

Whole Sky Imager Retrieval Guide

November 2003

Tim P. Tooman
Exploratory Systems Technologies
Sandia National Laboratories

Work supported by the U.S. Department of Energy,
Office of Energy Research, Office of Health and Environmental Research

Contents

1. Overview	1
1.1 Features of the Instrument	1
1.1.1 Description.....	1
1.1.2 Evolution.....	1
1.2 Instrument Deployment	2
1.2.1 Southern Great Plains	2
1.2.2 North Slope of Alaska.....	2
1.2.3 Tropical Western Pacific	3
1.2.4 Nomenclature.....	3
1.3 Data Collection Strategy	3
1.3.1 Discussion.....	3
1.3.2 Old Style	4
1.3.3 New Style.....	4
1.4 Data and Data Product Types	5
1.4.1 Raw Data Products.....	5
1.4.2 Visualization Data Products.....	6
1.4.3 Radiance Data Product.....	6
1.4.4 Sky Feature Data Product	6
1.4.5 Summary Data Products	6
1.4.6 Calibration Data Product	7
1.4.7 Utility Data Products	7
1.5 Data Quality Control.....	7
1.5.1 Items that Affect Data Quality	7
1.5.2 Data Quality Procedure.....	9
1.5.3 Potential Biases.....	10
1.6 Instrument Calibration	11
1.6.1 Calibration Factors.....	12
1.6.2 Laboratory Calibration.....	12
1.6.3 Field Calibration	13
2. Raw Products	13
2.1 Raw Data Files.....	13
2.1.1 Purpose	13
2.1.2 Generation and Storage.....	13
2.1.3 Naming Convention.....	14
2.1.4 Dimensions, Variables, and Coding.....	15
2.1.5 File Structure	15
2.1.6 Header Information.....	15
2.1.7 Algorithm.....	19
2.1.8 Usage Hints.....	19
2.1.9 Limitations.....	19
3. Visualization Products	20
3.1 Sky Picture File.....	20
3.1.1 Purpose	20

3.1.2	Naming Convention.....	20
3.1.3	Dimensions, Variables, and Coding.....	20
3.1.4	Algorithm.....	21
3.1.5	Usage Hints.....	21
3.1.6	Limitations.....	21
3.2	Movie File.....	22
3.2.1	Purpose	22
3.2.2	Naming Convention.....	23
3.2.3	Dimensions, Variables, and Coding.....	23
3.2.4	Algorithm.....	23
3.2.5	Usage Hints.....	23
3.2.6	Limitations.....	24
3.3	Sky Feature Spectral Retrieval Picture File.....	24
3.3.1	Purpose	24
3.3.2	Naming Convention.....	24
3.3.3	Dimensions, Variables, and Coding.....	24
3.3.4	Algorithm.....	25
3.3.5	Usage Hints.....	25
3.3.6	Limitations.....	25
3.4	Sky Feature Optical Density Retrieval Picture File.....	25
3.4.1	Purpose	25
3.4.2	Naming Convention.....	26
3.4.3	Dimensions, Variables, and Coding.....	27
3.4.4	Algorithm.....	27
3.4.5	Usage Hints.....	27
3.4.6	Limitations.....	28
4.	Radiance Product	29
4.1	Radiance Data File.....	29
4.1.1	Purpose	29
4.1.2	Naming Convention.....	29
4.1.3	Dimensions, Variables, and Coding.....	29
4.1.4	Algorithm.....	40
4.1.5	Usage Hints.....	48
4.1.6	Limitations.....	48
5.	Sky Feature Product.....	49
5.1	Sky Feature File.....	49
5.1.1	Purpose	49
5.1.2	Naming Convention.....	50
5.1.3	Dimensions, Variables, and Coding.....	50
5.1.4	Algorithm.....	56
5.1.5	Usage Hints.....	68
5.1.6	Limitations.....	69
6.	Summary Products	70
6.1	Patch Radiance File	70
6.1.1	Purpose	70
6.1.2	Naming Convention.....	71

6.1.3	Dimensions, Variables, and Coding.....	71
6.1.4	Algorithm.....	75
6.1.5	Usage Hints.....	75
6.1.6	Limitations.....	76
6.2	Sky Feature Summary File	76
6.2.1	Purpose	76
6.2.2	Naming Convention.....	76
6.2.3	Dimensions, Variables, and Coding.....	76
6.2.4	Algorithm.....	81
6.2.5	Usage Hints.....	81
6.2.6	Limitations.....	82
6.3	Sky Feature Plot File	82
6.3.1	Purpose	82
6.3.2	Naming Convention.....	83
6.3.3	Dimensions, Variables, and Coding.....	83
6.3.4	Algorithm.....	84
6.3.5	Usage Hints.....	84
6.3.6	Limitations.....	85
7.	Calibration Product	85
7.1	Calibration File	85
7.1.1	Purpose	85
7.1.2	Naming Convention.....	85
7.1.3	Dimensions, Variables, and Coding.....	86
7.1.4	Algorithm.....	96
7.1.5	Usage Hints.....	102
7.1.6	Limitations.....	103
8.	Utility Products	103
8.1	File Existence Plot File.....	103
8.1.1	Purpose	103
8.1.2	Naming Convention.....	104
8.1.3	Dimensions, Variables, and Coding.....	104
8.1.4	Algorithm.....	105
8.1.5	Usage Hints.....	105
8.1.6	Limitations.....	105
8.2	Log Files	107
8.2.1	Purpose	107
8.2.2	Naming Convention.....	107
8.2.3	Dimensions, Variables, and Coding.....	108
8.2.4	Algorithm.....	108
8.2.5	Usage Hints.....	109
8.2.6	Limitations.....	109

Figures

1	Sample sky picture	22
2	Sample sky feature spectral retrieval picture	26
3	Sample sky feature optical density retrieval picture	28
4	Density of radiance ratio values	60
5	Sample sky feature plot file.....	84
6	Sample file existence plot	106

Table

1	Data file name site codes.....	3
2	Sky feature spectral retrieval picture color codes	25
3	Sky feature optical density retrieval picture color codes	27
4	Radiance data file dimensions.....	30
5	Radiance data file variables	31
6	Interpretation of the red_flags characters.....	36
7	Interpretation of the yellow_flags characters	36
8	Sky feature file dimensions.....	51
9	Sky feature file variables.....	52
10	Sky feature classification encoding.....	52
11	Sky feature summation sky regions	53
12	Ad hoc radiance coefficients.....	58
13	Two-color sky interpretation graphical regions	60
14	Three-color sky interpretation graphical regions – 800 nm vs. 450 nm.....	61
15	Three-color sky interpretation graphical regions – 800 nm vs. 450 nm.....	61
16	NightFrac_vx hardwired constants.....	63
17	Patch summary file dimensions	71
18	Patch summary file variables	73
19	Sky feature summary file dimensions	77
20	Sky feature summary file variables.....	78
21	Sky feature plot color codes.....	83
22	Calibration file dimensions	87
23	Calibration file variables	899
24	File existence plot file encoding.....	104

1. Overview

The Whole Sky Imager is a hemispheric zenith viewing calibrated imaging instrument. As such its native product is sky radiance in three spectral bands with a spatial resolution of 30–35 microsteradians and a dynamic range extending from the brightness of the Milky Way at night to bright cirrus near the sun during the day. A rich set of retrievals have been implemented to transform these radiances into other useful products, such as sky feature delineation. These retrievals create many archived data sets, which are the subject of this document.

