1 character), where each identifier corresponds to a model component vector. It is used in conjunction with the MODEL_COMPONENT_n elements, where "n" is a number. The first id in the sequence corresponds to MODEL_COMPONENT_1, the second corresponds to MODEL_COMPONENT_2, etc. For example, for a geometric camera model with a value of "CAHV" for MODEL_TYPE, the MODEL_COMPONENT_ID would be (C, A, H, V). Please see the MODEL_COMPONENT_NAME data element for more details.	**Valid Values** 0 = "NONE" 1 = "(C,A,H,V)" 2 = "(C,A,H,V,O,R)" 3 = "(C,A,H,V,O,R,E,T,P)" **Type** string array **Units** n/a **Location** GEOMETRIC_CAMERA_MODEL (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** • Eng. Cameras "<IDPH DPO>:idph:cmod_mclass" • MMM Cameras "<IDPH DPO>:cidph:camera_model:mclass" **Type** enum
Ops Keyword MODEL_COMPONENT_NAME **PDS Keyword** same **Definition** Specifies a sequence of names, where each name identifies its corresponding model component vector. It is used in conjunction with the MODEL_COMPONENT_n elements, where "n" is a number. The first name in the sequence identifies MODEL_COMPONENT_1, the second identifies the MODEL_COMPONENT_2, etc. For example, for a geometric camera model with "CAHV" for MODEL_TYPE, the value for MODEL_COMPONENT_NAME would be (CENTER, AXIS, HORIZONTAL, VERTICAL). The three values of MODEL_COMPONENT_1 would describe the focal center vector; the three values of MODEL_COMPONENT_2 would describe the pointing direction (axis) vector; the three values of MODEL_COMPONENT_3 would describe the horizontal image plane vector, and the three values of the MODEL_COMPONENT_4 would describe the vertical image plane vector.	**Valid Values** 0 = "NONE" 1 = ("CENTER", "AXIS", "HORIZONTAL", "VERTICAL") 2 = ("CENTER", "AXIS", "HORIZONTAL", "VERTICAL", "OPTICAL", "RADIAL") 3 = ("CENTER", "AXIS", "HORIZONTAL", "VERTICAL", "OPTICAL", "RADIAL", "ENTRANCE", "MTYPE", "MPARM") **Type** string array **Units** n/a **Location** GEOMETRIC_CAMERA_MODEL (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** • Eng. Cameras "<IDPH DPO>:idph:cmod_mclass" • MMM Cameras "<IDPH DPO>:cidph:camera_model:mclass" **Type** enum
Ops Keyword ^MODEL_DESC **PDS Keyword** same	**Valid Values** "GEOMETRIC_CM.TXT" **Type** string	**Mode** Static Value **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
Definition Specifies a textual description of a model (or a pointer to a file containing the description). This is not intended to be a brief summary, but rather a detailed description of the model; at minimum, it should include a reference to a detailed description of the model in published literature. While other data elements such as CALIBRATION_SOURCE_ID, SOLUTION_ID, REFERENCE_COORD_SYSTEM_NAME, and MODEL_COMPONENT_NAME provide quick identifiers that distinguish how this model was generated, the details and data behind each of these identifiers should be explicitly included in the model description.	**Units** n/a **Location** GEOMETRIC_CAMERA_MODEL (Group)	**Type** n/a
Ops Keyword MODEL_TRANSFORM_QUATERNION **PDS Keyword** MSL:MODEL_TRANSFORM_QUATERNION **Definition** Specifies, along with MODEL_TRANSFORM_VECTOR, the transform used for the camera model in this image. Camera models created by the calibration process have associated with them a pose, comprised of the position (offset) and orientation (quaternion) of the camera at the time it was calibrated. The model is transformed ("pointed") for a specific image by computing, generally using articulation device kinematics, a final pose for the image. The camera model is then translated and rotated from the calibration to final pose. This keyword specifies the quaternion portion of the final pose. NOTE: Due to a flight software bug, the above description does not apply to early images. Prior to FSW version 10.6, these fields contained not the final pose, but the calibration pose. This discrepancy is reflected in the PDS labels. If keyword FLIGHT_SOFTWARE_VERSION_ID contains a value less than or equal to 141801503 (which corresponds to FSW 10.5.7), this keyword contains the calibration pose. The final pose can be computed for mast-mounted instruments using pointing correction procedures. Reverse-engineering the final value may also be possible for non-mast instruments given the calibration value and the camera	**Valid Values** n/a **Type** float array **Units** n/a **Location** GEOMETRIC_CAMERA_MODEL (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** • Eng. Cameras & Chemcam RMI "<IDPH DPO>:idph:cmod_trans:quaternion[3]", "<IDPH DPO>:idph:cmod_trans:quaternion[0]", "<IDPH DPO>:idph:cmod_trans:quaternion[1]", "<IDPH DPO>:idph:cmod_trans:quaternion[2]" • MMM Cameras "<IDPH DPO>:cidph:transform:quaternion[3]", "<IDPH DPO>:cidph:transform:quaternion[0]", "<IDPH DPO>:cidph:transform:quaternion[1]", "<IDPH DPO>:cidph:transform:quaternion[2]" **Type** F32[4]

<table>
<tr><th colspan="2">OUTPUT METADATA (PRODUCT LABEL)</th><th>INPUT METADATA (SOURCE)</th></tr>
<tr><th>• Ops Keyword
• PDS-Compliant Keyword
• Definition</th><th>• Valid Values (quoted)
• Type
• Units
• Keyword Location in Label</th><th>• Mode
• Metadata Field
• Type</th></tr>
<tr><td>model (or vice-versa). See keyword POINTING_MODEL_NAME.</td><td></td><td></td></tr>
<tr><td>Ops Keyword
MODEL_TRANSFORM_VECTOR

PDS Keyword
MSL:MODEL_TRANSFORM_VECTOR

Definition
Specifies, along with MODEL_TRANSFORM_QUATERNION, the transform used for the camera model in this image. Camera models created by the calibration process have associated with them a pose, comprised of the position (offset) and orientation (quaternion) of the camera at the time it was calibrated. The model is transformed ("pointed") for a specific image by computing, generally using articulation device kinematics, a final pose for the image. The camera model is then translated and rotated from the calibration to final pose. This keyword specifies the offset portion of the final pose.

NOTE: Due to a flight software bug, the above description does not apply to early images. Prior to FSW version 10.6, these fields contained not the final pose, but the calibration pose. This discrepancy is reflected in the PDS labels. If keyword FLIGHT_SOFTWARE_VERSION_ID contains a value less than or equal to 141801503 (which corresponds to FSW 10.5.7), this keyword contains the calibration pose. The final pose can be computed for mast-mounted instruments using pointing correction procedures. Reverse-engineering the final value may also be possible for non-mast instruments given the calibration value and the camera model (or vice-versa). See keyword POINTING_MODEL_NAME.</td><td>Valid Values
n/a

Type
float array

Units
n/a

Location
GEOMETRIC_CAMERA_MODEL (Group)</td><td>Mode
DPO in XML format (referenced to APID Name in Appendix E)

Field as “<xml name>:[<element>]:[<element>]:<field>”
• Eng. Cameras & Chemcam RMI
“<IDPH DPO>:idph:cmod_trans:position[0]”,
“<IDPH DPO>:idph:cmod_trans:position[1]”,
“<IDPH DPO>:idph:cmod_trans:position[2]”
• MMM Cameras
“<IDPH DPO>:cidph:transform:position[0]”,
“<IDPH DPO>:cidph:transform:position[1]”,
“<IDPH DPO>:cidph:transform:position[2]”

Type
F32[3]</td></tr>
<tr><td>Ops Keyword
MODEL_TYPE

PDS Keyword
same

Definition
Specifies an identifier for the type or kind of model. The value</td><td>Valid Values
0 = “NONE”
1 = “CAHV”
2 = “CAHVOR"
3 = “CAHVORE”

Type
string(63)</td><td>Mode
DPO in XML format (referenced to APID Name in Appendix E)

Field as “<xml name>:[<element>]:[<element>]:<field>”
• Eng. Cameras
“<IDPH DPO>:idph:cmod_mclass”
• MMM Cameras</td></tr>
</table>

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
should be one of a well defined set, providing an application program with sufficient information to know how to handle the rest of the parameters within the model. (CAHVORE is the only one that uses model component vectors 1-9.) For details on the definitions of the valid camera model types, see [Ref 21] through [Ref 27].	**Units** n/a **Location** GEOMETRIC_CAMERA_MODEL (Group)	"<IDPH DPO>:cidph:camera_model:mclass" **Type** enum
Ops Keyword ^MOSAIC_DESC **PDS Keyword** same **Definition** Specifies a brief textual description of a mosaic.	**Valid Values** n/a **Type** string **Units** n/a **Location** DERIVED_IMAGE_PARMS (Group)	**Mode** Static Value **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword N_SHOTS **PDS Keyword** MSL:N_SHOTS **Definition** For ChemCam on MSL, specifies the commanded number of laser shots for the LIBS instrument. The LIBS is operated in a burst of shots, and this keyword specifies the total in the burst.	**Valid Values** n/a **Type** integer **Units** n/a **Location** OBSERVATION_REQUEST_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** "<Ancillary DPO>:cmd_arguments:nshots" **Type** U8
Ops Keyword N_SHOTS_2_AVG **PDS Keyword** MSL:N_SHOTS_2_AVG **Definition** For ChemCam on MSL, specifies the number of groups of commanded laser shots for the LIBS instrument. The LIBS is operated in a burst of shots where the total number of shots is specified as N_SHOTS. However, the spectra gathered from each of these shots can be averaged in groups of shots specified in this keyword.	**Valid Values** n/a **Type** integer **Units** n/a **Location** OBSERVATION_REQUEST_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** "<Ancillary DPO>:cmd_arguments:nshots2avg" **Type** U8
Ops Keyword N_SHOTS_2_IGNORE **PDS Keyword** MSL:N_SHOTS_2_IGNORE	**Valid Values** n/a **Type** integer	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** "<Ancillary DPO>:cmd_arguments:nshots2ignor"

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
Definition For ChemCam on MSL, specifies the number of commanded laser shots for the LIBS instrument to ignore when gathering the spectra. The LIBS is operated in a burst of shots where the total number of shots is specified as N_SHOTS. The spectra collected during the first few shots can be ignored if a non-zero value is specified in this keyword.	**Units** n/a **Location** OBSERVATION_REQUEST_PARMS (Group)	**Type** U8
Ops Keyword NUM_SOFTWARE_KEYWORDS **PDS Keyword** MSL:NUM_SOFTWARE_KEYWORDS **Definition** Specifies the number of keywords that were supplied to the primary generating software named in SOFTWARE_MODULE_NAME. The PDS keywords that contain the software keyword names, values and types (if any) will be provided in SOFTWARE_KEYWORD_NAME, SOFTWARE_KEYWORD_TYPE, and SOFTWARE_KEYWORD_VALUE and are required for each software keyword supplied. For MSL, this keyword may be placed in a xxx_HISTORY_PARMS group, where "xxx" is the task name. There may be multiple groups to accommodate multiple software tasks.	**Valid Values** n/a **Type** integer **Units** n/a **Location** <xxx>_HISTORY_PARMS (Group) where, xxx = RDR software name	**Mode** RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword NUM_SOFTWARE_PARAMETERS **PDS Keyword** MSL:NUM_SOFTWARE_PARAMETERS **Definition** Specifies the number of parameters that were supplied to the primary generating software named in SOFTWARE_MODULE_NAME. The PDS keywords that contain the software parameter names, values and types (if any) will be provided in SOFTWARE_PARAMETER_NAME, SOFTWARE_PARAMETER_TYPE, and SOFTWARE_PARAMETER_VALUE and are required for each software keyword supplied. For MSL, this keyword may be placed in a xxx_HISTORY_PARMS group, where "xxx" is the task name. There may be multiple groups to accommodate multiple	**Valid Values** n/a **Type** integer **Units** n/a **Location** <xxx>_HISTORY_PARMS (Group) where, xxx = RDR software name	**Mode** RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
software tasks.		
Ops Keyword OBS_FROM_LIMIT_SWITCH **PDS Keyword** MSL:OBS_FROM_LIMIT_SWITCH **Definition** For ChemCam on MSL, when set to "1", specifies that the mast unit (MU) focus motor move to the limit switch position before moving to the specified focus position in the MANUAL or BASELINE focus options. For the AF_OFFSET focus option, it is ignored.	**Valid Values** n/a **Type** integer **Units** n/a **Location** OBSERVATION_REQUEST_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** "<Ancillary DPO>:cmd_parameters:obsFromLimitSwitch" **Type** U8
Ops Keyword OBSERVATION_ID **PDS Keyword** same **Definition** Specifies a unique identifier for a scientific observation within a data set. It is set via the data product context ID - which doesn't necessarily map to a specific object - it's just used to group various instrument data sets together via a common keyword.	**Valid Values** "UNK" **Type** string **Units** n/a **Location** IDENTIFICATION (Class)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword ODL_VERSION_ID **PDS Keyword** same **Definition** Specifies the version number of the PDS standards document used for an ODL format file. ODL3 files conform to the PDS3 standards document structurally and syntactically, the only difference being that keywords do not necessarily need to be defined in a PDS Data Dictionary Examples: ODL3	**Valid Values** "ODL<*version*>" **Type** string[6] **Units** n/a **Location** IDENTIFICATION (Class)	**Mode** PDS **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword OFFSET_MODE_ID **PDS Keyword** same **Definition** Specifies the analog value that is subtracted from the video signal prior to the analog/digital converters.	**Valid Values** • Eng. Cameras 1) "0" to "4095" • Chemcam 2) "0" to "4095" • MMM Cameras 3) a. "UNK" b. "0" to "4095"	**Mode** 1) DPO in XML format (referenced to APID Name in Appendix E) 2) DPO in XML format (referenced to APID Name in Appendix E) 3) Image DPO mini-header **Field as "<xml name>:[<element>]:[<element>]:<field>"** • Eng. Cameras 1) "<IDPH DPO>:idph:voff" • Chemcam

OUTPUT METADATA (PRODUCT LABEL)		**INPUT METADATA (SOURCE)**
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
For MSL, the value is a digital number (DN) which represents the offset used in a camera-specific manner. For MSL engineering cameras, this value is the video offset, and has a range 0-4095. For MSL ChemCam RMI, it specifies the commanded analog offset in the ChemCam Mast Unit (CCMU), in the range 0-255. See also GAIN_NUMBER. For MSL MMM cameras, it is the DC Offset as reported in the MMM mini-header.	**Type** string **Units** n/a **Location** 1) INSTRUMENT_STATE_PARMS (Group) 2) OBSERVATION_REQUEST_PARMS (Group) 3) a. INSTRUMENT_STATE_PARMS (Group) b. MINI_HEADER (Group)	2) "<Ancillary DPO>:cmd_parameters:AD_offset" • MMM Cameras 3) "MMM_Image_Mini_Header[52]", "MMM_Image_Mini_Header[53]", "MMM_Image_Mini_Header[54]", "MMM_Image_Mini_Header[55]" NOTES: • For MMM Cameras, value analogous to "DC_offset" is comprised of four bytes coming from Image DPO mini-header at byte offsets 52, 53, 54 and 55. **Type** 1) U16 2) U8 3) U8
Ops Keyword ORIGIN_OFFSET_VECTOR **PDS Keyword** same **Definition** Specifies the offset from the reference coordinate system's origin to the origin of the coordinate system being defined by the enclosing COORDINATE_SYSTEM group. In other words, it is the location of the current system's origin as measured in the reference system. For MSL, here is an example: In the case of the RSM_COORDINATE_SYSTEM group, ORIGIN_OFFSET_VECTOR describes the rotation of the RSM (camera head) boresight (about the ORIGIN_OFFSET_VECTOR) relative to the Rover frame.	**Valid Values** n/a **Type** float array[3] **Units** meters **Location** • For EDRs … c. ROVER_COORDINATE_SYSTEM (Group) d. RSM_COORDINATE_SYSTEM (Group) e. ARM_COORDINATE_SYSTEM (Group) f. INITIAL_STATE_PARMS (Group) • For RDRs … a. SITE_COORDINATE_SYSTEM (Group) b. LOCAL_LEVEL_COORDINATE_SYSTEM (Group) c. ROVER_COORDINATE_SYSTEM (Group) d. RSM_COORDINATE_SYSTEM (Group) e. ARM_COORDINATE_SYSTEM (Group)	**Mode** • Calculation: 1) a. PLACES (OPGS rover localization database) b. PLACES (OPGS rover localization database) • DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** • Eng. Cameras 2) c. "<IDPH DPO>:idph:rvr_p[3]" d. "<IDPH DPO>:idph:rsm_p[3]" e. "<IDPH DPO>:idph:arm_pos[3]" f. "<IDPH DPO>:idph:rvr_p[3]" • MMM Cameras 3) c. "<IDPH DPO>:cidph:rvr:position[0]", "<IDPH DPO>:cidph:rvr:position[1]", "<IDPH DPO>:cidph:rvr:position[2]" d. "<IDPH DPO>:cidph:rsm_state:position[0]", "<IDPH DPO>:cidph:rsm_state:position[1]", "<IDPH DPO>:cidph:rsm_state:position[2]" e. "<IDPH DPO>:cidph:arm_state:pos[0]", "<IDPH DPO>:cidph:arm_state:pos[1]", "<IDPH DPO>:cidph:arm_state:pos[2]" **Type** 1) U16[3] 2) U16[3] 3) F32
Ops Keyword ORIGIN_ROTATION_QUATERNION **PDS Keyword** same	**Valid Values** n/a **Type** float array[4]	**Mode** • Calculation: 1) PLACES (OPGS rover localization database) 2) PLACES (OPGS rover localization database) • DPO in XML format (referenced to APID Name in Appendix E)