1.1 Features of the Instrument

1.1.1 Description

A typical Whole Sky Imager has two major components, an optical assembly in a climate controlled box located outdoors to have a panoramic view of the sky and a computer control assembly located in a nearby convenient building. At the heart of the optical assembly is a 512×512 pixel 16 bit CCD camera whose focal plane is cooled to below -30°C for electronic noise reduction. On top of the camera is an optical stack that includes a fiber optic bundle to match image sizes, a mechanical shutter, neutral density and spectral filters, a precision 2π steradian fisheye lens, and a ground glass optical dome for environmental protection. A complete description of the Whole Sky Imager hardware may be found in the companion document, Whole Sky Imager Hardware Guide. A few items of particular relevance to data interpretation are discussed in the following paragraphs.

1.1.2 Evolution

The Day Night Whole Sky Imager EO 6 has evolved considerably since its ARM debut in 1995. As a result, some data features have changed as well; however, considerable effort has been expended to keep the data files as uniform as possible to ease data usage.

As originally deployed the Whole Sky Imager collected data in two spectral bands centered at 450 nm and 650 nm, each approximately 70 nm wide. Within two years the instruments began to be modified to collect data in a third spectral band as well centered at 800 nm. All instruments now deployed have all three bands.

The first hemispheric dome used was made from high quality molded Plexiglas that caused some optical distortion near the horizon and which would accumulate dew and frost. A recent instrument upgrade replaces that dome with an optical quality flattened ground glass dome with a heater built into its mount to preclude dew and frost. As of this writing, the improved dome has been installed on all instruments outside the tropics and soon will be there as well.

Given the sensitivity of the camera, which required to collect the full sky radiances, very bright light sources will saturate the optics and CCD array causing the formation of optical ghosts and saturated column streaks. A shadow must be cast over the entire dome for both sun and moon light, and artificial lighting near the horizon must be baffled as well during hours of darkness. Since the latter are associated with dynamic anthropomorphic events periods of time exist for each instrument with pollution from such horizon lights.

There are two different designs of occulter used to shadow the moon and sun. While sunlight can be occulted at any latitude with a fairly simple mechanical arrangement, occulting moonlight is more difficult due to the wide range of lunar positions within a year. At accessible sites (currently just the Southern Great Plains) and movable arc with north south axis is employed upon which rides a moveable trolley carrying the shade. This arrangement allows the shade to be placed at any zenith and azimuth angle combination. At the other sites just a movable arc is employed which mounts a broad shading strip cut to cover all possible sun and moon positions. This arrangement is less satisfactory as it obscures considerably more sky but is more reliable. Accurately controlling the occulter (and trolley) motion has been continuing problem at all sites and a new optically encoded assembly will soon be installed at the Southern Great Plains for testing. If successful it will be deployed to all sites. One added feature of this new occulter is that all of its apparatus will be out of the field of view of the camera when neither the sun nor moon is above the horizon. Another is an image footprint half the size of the current occulter.

1.2 Instrument Deployment

1.2.1 Southern Great Plains

A whole sky imager was installed as part of the optical group at the central facility near Lamont, Oklahoma and began collecting data on September 19, 1995. The ARM Archive contains all retrieval products for the time period since this date to the present with the exception of short spans when the instrument was inoperative and being repaired. The instrument has been modified, upgraded, occasionally replaced with a similar unit. When installed in 1995, the whole sky imager had only two spectral filters centered on 450 nm and 650 nm. The 800 nm filter was added April 4, 1997.

A second instrument was installed at the Blackwell-Tonkawa Airport approximately 20 km northeast of the central facility for the 2000 Cloud Intensive Observing Period. Data from this instrument are also available in the ARM Archive for the period of February 17 to April 10, 2000.

1.2.2 North Slope of Alaska

Two whole sky imagers are installed in Alaska, one as part of the ARM Barrow facility and one as part of the Atkasuk facility. Data from the Barrow instrument are in the ARM Archive for the period extending from October 16, 1998 to the present, with the exception of short periods of time when the instrument was inoperative and being repaired. Archived data from the Atkasuk instrument are available for the period from September 14, 2001 to the present, again with short gaps for instrument repair. Given the harsh climate, these instruments have performed remarkably well. The data are affected by artificial lighting during the hours and days of darkness, especially when ground fog is present. The aurora borealis have little effect on sky feature retrieval products.

A third whole sky imager was deployed as part of the SHEBA experiment, mostly aboard the research ship for the time period spanning November 12, 1996 and October 5, 1998. This was an earlier version of the instrument and only collected data in the 450 nm and 650 nm spectral bands. Retrieved data products from this deployment are not currently available from the ARM Archive, but can be made available to an experimenter by special arrangement with the instrument mentor.

1.2.3 Tropical Western Pacific

Three whole sky imagers are installed in the tropical western pacific, one at Manus, Papua New Guinea, one in the island nation of Nauru, and one near Darwin, Australia. Raw data from the Manus instrument are in the ARM Archive for the period from August 18, 1998 to the present. Raw data from the Nauru instrument have been similarly archived from November 30, 1998 to the present. Finally, processed data products for the Darwin instrument are available from the ARM Archive from the date of its installation, March 6, 2002 to the present.

An unfortunate and progressive degradation of the 450 nm and 650 nm filters corrupted the data from both the Manus and Nauru instruments from the original installation date until they were replaced. The replacement date for Manus was October 21, 2001 and for Nauru November 3, 2001. For the period of time following their replacement, good quality retrieved data products are in the ARM Archive for both instruments. An effort is underway to reclaim as much information as possible during the period of degradation by correcting the radiance values based on assumed clear sky values. When finished the resultant sky feature retrievals will be of good quality but the radiance products will still be of little value.

1.2.4 Nomenclature

All data in the ARM Archive have names which follow an established template. A field within that template identifies the site of origin. The following table identifies the values used within that field for the various whole sky imagers mentioned above.

Table 1. Data file name site codes.		
Designator	Region	Location
sgpC1	Southern Great Plains	Central Facility near Lamont, Oklahoma
sgpL3	Southern Great Plains	Blackwell-Tonkawa Airport, Oklahoma
nsaC1	North Slope of Alaska	Barrow, Alaska
nsaC2	North Slope of Alaska	Atqasuk, Alaska
shbC1	Arctic Ocean	On the SHEBA research vessel
twpC1	Tropical Western Pacific	Manus, Papua New Guinea
twpC2	Tropical Western Pacific	Nauru
twpC3	Tropical Western Pacific	Darwin, Australia

1.3 Data Collection Strategy

1.3.1 Discussion

The Whole Sky Imager collects one image at a time and that for a particular neutral density and spectral filter combination. Exposure times vary from milliseconds to 100 seconds depending on the ambient light level. After collection the image must be transferred to the host computer and the filter wheels spun to the proper setting for the next image, a process that takes approximately 20 additional seconds. This sequence is repeated until all the desired images in a set have been collected. Between image sets the occulter arc and trolley, if present, will be moved to a new location.

Different spectral images within a set are thus not collected at the same moment, but rather a few tens of seconds apart. This is of little concern when the apparent angular motion of the clouds is small, but can be noticeable for fast moving low clouds such as boundary layer cumulus on a windy day.

1.3.2 Old Style

Prior to year 2000 all deployed Whole Sky Imagers collected images based on a ten minute interval. The base time at the start of a collection chain was determined by the most recent boot of the controlling computer which occurred at irregular intervals, sometimes a few a day and sometimes many weeks apart. For example, the instrument might be collecting data at 1413 1423, 1433 ... hours for a time span and then suddenly switch to 1548, 1558, 1608 ... hours after a reboot.

At each interval during daylight, twilight, and night bright moonlight (i.e., near full moon) hours 450 nm, 650 nm, and 800 nm (when deployed, see [Evolution: Chapter 1: Overview](#)) images were collected. In addition, a closed shutter image was collected whenever the exposure changed from the previous collection set value. To normalize the images the camera's background count level must be determined from this closed shutter image, and it is dependent on exposure time.

At each interval during night dim moonlight and night moonless hours a null filter (i.e., no spectral filter) image was collected. As with bright hours, a closed shutter image was also collected whenever the exposure changed.

On rare short occasions for testing or an Intensive Operating Period, the exposure interval was reduced to as little as one minute, but still using the same collection strategy. Whenever the various activities of the collection took more that the specified interval time, they proceeded apace and the next collection interval would start at their conclusion.

1.3.3 New Style

In ARM science team discussions leading up to the year 2000, it was determined that the old style collection strategy was inadequate. Mainly, it was observed that for many cloud fields the sky's autocorrelation time is considerably less than ten minutes and thus the base interval was missing statistically relevant samples. Because of the size of the data files, the amount of computer time necessary to perform the retrievals, and the desire not to fundamentally redesign the instrument the discussion participants decided that a six minute interval would be adequate if sub interval samples were taken. This became the basis for a new collection style that was first tested at the Blackwell-Tonkawa Intensive Observing Period deployment in the spring of 2000 and then after improvement was spread to the Southern Great Plains Central Facility instrument and then onward to all sites.

During daylight and twilight hours a full spectral set of images are collected at times whose minutes are evenly divisible by six, e.g., 1400, 1406, 1412, ... hours. This set includes 450 nm, 650 nm, and 800 nm images as well as a closed shutter image if the exposure has changed since the previous full set. This image set can be utilized in three band sky radiance mapping and for the development of sky feature retrievals.

During the same daylight and twilight hours but at times not included in the full set whose minutes are evenly divisible by two, e.g., 1402, 1404, 1408, 1410, 1414, ... hours, a single 650 nm band image is collected. The exposure is never changed from that of the previous full set; hence no closed shutter image is ever collected at these times. In conjunction with the 650 nm image from the full set these images can be utilized in single band sky radiance mapping with greater temporal resolution.

During nighttime hours a null filter image is collected at times whose minutes are evenly divisible by two, e.g., 1400, 1402, 1404, 1406 ... hours. In keeping with the daylight scheme, exposure time is changed as necessary only at times whose minutes are evenly divisible by six, and if it is changed a closed shutter image is collected as well. Even though this data set would allow for sky feature retrievals at two-minute intervals, they are only performed at six-minute steps in keeping with the daytime practice.

Additionally during nighttime hours on the single night containing or closest to a full moon in a lunar month a full set of spectral images is collected. The three spectral images are collected as during daylight hours at times whose minutes are evenly divisible by six. Since considerably fewer photons pass the spectral filters than in the null filter condition, these images are taken with a different, longer exposure time than the null filter image within the same six-minute set. This spectral exposure time is maintained separately from the null filter exposure and whenever it changes an additional closed shutter image is acquired called a type 2 closed shutter image. Note that this type 2 image will only be collected on full moon nights. This data set is taken to accommodate comparison between spectral sky feature retrieval algorithms (full moonlight is just bright enough to allow such) and the optical density sky feature retrieval algorithms normally used at night.

Especially on full moon nights, but also occasionally at other times, the time allotted for a collection interval is shorter than the time required to acquire all the images, move the filter wheels, and position the occulting mechanism. Since the six-minute collection sets are more important than the intervening two-minute collections, the two-minute ones will be skipped as necessary. The controlling software predicts the time needed for any two-minute collection and associated movements. If the allotted time is insufficient to allow a subsequent six-minute collection to occur on schedule, then it is omitted.

1.4 Data and Data Product Types

This section provides an overview of the Whole Sky Imager data products that may be downloaded from the ARM Archive. Please refer to subsequent chapters in this guide for a detailed explanation of each product, as well as hints and warnings regarding its use.

1.4.1 Raw Data Products

There are six types of raw data files, representing three spectral band, one null filter, and two closed shutter images. These files are in the ARM Archive but are available to the general user only by approval of the instrument mentor. They are hidden, i.e., do not show in the archive's listing of data sets. They would be of use only to a sophisticated user who understands Whole Sky Imager calibration application. The raw files are in a binary format.

1.4.2 Visualization Data Products

There are four Whole Sky Imager visualization products available from the ARM Archive.

- The first are sky pictures in JPEG format based on two-minute (ten-minute old style collection, see [Data Collection Strategy; Chapter 1: Overview](#)) 650 nm filter daytime and null filter nighttime data packaged into a single UNIX style tar archive for each 24-hour Greenwich day. These are grayscale in intensity, with an intensity level bar for interpretation.
- The second is a MPEG format movie formed from these individual image frames on a day by day basis.
- The third are spectral sky feature retrieval maps in JPEG format based on six-minute spectral images (ten-minute old style collection) taken during daylight and twilight hours. As with the sky images these sky feature maps are packaged into a single tar archive for each 24-hour day.
- The last are similarly sky feature retrieval maps in JPEG format but are based on optical density retrievals using null filter nighttime data. These images are packaged in the same manner as the spectral maps.

1.4.3 Radiance Data Product

The primary Whole Sky Imager data product in the ARM Archive is the NetCDF file that contains calibrated radiance data with 30–35 microsteradian spatial resolution for the entire 2π zenith hemisphere. There is one such file for each six-minute (ten-minute, old style) collection that contains data from all relevant spectral bands. All such files for one 24-hour Greenwich day are packaged into a tar archive that is quite large.