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
Definition Specifies an array of four values that specifies the rotation of the coordinate system being defined by the enclosing COORDINATE_SYSTEM group, relative to the reference system. Mathematically this can be expressed as follows: Given a vector expressed in the current frame, multiplication by this quaternion will give the same vector as expressed in the reference frame. Quaternions are expressed as a set of four numbers in the order: (s, v1, v2, v3) where, s = cos(theta/2) v(n) = sin(theta/2)*a(n). theta = the angle of rotation a = (x,y,z) vector around which rotation occurs Note that quaternions have different component order conventions between flight and ground software. They are received in the order "(v1, v2, v3, s)". However, the ground order convention is "(s, v1, v2, v3)", and all values are converted to the ground order before being stored in the label. For MSL, the value for ORIGIN_ROTATION_QUATERNION that defines a coordinate frame like Rover frame is computed with respect to only the orientations of the frame's axes… regardless of whether POSITIVE_ELEVATION_DIRECTION is declared to be "UP" or "DOWN". For MSL, here is an example: In the case of the RSM_COORDINATE_SYSTEM group, ORIGIN_ROTATION_QUATERNION describes the rotation of the RSM (camera head on Mast) boresight (about the ORIGIN_OFFSET_VECTOR) relative to the Rover frame.	**Units** meters **Location** • For EDRs … 3) ROVER_COORDINATE_SYSTEM (Group) 4) RSM_COORDINATE_SYSTEM (Group) 5) ARM_COORDINATE_SYSTEM (Group) 6) INITIAL_STATE_PARMS (Group) • For RDRs … 1) SITE_COORDINATE_SYSTEM (Group) 2) LOCAL_LEVEL_COORDINATE_SYSTEM (Group) 3) ROVER_COORDINATE_SYSTEM (Group) 4) RSM_COORDINATE_SYSTEM (Group) 5) ARM_COORDINATE_SYSTEM (Group)	**Field as "<xml name>:[<element>]:[<element>]:<field>"** • Eng. Cameras 3) a. "<IDPH DPO>:idph:rvr_q[4]" 4) a. "<IDPH DPO>:idph:rsm_q[4]" 5) a. "<IDPH DPO>:idph:arm_quat[4]" 6) "<IDPH DPO>:idph:rvr_q[4]" • MMM Cameras 3) b. "<IDPH DPO>:cidph:rvr:quaternion[3]", "<IDPH DPO>:cidph:rvr:quaternion[0]", "<IDPH DPO>:cidph:rvr:quaternion[1]", "<IDPH DPO>:cidph:rvr:quaternion[2]" 4) b. "<IDPH DPO>:cidph:rsm_state:quaternion[3]", "<IDPH DPO>:cidph:rsm_state:quaternion[0]", "<IDPH DPO>:cidph:rsm_state:quaternion[1]", "<IDPH DPO>:cidph:rsm_state:quaternion[2]" 5) b. "<IDPH DPO>:cidph:arm_state:quat[3]", "<IDPH DPO>:cidph:arm_state:quat[0]", "<IDPH DPO>:cidph:arm_state:quat[1]", "<IDPH DPO>:cidph:arm_state:quat[2]" **Type** 1) F32[4] 2) F32[4] 3) a. F32[4] b. F32 4) a. F32[4] b. F32 5) a. F32[4] b. F32
Ops Keyword PDS_VERSION_ID **PDS Keyword** same **Definition** Specifies the version number of the PDS standards document that is valid when a data product label is created. Values for the PDS_version_id are formed by appending the integer for the latest version number to the letters 'PDS'. Examples: PDS3	**Valid Values** "PDS<*version*>" **Type** string[6] **Units** n/a **Location** IDENTIFICATION (Class)	**Mode** PDS **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword	**Valid Values**	**Mode**

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
PIXEL_AVERAGING_HEIGHT **PDS Keyword** same **Definition** Specifies the vertical dimension, in pixels, of the area over which pixels were averaged prior to image compression.	• Eng. Cameras 1) "0" to "1024" • MMM Cameras 2) a. "UNK" b. non-Thumbnail = "1" Thumbnail = "8" **Type** integer **Units** pixel **Location** 1) a. IMAGE_REQUEST_PARMS (Group) b. INSTRUMENT_STATE_PARMS (Group) 2) a. IMAGE_REQUEST_PARMS (Group) b. INSTRUMENT_STATE_PARMS (Group)	1) DPO in XML format (referenced to APID Name in Appendix E) 2) EMD in XML format, calculation **Field as "<xml name>:[<element>]:[<element>]:<field>"** • Eng. Cameras 1) a. "<IDPH DPO>:idph:params:res_rows" b. "<IDPH DPO>:idph:res_rows" • MMM Cameras 2) "MslEarthProductMetadata:MslProductMetadata: ProductName" NOTES: • For MMM Cameras, if product type is Thumbnail, set value to "8". Otherwise, set value to "1". **Type** U16
Ops Keyword PIXEL_AVERAGING_WIDTH **PDS Keyword** same **Definition** Specifies the horizontal dimension, in pixels, of the area over which pixels were averaged prior to image compression.	**Valid Values** • Eng. Cameras 1) "0" to "1024" • MMM Cameras 2) a. "UNK" b. non-Thumbnail = "1" Thumbnail = "8" **Type** integer **Units** pixel **Location** 1) a. IMAGE_REQUEST_PARMS (Group) b. INSTRUMENT_STATE_PARMS (Group) 2) a. IMAGE_REQUEST_PARMS (Group) b. INSTRUMENT_STATE_PARMS (Group)	**Mode** 1) DPO in XML format (referenced to APID Name in Appendix E) 2) EMD in XML format, calculation **Field as "<xml name>:[<element>]:[<element>]:<field>"** • Eng. Cameras 1) a. "<IDPH DPO>:idph:params:res_cols" b. "<IDPH DPO>:idph:res_cols" • MMM Cameras 2) "MslEarthProductMetadata:MslProductMetadata: ProductName" NOTES: • For MMM Cameras, if product type is Thumbnail, set value to "8". Otherwise, set value to "1". **Type** U16
Ops Keyword PIXEL_DOWNSAMPLE_OPTION **PDS Keyword** same **Definition** Specifies whether to downsample the image(s), and if so, which pixel resolution downsample method to use. Note that for MSL, the camera hardware could downsample	**Valid Values** • Eng. Cameras 0 = "NONE" 1 = "SW_MEAN" 2 = "HW_COND" 3 = "HW_SW" 4 = "SW_OUTREJ" 5 = "SW_MEDIAN" • MMM Cameras "UNK"	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** Eng. Cameras "<IDPH DPO>:idph:params:resolution" **Type** enum

<table>
<tr><th colspan="2">OUTPUT METADATA (PRODUCT LABEL)</th><th>INPUT METADATA (SOURCE)</th></tr>
<tr><th>• Ops Keyword
• PDS-Compliant Keyword
• Definition</th><th>• Valid Values (quoted)
• Type
• Units
• Keyword Location in Label</th><th>• Mode
• Metadata Field
• Type</th></tr>
<tr><td>entire rows 4-to-1, but software had to be used to do additional row-wise downsampling and any column downsampling.

For MSL, the meaning of the valid values follows:
a) “SW_MEAN” - Downsampling done in software by calculation of the mean.
b) “HW_COND” - Use hardware binning if downsampling (by mean calculation) and subframe arguments are consistent.
c) “HW_SW” - Use hardware binning by changing the commanded downsampling and subframe arguments to be consistent with hardware binning. Any subsequent downsampling is done in software by calculation of the mean.
d) “SW_OUTREJ” -Software pixel averaging with outlier rejection. The pixel whose value lies farthest away from the mean of the sample is rejected.
e) “SW_MEDIAN” - Software downsampling done by calculation of the median rather than the mean.</td><td>Type
string

Units
n/a

Location
IMAGE_REQUEST_PARMS (Group)</td><td></td></tr>
<tr><td>Ops Keyword
PLANET_DAY_NUMBER

PDS Keyword
same

Definition
Specifies the number of solar days elapsed since a reference day (e.g., the day on which a landing vehicle set down) for local mean solar time (LMST). Days are measured in rotations of the planet in question from midnight to midnight.

For MSL, the reference day is “0”, as Landing day is Sol 0. If before Landing day, value will be less than or equal to “0”.

See also LOCAL_MEAN_SOLAR_TIME and LOCAL_TRUE_SOLAR_TIME_SOL.</td><td>Valid Values
Mission Phase — Values
Cruise — less than or equal to 0
Surface — “0” to n

NOTE: Value will be uncalibrated if SPICE kernels are unavailable.

Type
integer

Units
n/a

Location
IDENTIFICATION (Class)</td><td>Mode
1) a. Calculation:
- SCLK Kernel
b. DPO in XML format (referenced to APID Name in Appendix E)
2) a. Image DPO mini-header, Calculation:
- SCLK Kernel
b. DPO in XML format (referenced to APID Name in Appendix E)

Field as “<xml name>:[<element>]:[<element>]:<field>”
• Eng. Cameras
1) b. “<IDPH DPO>:idph:sclk_seconds”, “<IDPH DPO>:idph:sclk_subseconds”
• MMM Cameras
2) a. “MMM_Image_Mini_Header[8]”, “MMM_Image_Mini_Header[9]”, “MMM_Image_Mini_Header[10]”, “MMM_Image_Mini_Header[11]”
b. “<IDPH DPO>:cidph:sclk:seconds”
c. “<Ancillary DPO>:sclk:seconds”

NOTES:
• For MMM recovered products, value is comprised of four bytes coming from Image DPO mini-header at byte offsets 8, 9, 10 and 11 per case “2a”.
• For MMM non-recovered data products, if IDPH is present, value comes from IDPH DPO per case “2b”.
• For MMM non-recovered data products, if IDPH is not present, value comes from Ancillary DPO per case “2c”.

Type</td></tr>
</table>

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
		U32
Ops Keyword POINTING_CORRECTION_FILE_NAME **PDS Keyword** MSL:POINTING_CORRECTION_FILE_NAME **Definition** Specifies the name of the file containing pointing correction parameters for each input file of a mosaic. The file is in XML format, and is often called a "nav" file in operations. See also SOLUTION_ID.	**Valid Values** n/a **Type** string **Units** n/a **Location** DERIVED_IMAGE_PARMS (Group)	**Mode** RDR-generating software **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a
Ops Keyword POINTING_MODEL_NAME **PDS Keyword** MSL:POINTING_MODEL_NAME **Definition** Specifies which of several "pointing models" were used to transform the camera model based on updated pointing information. These updates are typically derived from mosaic seam corrections. This keyword and the associated POINTING_MODEL_PARAMS keyword define what the updated pointing information is, providing enough information to re-create the camera model from calibration data. If present, this keyword overrides the default pointing based on telemetry. The special value "NONE" shall be interpreted the same as if the keyword is absent (i.e. the default pointing model should be used). New model names can be created at any time; the models themselves should be described in the camera model ancillary file “msl_pointing_models.txt” included in the Archive Volume. See also SOLUTION_ID.	**Valid Values** “MSLmastCamera”, “MSLmastCamera3dof”, “MSLbodyFixedCamera”, “MSLmahliCamera”, “MSLmahliCamera6dof”, “NONE” **Type** string **Units** n/a **Location** DERIVED_IMAGE_PARMS (Group)	**Mode** RDR-generating software **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a
Ops Keyword POINTING_MODEL_PARAMS **PDS Keyword** MSL:POINTING_MODEL_PARAMS **Definition** Specifies the numeric parameters needed by the pointing model identified in POINTING_MODEL_NAME. The meaning of any given parameter is defined by the pointing model.	**Valid Values** n/a **Type** float **Units** n/a **Location**	**Mode** RDR-generating software **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
	DERIVED_IMAGE_PARMS (Group)	
Ops Keyword POINTING_MODEL_PARAMS_NAME **PDS Keyword** MSL:POINTING_MODEL_PARAMS_NAME **Definition** This keyword specifies the name assigned by the pointing model for each parameter specified in POINTING_MODEL_PARAMS. For example, for the MSL model "MSLmastCamera3dof" the values would be "AZIMUTH", "ELEVATION", "TWIST" in that order.	**Valid Values** Pointing Model — Parameter Name Values (in order) MSLmastCamera — (“AZIMUTH”, “ELEVATION”) MSLmastCamera3dof — (“AZIMUTH”, “ELEVATION”, “TWIST”) MSLbodyFixedCamera — none MSLmahliCamera — (“S”, “V1”, “V2”, “V3”, “X”, “Y”, “Z”) MSLmahliCamera6dof — (“AZIMUTH”, “ELEVATION”, “TWIST”, “X”, “Y”, “Z”) **Type** string array **Units** n/a **Location** DERIVED_IMAGE_PARMS (Group)	**Mode** RDR-generating software **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a
Ops Keyword POSITIVE_AZIMUTH_DIRECTION **PDS Keyword** same **Definition** Specifies the direction in which azimuth is measured in positive degrees for an observer on the surface of a body. The azimuth is measured with respect to the elevational reference plane. A value of CW indicates that azimuth is measured positively Clockwise, and CCW indicates that azimuth increases positively Counter-clockwise. For the MSL operational coordinate frames, which follow the Mars Pathfinder convention, increasing azimuth moves in a clockwise (“CLOCKWISE”) direction as viewed from above. NOTE: For MSL, early in the mission the value for only the instance of keyword SOLAR_AZIMUTH in label group SITE_DERIVED_GEOMETRY_PARMS was being incorrectly reported as the inverse of what it should have been. For the Ops (ODL) label, the value is flagged “correct” if POSITIVE_AZIMUTH_DIRECTION resides in the SITE_DERIVED_GEOMETRY_PARMS label group. For the PDS label, the value is correct for all files as of archive Volume 2. The value for the instance of SOLAR_AZIMUTH in the label group ROVER_DERIVED_GEOMETRY_PARMS is always correct for the Ops and PDS labels.	**Valid Values** “CLOCKWISE” **Type** string **Units** n/a **Location** • For EDRs … - ROVER_COORDINATE_SYSTEM (Group) - RSM_COORDINATE_SYSTEM (Group) - ARM_COORDINATE_SYSTEM (Group) - SITE_DERIVED_GEOMETRY_PARMS (Group) • For RDRs … - ROVER_COORDINATE_SYSTEM (Group) - RSM_COORDINATE_SYSTEM (Group) - ARM_COORDINATE_SYSTEM (Group) - SITE_COORDINATE_SYSTEM (Group) - LOCAL_LEVEL_COORDINATE_SYSTEM (Group) - SITE_DERIVED_GEOMETRY_PARMS (Group)	**Mode** Static Value: - Determined by Coordinate Frame definitions **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a
Ops Keyword	**Valid Values**	**Mode**

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
POSITIVE_ELEVATION_DIRECTION **PDS Keyword** same **Definition** Specifies the direction in which elevation is measured in positive degrees for an observer on the surface of a body. The elevation is measured with respect to the azimuthal reference plane. A value of "UP" indicates that elevation is measured positively upwards, i.e., the zenith point would be at +90 degrees and the nadir point at -90 degrees. "DOWN" indicates that the elevation is measured positively downwards; the zenith point would be at -90 degrees and the nadir point at +90 degrees. For the MSL operational coordinate frames, which follow the Mars Pathfinder convention, increasing elevation ("UP") moves towards the negative Z axis.	"UP" **Type** string **Units** n/a **Location** • For EDRs … - ROVER_COORDINATE_SYSTEM (Group) - RSM_COORDINATE_SYSTEM (Group) - ARM_COORDINATE_SYSTEM (Group) • For RDRs … - ROVER_COORDINATE_SYSTEM (Group) - RSM_COORDINATE_SYSTEM (Group) - ARM_COORDINATE_SYSTEM (Group) - SITE_COORDINATE_SYSTEM (Group) - LOCAL_LEVEL_COORDINATE_SYSTEM (Group)	Static Value: - Determined by Coordinate Frame definitions **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword PROCESSING_HISTORY_TEXT **PDS Keyword** same **Definition** Specifies an entry for each processing step and program used in generating a particular data file.	**Valid Values** "CODMAC LEVEL 1 TO LEVEL 2 CONVERSION VIA JPL/MIPL MSLEDRGEN" **Type** string **Units** n/a **Location** • For EDRs … - MSLEDRGEN_HISTORY_PARMS (Group) • For RDRs … - <xxx>_HISTORY_PARMS (Group) where, xxx = RDR software name	**Mode** Dependent on EDR/RDR state: - For EDRs, static - For RDRs, RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword PROCESSING_INFO **PDS Keyword** same **Definition** Specifies information about the processing used to generate the RDR that is not covered by other PDS label keywords.	**Valid Values** n/a **Type** string array **Units** n/a **Location** DERIVED_IMAGE_PARMS (Group)	**Mode** RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword PRODUCER_INSTITUTION_NAME	**Valid Values** "MULTIMISSION INSTRUMENT PROCESSING LAB, JET	**Mode** Static Value

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
PDS Keyword same **Definition** Specifies the identity of a university, research center, NASA center or other institution associated with the production of a data set. This would generally be an institution associated with the element PRODUCER_FULL_NAME.	PROPULSION LAB" **Type** string(60) **Units** n/a **Location** IDENTIFICATION (Class)	**Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword PRODUCT_COMPLETION_STATUS **PDS Keyword** MSL:PRODUCT_COMPLETION_STATUS **Definition** Specifies the completion status of a product, specifying for example, if all portions have been downlinked and received correctly, if it is a partial product, or if it contains transmission errors. The specific valid values may be mission-dependent. For MSL, the valid values indicate whether it was a complete or partial product as it came out of MPCS, and whether the checksum passed, failed, or was missing. Note that partial products with missing metadata may not have standard filenames or label values (e.g. seq ID will default to "ACS_00000"). Once the full product is eventually downlinked, the ground product will be regenerated with the correct filename and label. Data sent to PDS Archives is the complete version available for that product.	**Valid Values** "PARTIAL", "PARTIAL_CHECKSUM_FAIL", "COMPLETE_CHECKSUM_PASS", "COMPLETE_NO_CHECKSUM", "COMPLETE_CHECKSUM_FAIL" **Type** string **Units** n/a **Location** TELEMETRY (Class)	**Mode** EMD in XML format **Field as "<xml name>:[<element>]:[<element>]:<field>"** "MslEarthProductMetadata:MslProductMetadata:GroundStatus" **Type** n/a
Ops Keyword PRODUCT_CREATION_TIME **PDS Keyword** same **Definition** Specifies the UTC system format for the time when a product was created.	**Valid Values** <YYYY>-<MM>-<DD>T<hh>:<mm>:<ss>[.<fff>] **Type** time **Units** n/a **Location** IDENTIFICATION (Class)	**Mode** Calculation **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a

Ops Keyword
PRODUCT_ID
 	
PDS Keyword
same
 	
Definition
	Valid Values
	n/a
	
	Type
	string(40)
	
	Units
		Mode
		• EDR-generating software:
		  - Filename of output EDR product, minus extension
		• RDR-generating software:
		  - Filename of output RDR product, minus extension
		
		Field as "<xml name>:[<element>]:[<element>]:<field>"

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
Specifies a permanent, unique identifier assigned to a data product by its producer. For MSL, it is the filename minus the extension. NOTES: In the PDS, the value assigned to product_id must be unique within its data set. The PRODUCT_ID can describe the lowest-level Data object that has a PDS label.	n/a **Location** IDENTIFICATION (Class)	n/a **Type** n/a
Ops Keyword PRODUCT_TAG **PDS Keyword** MSL:PRODUCT_TAG **Definition** Data Product Tag. NOTE: For MSL, use of this tag is defined separately for Individual product types. It is anticipated that this tag may be used to associate multiple products for later processing; it may also be used to indicate instrument FSW versions, or other uses.	**Valid Values** n/a **Type** string **Units** n/a **Location** TELEMETRY (Class)	**Mode** EMD in XML format **Field as “<xml name>:[<element>]:[<element>]:<field>”** “MslEarthProductMetadata:MslProductMetadata:ProductTag” **Type** n/a
Ops Keyword PRODUCT_VERSION_ID **PDS Keyword** same **Definition** Specifies the version of an individual product within a data set. PRODUCT_VERSION_ID is intended for use within AMMOS to identify separate iterations of a given product, which will also have a unique FILE_NAME. For MSL, this is a Version field that begins with “V” followed by a decimal number denoting the product’s iteration (i.e., version). Example: “V2.0” When appearing in the attached ODL (operations) label, keyword value represents the version incorporated in the Version field of the product’s filename. This does not correspond to the keyword value appearing in the detached PDS (archive) label. When appearing in the detached PDS (archive) label, keyword	**Valid Values** n/a **Type** string(12) **Units** n/a **Location** IDENTIFICATION (Class)	**Mode** User specified parameter value **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
value represents the product version within the Archive data set, which itself is identified by an element of DATA_SET_ID (note that the publicly released version of DATA_SET_ID is identified by RELEASE_ID).		
Ops Keyword PROJECTION_AXIS_OFFSET **PDS Keyword** MSL:PROJECTION_AXIS_OFFSET **Definition** Specifies an offset from a projection axis in a map projection. For the Cylindrical-Perspective projection, this is the radius of a circle which represents the rotation around the projection origin of the synthetic camera used to calculate each column.	**Valid Values** n/a **Type** float **Units** meters (<m> unit tag required) **Location** SURFACE_PROJECTION_PARMS (Group)	**Mode** RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword PROJECTION_AZIMUTH **PDS Keyword** same **Definition** Specifies the azimuth, in degrees, of the horizontal center of projection for the PERSPECTIVE projection (loosely, where the camera model is pointing).	**Valid Values** n/a **Type** float **Units** deg (<deg> unit tag required) **Location** SURFACE_PROJECTION_PARMS (Group)	**Mode** RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword PROJECTION_ELEVATION **PDS Keyword** same **Definition** Specifies the elevation, in degrees, of the vertical center of projection (loosely, where the camera is pointing). For Perspective projections, this applies to the single output camera model; for Cylindrical-Perspective it applies to each column's output camera model, before the rotation specified by PROJECTION_AXIS_Z_VECTOR.	**Valid Values** n/a **Type** float **Units** deg (<deg> unit tag required) **Location** SURFACE_PROJECTION_PARMS (Group)	**Mode** RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword PROJECTION_ELEVATION_LINE **PDS Keyword** same **Definition** Specifies the image line which corresponds to PROJECTION_ELEVATION for each column of the Cylindrical-	**Valid Values** n/a **Type** float **Units** pixel (<pixel> unit tag required)	**Mode** RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
Perspective projection, before the rotation specified by PROJECTION_AXIS_Z_VECTOR.	**Location** SURFACE_PROJECTION_PARMS (Group)	
Ops Keyword PROJECTION_ORIGIN_VECTOR **PDS Keyword** same **Definition** Specifies the location of the origin of the projection. For Polar and Cylindrical projections, this is the XYZ point from which all the azimuth/elevation rays emanate. For the Cylindrical-Perspective projection, this defines the center of the circle around which the synthetic camera orbits. For Orthographic, Orthorectified, and Vertical projections, this optional keyword specifies the point on the projection plane that serves as the origin of the projection (i.e. all points on a line through this point in the direction of PROJECTION_Z_AXIS_VECTOR will be located at X=Y=0 in the projection). If not present, (0,0,0) should be assumed. This translation is generally not necessary and not often used; the (X|Y)_AXIS_MINIMUM and (X|Y)_AXIS_MAXIMUM fields allow the mosaic to be located arbitrarily in the projection plane.	**Valid Values** n/a **Type** float array[3] **Units** meters (<m> unit tag required) **Location** SURFACE_PROJECTION_PARMS (Group)	**Mode** RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword PROJECTION_X_AXIS_VECTOR **PDS Keyword** MSL:PROJECTION_X_AXIS_VECTOR **Definition** Specifies a unit vector defining the X-axis for a given projection. For Orthographic, Orthorectified, and Vertical projections, this vector defines how the * axis in the mosaic is oriented in space. The X and Y axis vectors together define the rotation of the projection plane around the projection axis.	**Valid Values** n/a **Type** float array[3] **Units** n/a **Location** SURFACE_PROJECTION_PARMS (Group)	**Mode** RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword PROJECTION_Y_AXIS_VECTOR **PDS Keyword** MSL:PROJECTION_Y_AXIS_VECTOR **Definition** Specifies a unit vector defining the Y-axis for a given projection. For Orthographic, Orthorectified, and Vertical projections, this	**Valid Values** n/a **Type** float array[3] **Units** n/a **Location** SURFACE_PROJECTION_PARMS (Group)	**Mode** RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
vector defines how the * axis in the mosaic is oriented in space. The X and Y axis vectors together define the rotation of the projection plane around the projection axis.		
Ops Keyword PROJECTION_Z_AXIS_VECTOR **PDS Keyword** MSL:PROJECTION_Z_AXIS_VECTOR **Definition** Specifies a unit vector defining the Z axis for a given projection. For Orthographic, Orthorectified, and Vertical projections, this vector defines the projection axis for the mosaic. All points along a line parallel to this axis are projected to the same spot in the projection plane. For the Cylindrical-Perspective projections, this defines the new axis of the circle around which the synthetic camera orbits (i.e. the normal to the circle), after the cameras have been rotated to correct for rover tilt. CAMERA_ROTATION_AXIS_VECTOR contains the axis before rotation; the difference in these two indicate the rotation amount.	**Valid Values** n/a **Type** float array[3] **Units** n/a **Location** SURFACE_PROJECTION_PARMS (Group)	**Mode** RDR-generating software **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a
Ops Keyword QUATERNION_MEASUREMENT_METHOD **PDS Keyword** same **Definition** Specifies the quality of the rover orientation estimate. Valid values are: a) “UNKNOWN” - The attitude should simply not be trusted. This is the initial grade given on Landing, for example. b) “TILT_ONLY” - The attitude estimate is only good for tilt determination (2-axis knowledge). Activities which require azimuth knowledge should be careful. c) “FINE” - Sun identification or other attitude determination has completed successfully, and the attitude estimate is sufficient for pointing HGA (or “ThreeAxisFine”).	**Valid Values** • ALL Cameras 0 = “UNKNOWN” 1 = “TILT_ONLY” 2 = “FINE” • MMM Cameras only no IDPH = “UNK” **Type** string **Units** n/a **Location** • ROVER_COORDINATE_SYSTEM (Group) • INITIAL_STATE_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E), Table Lookup **Field as “<xml name>:[<element>]:[<element>]:<field>”** • Eng. Cameras 1) “<IDPH DPO>:idph:rvr_quality” • MMM Cameras 2) “<IDPH DPO>:cidph:rvr:quality” **Type** 1) enum 2) I32
Ops Keyword RADIANCE_OFFSET **PDS Keyword** MSL:RADIANCE_OFFSET **Definition** Specifies the constant value by which a stored radiance is added.	**Valid Values** n/a **Type** float **Units** • RAD	**Mode** RDR-generating software **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
Expressed as an equation as follows: true_radiance_value = radiance_offset + radiance_scaling_factor * stored_radiance value There are 3 types of radiometric corrections: Radiance-calibrated RDRs These "RAD" (and "RAL") RDRs have been scaled to absolute radiance units using either pre-flight radiometric calibration coefficients or calibration coefficients derived from in-flight observations of the calibration target. The units on these files are (W/m^2/sr/nm). Radiance factor-calibrated RDRs These "IOF" (and "IOL") RDRs are unitless but have values in the range of 0.0 to 1.0 (for example, average bright Mars soils exhibit I/F ~ 0.35 at 750 nm and I/F ~ 0.05 at 410 nm). Instrumentally-calibrated RDRs These "CCD" (and "CCL") RDRs have had no radiance scaling applied, so the units on these files are "corrected" DN.	WATT* M**-2* SR**-1* NM**-1 • IOF Unitless I/F • CCD DN **Location** DERIVED_IMAGE_PARMS (Group)	
Ops Keyword RADIANCE_SCALING_FACTOR **PDS Keyword** MSL:RADIANCE_SCALING_FACTOR **Definition** Specifies the constant value by which a stored radiance is multiplied. NOTE: Expressed as an equation: true_radiance_value = radiance_offset + radiance_scaling_factor * stored_radiance value There are 3 types of radiometric corrections: Radiance-calibrated RDRs These "RAD" (and "RAL") RDRs have been scaled to absolute radiance units using either pre-flight radiometric calibration coefficients or calibration coefficients derived from in-flight observations of the calibration target. The units on these files are (W/m^2/sr/nm). Radiance factor-calibrated RDRs These "IOF" (and "IOL") RDRs are unitless but have values in	**Valid Values** n/a **Type** float **Units** n/a **Location** • DERIVED_IMAGE_PARMS (Group)	**Mode** RDR-generating software **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
the range of 0.0 to 1.0 (for example, average bright Mars soils exhibit I/F ~ 0.35 at 750 nm and I/F ~ 0.05 at 410 nm). Instrumentally-calibrated RDRs These "CCD" (and "CCL") RDRs have had no radiance scaling applied, so the units on these files are "corrected" DN.		
Ops Keyword RADIOMETRIC_CORRECTION_TYPE **PDS Keyword** same **Definition** Identifies the method used for radiometric correction. Values include ""CHEMRAD" for the correction done by the ChemCam team and "MIPLRAD" for the MIPL correction (flat-field, exposure and temperature only), or "NONE" for the case when no radiometric correction has been performed.	**Valid Values** "CHEMRAD", "MIPLRAD", "NONE" **Type** string **Units** n/a **Location** DERIVED_IMAGE_PARMS (Group)	**Mode** RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword RANGE_ORIGIN_VECTOR **PDS Keyword** MSL:RANGE_ORIGIN_VECTOR **Definition** Specifies the 3-D space from which the Range values are measured in a Range RDR. This will normally be the same as the C point of the camera. It is expressed in the coordinate system specified by the REFERENCE_COORD_SYSTEM_* keywords in the enclosing DERIVED_IMAGE_PARMS group.	**Valid Values** n/a **Type** float array[3] **Units** meters (<m> unit tag required) **Location** DERIVED_IMAGE_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** "<IDPH DPO>:idph:cmod_c[3]" **Type** F64[3]
Ops Keyword RECEIVED_PACKETS **PDS Keyword** same **Definition** Specifies the total number of telemetry packets which constitute a reconstructed data product. NOTE: For MSL, telemetry data processing does not track "packets", but instead data product "parts".	**Valid Values** n/a **Type** integer **Units** n/a **Location** TELEMETRY (Class)	**Mode** EMD in XML format **Field as "<xml name>:[<element>]:[<element>]:<field>"** "MslEarthProductMetadata:MslProductMetadata:PartList: TotalReceived" **Type** n/a
Ops Keyword RECORD_BYTES **PDS Keyword** same	**Valid Values** "0" to n **Type** integer	**Mode** Calculation **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a

OUTPUT METADATA (PRODUCT LABEL)		**INPUT METADATA (SOURCE)**
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
Definition Specifies the number of bytes in a physical file record, including record terminators and separators. NOTE: In the PDS, the use of record_bytes, along with other file-related data elements is fully described in the Standards Reference.	**Units** n/a **Location** FILE (Class)	**Type** n/a
Ops Keyword RECORD_TYPE **PDS Keyword** same **Definition** Specifies the record format of a file. NOTE: In the PDS, when record_type is used in a detached label file it always describes its corresponding detached data file, not the label file itself. The use of record_type along with other file-related data elements is fully described in the PDS Standards Reference.	**Valid Values** “FIXED_LENGTH” **Type** string(20) **Units** n/a **Location** FILE (Class)	**Mode** Calculation **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a
Ops Keyword REFERENCE_AZIMUTH **PDS Keyword** same **Definition** Specifies the azimuth of the line extending from the center of the image to the top center of the image with respect to a polar projection.	**Valid Values** n/a **Type** float **Units** deg (<deg> unit tag required) **Location** SURFACE_PROJECTION_PARMS (Group)	**Mode** RDR-generating software **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a
Ops Keyword REFERENCE_COORD_SYSTEM_INDEX **PDS Keyword** same **Definition** Specifies which instance of the coordinate system named by REFERENCE_COORD_SYSTEM_NAME is the reference coordinate system for the group in which the keyword occurs. This index is a set of integers which serve to identify coordinate system instances in a mission-specific manner. For MSL, these indices are based on the ROVER_MOTION_COUNTER and are in the same order as specified by ROVER_MOTION_COUNTER_NAME. • EDRs will contain 1 or 10, depending on the group in which the keyword occurs.	**Valid Values** n/a **Type** integer array[10] **Units** n/a **Location** • For EDRs … 1) ROVER_COORDINATE_SYSTEM (Group) 2) RSM_COORDINATE_SYSTEM (Group) 3) ARM_COORDINATE_SYSTEM (Group) 4) GEOMETRIC_CAMERA_MODEL (Group) 5) ROVER_DERIVED_GEOMETRY_PARMS (Group) 6) SITE_DERIVED_GEOMETRY_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E), Calculation **Field as “<xml name>:[<element>]:[<element>]:<field>”** • Eng. Cameras 1) “<IDPH DPO>:idph:rmc_site” 2) “<IDPH DPO>:idph:rmc_site”, “<IDPH DPO>:idph:rmc_drive”, “<IDPH DPO>:idph:rmc_pose”, “<IDPH DPO>:idph:rmc_arm”, “<IDPH DPO>:idph:rmc_chimra”, “<IDPH DPO>:idph:rmc_drill”, “<IDPH DPO>:idph:rmc_rsm”, “<IDPH DPO>:idph:rmc_hga”, “<IDPH DPO>:idph:rmc_drt”, “<IDPH DPO>:idph:rmc_ic”