1.4.4 Sky Feature Data Product

A further retrieval of the radiance data just described leads to sky feature estimates of varying spatial resolution using a spectrally based algorithm during daytime and an optical density based algorithm at nighttime. These are contained in a sky feature NetCDF file available from the ARM Archive. As with the radiance data there is one such file for each six-minute (ten minute, old style) collection, and all files for a 24-hour Greenwich day are packaged in a single tar archive. This archive is quite large.

1.4.5 Summary Data Products

There are three Whole Sky Imager summary data products. Each contains information for an entire 24-hour Greenwich day. These files are many times smaller than radiance and sky feature data files from which they are derived. They are for the user interested in long term averages rather than high resolution details.

- The first is a NetCDF file that contains 650 nm radiance data at two minute time intervals (ten minute old style) for three $5^\circ \times 5^\circ$ sky patches at zenith and to the east and west. It is useful for comparing Whole Sky Imager data with other narrow field of view zenith pointing instruments.

- The second is a NetCDF file containing sky features averaged over ten large sky regions. The averages are based on six-minute (ten minute old style) sky feature retrievals using either the spectral or optical density technique.
- The third is a GIF or PNG file containing a color coded plot of the sky feature averages just mentioned for the entire sky dome.

1.4.6 Calibration Data Product

The calibration NetCDF file contains the constants required to transform the raw images into the various retrievals, as well as information necessary for the sophisticated interpretation of those retrievals. A series of calibration files exist for each instrument that are named to allow identification of the particular file used to build any historical retrieval. The series is expanded each time an instrument parameter changes, such as its physical orientation or a spectral filter constant.

1.4.7 Utility Data Products

The data utility products are for the sophisticated user of Whole Sky Imager data and are designed to aid in the interpretation or usage of other data products. In themselves they contain no useful scientific information. There are two such products.

- The first is a GIF or PNG file containing a pictorial representation of extant Whole Sky Imager files for an instrument for an entire Greenwich day. Since there are 3000 – 3500 of these (hence the extensive use of tar archives), it is useful to check this plot to ensure the exact files of interest are extant before requesting a day's worth of data from the ARM Archive.
- The second is a tar archive containing all the informational log files produced by the various retrieval programs in the course of processing a Greenwich day's worth of data. It is useful in detecting processing anomalies. Like the raw files it is maintained at the ARM Archive but is hidden from the archive's user interface. It is available only by special request and with the instrument mentor's concurrence.

1.5 Data Quality Control

Before Whole Sky Imager retrieved data products are released to the ARM Archive they are subjected to reasonably thorough quality control process to eliminate bad data. All faulty records are removed from the set being archived in contrast to simply flagging the bad data as is the practice for some other instruments. The user can have considerable confidence in archived Whole Sky Imager data.

1.5.1 Items that Affect Data Quality

Many different kinds of things can affect data quality including hardware, the environment, sky conditions, and retrieval (software) anomalies. Considerable experience has been acquired by the mentor staff in recognizing and categorizing causes of poor quality data; those that frequently recur are listed below.

- If the occulter is misaligned such that it does not fully shade the dome from either moonlight or sunlight, then (benignly) optical ghosts will appear in the raw data or (severely) columns of pixels will be saturated by electronic bleed. In either case, the radiance retrieval is corrupted as well as all downstream retrievals such as sky feature. This is one of the three leading causes of lost retrievals, but should be significantly reduced by the newly designed optically encoded occulter mentioned in [Evolution; Features of the Instrument; Chapter 1: Overview](#).
- If nighttime artificial lights near the horizon are sufficiently bright, then optical ghosts or saturated columns of pixels will occur in the raw data just as in the case of a misaligned occulter. This also corrupts the radiance retrieval and downstream products such as sky feature. At each site small baffles are erected a few meters from the instrument to shade the dome from such lights, however new lights occasionally are installed resulting in lost retrievals (often for entire nights) until they are identified and baffled by site personnel.
- At nighttime any unheated Whole Sky Imager dome will cool radiatively below atmospheric temperature and, if the humidity is high enough, will collect dew or frost that can persist well into the following daylight period. Of course the radiance retrieval is affected and downstream retrievals such as sky feature are corrupted. This is historically a leading cause of lost retrievals, but is being significantly reduced by the installation of dome heating kits as mentioned in [Evolution; Chapter 1: Overview](#). All deployed Whole Sky Imagers will have dome heaters by 2004.
- The daytime spectral sky feature retrieval algorithm identifies optically thin clouds only poorly as discussed in detail below in [Daytime Spectral Retrieval; Algorithm; Sky Feature File; Chapter 5: Sky Feature Product](#). Radiance retrievals are not affected; just the sky feature retrievals are corrupted. This is the second of the three leading causes of lost retrievals. There are known algorithms to greatly improve these retrievals, but they are somewhat complex involving atmospheric aerosol loading, ice habit (most of the clouds in question are cirrus), and sky viewing angle vis-à-vis solar illumination angle. Because of their intricacy these algorithms have yet to be implemented, but when they are they could be applied to historical data and the lost thin cloud retrievals could be restored.
- The spectral sky feature retrieval similarly does not perform well when the sun is near the horizon, either above or below. As with the thin cloud case, radiance retrievals are not affected; just the sky feature retrievals are corrupted. This is the third of the three leading causes of lost retrievals. The spectral algorithm uses a reference clear sky as a base for feature calculations. The current reference sky was developed from the SBDART radiance model which assumes a flat earth. This assumption is adequate for low and moderate solar zenith angles, but at zenith angles above 80° curved earth effects have significant impact on sky radiance. If the current algorithm were upgraded with reference radiances appropriate for this condition, then this cause of corrupted retrievals would be ameliorated. In fact, the upgraded algorithm could be applied to historical data and the lost high sun angle retrievals could be restored. The needed reference radiances could be calculated from a curved earth model or developed empirically.
- As discussed in the previous section [Tropical Western Pacific; Instrument Deployment; Chapter 1: Overview](#), the Whole Sky Imager instruments have suffered two cases of spectral filter degradation. Of course, this results in corrupted radiance retrievals and affects all downstream retrievals such as

sky feature. By comparing retrieved corrupted clear sky radiances and estimated correct clear sky values an approximate filter correction factor can be derived. This factor is not sufficiently accurate to recover radiance data, but is sufficient to recover sky feature retrievals. This process is, as of this writing, underway for the historical corrupted Manus retrievals and will be applied to the Nauru retrievals in the future.

On infrequent occasion, retrievals are corrupted by other anomalies, such as a failing mechanical shutter, very bright aurora borealis, insufficient camera focal plane cooling, animals (or people) in the field of view, and the like. Historically, these have caused the loss of but a few retrievals. Site personnel are quite efficient in effecting instrument repair which is a great help in this minimization.