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
• For RDRs, the number of indices can be anything from 1 (used for SITE_FRAME) up to 10; however only 1, 2, 3, and 10 indices are common in RDRs. See also REFERENCE_COORD_SYSTEM_NAME and COORDINATE_SYSTEM_INDEX. The distinction is that COORDINATE_SYSTEM_INDEX is used when defining a coordinate system group to indicate what is being defined, while REFERENCE_COORD_SYSTEM_INDEX is used when referring to one.	7) ROVER_COORDINATE_SYSTEM (Group) 8) RSM_COORDINATE_SYSTEM (Group) 9) ARM_COORDINATE_SYSTEM (Group) 10) GEOMETRIC_CAMERA_MODEL (Group) 11) ROVER_DERIVED_GEOMETRY_PARMS (Group) 12) SITE_DERIVED_GEOMETRY_PARMS (Group) • For RDRs … - ROVER_COORDINATE_SYSTEM (Group) - RSM_COORDINATE_SYSTEM (Group) - ARM_COORDINATE_SYSTEM (Group) - GEOMETRIC_CAMERA_MODEL (Group) - ROVER_DERIVED_GEOMETRY_PARMS (Group) - SITE_DERIVED_GEOMETRY_PARMS (Group) - SITE_COORDINATE_SYSTEM (Group) - LOCAL_LEVEL_COORDINATE_SYSTEM (Group) - DERIVED_IMAGE_PARMS (Group) - SURFACE_PROJECTION_PARMS (Group) - SURFACE_MODEL_PARMS (Group)	3) same as #2 above 4) same as #2 above 5) same as #2 above 6) same as #1 above • MMM Cameras 7) "<IDPH DPO>:cidph:rmc_site" 8) "<IDPH DPO>:cidph:rmc_site", "<IDPH DPO>:cidph:rmc_drive", "<IDPH DPO>:cidph:rmc_pose", "<IDPH DPO>:cidph:rmc_arm", "<IDPH DPO>:cidph:rmc_chimra", "<IDPH DPO>:cidph:rmc_drill", "<IDPH DPO>:cidph:rmc_rsm", "<IDPH DPO>:cidph:rmc_hga", "<IDPH DPO>:cidph:rmc_drt", "<IDPH DPO>:cidph:rmc_ic" 9) same as #8 above 10) same as #8 above 11) same as #8 above 12) same as #7 above **Type** U16
Ops Keyword REFERENCE_COORD_SYSTEM_NAME **PDS Keyword** same **Definition** Specifies the full name of the reference coordinate system (CS) for the group in which the keyword occurs. All vectors and positions relating to 3-D space within the enclosing group are expressed using this reference coordinate system. For rover or lander missions with non-unique coordinate systems (such as "SITE"), the CS name and index (see (REFERENCE_COORD_SYSTEM_INDEX) together, along with the solution id (see REFERENCE_COORD_SYSTEM_SOLN_ID), completely specify the reference CS. The values appearing in a reference index depend on what the reference name is. The valid coordinate frame names vary by project, but are generally the same as for COORDINATE_SYSTEM_FRAME. For MSL, EDRs use a standard, predefined frame name for each occurrence. However, RDRs can use any value available in COORDINATE_SYSTEM_NAME. Despite that, only a few	**Valid Values** • For EDRs … 1) "SITE_FRAME" 2) "ROVER_NAV_FRAME" 3) "ROVER_NAV_FRAME" 4) "ROVER_NAV_FRAME" 5) "ROVER_NAV_FRAME" 6) "SITE_FRAME" • For RDRs … Any value from COORDINATE_SYSTEM_NAME is valid. **Type** string(20) **Units** n/a **Location** • For EDRs … 1) ROVER_COORDINATE_SYSTEM (Group) 2) RSM_COORDINATE_SYSTEM (Group) 3) ARM_COORDINATE_SYSTEM (Group) 4) GEOMETRIC_CAMERA_MODEL (Group) 5) ROVER_DERIVED_GEOMETRY_PARMS (Group) 6) SITE_DERIVED_GEOMETRY_PARMS (Group)	**Mode** Static values: 1) for 1 RMC element defining Site Frame 2) for 10 RMC elements defining Rover Navigation Frame 3) for 10 RMC elements defining Rover Navigation Frame 4) for 10 RMC elements defining Rover Navigation Frame 5) for 10 RMC elements defining Rover Navigation Frame 6) for 1 RMC element defining Site Frame **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a

<table>
<tr><th colspan="2">OUTPUT METADATA (PRODUCT LABEL)</th><th>INPUT METADATA (SOURCE)</th></tr>
<tr><th>• Ops Keyword
• PDS-Compliant Keyword
• Definition</th><th>• Valid Values (quoted)
• Type
• Units
• Keyword Location in Label</th><th>• Mode
• Metadata Field
• Type</th></tr>
<tr><td>frame names are commonly used. “SITE_FRAME” is used for most SITE, ROVER, and LOCAL_LEVEL CS definitions, as well as for XYZ data and many mosaics. “ROVER_NAV_FRAME” is used for most other CS definitions, surface normals, camera models, and some mosaics. “LOCAL_LEVEL_FRAME” is used for some mosaics.

See also COORDINATE_SYSTEM_NAME. The distinction is that COORDINATE_SYSTEM_NAME is used when defining a coordinate system group to indicate what is being defined, while REFERENCE_COORD_SYSTEM_NAME is used when referring to one.</td><td>• For RDRs …
- ROVER_COORDINATE_SYSTEM (Group)
- RSM_COORDINATE_SYSTEM (Group)
- ARM_COORDINATE_SYSTEM (Group)
- GEOMETRIC_CAMERA_MODEL (Group)
- ROVER_DERIVED_GEOMETRY_PARMS (Group)
- SITE_DERIVED_GEOMETRY_PARMS (Group)
- SITE_COORDINATE_SYSTEM (Group)
- LOCAL_LEVEL_COORDINATE_SYSTEM (Group)
- DERIVED_IMAGE_PARMS (Group)
- SURFACE_PROJECTION_PARMS (Group)
- SURFACE_MODEL_PARMS (Group)</td><td></td></tr>
<tr><td><u>Ops Keyword</u>
REFERENCE_COORD_SYSTEM_SOLN_ID

<u>PDS Keyword</u>
MSL:REFERENCE_COORD_SYSTEM_SOLN_ID

<u>Definition</u>
See SOLUTION_ID. SOLUTION_ID is used when defining a coordinate system group to indicate what is being defined, while REFERENCE_COORD_SYSTEM_SOLN_ID is used when referring to one. Appears in labels of RDRs only, not EDR products.</td><td><u>Valid Values</u>
<u>n/a</u>

<u>Type</u>
string

<u>Units</u>
n/a

<u>Location</u>
For RDRs only …
- ROVER_COORDINATE_SYSTEM (Group)
- RSM_COORDINATE_SYSTEM (Group)
- ARM_COORDINATE_SYSTEM (Group)
- GEOMETRIC_CAMERA_MODEL (Group)
- ROVER_DERIVED_GEOMETRY_PARMS (Group)
- SITE_DERIVED_GEOMETRY_PARMS (Group)
- SITE_COORDINATE_SYSTEM (Group)
- LOCAL_LEVEL_COORDINATE_SYSTEM (Group)
- DERIVED_IMAGE_PARMS (Group)
- SURFACE_PROJECTION_PARMS (Group)
- SURFACE_MODEL_PARMS (Group)</td><td><u>Mode</u>
Software dependent

<u>Field as “<xml name>:[<element>]:[<element>]:<field>”</u>
n/a

<u>Type</u>
n/a</td></tr>
<tr><td><u>Ops Keyword</u>
REFERENCE_PIXEL_IMAGE_ID

<u>PDS Keyword</u>
MSL:REFERENCE_PIXEL_IMAGE_ID

<u>Definition</u>
Specifies the value of IMAGE_ID or PRODUCT_ID for the reference pixel EDR that was used to remove bias in generating the RDR.

NOTE: If the model rather than a reference pixel EDR was used to remove the bias then this keyword is not included in the PDS label.</td><td><u>Valid Values</u>
n/a

<u>Type</u>
string

<u>Units</u>
n/a

<u>Location</u>
DERIVED_IMAGE_PARMS (Group)</td><td><u>Mode</u>
RDR-generating software

<u>Field as “<xml name>:[<element>]:[<element>]:<field>”</u>
n/a

<u>Type</u>
n/a</td></tr>
</table>

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
Ops Keyword REGION_COUNT **PDS Keyword** None **Definition** Number of regions in the AEGIS AGS mask.	**Valid Values** n/a **Type** Integer **Units** n/a **Location** DERIVED_IMAGE_PARMS (Group)	**Mode** RDR-generating software **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a
Ops Keyword RELEASE_ID **PDS Keyword** same **Definition** Specifies the unique identifier associated with the release to the public of all or part of a data set. The release number is associated with the data set, not the mission. When a data set is released incrementally, such as every three months during a mission, the RELEASE_ID is updated each time part of the data set is released. The first release of a data set in the mission should have a value of "0001". For example, on MSL the first release of the EDR data set will have RELEASE_ID = "0001". The next EDR release will have RELEASE_ID = "0002".	**Valid Values** n/a **Type** string **Units** n/a **Location** IDENTIFICATION (Class)	**Mode** User parameter input **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a
Ops Keyword REQUEST_ID **PDS Keyword** MSL:REQUEST_ID **Definition** Specifies the ground-assigned Request ID associated with the data product. The general operational usage of REQUEST_ID is to group related datasets together by science or engineering application or theme, such as frames in a mosaic.	**Valid Values** n/a **Type** string **Units** **Location** IDENTIFICATION (Class)	**Mode** EMD in XML format **Field as “<xml name>:[<element>]:[<element>]:<field>”** “MslEarthProductMetadata:MslProductMetadata:RequestId” **Type** U16
Ops Keyword RESPONSIVITY_CONSTANTS **PDS Keyword** MSL:RESPONSIVITY_CONSTANTS **Definition** Specifies the array of responsivity constants used in generating	**Valid Values** n/a **Type** float array **Units** n/a	**Mode** RDR-generating software **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
the RDR.	**Location** DERIVED_IMAGE_PARMS (Group)	
Ops Keyword RESPONSIVITY_CONSTANTS_FILE **PDS Keyword** MSL:RESPONSIVITY_CONSTANTS_FILE **Definition** Specifies the name of the responsivity constants file used in generating the RDR.	**Valid Values** n/a **Type** string **Units** n/a **Location** DERIVED_IMAGE_PARMS (Group)	**Mode** RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword ROVER_MOTION_COUNTER **PDS Keyword** same **Definition** Specifies a set of integers which describe a (potentially) unique location (position/orientation) for a rover. Each time something happens that moves, or could potentially move, the rover, a new motion counter value is created. This includes intentional motion due to drive commands, as well as potential motion due to other articulating devices, such as arms or antennae. This motion counter (or part of it) is used as a reference to define instances of coordinate systems which can move such as SITE or ROVER frames. The motion counter is defined in a mission-specific manner. Although the original intent was to have incrementing indices (e.g. MER), the motion counter could also contain any integer values which conform to the above definition, such as time or spacecraft clock values. For MSL, the motion counter consists of ten values. In order, they are "Site", "Drive", "Pose", "Arm", "CHIMRA", "Drill", "RSM", "HGA", "DRT", and "IC". The Site value increments whenever a new major Site frame is declared. The Drive value increments any time intentional driving is done. Each of those, along with Pose, resets all later indices to 0 when they increment. The Arm, CHIMRA, DRILL, RSM, HGA, DRT and IC increment whenever the corresponding articulation device moves. These all increment independently of each other; they are reset to zero only when the SITE or DRIVE or POSE changes.	**Valid Values** n/a **Type** integer array[10] **Units** n/a **Location** • IDENTIFICATION (Class) • INITIAL_STATE_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** • Eng. Cameras "<IDPH DPO>:idph:rmc_site", "<IDPH DPO>:idph:rmc_drive", "<IDPH DPO>:idph:rmc_pose", "<IDPH DPO>:idph:rmc_arm", "<IDPH DPO>:idph:rmc_chimra", "<IDPH DPO>:idph:rmc_drill", "<IDPH DPO>:idph:rmc_rsm", "<IDPH DPO>:idph:rmc_hga", "<IDPH DPO>:idph:rmc_drt", "<IDPH DPO>:idph:rmc_ic" • MMM Cameras "<IDPH DPO>:cidph:rmc:site", "<IDPH DPO>:cidph:rmc:drive", "<IDPH DPO>:cidph:rmc:pose", "<IDPH DPO>:cidph:rmc:arm", "<IDPH DPO>:cidph:rmc:chimra", "<IDPH DPO>:cidph:rmc:drill", "<IDPH DPO>:cidph:rmc:rsm", "<IDPH DPO>:cidph:rmc:hga", "<IDPH DPO>:cidph:rmc:drt", "<IDPH DPO>:cidph:rmc:ic" **Type** U16
Ops Keyword ROVER_MOTION_COUNTER_NAME	**Valid Values** ("SITE", "DRIVE", "POSE", "ARM", "CHIMRA", "DRILL", "RSM", "HGA", "DRT", "IC")	**Mode** Static value: - Single value representing array of 10 RMC elements from

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
PDS Keyword same **Definition** Specifies an array that provides the formal names identifying each integer in ROVER_MOTION_COUNTER.	**Type** string array[10] **Units** n/a **Location** • IDENTIFICATION (Class) • INITIAL_STATE_PARMS (Group)	ROVER_MOTION_COUNTER. **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a
Ops Keyword SAMPLE_BITS **PDS Keyword** same **Definition** Specifies the stored total number of bits, or units of binary information, contained in a LINE_SAMPLE value. As an example, a SAMPLE_BIT_MASK value of "2#0000111111111111", expressing 12-bit data in the case of an unscaled EDR or ILUT RDR (see SAMPLE_BIT_METHOD), has a SAMPLE_BITS value of “16”.	**Valid Values** “1”, “2”, “4”, “8”, “16”, “32”, “64” **Type** integer **Units** n/a **Location** IMAGE (Object)	**Mode** DPO in XML format (referenced to APID Name in Appendix E), Calculation **Field as “<xml name>:[<element>]:[<element>]:<field>”** “<IDPH DPO>:idph:scale” **Type** enum
Ops Keyword SAMPLE_BIT_MASK **PDS Keyword** same **Definition** Specifies the active bits in a sample. Any bit mask is valid in an RDR. However, only a few are generated for products described in this SIS. Any 8-bit product, whether a scaled EDR or an RDR, will have the value "2#11111111" and be stored in one byte. Any 12-bit product, whether an unscaled EDR, or an ILUT RDR (see SAMPLE_BIT_METHOD), will have the value "2#0000111111111111" and be stored in two bytes. A 15-bit product (e.g. Radiometrically-corrected RDR product type) will have the value "2#0111111111111111" and be stored in two bytes. Any 32-bit integer product (e.g. Histogram EDR) will have the value "2#11111111111111111111111111111111" and be stored in four bytes.	**Valid Values** • For EDRs … - Eng. Cameras “2#11111111#”, “2#0000111111111111#”, “2#11111111111111111111111111111111#” - MMM Cameras “2#11111111#” • For RDRs … - Value Any combination of 0’s and 1’s is valid **Type** non-decimal **Units** n/a **Location** IMAGE (Object)	**Mode** DPO in XML format (referenced to APID Name in Appendix E), Calculation **Field as “<xml name>:[<element>]:[<element>]:<field>”** Eng. Cameras “<IDPH DPO>:idph:scale” NOTES: • For EDRs … - Eng. Cameras: a. If data type is BYTE … scale = 1 thru 5, 7-11 or 127 and valid value is “2#11111111#” b. If data type is HALF … scale = 0 and valid value is “2#0000111111111111#” c. If data type is FULL … valid value is “2#11111111111111111111111111111111#” **Type** enum

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
For floating-point data, SAMPLE_BIT_MASK is not valid and may be absent. If present, it should be ignored. NOTE: In the PDS, the domain of SAMPLE_BIT_MASK is dependent upon the currently-described value in the SAMPLE_BITS element and only applies to integer values.		
Ops Keyword SAMPLE_BIT_METHOD **PDS Keyword** same **Definition** Specifies the method in which bit scaling is performed. For MSL, the bit scaling is a 12-bit to 8-bit scaling and can be performed onboard via hardware and/or software , or on the ground in an inverse operation. As valid values, "SOFTWARE_INVERTED" and "HARDWARE_INVERTED" indicate that an Inverse Lookup Table (ILUT) was applied during ground processing to 8-bit data, scaling the 8 bits in the EDR to the lowest 12 bits in the signed 16-bit integer RDR. This characterizes the Inverse LUT RDR (OPGS), which is the inverse of the "12 to 8-bit" scaled version of onboard 12-bit data (via LUT).	**Valid Values** • For EDRs … - Eng. Cameras & RMI "NONE", "HARDWARE", "SOFTWARE" - MMM Cameras "NONE", "HARDWARE" • For RDRs … - Eng. Cameras & RMI "NONE", "HARDWARE", "SOFTWARE", "SOFTWARE_INVERTED", "HARDWARE_INVERTED" - MMM Cameras "NONE", "HARDWARE", "HARDWARE_INVERTED" **Type** string **Units** n/a **Location** INSTRUMENT_STATE_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E), Calculation **Field as "<xml name>:[<element>]:[<element>]:<field>"** • Eng. Cameras & RMI "<IDPH DPO>:idph:scale", "<IDPH DPO>:idph:hw_scale" • MMM Cameras "<Ancillary DPO>:image_id_data:comp_companding_mode" NOTES: • For EDRs …. - Eng. Cameras: a. If scale = 0, value is "NONE" b. If scale > 0 and hw_scale=0 (false), value is "SOFTWARE" c. If scale > 0 and hw_scale=1 (true), value is "HARDWARE" - MMM Cameras: a. If comp_companding_mode <= 32, value is "HARDWARE" b. If comp_companding_mode = 255, value is "NONE" c. If comp_companding_mode not "a" or "b", value is "UNK" • For RDRs …. - Any of the EDR values are valid. - If ILUT (inverse LUT) operation, then append "_INVERTED" to the EDR value, resulting in "SOFTWARE_INVERTED" or "HARDWARE_INVERTED" as the case may be. **Type** enum
Ops Keyword SAMPLE_BIT_MODE_ID **PDS Keyword** same **Definition** Specifies the type of pixel scaling performed.	**Valid Values** • Eng. Cameras & RMI 0 = "NONE" 1 = "LUT1" 2 = "LUT2" 3 = "LUT3" 4 = "LUT4" 5 = "LUT5" 7 = "MSB_BIT7"	**Mode** DPO in XML format (referenced to APID Name in Appendix E), Calculation **Field as "<xml name>:[<element>]:[<element>]:<field>"** • Eng. Cameras 1) "<IDPH DPO>:idph:scale" 2) "<IDPH DPO>:idph:params:scale"

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
For MSL, pixel scaling is accomplished by using onboard lookup tables or by shifting a specified bit into the most significant bit. Descriptions of the valid values for Eng. Cameras and RMI follow: a) “NONE” - No scaling; keep as 12-bit (if available) b) “LUT1” - Use lookup table 1 c) “LUT2” - Use lookup table 2 d) “LUT3” - Use lookup table 3 e) “LUT4” - Use lookup table 4 f) “LUT5” - Use lookup table 5 g) “MSB_BIT7” - No scaling; make bit 7 most significant bit h) “MSB_BIT8” - Shift to make bit 8 most significant bit i) “MSB_BIT9” - Shift to make bit 9 most significant bit j) “MSB_BIT10” - Shift to make bit 10 most significant bit k) “MSB_BIT11” - Shift to make bit 11 most significant bit l) “AUTOSHIFT” - Auto-shift to keep highest value Descriptions of the valid values for MMM Cameras follow: a) “NONE” - no scaling b) “MMM_LUT_<n>” - valid LUT, with n = 0 to 32 NOTE: For MSL, the instance of SAMPLE_BIT_MODE_ID in the PDS label Group INSTRUMENT_STATE_PARMS will never have the value “AUTOSHIFT”.	8 = “MSB_BIT8” 9 = “MSB_BIT9” 10 = “MSB_BIT10” 11 = “MSB_BIT11” 127 = “AUTOSHIFT” • MMM Cameras “MMM_LUT_<n>”, “NONE”, “UNK” where, n = 0 to 32 **Type** string **Units** n/a **Location** 1) INSTRUMENT_STATE_PARMS (Group) 2) IMAGE_REQUEST_PARMS (Group) 3) THUMBNAIL_REQUEST_PARMS (Group) 4) MINI_HEADER (Group) ** ** Except MAHLI Z-Stack & Range Map	3) “<IDPH DPO>:idph:params:thumb_scale” • MMM Cameras 1) “<Ancillary DPO>:image_id_data:comp_companding_mode” 4) “MMM_Image_Mini_Header[TBD]” NOTES: • For MMM Cameras … a. If comp_companding_mode <= 32, value is “MMM_LUT_<n>” b. If comp_companding_mode = 255, value is “NONE” c. If comp_companding_mode not “a” or “b”, value is “UNK” **Type** enum
Ops Keyword SAMPLE_CAMERA_MODEL_OFFSET **PDS Keyword** same **Definition** Specifies the location of the image origin with respect to the camera model's origin. For CAHV/CAHVOR models, this origin is not the center of the camera, but is the upper-left corner of the "standard"-size image, which is encoded in the CAHV vectors. (MIPL Projections - Perspective)	**Valid Values** n/a **Type** float **Units** pixel (<pixel> unit tag required) **Location** SURFACE_PROJECTION_PARMS (Group)	**Mode** RDR-generating software **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a
Ops Keyword SAMPLE_PROJECTION_OFFSET **PDS Keyword** same **Definition** Specifies the sample coordinate of the location in the image of the "special" point of the mosaic. For Polar projections, this is the nadir of the polar projection. For Vertical, Orthographic and Orthorectified projections, this is the origin of the projected coordinate system (corresponding to	**Valid Values** n/a **Type** float **Units** pixel (<pixel> unit tag required) **Location** SURFACE_PROJECTION_PARMS (Group)	**Mode** RDR-generating software **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
PROJECTION_ORIGIN_VECTOR), and may be off the image. Not applicable to other projections.		
Ops Keyword SAMPLE_TYPE **PDS Keyword** same **Definition** Specifies the data storage representation of sample value. The valid values are platform dependent. JAVA, Suns and PPC-based Macs or Intel-based Macs are MSB and IEEE_REAL. Intel-based machines usually running Windows or Linux are LSB integers and PC_REAL.	**Valid Values** "IEEE_REAL", "MSB_INTEGER", "MSB_UNSIGNED_INTEGER", "PC_REAL", "LSB_INTEGER", "LSB_UNSIGNED_INTEGER" **Type** string(30) **Units** n/a **Location** IMAGE (Object)	**Mode** DPO in XML format (referenced to APID Name in Appendix E), depends on host platform type **Field as "<xml name>:[<element>]:[<element>]:<field>"** 1) "<IDPH DPO>:idph:scale" 2) "<IDPH DPO>:idph:hw_scale" **Type** 1) enum 2) boolean
Ops Keyword SEQUENCE_EXECUTION_COUNT **PDS Keyword** MSL:SEQUENCE_EXECUTION_COUNT **Definition** Specifies how many times this sequence has executed since the last reset of the flight computer. For MSL, this means RCE (Rover Compute Element) start-up.	**Valid Values** n/a **Type** integer **Units** n/a **Location** TELEMETRY (Class)	**Mode** EMD in XML format **Field as "<xml name>:[<element>]:[<element>]:<field>"** "MslEarthProductMetadata:MslProductMetadata: SequenceExecutionCounter" **Type** n/a
Ops Keyword SEQUENCE_ID **PDS Keyword** same **Definition** Specifies the spacecraft sequence associated with the given product.	**Valid Values** n/a **Type** string(30) **Units** n/a **Location** IDENTIFICATION (Class)	**Mode** EMD in XML format **Field as "<xml name>:[<element>]:[<element>]:<field>"** "MslEarthProductMetadata:MslProductMetadata:SequenceId" **Type** n/a
Ops Keyword SEQUENCE_VERSION_ID **PDS Keyword** same **Definition** Specifies the version identifier for a particular observation sequence used during planning or data processing.	**Valid Values** n/a **Type** string(30) **Units** n/a **Location** IDENTIFICATION (Class)	**Mode** EMD in XML format **Field as "<xml name>:[<element>]:[<element>]:<field>"** "MslEarthProductMetadata:MslProductMetadata:SequenceVersion" **Type** n/a
Ops Keyword	**Valid Values**	**Mode**

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
SHUTTER_CORRECTION_MODE_ID **PDS Keyword** same **Definition** Specifies whether shutter subtraction will be performed.	• Eng. Cameras 0 = “NONE” 1 = “CONDITIONAL” 2 = “ALWAYS” • MMM Cameras “N/A” **Type** string **Units** n/a **Location** OBSERVATION_REQUEST_PARMS (Group)	DPO in XML format (referenced to APID Name in Appendix E) **Field as “<xml name>:[<element>]:[<element>]:<field>”** “<IDPH DPO>:idph:params:shutter” **Type** enum
Ops Keyword SHUTTER_CORRECT_THRESH_COUNT **PDS Keyword** same **Definition** Specifies the exposure time threshold for conditional shutter subtraction. For MSL, the count was in increments of 5.1 ms.	**Valid Values** • Eng. Cameras “0” to “65535” • MMM Cameras “N/A” **Type** unsigned integer **Units** n/a **Location** OBSERVATION_REQUEST_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as “<xml name>:[<element>]:[<element>]:<field>”** “<IDPH DPO>:idph:params:shutter_thresh” **Type** U16
Ops Keyword SHUTTER_EFFECT_CORRECTION_FLAG **PDS Keyword** same **Definition** Specifies whether or not a shutter effect correction was applied to the image. The shutter effect correction involves the removal from the image of the shutter, or fixed-pattern.	**Valid Values** • Eng. Cameras 0 = “FALSE” 1 = “TRUE” • MMM Cameras “UNK” **Type** string(5) **Units** n/a **Location** INSTRUMENT_STATE_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as “<xml name>:[<element>]:[<element>]:<field>”** “<IDPH DPO>:idph:shutter” **Type** boolean
Ops Keyword SOFTWARE_KEYWORD_NAME	**Valid Values** n/a	**Mode** RDR-generating software

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
PDS Keyword MSL:SOFTWARE_KEYWORD_NAME **Definition** Specifies the value of all keyword names used as input to the primary generating software named in SOFTWARE_MODULE_NAME. For MSL, this keyword may be placed in a xxx_HISTORY_PARMS group, where "xxx" is the task name. There may be multiple groups to accommodate multiple software tasks.	**Type** string array **Units** n/a **Location** <xxx>_HISTORY_PARMS (Group) where, xxx = RDR software name	**Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword SOFTWARE_KEYWORD_TYPE **PDS Keyword** MSL:SOFTWARE_KEYWORD_TYPE **Definition** Specifies the value of a keyword type used as input to the primary generating software named in SOFTWARE_MODULE_NAME. For MSL, this keyword may be placed in a xxx_HISTORY_PARMS group, where "xxx" is the task name. There may be multiple groups to accommodate multiple software tasks. Use of this keyword is optional. NOTE: The value of this keyword will be relevant to SOFTWARE_LANGUAGE.	**Valid Values** n/a **Type** string array **Units** n/a **Location** <xxx>_HISTORY_PARMS (Group) where, xxx = RDR software name	**Mode** RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword SOFTWARE_KEYWORD_VALUE **PDS Keyword** MSL:SOFTWARE_KEYWORD_VALUE **Definition** Specifies the value of a keyword value used as input to the primary generating software named in SOFTWARE_MODULE_NAME. Array values can be nested. Example: "SOFTWARE_KEYWORD_VALUE = (value 1, (value2A, value2b), value3)." For MSL, this keyword may be placed in a xxx_HISTORY_PARMS group, where "xxx" is the task name. There may be multiple groups to accommodate multiple software tasks.	**Valid Values** n/a **Type** string array **Units** n/a **Location** <xxx>_HISTORY_PARMS (Group) where, xxx = RDR software name	**Mode** RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword	**Valid Values**	**Mode**

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
SOFTWARE_LANGUAGE **PDS Keyword** MSL:SOFTWARE_LANGUAGE **Definition** Specifies the programming language that the primary RDR generating software is written in (eg: IDL)	n/a **Type** string **Units** n/a **Location** <xxx>_HISTORY_PARMS (Group) where, xxx = RDR software name	RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword SOFTWARE_MODULE_NAME **PDS Keyword** MSL:SOFTWARE_MODULE_NAME **Definition** Specifies the name of the primary software module used to generate this product. This is the module to which the PDS label entries SOFTWARE_PARAMETER_* and SOFTWARE_KEYWORD_* apply. For MSL, this keyword may be placed in a xxx_HISTORY_PARMS group, where "xxx" is the task name. There may be multiple groups to accommodate multiple software tasks.	**Valid Values** n/a **Type** string **Units** n/a **Location** <xxx>_HISTORY_PARMS (Group) where, xxx = RDR software name	**Mode** RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword SOFTWARE_MODULE_TYPE **PDS Keyword** MSL:SOFTWARE_MODULE_TYPE **Definition** Specifies the type of the primary software module named in SOFTWARE_MODULE_NAME.	**Valid Values** "PROCEDURE", "FUNCTION", "SCRIPT" **Type** string **Units** n/a **Location** <xxx>_HISTORY_PARMS (Group) where, xxx = RDR software name	**Mode** RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword SOFTWARE_NAME **PDS Keyword** same **Definition** Specifies the name of data processing software such as a program or a program library.	**Valid Values** "MSLEDRGEN", other **Type** string(60) **Units** n/a **Location**	**Mode** Dependent on EDR/RDR state: - For EDRs, static - For RDRs, RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
	• For EDRs … - MSLEDRGEN_HISTORY_PARMS (Group) • For RDRs … - <xxx>_HISTORY_PARMS (Group) where, xxx = RDR software name	
Ops Keyword SOFTWARE_PARAMETER_NAME **PDS Keyword** MSL:SOFTWARE_PARAMETER_NAME **Definition** Specifies the value of a parameter name used as input to the primary generating software named in SOFTWARE_MODULE_NAME. For MSL, this keyword may be placed in a xxx_HISTORY_PARMS group, where “xxx” is the task name. There may be multiple groups to accommodate multiple software tasks.	**Valid Values** n/a **Type** string array **Units** n/a **Location** <xxx>_HISTORY_PARMS (Group) where, xxx = RDR software name	**Mode** RDR-generating software **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a
Ops Keyword SOFTWARE_PARAMETER_TYPE **PDS Keyword** MSL:SOFTWARE_PARAMETER_TYPE **Definition** Specifies the value of a parameter type used as input to the primary generating software named in SOFTWARE_MODULE_NAME. For MSL, this keyword may be placed in a xxx_HISTORY_PARMS group, where “xxx” is the task name. There may be multiple groups to accommodate multiple software tasks. Use of this keyword is optional. NOTE: The value of this keyword will be relevant to SOFTWARE_LANGUAGE.	**Valid Values** n/a **Type** string array **Units** n/a **Location** <xxx>_HISTORY_PARMS (Group) where, xxx = RDR software name	**Mode** RDR-generating software **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a
Ops Keyword SOFTWARE_PARAMETER_VALUE **PDS Keyword** MSL:SOFTWARE_PARAMETER_VALUE **Definition** Specifies the value of a parameter value used as input to the primary generating software named in SOFTWARE_MODULE_NAME. Array values can be nested.	**Valid Values** n/a **Type** array **Units** n/a **Location** <xxx>_HISTORY_PARMS (Group)	**Mode** RDR-generating software **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
Example: "SOFTWARE_PARAMETER_VALUE = (value 1, (value2A, value2b),value3)." For MSL, this keyword may be placed in a xxx_HISTORY_PARMS group, where "xxx" is the task name. There may be multiple groups to accommodate multiple software tasks.	where, xxx = RDR software name	
Ops Keyword SOFTWARE_VERSION_ID **PDS Keyword** same **Definition** Specifies the version (development level) of a program or a program library.	**Valid Values** n/a **Type** string(20) **Units** n/a **Location** • For EDRs … - MSLEDRGEN_HISTORY_PARMS (Group) • For RDRs … - <xxx>_HISTORY_PARMS (Group) where, xxx = RDR software name	**Mode** Dependent on EDR/RDR state: - For EDRs, static - For RDRs, RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword SOLAR_AZIMUTH **PDS Keyword** same **Definition** Specifies one of two angular measurements indicating the direction to the Sun as measured from a specific point on the surface of a planet (ex., from a lander or rover). The positive direction of the elevation is set by the POSITIVE_AZIMUTH_DIRECTION data element. The azimuth is measured positively in the clockwise direction (as viewed from above) with the meridian passing through the positive spin axis of the planet (i.e., the north pole) defining the zero reference. 0 <= SOLAR_AZIMUTH <= 360. Units are degrees. For MSL, the value in the SITE_DERIVED_GEOMETRY group is calculated using SPICE based on the time of the observation. The value in the ROVER_DERIVED_GEOMETRY group reflects what was sent in telemetry (as az/el, converted from the telemetered unit vector). Thus, even after they are converted to a common frame, the values will likely differ by a small amount, representing the difference between the rover's knowledge and the (more accurate) SPICE computation.	**Valid Values** "0.0" to "359.99" "N/A" if any SPICE kernel is unavailable. **Type** float **Units** deg (<deg> unit tag required) **Location** 1) SITE_DERIVED_GEOMETRY_PARMS (Group) 2) ROVER_DERIVED_GEOMETRY_PARMS (Group)	**Mode** 1) a. Calculation using SPICE: - EK Kernel - SCLK Kernel - Leapsecond Kernel - SPK Kernel - PCK Kernel - Surface Kernel b. DPO in XML format (referenced to APID Name in Appendix D), Calculation 2) DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** 1) b. "<IDPH DPO>:idph:sclk_seconds", "<IDPH DPO>:idph:sclk_subseconds" 2) "<IDPH DPO>:idph:sun_dir" NOTE: For MMM products, see details under keyword description for SPACECRAFT_CLOCK_START_COUNT **Type** U32

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
NOTE: For MSL, early in the mission the value for only the instance of this keyword in the label group SITE_DERIVED_GEOMETRY_PARMS was being incorrectly reported as the inverse of what it should have been. For the Ops (ODL) label, the value is flagged as correct by the presence of another keyword (see POSITIVE_AZIMUTH_DIRECTION) in group SITE_DERIVED_GEOMETRY_PARMS. For the PDS label, the value is correct for all files as of archive Volume 2. The instance of this keyword in group ROVER_DERIVED_GEOMETRY_PARMS is always correct for the Ops and PDS labels.		
Ops Keyword SOLAR_ELEVATION **PDS Keyword** same **Definition** Specifies one of two angular measurements indicating the direction to the Sun as measured from a specific point on the surface of a planet (ex., from a lander or rover). The positive direction of the elevation is set by the POSITIVE_ELEVATION_DIRECTION data element. The elevation is measured from the plane which is normal to the line passing between the surface point and the planet's center of mass, and which intersects the surface point. -90 <= SOLAR_ELEVATION <= 90. Units are degrees. For MSL, the value in the SITE_DERIVED_GEOMETRY group is calculated using SPICE based on the time of the observation. The value in the ROVER_DERIVED_GEOMETRY group reflects what was sent in telemetry (as az/el, converted from the telemetered unit vector). Thus, even after they are converted to a common frame, the values will likely differ by a small amount, representing the difference between the rover's knowledge and the (more accurate) SPICE computation.	**Valid Values** "-90.0" to "90.0" "N/A" if any SPICE kernel is unavailable. **Type** float **Units** deg (<deg> unit tag required) **Location** 1) SITE_DERIVED_GEOMETRY_PARMS (Group) 2) ROVER_DERIVED_GEOMETRY_PARMS (Group)	**Mode** 1) a. Calculation using SPICE: - EK Kernel - SCLK Kernel - Leapsecond Kernel - SPK Kernel - PCK Kernel - Surface Kernel b. DPO in XML format (referenced to APID Name in Appendix D), Calculation 2) DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** 1) b. "<IDPH DPO>:idph:sclk_seconds", "<IDPH DPO>:idph:sclk_subseconds" 2) "<IDPH DPO>:idph:sun_dir" NOTE: For MMM products, see details under keyword description for SPACECRAFT_CLOCK_START_COUNT **Type** U32
Ops Keyword SOLAR_LONGITUDE **PDS Keyword** same **Definition** Specifies the value of the angle between the body_Sun line at the time of interest and the body_Sun line at the vernal equinox. This provides a measure of season on a target body, with values of 0 to 90 degrees representing northern spring, 90 to 180 degrees representing northern summer, 180 to 270	**Valid Values** "0.0" to "359.99" "N/A" if any SPICE kernel is unavailable. **Type** float **Units** deg (<deg> unit tag required) **Location**	**Mode** • Calculation: - SCLK Kernel - Landing Site Kernel - P Kernel • DPO in XML format (referenced to APID Name in Appendix E), Calculation **Field as "<xml name>:[<element>]:[<element>]:<field>"** "<IDPH DPO>:idph:sclk_seconds", "<IDPH DPO>:idph:sclk_subseconds