1.5.2 Data Quality Procedure

Image based data streams are considerably harder to quality check than those for instruments which produce a single value per time increment such as radiometers. In the latter case the knowledge of minimum, maximum, and allowed change often provides an adequate level of quality control. For image data an equivalent level of quality control involves an analysis of the image either by software or human viewing. While appropriate software can be developed to accomplish this task, given the range of possible anomalies such software would be very complex and difficult to make sufficiently robust to execute in a production environment. On the other hand, a trained human can determine the quality of an image in a fraction of a second. The current quality control procedure is based on a combination of computer analysis of intrinsic data values and computer aided human scanning.

The computer analysis considers such factors as camera focal plane temperature, consistency between collections within a set, and number of suspect pixel level retrievals to assign a data quality value to a particular retrieval. Failed data is kept, but may cause downstream retrievals to be skipped. This analysis is discussed in great detail in [Quality Checking; Algorithm; Radiance Data File; Chapter 4: Radiance Product](#) and [Quality Checking; Algorithm; Sky Feature File; Chapter 5: Sky Feature Product](#).

The current human scanning involves student employees at Sandia National Laboratories and computer processing at the ARM Archive. Data is processed through two phases, and that which passes is permanently archived. Failed data is deleted.

The first phase begins with the archive processing raw data and producing image, radiance, and sky feature retrievals by Greenwich day by site. The sky picture, sky feature spectral retrieval picture, and sky feature optical density retrieval picture files are uploaded to Sandia and reviewed frame by frame. Excellent software aids have been developed to expedite this process, so the student views six frames simultaneously on a high resolution computer screen and clicks user interface tabs to select various actions to accept or reject retrievals. An experienced student can review an entire day's worth of retrievals in five to ten minutes, where the shorter time is needed for problem free days. With the sky feature pictures, the goal is to reject those that appear to be more than 10% different than the sky picture would indicate. Phase one ends with the automatic production of a list of corrupted retrievals to be deleted (if any) which is downloaded to the ARM Archive.

Phase two begins with the archive deleting noted files and then finishing the retrieval chain by producing the summary data products. These are uploaded to Sandia and scanned for inconsistencies.

Since summary products each cover an entire Greenwich day in one movie or plot, this process is quite rapid. If the data appear correct the archive is asked to permanently store all products; if there is an error then phase one is repeated. Currently the rejection rate at this point is less than 5% for an experienced student. The most common causes for rejection are the subtle deposition of dew and the transit of very thin clouds. Each of these is more easily seen in the animation associated with the movie than by frame analysis.

In the future, instrument and algorithm improvements will allow a considerably streamlined version of this process that selectively eliminates phase one activities. When the causes of data corruption associated with occulter misalignment, artificial lighting, dome moisture deposition, and high solar zenith angle are corrected as discussed in [Items that Affect Data Quality; Data Quality Control; Chapter 1: Overview](#) then the data would be reviewed by first quickly scanning the phase two summary products. Most days will pass in entirety and be immediately archived, but those that don't would be treated to the full process outlined above. Most of these cases would involve thin cloud situations; when that algorithm is similarly improved even less quality checking will be needed.

It is important to understand that this quality control process does not directly check for filter degradation, flat field changes, camera quantum efficiency, roll off changes, or geometric translation. These items are associated with instrument calibration and are discussed in more detail [Chapter 7: Calibration Product](#). A few of these will be noticed in extreme cases by derivative effect, e.g., a geometric translation of more than a degree or two will cause the optical density sky feature retrieval to fail.

1.5.3 Potential Biases

Any quality control process can introduce bias into data collections, and in particular into summary and average data products. The potential for this is analyzed in this section for each of the major causes of data corruption discussed in [Items that Affect Data Quality; Data Quality Control; Chapter 1: Overview](#) and in the same order as presented there. Those anomalies that occur infrequently are assumed to not have significant effect. As the underlying causes of rejected data are eliminated, then the biases listed here disappear, e.g., heaters installed on the domes inhibit dew formation and hence the dew or frost bias mentioned below becomes important only in the historical sense. One must also keep in mind that the data rejected in quality control is a small subset of the overall data set, and those in each item listed below are small subsets of that small subset.

- Occulter misalignment will not be observed unless the sun or moon is visible in the sky at the Whole Sky Imager location. Three cases show the potential for bias: (1) with an optically dense overcast sky no data will be deleted for misalignment, (2) with a broken optically dense cloud deck nearly random data will be deleted when a break in the clouds aligns with the moon or sun, and (3) under a clear sky all data will be deleted as long as the sun or moon is above the horizon. Thus long term averages of radiance and sky feature will be biased toward cloudy skies.
- The appearance of nighttime artificial lights is in general independent of sky conditions and therefore should not cause and bias in either radiance or sky feature. The exception to this is in the presence of

ground fog, as occurs regularly in the arctic and which causes a blooming effect around the light that can defeat any installed baffle. In this special case the long term averages of radiance and sky feature will be biased away from fog conditions.

- Dew or frost formation begins generally in the later nighttime hours and can persist into early daylight hours. In the arctic frost can persist all day. Any sky feature that is in phase with the diurnal cycle will be biased against those hours. This is probably a minor effect. There is a more important bias however. Dew or frost is far more likely to form under clear skies that allow strong radiative cooling than cloudy skies. Because of this, long term averages of radiance and sky feature will be biased toward cloudy skies.
- Obviously, the spectral sky feature algorithm limitations concerning thin clouds (see [Daytime Spectral Retrieval; Algorithm; Sky Feature File; Chapter 5: Sky Feature Product](#)) are of importance only when thin clouds are present in the sky to a greater extent than the 10% criterion used to reject poor quality data. Since the algorithm does correctly identify most of the “thicker” of the thin clouds, there is a preferential rejection of collection times with very thin clouds covering more than 10% of the sky dome. This leads to a bias in long term averages of sky feature against this situation; there is no bias produced in radiance data as that is not rejected for this anomaly. Whether the sky feature bias is in favor of cloudy or clear depends on whether such very thin clouds have a “cloudy” impact on one’s research.
- The high solar zenith angle limitation on the spectral sky feature retrieval algorithm affects retrievals for only those times of the day near sunrise and sunset. It does not influence radiance data, which is accurately archived for these times. Any sky feature that is in phase with the diurnal cycle will be biased against those hours. Any large scale sky feature will be well represented by interpolation between the coverage on either side of the high zenith angle break and will have no bias.
- Following recovery of retrievals for the two cases of spectral filter degradation the archived radiance values will be suspect but will have no particular bias. The sky feature data will be valid and have no particular bias in this regard either.

1.6 Instrument Calibration

The calibration of an imaging sensor is quite difficult, as much of an art form as scientific endeavor. Most of the techniques used to calibrate the Whole Sky Imager have been established by the instrument developer, the Marine Physical Laboratory of Scripps Oceanographic Institute. They are well documented elsewhere and simply reviewed here; see [Algorithm; Calibration File; Chapter 7: Calibration Product](#) for citations.