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
degrees representing northern autumn and 270 to 360 degrees representing northern winter. Example: For IRAS, the geocentric ecliptic longitude (B1950) of the Sun at the start of a scan.	IDENTIFICATION (Class)	NOTE: For MMM products, see details under keyword description for SPACECRAFT_CLOCK_START_COUNT **Type** U32
Ops Keyword SOLUTION_ID **PDS Keyword** MSL:SOLUTION_ID **Definition** Specifies the unique identifier for the solution set to which the values in the group belong. For certain kinds of information, such as pointing correction (pointing models) and rover localization (coordinate system definitions), the "true" value is unknown and only estimates of the true value exist. Thus, more than one set of estimates may exist simultaneously, each valid for its intended purpose. Each of these sets is called a "solution" to the unknown true value. SOLUTION_ID is used to identify which solution is being expressed by the containing group. No specific naming convention is defined here, however it is recommended that projects adopt one. The intent is to be able to identify who created the solution, and why. Possible components of the naming convention include user, institution, purpose, request ID, version, program, date/time. The value is not case-sensitive. For MSL, when in a COORDINATE_SYSTEM group, the SOLUTION_ID specifies the ID of the coordinate system being defined in that group. It must be globally unique across all coordinate system instances, i.e. it cannot be reused to define the same coordinate system instance differently. Different coordinate system instances (for example, different values of the RMC index) may share the same SOLUTION_ID. See also REFERENCE_COORD_SYSTEM_NAME and REFERENCE_COORD_SYSTEM_SOLN_ID. The SOLUTION_ID should be the same identifier used in the PLACES rover localization database. The special name "telemetry" is used for values telemetered from the rover. If SOLUTION_ID is absent, "telemetry" should be assumed. For MSL, when in the DERIVED_GEOMETRY_PARMS group, SOLUTION_ID specifies the identifier of the pointing correction solution used to derive the model specified via POINTING_MODEL_PARAMS et al. This identifier should also	**Valid Values** • EDR-generating software "TELEMETRY" • RDR-generating software any string value **Type** string **Units** n/a **Location** • For EDRs … - ROVER_COORDINATE_SYSTEM (Group) - RSM_COORDINATE_SYSTEM (Group) - ARM_COORDINATE_SYSTEM (Group) • For RDRs … - ROVER_COORDINATE_SYSTEM (Group) - RSM_COORDINATE_SYSTEM (Group) - ARM_COORDINATE_SYSTEM (Group) - SITE_COORDINATE_SYSTEM (Group) - LOCAL_LEVEL_COORDINATE_SYSTEM (Group) - DERIVED_IMAGE_PARMS (Group)	**Mode** • EDR-generating software: - Static Value • RDR-generating software: - Determined **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
appear in the pointing correction file referenced by POINTING_CORRECTION_FILE_NAME. If there is only one identifier in the correction file, then SOLUTION_ID may be omitted. SOLUTION_ID may be reused in the context of pointing corrections, although uniqueness is recommended. The pointing correction solution ID namespace is separate from the coordinate system namespace.		
Ops Keyword SOURCE_ID **PDS Keyword** same **Definition** Specifies an identifier for the source. For MSL, it identifies the FSW element that requested the image, i.e. what was the source of the command. The field is based on the value for IMAGE_ID and the mappings are per convention; there is no guarantee that the mappings are used in this way during operations.	**Valid Values** • Eng. Cameras … Value / Range / Command Source "GND" 000000000-099999999 Ground "NAVF" 100000000-199999999 Navigation, Front (left/right camera) "NAVR" 200000000-299999999 Navigation, Rear (left/right camera) "NAVS" 300000000-399999999 Navigation, Steerable (left/right camera) "HAFIQ" 400000000-499999999 Hazard Avoidance, Fault Image Queue "SUN" 500000000-599999999 Sun Finding "FAULT" 600000000-699999999 Fault Protection/ Response "VTT" 700000000-799999999 Target Tracking (left/right) "ARMC" 800000000-899999999 Touch & Go, coarse res (Penultimate left/right) "ARMF" 900000000-999999999 Touch & Go, fine res (Final left/right) "WATCH" 1000000000-1099999999 Dust Devil detection "VISODOM" 1100000000-1199999999 Visual Odometry (left/right) "SPARE1" 1200000000-1299999999 Spare Param group 1 "SPARE2" 1300000000-1399999999 Spare Param group 2 "SPARE3" 1400000000-1499999999 Spare Param group 3 "SPARE4" 1500000000-1599999999 Spare Param group 4 • MMM Cameras … Value "UNK"	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** "<IDPH DPO>:idph:params:imgid" **Type** U32

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
	Type string **Units** n/a **Location** • OBSERVATION_REQUEST_PARMS (Group) • IMAGE_REQUEST_PARMS (Group) • SUBFRAME_REQUEST_PARMS (Group) • THUMBNAIL_REQUEST_PARMS (Group) • REFERENCE_PIXEL_REQUEST_PARMS (Group) • ROW_SUM_REQUEST_PARMS (Group) • COLUMN_SUM_REQUEST_PARMS (Group) • HISTOGRAM_REQUEST_PARMS (Group)	
Ops Keyword SOURCE_PRODUCT_ID **PDS Keyword** same **Definition** Identifies a product used as input to create a new product. The source_product_id may be based on a file name. See also: PRODUCT_ID. For MSL, this keyword indicates the PRODUCT_ID (filename minus extension) of the EDRs (not RDRs) that were used to create this product. In an EDR, this keyword exists and refers to itself; i.e. it is equivalent to PRODUCT_ID. See also INPUT_PRODUCT_ID.	**Valid Values** n/a **Type** string array **Units** n/a **Location** IDENTIFICATION (Class)	**Mode** • EDR-generating software: - Filename of output EDR product, minus extension • RDR-generating software: - Filename of root input EDR product, minus extension **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a
Ops Keyword SOURCE_PRODUCT_WAVELENGTH **PDS Keyword** MSL:SOURCE_PRODUCT_WAVELENGTH **Definition** Specifies the effective wavelength of the corresponding images listed in SOURCE_PRODUCT_ID.	**Valid Values** n/a **Type** integer array **Units** n/a **Location** DERIVED_IMAGE_PARMS (Group)	**Mode** RDR-generating software **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a
Ops Keyword SPACECRAFT_CLOCK_CNT_PARTITION **PDS Keyword** same	**Valid Values** “1” **Type** integer	**Mode** Static Value **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a

OUTPUT METADATA (PRODUCT LABEL)		**INPUT METADATA (SOURCE)**
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
Definition Specifies the clock partition active for the SPACECRAFT_CLOCK_START_COUNT and SPACECRAFT_CLOCK_STOP_COUNT elements.	**Units** n/a **Location** IDENTIFICATION (Class)	**Type** n/a
Ops Keyword SPACECRAFT_CLOCK_START_COUNT **PDS Keyword** same **Definition** Specifies the value of the spacecraft clock at the beginning of a time period of interest. Format is “sssssssss.mmm”, stored as a floating point number where, “sssssssss” = seconds converted from clock’s coarse counter “mmm” = milliseconds converted from clock’s fine counter For MSL, the time period of interest is the beginning of data acquisition. The fractional component “mmm” is computed as follows: [(shift right 12 bits) / 2**20] * 1000	**Valid Values** sssssssss.mmm **Type** string(30) **Units** n/a **Location** 1) IDENTIFICATION (Class) 2) IDENTIFICATION (Class) 3) a. IDENTIFICATION (Class) b. IDENTIFICATION (Class) c. MINI_HEADER (Group)	**Mode** 1) DPO in XML format (referenced to APID Name in Appendix E) 2) DPO in XML format (referenced to APID Name in Appendix E) 3) DPO in XML format (referenced to APID Name in Appendix E) or Image DPO mini-header **Field as “<xml name>:[<element>]:[<element>]:<field>”** 1) Eng. Cameras “<IDPH DPO>:idph:sclk_seconds”, “<IDPH DPO>:idph:sclk_subseconds” 2) Chemcam “<Ancillary DPO>:image_sclk”, “<Ancillary DPO>:soh_sclk” 3) MMM Cameras a. “MMM_Image_Mini_Header[8]”, “MMM_Image_Mini_Header[9]”, “MMM_Image_Mini_Header[10]”, “MMM_Image_Mini_Header[11]” b. “<IDPH DPO>:cidph:sclk:seconds” c. “<Ancillary DPO>:sclk:seconds” NOTES: • For ChemCam, if APID Name contains string “Soh” in it, use field “soh_sclk”. Otherwise, use field “image_sclk”. • For MMM recovered data products, value is comprised of four bytes coming from Image DPO mini-header at byte offsets 8, 9, 10, and 11 per case “3a”. • For MMM non-recovered data products, if IDPH is present, value comes from IDPH DPO per case “3b”. • For MMM non-recovered data products, if IDPH is not present, value comes from Ancillary DPO per case “3c”. **Type** U32
Ops Keyword SPACECRAFT_CLOCK_STOP_COUNT **PDS Keyword** same **Definition** Specifies the value of the spacecraft clock at the end of a time period of interest.	**Valid Values** sssssssss.mmm **Type** string(30) **Units** n/a	**Mode** 1) DPO in XML format (referenced to APID Name in Appendix E), Calculation: - Stop SCLK Count = Start SCLK Count + (exp_time * 5.12) 2) DPO in XML format (referenced to APID Name in Appendix E) or Image DPO mini-header, Calculation: - Stop SCLK Count = Start SCLK Count + (exp_time / 1000) **Field as “<xml name>:[<element>]:[<element>]:<field>”**

<table>
<tr><th colspan="2">OUTPUT METADATA (PRODUCT LABEL)</th><th>INPUT METADATA (SOURCE)</th></tr>
<tr><th>• Ops Keyword
• PDS-Compliant Keyword
• Definition</th><th>• Valid Values (quoted)
• Type
• Units
• Keyword Location in Label</th><th>• Mode
• Metadata Field
• Type</th></tr>
<tr><td>Format is “ssssssssss.mmm”, stored as a floating point number where,
“ssssssssss” = seconds converted from clock’s coarse counter
“mmm” = milliseconds converted from clock’s fine counter

For MSL, the time period of interest is the end of data acquisition. The fractional component “mmm” is computed as follows:
[(shift right 12 bits) / 2**20] * 1000</td><td><u>Location</u>
IDENTIFICATION (Class)</td><td>1) <u>Eng. Cameras & Chemcam</u>
“<IDPH DPO>:idph:sclk_seconds”,
“<IDPH DPO>:idph:sclk_subseconds”,
“<IDPH DPO>:idph:exp_time”
2) <u>MMM Cameras</u>
a. “MMM_Image_Mini_Header[8]”,
“MMM_Image_Mini_Header[9]”,
“MMM_Image_Mini_Header[10]”,
“MMM_Image_Mini_Header[11]”
b. “<IDPH DPO>:cidph:sclk:seconds”
c. “<Ancillary DPO>:sclk:seconds”

NOTES:
• For Eng. Cameras and Chemcam, “exp_time” is in raw counts, with each count translating to 5.12 ms.
• For MMM recovered data products, value is comprised of four bytes coming from Image DPO mini-header at byte offsets 8, 9, 10, and 11 per case “2a”.
• For MMM non-recovered and non-Z-stack data products, if IDPH is present, value comes from IDPH DPO per case “2b”.
• For MMM non-recovered and non-Z-stack data products, if IDPH is not present, value comes from Ancillary DPO per case “2c”.
• For MMM Z-stack data products, value comes from EMD.

<u>Type</u>
U32</td></tr>
<tr><td><u>Ops Keyword</u>
SPECIAL_LINE

<u>PDS Keyword</u>
MSL:SPECIAL_LINE

<u>Definition</u>
For ChemCam on MSL, specifies the line number in a particular RMI image that has a special meaning. The context of the meaning is defined by SPECIAL_NAME.</td><td><u>Valid Values</u>
“497”

<u>Type</u>
integer

<u>Units</u>
n/a

<u>Location</u>
INSTRUMENT_STATE_PARMS (Group)</td><td><u>Mode</u>
Static value

<u>Field as “<xml name>:[<element>]:[<element>]:<field>”</u>
n/a

<u>Type</u>
U8</td></tr>
<tr><td><u>Ops Keyword</u>
SPECIAL_NAME

<u>PDS Keyword</u>
MSL:SPECIAL_NAME

<u>Definition</u>
Specifies how the coordinate defined as “(SPECIAL_LINE, SPECIAL_SAMPLE)” is to be interpreted; i.e. why that coordinate is special.

For ChemCam on MSL, identifies the LIBS laser “hot spot” in a</td><td><u>Valid Values</u>
“Location of the LIBS laser spot in the RMI image described by the camera model”

<u>Type</u>
string

<u>Units</u>
n/a

<u>Location</u>
INSTRUMENT_STATE_PARMS (Group)</td><td><u>Mode</u>
Static value

<u>Field as “<xml name>:[<element>]:[<element>]:<field>”</u>
n/a

<u>Type</u>
U8</td></tr>
</table>

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
particular RMI image. The text string used is “Location of the LIBS laser spot in the RMI image described by the camera model”.		
Ops Keyword SPECIAL_SAMPLE **PDS Keyword** MSL:SPECIAL_SAMPLE **Definition** For ChemCam on MSL, specifies the sample number in a particular RMI image that has a special meaning. The context of the meaning is defined by SPECIAL_NAME.	**Valid Values** “532” **Type** integer **Units** n/a **Location** INSTRUMENT_STATE_PARMS (Group)	**Mode** Static value **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** U8
Ops Keyword SPECTROMETER_CONTROL_BYTE **PDS Keyword** MSL:SPECTROMETER_CONTROL_BYTE **Definition** For ChemCam on MSL, 0x02 enables clocks, 0x04 enables temperature checks for the LIBS instrument.	**Valid Values** n/a **Type** integer **Units** n/a **Location** OBSERVATION_REQUEST_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as “<xml name>:[<element>]:[<element>]:<field>”** “<Ancillary DPO>:cmd_parameters:spectControlByte” **Type** U8
Ops Keyword SPECTROMETER_SELECT **PDS Keyword** MSL:SPECTROMETER_SELECT **Definition** For ChemCam on MSL, specifies the selected LIBS spectrometer and its mode of operation. For each bit, 1 = enable, 0 = disable. Descriptions of the valid values follow: a) “2D_ENABLE” (Hex 0x1) - enable 2D b) “UV” (Hex 0x2) - UV only c) “VIS” (Hex 0x4) - VIS only d) “VNIR” (Hex 0x8) - VNIR only The value is usually ““VNIR, VIS and UV” (0xE), currently not supported at this time.	**Valid Values** “2D_ENABLE”, “UV”, “VIS”, “VNIR”, “UNK” **Type** string **Units** n/a **Location** OBSERVATION_REQUEST_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as “<xml name>:[<element>]:[<element>]:<field>”** “<Ancillary DPO>:cmd_parameters:spectrometerSelect” **Type** U8
Ops Keyword SPECTROMETER_SERIAL_CLOCK **PDS Keyword**	**Valid Values** “0” = 2750 kHz “1” = 1380 kHz “2” = 922 kHz	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as “<xml name>:[<element>]:[<element>]:<field>”**

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
MSL:SPECTROMETER_SERIAL_CLOCK **Definition** For ChemCam on MSL, specifies the Spectrometer Serial Clock select for the LIBS instrument. The value is usually “2” or “3”.	“3” = 692 kHz “4” = 554 kHz “5” = 461 kHz “6” = 396 kHz “7” = 346 kHz “8” = 308 kHz “9” = 278 kHz “10” = 252 kHz “11” = 231 kHz “12” = 218 kHz “13” = 198 kHz “14” = 185 kHz “15” = 173 kHz **Type** integer **Units** n/a **Location** OBSERVATION_REQUEST_PARMS (Group)	“<Ancillary DPO>:cmd_parameters:spectrometerSerialClock” **Type** U8
Ops Keyword SPEC_AD_CONVERTUV **PDS Keyword** MSL:SPEC_AD_CONVERTUV **Definition** For ChemCam on MSL, specifies the adjustment in time between when the analog signal from the ChemCam UV channel CCD is clamped (held at its current level or higher) and when it is digitized. Typically adjusted to yield the maximum amplitude signal from the UV CCD.	**Valid Values** n/a **Type** integer **Units** n/a **Location** OBSERVATION_REQUEST_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as “<xml name>:[<element>]:[<element>]:<field>”** “<Ancillary DPO>:cmd_parameters:specAdConvertUV” **Type** U8
Ops Keyword SPEC_AD_CONVERTVIS **PDS Keyword** MSL:SPEC_AD_CONVERTVIS **Definition** For ChemCam on MSL, specifies the adjustment in time between when the analog signal from the ChemCam VIS channel CCD is clamped (held at its current level or higher) and when it is digitized. Typically adjusted to yield the maximum amplitude signal from the VIS CCD.	**Valid Values** n/a **Type** integer **Units** n/a **Location** OBSERVATION_REQUEST_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as “<xml name>:[<element>]:[<element>]:<field>”** “<Ancillary DPO>:cmd_parameters:specAdConvertVIS” **Type** U8
Ops Keyword SPEC_AD_CONVERTVNIR	**Valid Values** n/a **Type**	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as “<xml name>:[<element>]:[<element>]:<field>”**