Historically each Whole Sky Imager was calibrated for radiance at Scripps prior to delivery to ARM and that calibration has been considered valid until factors conspired to have the instrument returned to Scripps for refurbishment. The filter degradation previously discussed on two instruments in the Tropical Western Pacific, as well as strong indication of individual pixel degradation on other instruments, showed that this assumption was suspect. Since returning each Whole Sky Imager to Scripps on a frequent (perhaps annual) basis for calibration is impractical, ARM funded the development of a field calibration

unit that can be transported to each site. The first calibrations with this unit on each of the six deployed Whole Sky Imagers occurred in the year 2003.

Currently annual field recalibrations of each instrument are planned. It may be that a few years experience will show a more relaxed schedule would serve as well, but as of this writing that is not apparent. After an instrument is recalibrated, its calibration file will be upgraded with new constants. These may cause a slight discontinuity in retrieved values, particularly in radiances.

1.6.1 Calibration Factors

There are several factors to be considered in the conversion of a Whole Sky Imager raw image into sky radiances and then further into sky features. The entire set is listed in this document's [Chapter 7: Calibration Product](#). The list below encompasses a few of the more important. In addition various instrument size and location parameters must be assayed.

- The absolute constants are used to convert a sensed intensity near zenith to physical radiance units. There is an absolute constant for each spectral and neutral density filter combination.
- The flat field array is used to adjust for pixel to pixel variation in sensitivity. This may be caused by slightly different quantum efficiencies between the pixels or by variation in the fiber optic bundle.
- The roll off array is used to adjust for lessening efficiency of the optic train lenses as the light sources moves from their center to their horizon. As this effect is wavelength dependent, there is a different roll off array for each spectral filter.
- The dark current array is used to correct the sensed intensity for the no illumination electronic signal.
- The geometric arrays are used to convert from pixel row and column to sky zenith and azimuth, and vice versa.
- The radiance arrays are used to calculate reference pristine sky radiance values for use in spectral sky feature retrievals.
- The ephemeris data are used to calculate star positions for use in optical density sky feature retrievals.

1.6.2 Laboratory Calibration

Laboratory calibration is performed at the Marine Physical Laboratory near San Diego, California prior to initial instrument delivery and each time an instrument is returned for refurbishment. The optical train and camera are demounted from the instrument and secured to an optical rail in a dark room where measurements are made leading to absolute constants, a flat field array, and a roll off arrays that are traceable to NIST standards. To complete the calibration file, data from the instrument after fielding is used to complete the closed shutter and geometric arrays. The radiance arrays are produced by modeling and the ephemeris data are from the Yale star chart.

1.6.3 Field Calibration

Field calibrations are performed with the field calibration device mentioned earlier in this section. No disassembly of the instrument is needed as the calibration is performed outdoors at the box containing the camera. One attachment using NIST traceable standard lamps allows the measurement of absolute constants and another using an integrating sphere allows the measurement of flat field and roll off effects. A third attachment provides near horizon geometric information that will later be combined with star data to develop the geometric arrays; this is a considerable improvement on the standard laboratory calibration method. The dark current and radiance arrays are produced in the same manner as in the laboratory calibration, as is the ephemeris data.

A difference of several percent has been noticed between laboratory and field calibration derived absolute constants; this difference is documented in [Radiance Variables; Algorithm; Calibration File; Chapter 7: Calibration Product](#). As a result there will be a decrease in retrieved radiance values in the historical record following the first field calibration of each instrument.

2. Raw Products

There are six types of raw data products, but they do not differ in purpose or format, and so are treated in this document as variants of a single type.

2.1 Raw Data Files

2.1.1 Purpose

Raw data files are at the base of the Whole Sky Imager retrieval scheme. They are not formatted conveniently for general use; in fact they are binary files following the Little Endian encoding pattern. More recent raw files are compressed as well to eliminate useless data near image corners. The raw files are in the ARM Archive but are available to the general user only by approval of the instrument mentor. They are hidden, i.e., do not show in the archive's listing of data sets. Raw data would be of use only to a sophisticated analyst who understands Whole Sky Imager calibration application.

2.1.2 Generation and Storage

Raw data files are generated by the instrument during the data collection process at a rate of one file for each sky image. They are passed through the ARM data system to the ARM Archive and are in permanent storage there.

There are six different types of raw files:

1. an image with the 450 nm spectral filter
2. an image with the 650 nm spectral filter
3. an image with the 800 nm spectral filter
4. an image with no spectral filter, also called the null filter image
5. an image with the mechanical shutter closed taken with an exposure equivalent to that of the data for normal retrieval, also called the shutter closed image

6. an image with the mechanical shutter closed taken with an exposure equivalent to that of the spectral images taken on the night of a full moon, also called the shutter closed type 2 image.

For an idealized day without a full moon and with 13 hours of daylight plus twilight and 11 hours of darkness a Whole Sky Imager would generate 130 raw files of the 450 nm type (one every 6 minutes for 13 hours), 390 files of the 650 nm type (one every 2 minutes for 13 hours), 130 files of the 800 nm type (one every 6 minutes for 13 hours), 330 null filter files (one every 2 minutes for 11 hours), and approximately 30 closed shutter files (one for each exposure change for 24 hours). Thus for every 24-hour Greenwich period each deployed instrument produces something on the order of 1010 raw files. These are stored in the ARM Archive in five UNIX style tar archives, one for each file type.

2.1.3 Naming Convention

Prior to the deployment of the upgraded Whole Sky Imager at the Southern Great Plains in 2003, all raw files were named upon generation with a DOS 8.3 style name. The name follows the format of:

ydddhhmm.ttt

where

- y = the last digit of the current year
- ddd = the numeric Julian day of the year with the range of 1 to 366
- hh = the hour with the range of 0 to 23
- mm = the minute with the range of 0 to 59
- ttt = the spectral file type with a value from the set {blu, red, nir, clr, drk, dr2} for {450 nm, 650 nm, 800 nm, null filter, closed shutter, closed shutter type 2} respectively.

Obviously, this scheme has some severe deficiencies, specifically it is not unique beyond ten years and it is not unique beyond one instrument. In recent years, all raw files are renamed immediately upon collection using the following name based on standard ARM naming practices. The upgraded unit will name files upon generation in this manner as well.

sssff.00.yyyymmdd.hhmmss.raw.ttt

where the blue characters are literal and

- sss ff = the site and facility identifier as listed in Table 1
- yyymmdd.hhmmss = the standard 15 character ARM date time stamp showing year, month, day, hour, minute, and second
- ttt = the spectral file type with a value from the set {blu, red, nir, clr, drk, dr2} for {450 nm, 650 nm, 800 nm, null filter, closed shutter, closed shutter type 2}, respectively.

Of the historical files in the ARM Archive, only those from the SHEBA experiment should still have names in the 8.3 format. All other files that had initially been stored with 8.3 names have since been renamed to the ARM standard.

The UNIX style tar archives associated with these files use the same basic name but have “.tar” appended to the end.