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
PDS Keyword MSL:SPEC_AD_CONVERTVNIR **Definition** For ChemCam on MSL, specifies the adjustment in time between when the analog signal from the ChemCam VNIR channel CCD is clamped (held at its current level or higher) and when it is digitized. Typically adjusted to yield the maximum amplitude signal from the VNIR CCD.	integer **Units** n/a **Location** OBSERVATION_REQUEST_PARMS (Group)	"<Ancillary DPO>:cmd_parameters:specAdConvertVNIR" **Type** U8
Ops Keyword SPEC_IMAGE_TYPE **PDS Keyword** MSL:SPEC_IMAGE_TYPE **Definition** For ChemCam on MSL, specifies the type of spectroscopy data acquired by the LIBS instrument. Descriptions of the valid values follow: a) "1D_DARK_AVG" - 1-D dark current averaging b) "1D_DARK_SINGLE" - 1-D dark current single c) "1D_PASS_SINGLE" - 1-D passive single d) "1D_AVG" - 1-D average e) "1D_SINGLE" - 1-D single f) "2D_DIAG" - 2-D diagnostic g) "RESERVED" - Reserved NOTE: The "dark", "passive" and "diagnostic" are ineffective descriptors. Only the Ds and single/average matter. The value is usually "2".	**Valid Values** 0 = "1D_DARK_AVG" 1 = "1D_DARK_SINGLE" 2 = "1D_PASS_SINGLE" 3 = "1D_AVG" 4 = "1D_SINGLE" 5 = "2D_DIAG" 6 = "RESERVED" **Type** integer **Units** n/a **Location** OBSERVATION_REQUEST_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** "<Ancillary DPO>:cmd_parameters:spectImageType" **Type** U8
Ops Keyword SPEC_VERT_CLK **PDS Keyword** MSL:SPEC_VERT_CLK **Definition** For ChemCam on MSL, specifies the Vertical Clock select for the LIBS instrument. The value is usually "2".	**Valid Values** n/a **Type** integer **Units** n/a **Location** OBSERVATION_REQUEST_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** "<Ancillary DPO>:cmd_parameters:specVertClk" **Type** U8
Ops Keyword SPICE_FILE_ID **PDS Keyword** same **Definition**	**Valid Values** n/a **Type** string **Units**	**Mode** User parameter input **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type**

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
Specifies an abbreviated name or acronym which identifies a particular SPICE file.	n/a **Location** TELEMETRY (Class)	n/a
Ops Keyword SPICE_FILE_NAME **PDS Keyword** same **Definition** Specifies the names of the SPICE files used in processing the data.	**Valid Values** n/a **Type** string(180) **Units** n/a **Location** TELEMETRY (Class)	**Mode** User parameter input **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword STACK_1_LEVEL **PDS Keyword** MSL:STACK_1_LEVEL **Definition** For ChemCam on MSL, specifies the maximum current stack level for Stack 1 for the LIBS instrument.	**Valid Values** n/a **Type** integer **Units** n/a **Location** OBSERVATION_REQUEST_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** "<Ancillary DPO>:cmd_parameters:stack1Level" **Type** F32
Ops Keyword STACK_2_LEVEL **PDS Keyword** MSL:STACK_2_LEVEL **Definition** For ChemCam on MSL, specifies the maximum current stack level for Stack 2 for the LIBS instrument.	**Valid Values** n/a **Type** integer **Units** n/a **Location** OBSERVATION_REQUEST_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** "<Ancillary DPO>:cmd_parameters:stack2Level" **Type** F32
Ops Keyword STACK_3_LEVEL **PDS Keyword** MSL:STACK_3_LEVEL **Definition** For ChemCam on MSL, specifies the maximum current stack level for Stack 3 for the LIBS instrument.	**Valid Values** n/a **Type** integer **Units** n/a **Location** OBSERVATION_REQUEST_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** "<Ancillary DPO>:cmd_parameters:stack3Level" **Type** F32
Ops Keyword	**Valid Values**	**Mode**

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
STACK_DURATION **PDS Keyword** MSL:STACK_DURATION **Definition** For ChemCam on MSL, specifies the stack duration (in µsec) for the LIBS instrument.	n/a **Type** integer **Units** n/a **Location** OBSERVATION_REQUEST_PARMS (Group)	DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** "<Ancillary DPO>:cmd_parameters:stackDuration" **Type** U8
Ops Keyword STANDARD_DEVIATION **PDS Keyword** same **Definition** Specifies the standard deviation of the DN values in the image array.	**Valid Values** n/a **Type** float **Units** n/a **Location** IMAGE (Object)	**Mode** Calculation **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword START_AZIMUTH **PDS Keyword** same **Definition** Specifies the angular distance from a fixed reference position at which an image or observation starts. Azimuth is measured in a spherical coordinate system, in a plane normal to the principal axis. Azimuth values increase according to the right hand rule relative to the positive direction of the principal axis of the spherical coordinate system. When in a SURFACE_PROJECTION or SITE_DERIVED_GEOMETRY group, specifies the azimuth of the left edge of the output map. Applies to Cylindrical and Cylindrical-Perspective projections only. Mosaics only: If START_AZIMUTH is equal to STOP_AZIMUTH, that azimuth value applies to the left and right edge of the mosaic that spans a full 360 degrees.	**Valid Values** "0" to "360" **Type** float **Units** deg (<deg> unit tag required) **Location** • SURFACE_PROJECTION_PARMS (Group) • SITE_DERIVED_GEOMETRY_PARMS (Group)	**Mode** RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword START_IMAGE_ID **PDS Keyword** MSL:START_IMAGE_ID **Definition**	**Valid Values** n/a **Type** integer **Units**	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** 1) Chemcam RMI "<IDPH DPO>:cmd_parameters:startImageID" 2) MAHLI Range Map, Z-stack

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
For ChemCam on MSL, specifies the commanded image start in case the SRAM on the ChemCam Mast Unit (CCMU) goes bad.	n/a **Location** 1) OBSERVATION_REQUEST_PARMS (Group) 2) MINI_HEADER (Group) ** ** Only MAHLI Z-Stack & Range Map	"MMM_Image_Mini_Header[TBD]" **Type** U8
Ops Keyword START_ROW_UV **PDS Keyword** MSL:START_ROW_UV **Definition** For ChemCam on MSL, specifies the beginning row of the ChemCam CCD over whith UV integration occurs.	**Valid Values** n/a **Type** integer **Units** n/a **Location** OBSERVATION_REQUEST_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** "<Ancillary DPO>:cmd_parameters:startRowUV" **Type** U16
Ops Keyword START_ROW_VIS **PDS Keyword** MSL:START_ROW_VIS **Definition** For ChemCam on MSL, specifies the beginning row of the ChemCam CCD over whith VIS integration occurs.	**Valid Values** n/a **Type** integer **Units** n/a **Location** OBSERVATION_REQUEST_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** "<Ancillary DPO>:cmd_parameters:startRowVIS" **Type** U16
Ops Keyword START_ROW_VNIR **PDS Keyword** MSL:START_ROW_VNIR **Definition** For ChemCam on MSL, specifies the beginning row of the ChemCam CCD over whith VNIR integration occurs.	**Valid Values** n/a **Type** integer **Units** n/a **Location** OBSERVATION_REQUEST_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** "<Ancillary DPO>:cmd_parameters:startRowVNIR" **Type** U16
Ops Keyword START_TIME **PDS Keyword** same **Definition** Specifies the date and time of the beginning of an event or observation (whether it be a spacecraft, ground-based, or system event) in UTC system format.	**Valid Values** <YYYY>-<MM>-<DD>T<hh>:<mm>:<ss>[.<fff>] NOTE: Value will be uncalibrated if SPICE kernels unavailable. **Type** time **Units** n/a	**Mode** 1) a. DPO in XML format (referenced to APID Name in Appendix E), Calculation using SPICE Kernels: - SCLK b. EMD in XML format, Calculation using SPICE Kernels: - SCLK 2) DPO in XML format (referenced to APID Name in Appendix E), Calculation using SPICE Kernels: - SCLK 3) DPO in XML format (referenced to APID Name in Appendix E) or Image DPO mini-header, Calculation using SPICE Kernels:

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
For MSL, the time period of interest is returned from SPICE suburoutines and based on the beginning of data acquisition.	**Location** IDENTIFICATION (Class)	- SCLK **Field as “<xml name>:[<element>]:[<element>]:<field>”** • Eng. Cameras 1) a. “<IDPH DPO>:idph:sclk_seconds”, “<IDPH DPO>:idph:sclk_subseconds” b. “MslEarthProductMetadata:MslProductMetadata: OnboardCreationTime” • Chemcam 2) “<Ancillary DPO>:image_sclk”, “<Ancillary DPO>:soh_sclk” • MMM Cameras 3) a. “MMM_Image_Mini_Header[8]”, “MMM_Image_Mini_Header[9]”, “MMM_Image_Mini_Header[10]”, “MMM_Image_Mini_Header[11]” b. “<IDPH DPO>:cidph:sclk:seconds” c. “<Ancillary DPO>:sclk:seconds” NOTES: • For ChemCam, if APID Name contains string “Soh” in it, use field “soh_sclk”. Otherwise, use field “image_sclk”. • For MMM recovered data products, value is comprised of four bytes coming from Image DPO mini-header atbyte offsets 8, 9, 1 0 and 11 per case “3a”. • For MMM non-recovered data products, if IDPH is present, value comes from IDPH DPO per case “3b”. • For MMM non-recovered data products, if IDPH is not present, value comes from Ancillary DPO per case “3c”. **Type** 1) a. U32 b. n/a 2) U32 3) U32
Ops Keyword STEREO_BASELINE **PDS Keyword** none **Definition** Specifies the separation, in meters, between the two cameras used for processing of the stereo image.	**Valid Values** n/a **Type** float **Units** n/a **Location** DERIVED_IMAGE_PARMS (Group)	**Mode** RDR-generating software **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a
Ops Keyword STEREO_PRODUCT_ID	**Valid Values** n/a	**Mode** RDR-generating software

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
PDS Keyword MSL:STEREO_PRODUCT_ID **Definition** Specifies the PRODUCT_ID for the stereo or coregistration partner that this image was correlated against. The pixel values in the correlation product represent line/sample coordinates in the image identified by STEREO_PRODUCT_ID.	**Type** string **Units** n/a **Location** DERIVED_IMAGE_PARMS (Group)	**Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a
Ops Keyword STOP_AZIMUTH **PDS Keyword** same **Definition** Specifies the angular distance from a fixed reference position at which an image or observation stops. Azimuth is measured in a spherical coordinate system, in a plane normal to the principal axis. Azimuth values increase according to the right hand rule relative to the positive direction of the principal axis of the spherical coordinate system. When in a SURFACE_PROJECTION or SITE_DERIVED_GEOMETRY group, specifies the azimuth of the right edge of the output map. Applies to Cylindrical and Cylindrical-Perspective projections only. Mosaics only: If STOP_AZIMUTH is equal to START_AZIMUTH, that azimuth value applies to the left and right edge of the mosaic that spans a full 360 degrees.	**Valid Values** “0” to “360” **Type** float **Units** deg (<deg> unit tag required) **Location** • SURFACE_PROJECTION_PARMS (Group) • SITE_DERIVED_GEOMETRY_PARMS (Group)	**Mode** RDR-generating software **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a
Ops Keyword STOP_ROW_UV **PDS Keyword** MSL:STOP_ROW_UV **Definition** For ChemCam on MSL, specifies the end row of the ChemCam CCD over whith UV integration occurs.	**Valid Values** n/a **Type** integer **Units** n/a **Location** OBSERVATION_REQUEST_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as “<xml name>:[<element>]:[<element>]:<field>”** “<Ancillary DPO>:cmd_parameters:stopRowUV” **Type** U16
Ops Keyword STOP_ROW_VIS **PDS Keyword** MSL:STOP_ROW_VIS **Definition**	**Valid Values** n/a **Type** integer **Units**	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as “<xml name>:[<element>]:[<element>]:<field>”** “<Ancillary DPO>:cmd_parameters:stopRowVIS” **Type**

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
For ChemCam on MSL, specifies the end row of the ChemCam CCD over whith VIS integration occurs.	n/a **Location** OBSERVATION_REQUEST_PARMS (Group)	U16
Ops Keyword STOP_ROW_VNIR **PDS Keyword** MSL:STOP_ROW_VNIR **Definition** For ChemCam on MSL, specifies the end row of the ChemCam CCD over whith VNIR integration occurs.	**Valid Values** n/a **Type** integer **Units** n/a **Location** OBSERVATION_REQUEST_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** "<Ancillary DPO>:cmd_parameters:stopRowVNIR" **Type** U16
Ops Keyword STOP_TIME **PDS Keyword** same **Definition** Specifies the date and time of the end of an event or observation (whether it be a spacecraft, ground-based, or system event) in UTC system format.	**Valid Values** <YYYY>-<MM>-<DD>T<hh>:<mm>:<ss>[.<fff>] NOTE: Value will be uncalibrated if SPICE kernels unavailable. **Type** time **Units** n/a **Location** IDENTIFICATION (Class)	**Mode** 1) DPO in XML format (referenced to APID Name in Appendix E), Calculation using SPICE Kernels: - SCLK 2) DPO in XML format (referenced to APID Name in Appendix E) or Image DPO mini-header, Calculation using SPICE Kernels: - SCLK **Field as "<xml name>:[<element>]:[<element>]:<field>"** • Eng. Cameras & Chemcam 1) a. "<IDPH DPO>:idph:sclk_seconds" b. "<IDPH DPO>:idph:sclk_subseconds" c. "<IDPH DPO>:idph:exp_time" • MMM Cameras 2) a. "MMM_Image_Mini_Header[8]", "MMM_Image_Mini_Header[9]", "MMM_Image_Mini_Header[10]", "MMM_Image_Mini_Header[11]" b. "<IDPH DPO>:cidph:sclk:seconds" c. "<Ancillary DPO>:sclk:seconds" NOTES: • For Eng. Cameras and Chemcam, "exp_time" is in raw counts, with each count translating to 5.12 ms. • For MMM recovered data products, value is comprised of four bytes coming from Image DPO mini-header at byte offsets 8, 9, 10 and 11 per case "2a". • For MMM non-recovered and non-Z-stack data products, if IDPH is present, value comes from IDPH DPO per case "2b". • For MMM non-recovered and non-Z-stack data products, if IDPH is not present, value comes from Ancillary DPO per case "2c". • For MMM Z-stack data products, value comes from EMD.

<table>
<tr><th colspan="2">OUTPUT METADATA (PRODUCT LABEL)</th><th>INPUT METADATA (SOURCE)</th></tr>
<tr><th>• Ops Keyword
• PDS-Compliant Keyword
• Definition</th><th>• Valid Values (quoted)
• Type
• Units
• Keyword Location in Label</th><th>• Mode
• Metadata Field
• Type</th></tr>
<tr><td></td><td></td><td><u>Type</u>
1) a. U32
b. U32
c. U16
2) U32</td></tr>
<tr><td><u>Ops Keyword</u>
STRIPING_COUNT

<u>PDS Keyword</u>
MSL:STRIPING_COUNT

<u>Definition</u>
Specifies the number of stripes (N) used during dark current mitigation within image acquisition.

Image “striping” is comprised of reading out the image in N different parts (“stripes”) using hardware windowing mode, (where N is a commanded number from 0 to 1024), using N separate exposures (with identical exposure times). These successive stripes correspond to physically different locations on the CCD. As part of the striping exposure technique, also include a commandable overlap (M rows) to allow each successive row to “cover” the image pixels towards the readout region.</td><td><u>Valid Values</u>
n/a

<u>Type</u>
integer

<u>Units</u>
n/a

<u>Location</u>
TELEMETRY (Class)</td><td><u>Mode</u>
DPO in XML format (referenced to APID Name in Appendix E)

<u>Field as “<xml name>:[<element>]:[<element>]:<field>”</u>
“<IDPH DPO>:idph:stripes”

<u>Type</u>
U16</td></tr>
<tr><td><u>Ops Keyword</u>
STRIPING_OVERLAP_ROWS

<u>PDS Keyword</u>
MSL:STRIPING_OVERLAP_ROWS

<u>Definition</u>
Specifies the number of rows (M) of striping overlap used during dark current mitigation within image acquisition.