2.1.4 Dimensions, Variables, and Coding

Since the raw data files are binary rather than NetCDF they do not have dimensions, attributes, and variables in the manner of normal ARM usage.

2.1.5 File Structure

Prior to the year 2001 all raw files contained 524288 bytes based on a 512×512 pixel image with each pixel represented by an unsigned short integer (2 bytes). The first integer is the extreme southwest point in the image. Subsequent integers represent pixels progressing eastward on the first row until the 512th integer represents the extreme southeastern image point. The next integer is the western point of the next row; the mapping continues until the last integer in the file represents the northeastern corner pixel of the image. The first two rows of image data (2048 bytes) are overlaid with instrument and image information in ASCII coding following a {key word, value} format. They are described in the next section.

A complete Whole Sky Imager raw data set for a 24-hour period will exceed a half gigabyte in size – a considerable amount of data to transfer from remote ARM sites. The image appears as a circle in the square image; therefore the corners of the image contain information of negligible value. In recognition of this and as an effort to reduce overall raw data volume a new raw data format was developed in 2001 and gradually introduced to all deployed instruments. In this new format 705 bytes of header information are ASCII encoded followed by 205859 unsigned short integers for a total file size of 412423 bytes, a 21.3% reduction. The header contains the same information as in the previous format, but without as many blanks. The string of integers are values for first an increasing and then decreasing number of columns for each row progressing from south to north. The presentation of the list of the beginning and ending column numbers of data saved for each row necessary to reconstruct a 512×512 pixel image is beyond the scope of this document, but may be obtained from the instrument mentor.

2.1.6 Header Information

A brief discussion of each item included here. Most of these, e.g., red flags, have NetCDF variable counterparts in the radiance file as discussed in [Variables; Dimensions, Variables, and Coding; Chapter 4: Radiance Product](#), and are discussed there as well. Additional information may be obtained from the instrument mentor by special request.

The **site** item – keyword: Site: – indicates the location of the instrument. The value is retrieved from a lookup table provided by the instrument operator, and is as valid as his input. The values can vary, but will be approximately one of the set {SHB, NS1, NS2, TW1, TW2, TW3, SG1, SG2}. This set corresponds to the designators in Table 1 as follows {shbC1, nsaC1, nsaC2, twpC1, twpC2, twpC3, sgpC1, sgpL3}, respectively.

The **latitude** item – keyword: la= – indicates the latitude of the instrument’s location in decimal degrees. For fixed base instruments the value is retrieved from a lookup table provided by the instrument

operator, and is as valid as his input; a better value is found in the calibration file (see [Variables; Dimensions, Variables, and Coding; Calibration File; Chapter 7: Calibration Product](#)). For moveable base instruments, see the discussion for the variable lat in [Variables; Dimensions, Variables, and Coding; Radiance Data File; Chapter 4: Radiance Product](#).

The **longitude** item – keyword: lon= – indicates the longitude of the instrument’s location in decimal degrees. The comments for this item are identical to those above for latitude.

The **file** item – keyword: file: – indicates the image type of this raw file from the six possible types. The values can vary, but will be approximately one of the set {Blue Image, Red Image, NIR Image, Clear Image, Dark Image, Dr2 Image} for {450 nm, 650 nm, 800 nm, null filter, closed shutter, closed shutter type 2} respectively.

The **day** item – keyword: Day= – indicates the numeric day that this raw file was collected. Its value is from a precision time source and is valid.

The **month** item – keyword: Month= – indicates the numeric month that this raw file was collected. Its value is from a precision time source and is valid.

The **year** item – keyword: Year= – indicates the four digit numeric year that this raw file was collected. Its value is from a precision time source and is valid.

The **zulu time** item – keyword: Time = – indicates the four digit military style Greenwich hour that this raw file was collected. The value is followed by the character “Z”. It is from a precision time source and is valid.

The **time source** item – follows zulu time without a keyword – indicates the precision source from which the values of day, month, year, and zulu time are derived. It may have a value from the set {"@", "W", "B", "S", "G", "F"} with the assigned meaning of @ = null character, W = WWV, B = BIOS, S = system clock, G = GPS, and F = none. Its value is measured and therefore valid.

The **exposure** item – keyword: Exposure= – indicates the camera exposure time in milliseconds for this raw file. The value is followed by the characters “ms”. The value is set by the instrument’s control software and therefore is valid.

The **neutral filter** item – keyword: ND= – indicates the neutral filter in place for this raw file. It may have any value from the set {0, 1, 2, 3, 4} where 0 = null value and the other numbers indicate approximate filter transmission thus: 1 = 1.0, 2 = 0.01, 3 = 0.001, and 4 = 1.0. The value is set by the instrument’s control software and therefore is valid.

The **spectral filter** item – keyword: SP= – indicates the spectral filter in place for this raw file. It may have a value from the set {0, 1, 2, 3, 4} where 0 = null value and the other numbers indicate the approximate filter center wavelength and bandpass thus: 1 = 800 nm and 70 nm FWHM, 2 = null filter, 3 = 650 nm and 70 nm FWHM, and 4 = 450 nm and 70 nm FWHM. The value is set by the instrument’s control software and therefore is valid.

The **arc destination** item – keyword: Occultor Destination: Arc= – indicates the calculated, desired position for the occultor arc in decimal degrees. The value is set by the instrument’s control software and therefore is valid.

The **trolley destination** item – keyword: Trolley= – indicates the calculated, desired position for the occultor trolley in decimal degrees. The value is set by the instrument’s control software and therefore is valid.

The **housing temperature** item – keyword: Housing Temp= – indicates the temperature of the camera housing inside the environmental housing in degrees Celsius. Its value is measured and therefore valid. The hardware version item – keyword: Hardware Ver: – indicates the hardware revision of the instrument. The value is retrieved from a lookup table provided by the instrument operator, and is as valid as his input.

The **software version** item – keyword: Software Ver: – indicates the revision of the instrument’s control software. The value is retrieved from a lookup table provided by the instrument operator, and is as valid as his input.

The **time status** item – keyword: Time Stat: – indicates the diagnostic code for the clock. It may have a value from the set {0, 1, 2, 3, 4, 9, A}. These change meanings with the value of the time source item as listed below.

- If the time source is W or G then 1 = first time grab, 2 = match with BIOS, 3 = match with system, 9 = mismatch with BIOS, and A = mismatch with system.
- If the time source is S then 1 = only system clock valid.
- If the time source is B then 4 = WWV or GPS invalid.
- If the time source is F then 0 = no valid clocks.

The **nitrogen pressure** item – keyword: N2 pressure= – indicates the differential pressure of the dry nitrogen fill inside the camera housing inside the environmental housing in pounds per square inch. Its value is measured and therefore valid.

The **flow rate** item – keyword: Flow rate= – indicates the flow rate of the camera CCD coolant in gallons per minute. Its value is measured and therefore valid.