Image “striping” is comprised of reading out the image in N different parts (“stripes”) using hardware windowing mode, (where N is a commanded number from 0 to 1024), using N separate exposures (with identical exposure times). These successive stripes correspond to physically different locations on the CCD. As part of the striping exposure technique, also include a commandable overlap (M rows) to allow each successive row to “cover” the image pixels towards the readout region.</td><td><u>Valid Values</u>
• <u>Eng. Cameras</u>
n/a
• <u>MMM Cameras</u>
“UNK”

<u>Type</u>
integer

<u>Units</u>
n/a

<u>Location</u>
TELEMETRY (Class)</td><td><u>Mode</u>
DPO in XML format (referenced to APID Name in Appendix E)

<u>Field as “<xml name>:[<element>]:[<element>]:<field>”</u>
“<IDPH DPO>:idph:overlap”

<u>Type</u>
U16</td></tr>
<tr><td><u>Ops Keyword</u>
SUBFRAME_TYPE

<u>PDS Keyword</u>
same</td><td><u>Valid Values</u>
• <u>Eng. Cameras</u>
0 = “NONE”
1 = “SW_ONLY”
2 = “HW_COND”</td><td><u>Mode</u>
1) DPO in XML format (referenced to APID Name in Appendix E)
2) DPO in XML format (referenced to APID Name in Appendix E) or Image DPO mini-header

<u>Field as “<xml name>:[<element>]:[<element>]:<field>”</u></td></tr>
</table>

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
Definition Specifies the method of subframing performed on the image. Descriptions of the valid values follow: a) "NONE" - No subframing requested. b) "SW_ONLY" - Software processsing only. c) "HW_COND" - Use hardware only if compatible. d) "HW_SW" - Use hardware then software. e) "SUN_NO_IMG" - If the sun is found, send a subframed image of the sun. If sun is not found, send back no image. f) "SUN_FULL" - If the sun is found, send a subframed image of the sun. If the sun is not found, send back the entire image.	3 = "HW_SW" 4 = "SUN_NO_IMG" 5 = "SUN_FULL" • MMM Cameras - If width is 1648 and height is 1200, then "NONE" - Otherwise, "HW_ONLY" **Type** string **Units** n/a **Location** SUBFRAME_REQUEST_PARMS (Group)	• Eng. Cameras 1) "<IDPH DPO>:idph:params:subframe" • MMM Cameras 2) a. "<Ancillary DPO>:cmd_arguments_image:window_width" b. "<Ancillary DPO>:cmd_arguments_image:window_height" c. "MMM_Image_Mini_Header[22]" d. "MMM_Image_Mini_Header[23]" NOTES: • For MMM non-recovered data products (cases "a" and "b"), float values come from the Ancillary DPO. • For MMM recovered data products (cases "c" and "d"), values analogous to cases "a" (window_width) and "b" (window_height) are each comprised of one byte coming from Image DPO mini-header at byte offsets 22 and 23, respectively. • For MMM, parm "window_width" is number of image samples. • For MMM, parm "window_height" is number of image lines. • For MMM non-Thumbnail data products, multiply by factor of 8 to convert to correct line/sample value. **Type** 1) enum 2) U8
Ops Keyword SUN_VIEW_DIRECTION **PDS Keyword** same **Definition** Specifies a unit vector identifying the sun viewing direction.	**Valid Values** n/a **Type** float array[3] **Units** n/a **Location** ROVER_DERIVED_GEOMETRY_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** • Eng. Cameras & Chemcam "<IDPH DPO>:idph:sun_dir[3]" **Type** F32[3]
Ops Keyword SURFACE_GROUND_LOCATION **PDS Keyword** same **Definition** Specifies any point on the surface (for SURFACE_MODEL_TYPE of "PLANE"), or the center of the sphere (for the three "SPHERE" types). This point is measured in the coordinates specified by the REFERENCE_COORD_SYSTEM_* keywords in the same group.	**Valid Values** n/a **Type** float array[3] **Units** meters **Location** SURFACE_MODEL_PARMS (Group)	**Mode** RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword SURFACE_MODEL_FILE_NAME	**Valid Values** n/a	**Mode** RDR-generating software

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
PDS Keyword MSL:SURFACE_MODEL_FILE_NAME **Definition** Specifies the name of an XYZ, Z-component or other RDR used as a digital elevation model onto which the data were projected.	**Type** string **Units** n/a **Location** SURFACE_MODEL_PARMS (Group)	**Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword SURFACE_MODEL_TYPE **PDS Keyword** same **Definition** Specifies the type of surface used for the reprojection performed during the mosaicking process. The valid values are defined as: a) "INFINITY" - refers to an infinitely distant "surface" in all directions and has no parameters. b) "PLANE" - refers to a flat plane and require the SURFACE_NORMAL_VECTOR and SURFACE_GROUND_LOCATION keywords as parameters. c) "SPHERE" – refers to a spherical model where the camera is at the center of the sphere. The origin is specified by SURFACE_GROUND_LOCATION, and the radius by the first element of SURFACE_NORMAL_VECTOR. d) "SPHERE1" – refers to a general sphere model, whose center is defined by SURFACE_GROUND_LOCATION, and radius by the first element of SURFACE_NORMAL_VECTOR. If the camera is outside the sphere, the first intersection with the sphere is used; this makes it useful for modeling hills or rocks. e) "SPHERE2" – just like SPHERE1, except the second intersection with the sphere is used; this makes it useful for modeling craters.	**Valid Values** "INFINITY", "PLANE", "SPHERE", "SPHERE1", "SPHERE2" **Type** string **Units** n/a **Location** SURFACE_MODEL_PARMS (Group)	**Mode** RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword SURFACE_NORMAL_VECTOR **PDS Keyword** same **Definition** Specifies a vector normal to the surface (for SURFACE_MODEL_TYPE of "PLANE"). This vector is measured in the coordinates specified by the	**Valid Values** n/a **Type** float array[3] **Units** n/a **Location**	**Mode** RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a

OUTPUT METADATA (PRODUCT LABEL)		**INPUT METADATA (SOURCE)**
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
REFERENCE_COORD_SYSTEM_* keywords in the same group. For the "SPHERE" surface model types, the first element is used to specify the radius; the other two elements are unused. This is a misuse of this keyword's definition, which is retained for historical reasons.	SURFACE_MODEL_PARMS (Group)	
Ops Keyword TARGET_NAME **PDS Keyword** same **Definition** Specifies a target. The target may be a planet, satellite, ring, region, feature, asteroid or comet. See TARGET_TYPE.	**Valid Values** For EDRs … Value “MARS”, “CALIBRATION” For RDRs … Value any **Type** string(30) **Units** n/a **Location** IDENTIFICATION (Class)	**Mode** Calculation: - By algorithm to determine if looking at calibration target … if not, then value is “MARS” **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a
Ops Keyword TARGET_TYPE **PDS Keyword** same **Definition** Specifies the type of a named target.	**Valid Values** “CALIBRATION”, “DUST”, “SUN”, “PLANET”, “SATELLITE”, “N/A” **Type** string **Units** n/a **Location** IDENTIFICATION (Class)	**Mode** Static Value **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a
Ops Keyword TELEMETRY_PROVIDER_ID **PDS Keyword** same **Definition** Specifies the provider and version of the telemetry data used in the generation of this data.	**Valid Values** “MPCS_MSL_DP” **Type** string **Units** n/a **Location** TELEMETRY (Class)	**Mode** User parameter input **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a
Ops Keyword TELEMETRY_SOURCE_CHECKSUM **PDS Keyword**	**Valid Values** n/a **Type**	**Mode** EMD in XML format **Field as “<xml name>:[<element>]:[<element>]:<field>”**

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
MSL:TELEMETRY_SOURCE_CHECKSUM **Definition** Checksum for the source product from which this product was derived. For MSL, it is the sum of each (unsigned) byte in the data areas of all DPOs. It does not include the DPO headers.	integer **Units** n/a **Location** TELEMETRY (Class)	"MslEarthProductMetadata:MslProductMetadata:ProductChecksum" **Type** n/a
Ops Keyword TELEMETRY_SOURCE_HOST_NAME **PDS Keyword** MSL:TELEMETRY_SOURCE_HOST_NAME **Definition** Specifies the name of the host venue that provides the telemetry source used in creation of this data set. For MSL, example is "mslmstbgds1". See also TELEMETRY_SOURCE_NAME.	**Valid Values** n/a **Type** string **Units** n/a **Location** TELEMETRY (Class)	**Mode** EMD in XML format **Field as "<xml name>:[<element>]:[<element>]:<field>"** "MslEarthProductMetadata:SessionInformation:Venue:Host" **Type** n/a
Ops Keyword TELEMETRY_SOURCE_NAME **PDS Keyword** same **Definition** Specifies the name of the telemetry source used in creation of this data set. For MSL, example is "mslgdsdev2". See also TELEMETRY_SOURCE_HOST_NAME.	**Valid Values** n/a **Type** string **Units** n/a **Location** TELEMETRY (Class)	**Mode** EMD in XML format **Field as "<xml name>:[<element>]:[<element>]:<field>"** "MslEarthProductMetadata:MslProductMetadata:DataFileName" **Type** n/a
Ops Keyword TELEMETRY_SOURCE_SCLK_START **PDS Keyword** MSL:TELEMETRY_SOURCE_SCLK_START **Definition** Specifies the value of the spacecraft clock (in seconds) at the creation time of the source product from which this product was derived. This differs from SPACECRAFT_CLOCK_START_COUNT, which is the time the instrument acquired the data. For MSL, it refers to the creation time (DVT) of the onboard DPO and comes from the secondary packet header. Note that this is the SCLK used by Data Management operationally to	**Valid Values** sssssssss.mmm **Type** string(30) **Units** n/a **Location** TELEMETRY (Class)	**Mode** EMD in XML format **Field as "<xml name>:[<element>]:[<element>]:<field>"** "MslEarthProductMetadata:MslProductMetadata:DvtCoarse", "MslEarthProductMetadata:MslProductMetadata:DvtFine" **Type** n/a

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
identify data products.		
Ops Keyword TELEMETRY_SOURCE_SIZE **PDS Keyword** MSL:TELEMETRY_SOURCE_SIZE **Definition** Specifies the length in bytes of the source product from which this product was derived. For MSL, it is the length of the user portion of the Data Product Object (DPO).	**Valid Values** n/a **Type** integer **Units** n/a **Location** TELEMETRY (Class)	**Mode** EMD in XML format **Field as "<xml name>:[<element>]:[<element>]:<field>"** "MslEarthProductMetadata:MslProductMetadata:ProductFileSize" **Type** n/a
Ops Keyword TELEMETRY_SOURCE_START_TIME **PDS Keyword** MSL:TELEMETRY_SOURCE_START_TIME **Definition** Specifies the creation time of the source product from which this product was derived. It is the same as TELEMETRY_SOURCE_SCLK_START converted to Spacecraft Event Time (SCET).	**Valid Values** <YYYY>-<DDD>T<hh>:<mm>:<ss>[.<fff>] **Type** time **Units** n/a **Location** IDENTIFICATION (Class)	**Mode** EMD in XML format **Field as "<xml name>:[<element>]:[<element>]:<field>"** "MslEarthProductMetadata:MslProductMetadata:Partlist:Part:Scet" **Type** n/a
Ops Keyword TELEMETRY_SOURCE_TYPE **PDS Keyword** MSL:TELEMETRY_SOURCE_TYPE **Definition** Specifies the classification of the source of the telemetry used in creating this data set.	**Valid Values** "DATA PRODUCT" **Type** string(12) **Units** n/a **Location** TELEMETRY (Class)	**Mode** User parameter input **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword TIME_BETWEEN_SHOTS **PDS Keyword** MSL:CCAM_TIME_BETWEEN_SHOTS **Definition** For ChemCam on MSL, specifies the inverse of the LIBS laser shot repetition rate in Hertz (1/shot frequency).	**Valid Values** n/a **Type** integer **Units** n/a **Location** OBSERVATION_REQUEST_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** "<Ancillary DPO>:cmd_parameters:timeBetweenShots" **Type** U8
Ops Keyword TRANSMISSION_PATH	**Valid Values** n/a	**Mode** EMD in XML format

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
PDS Keyword MSL:TRANSMISSION_PATH **Definition** Routing status at time of MPDU (Metadata PDU) generation. Indicates the actual transmission paths (routes) of the Data Product.	**Type** string **Units** n/a **Location** TELEMETRY (Class)	**Field as "<xml name>:[<element>]:[<element>]:<field>"** "MslEarthProductMetadata:MslProductMetadata: TransmissionStatus" **Type** n/a
Ops Keyword VALID_MAXIMUM_PIXEL **PDS Keyword** MSL:VALID_MAXIMUM_PIXEL **Definition** Specifies the maximum pixel DN value that is considered valid for some use. For MSL Chemcam, specifies the threshold for pixels used by the autoexposure algorithm.	**Valid Values** "0" to "1023" **Type** integer **Units** n/a **Location** OBSERVATION_REQUEST_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** Chemcam "<Ancillary DPO>:cmd_parameters:upperThreshold" **Type** U16
Ops Keyword VALID_MINIMUM_PIXEL **PDS Keyword** MSL:VALID_MINIMUM_PIXEL **Definition** Specifies the minimum pixel DN value that is considered valid for some use. For MSL Chemcam, specifies the threshold for pixels used by the autoexposure algorithm.	**Valid Values** "0" to "1023" **Type** integer **Units** n/a **Location** OBSERVATION_REQUEST_PARMS (Group)	**Mode** DPO in XML format (referenced to APID Name in Appendix E) **Field as "<xml name>:[<element>]:[<element>]:<field>"** Chemcam "<Ancillary DPO>:cmd_parameters:lowerThreshold" **Type** U16
Ops Keyword VIRTUAL_CHANNEL_ID **PDS Keyword** MSL:VIRTUAL_CHANNEL_ID **Definition** The Virtual Channel Identifier is used by MSL to identify the RCE string generating the Transfer Frame, and to indicate the type of data flowing in the telemetry virtual channel. RCE String A is indicated by all Virtual Channel Identifier values having a '0' as the high bit (e.g., virtual channels 0 to 31); RCE String B is indicated by all Virtual Channel Identifier values having a '1' for the high bit (e.g., virtual channels 32 to 63).	**Valid Values** "0" to "63" **Type** string **Units** n/a **Location** TELEMETRY (Class)	**Mode** EMD in XML format **Field as "<xml name>:[<element>]:[<element>]:<field>"** "MslEarthProductMetadata:MslProductMetadata:Vcid" **Type** n/a
Ops Keyword X_AXIS_MAXIMUM	**Valid Values** n/a	**Mode** RDR-generating software

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
PDS Keyword same **Definition** Specifies the value of the X coordinate (measured in the projection frame) of a Vertical, Orthographic or Orthorectified projection at the top of the image. Note that +X is at the top of the image and +Y is at the right, so +X corresponds to North in the Vertical projection.	**Type** float **Units** meters (<m> unit tag required) **Location** SURFACE_PROJECTION_PARMS (Group)	**Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword X_AXIS_MINIMUM **PDS Keyword** same **Definition** Specifies the value of the X coordinate (measured in the projection frame) of a Vertical, Orthographic or Orthorectified projection at the bottom of the image.	**Valid Values** n/a **Type** float **Units** meters (<m> unit tag required) **Location** SURFACE_PROJECTION_PARMS (Group)	**Mode** RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword Y_AXIS_MAXIMUM **PDS Keyword** same **Definition** Specifies the value of the Y coordinate (measured in the projection frame) of a Vertical, Orthographic or Orthorectified projection at the right edge of the image.	**Valid Values** n/a **Type** float **Units** meters (<m> unit tag required) **Location** SURFACE_PROJECTION_PARMS (Group)	**Mode** RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword Y_AXIS_MINIMUM **PDS Keyword** same **Definition** Specifies the value of the Y coordinate (measured in the projection frame) of a Vertical, Orthographic or Orthorectified projection at the left edge of the image.	**Valid Values** n/a **Type** float **Units** meters (<m> unit tag required) **Location** SURFACE_PROJECTION_PARMS (Group)	**Mode** RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a
Ops Keyword ZERO_ELEVATION_LINE **PDS Keyword** same **Definition** Specifies the image line representing 0.0 degree elevation.	**Valid Values** n/a **Type** float **Units** pixel (<pixel> unit tag required)	**Mode** RDR-generating software **Field as "<xml name>:[<element>]:[<element>]:<field>"** n/a **Type** n/a

OUTPUT METADATA (PRODUCT LABEL)		INPUT METADATA (SOURCE)
• Ops Keyword • PDS-Compliant Keyword • Definition	• Valid Values (quoted) • Type • Units • Keyword Location in Label	• Mode • Metadata Field • Type
Applies to Cylindrical projections.	**Location** SURFACE_PROJECTION_PARMS (Group)	
Ops Keyword ZERO_EXPOSURE_IMAGE_ID **PDS Keyword** MSL:ZERO_EXPOSURE_IMAGE_ID **Definition** Specifies the value of PRODUCT_ID for the zero-exposure EDR that is subtracted during RDR generation to account for shutter smear and masked-region dark current. NOTE: This keyword is only placed in the PDS label if a zero-exposure image EDR product was used during on-ground calibration. It is not set when on-board shutter-subtraction is done.	**Valid Values** n/a **Type** string **Units** n/a **Location** DERIVED_IMAGE_PARMS (Group)	**Mode** RDR-generating software **Field as “<xml name>:[<element>]:[<element>]:<field>”** n/a **Type** n/a

APPENDIX G – Label Keywords Omitted from PDS Label

Label Keyword Name Omitted from PDS label	ODL Label Location
ODL_VERSION_ID	1st label record
STEREO_BASELINE	Group DERIVED_IMAGE_PARMS
CAMERA_SERIAL_NUMBER	Group GEOMETRIC_CAMERA_MODEL
MAGIC_NUMBERS	Group MINI_HEADER
INST_CMPRS_DEFERRED_FLAG	Groups IMAGE_REQUEST_PARMS, THUMBNAIL_REQUEST_PARMS, REFERENCE_PIXEL_REQUEST_PARMS, COMPRESSION_PARMS
INST_CMPRS_COLOR_MODE	Group COMPRESSION_PARMS
CLASSIFIER_BAND_HUE	Group DERIVED_IMAGE_PARMS
CLASSIFIER_BAND_INDEX	Group DERIVED_IMAGE_PARMS
CLASSIFIER_BAND_INDEX_NAME	Group DERIVED_IMAGE_PARMS
CLASSIFIER_LABEL_HUE	Group DERIVED_IMAGE_PARMS
CLASSIFIER_LABEL_INDEX	Group DERIVED_IMAGE_PARMS
CLASSIFIER_LABEL_INDEX_NAME	Group DERIVED_IMAGE_PARMS
CLASSIFIER_MAX_RANGE	Group DERIVED_IMAGE_PARMS
CLASSIFIER_PROJECTION_SCALE	Group DERIVED_IMAGE_PARMS
CLASSIFIER_VERSION	Group DERIVED_IMAGE_PARMS
REGION_COUNT	Group DERIVED_IMAGE_PARMS
LINEARIZATION_PRODUCT_ID	Group DERIVED_IMAGE_PARMS