The **environmental housing temperature** item – keyword: Env. Housing Temp= – indicates the temperature in degrees Celsius of the environmental housing containing the outdoor portion of the Whole Sky Imager. Its value is measured and therefore valid.

The **CCD chip temperature** item – keyword: CCD Chip Temp= – indicates the temperature of the camera CCD chip in degrees Celsius. Its value is measured and therefore valid.

The **arc position** item – keyword: Occultor Position: Arc= – indicates the measured position of the occultor arc in decimal degrees. The value is only as good as the measurement, which in some cases has a few degrees of error.

The **trolley position** item – keyword: Trolley= – indicates the measured position of the occulter trolley in decimal degrees. The value is only as good as the measurement, which in some cases has a few degrees of error.

The **relative humidity** item – keyword: Rel. Humidity= – indicates the relative humidity inside the environmental housing in percent. Its value is measured and therefore valid.

The **red flags** item – keyword: Red Flags= – indicates the results of internal software checks for error conditions. For an interpretation of this item see the discussion of the variable red_flags in [Variables; Dimensions, Variables, and Coding; Radiance Data File; Chapter 4: Radiance Product](#).

The **yellow flags** item – keyword: **No Keyword!** – indicates the results of internal software checks for warning conditions. For an interpretation of this item see the discussion of the variable yellow_flags in [Variables; Dimensions, Variables, and Coding; Radiance Data File; Chapter 4: Radiance Product](#).

The **sun azimuth** item – keyword: Sun Position: Azimuth= – indicates the calculated sun azimuth angle based on the time and position of the instrument. As the value is calculated it is as accurate as the input data, which in this case is limited by the accuracy of latitude and longitude.

The **sun zenith** item – keyword: Zenith= – indicates the calculated sun zenith angle. The comments for this item are identical to those above for sun azimuth.

The **moon azimuth** item – keyword: Moon Position: Azimuth= – indicates the calculated moon azimuth angle. The comments for this item are identical to those above for sun azimuth.

The **moon zenith** item – keyword: Zenith= – indicates the calculated moon zenith angle. The comments for this item are identical to those above for sun azimuth.

The **source** item – keyword: Source= – indicates the dominant light source in the sky. It may have a value from the set {sun, moon, none} which are self explanatory.

The **camera azimuth offset** item – keyword: Azimuth Offsets: Camera= – indicates the angular difference in decimal degrees between the line along the camera’s pixel columns and the environmental housing’s north south alignment fiducial. The value is retrieved from a lookup table provided by the instrument operator, and is as valid as his input.

The **field azimuth offset** item – keyword: Field= – indicates the angular difference in decimal degrees between the environmental housing’s north south alignment fiducial and the direction of geographic north. For fixed base instruments the value is retrieved from a lookup table provided by the instrument operator, and is as valid as his input. For moveable base instruments the value is determined from GPS instrumentation and should be quite accurate.

The **occultor azimuth offset** item – keyword: Occultor Offsets: Azimuth= – indicates the angular difference in decimal degrees between the occultor’s pivot axis and the direction of geographic north. The value is retrieved from a lookup table provided by the instrument operator, and is as valid as his input.

The **occultor zenith offset** item – keyword: Zenith= – indicates the angular difference in decimal degrees between the occultor’s as installed zenith position and true zenith. The value is retrieved from a lookup table provided by the instrument operator, and is as valid as his input.

The **image center column** item – keyword: Image Center: Col= – indicates the camera CCD column upon which the image of the sky’s zenith point falls. The value is retrieved from a lookup table provided by the instrument operator, and is as valid as his input; a better value is found in the calibration file (see [Variables; Dimensions, Variables, and Coding; Calibration File; Chapter 7: Calibration Product](#)).

The **image center row** item – keyword: Row= – indicates the camera CCD row upon which the image of the sky’s zenith point falls. The comments for this item are identical to those above for image center column.

The **image radius** item – keyword: Radius= – indicates the number of camera CCD pixels between the one upon which the image of the sky’s zenith point falls and one upon which the image of a point on the horizon (90° zenith angle) falls. The comments for this item are identical to those above for image center column.

The **field azimuth update time** item – keyword: Field Azm. update time: – indicates the time of the last GPS update of the field azimuth offset item. It is valid only for moveable base instruments such as the Whole Sky Imager deployed on the SHEBA ship; for fixed base instruments it has no meaning. The format is hhmm:ss where each letter stands for a numeral representing (in order) hour minute : second of the day of image collection.

2.1.7 Algorithm

The raw data files are archived in the same format as they are produced by the instrument; no transformation algorithms are used.

2.1.8 Usage Hints

As mentioned above, only a sophisticated analyzer of Whole Sky Imager data should attempt to user raw file data. One reason to attempt such would be to apply a different calibration. ARM Archive personnel have occasionally had to use header information to determine the date and time an image was captured to correct an improperly generated ARM style file name date time group.

2.1.9 Limitations

All raw data files acquired are in the ARM Archive, without regard to quality. This is in contrast to the retrieved data products which are only archived if valid. Therefore the user of raw data files must perform quality assessment himself.

As noted some of the header information is from a lookup table supplied (most often) by the person installing the instrument. These tables are not necessarily upgraded when instrument software and hardware changes are implemented. The calibration file always contains the correct value for some of these items, and should be used in lieu of the header information when possible.

3. Visualization Products

There are four Whole Sky Imager visualization products available from the ARM Archive. The first are gray scale picture frames with a hemispheric sky view. The second is a 24-hour movie made from those frames. The last two are picture frames with the spectral and optical density sky feature retrievals depicted.

3.1 Sky Picture File

3.1.1 Purpose

The sky picture file is a JPEG format hemispheric sky view picture based on two-minute (ten-minute old style collection, see [Data Collection Strategy, Chapter 1: Overview](#)) 650 nm daytime and null filter nighttime data packaged into a single UNIX style tar archive for each 24-hour Greenwich day. These are grayscale in intensity, with an intensity level bar for interpretation.

These pictures are invaluable in diagnosing peculiarities in radiance or sky feature retrievals. They can be very helpful as well in finding times within a day that have a particular kind of cloud cover for detailed study.

3.1.2 Naming Convention

The sky picture files have the following name based on standard ARM naming practices.

`sssiwsipttradjpgff.a1.yyyymmdd.hhmmss.jpg`

or

`sssiwsipttradjpgff.b1.yyyymmdd.hhmmss.jpg`

where the blue characters are literal and

<code>sss ff</code>	= the site and facility identifier as listed in Table 1
<code>ii</code>	= the time interval of production from the set {02, 10}
<code>yyymmdd.hhmmss</code>	= the standard 15 character ARM date time stamp showing year, month, day, hour, minute, and second.

As a result of a change in the ARM data level standard, the “a1” found in earlier files has been replaced by “b1” in later ones. There is no difference in content or structure between the two. The UNIX style tar archives associated with these files use the same basic name but have “.tar” appended to the end.

3.1.3 Dimensions, Variables, and Coding

Since the sky picture files are JPEG rather than NetCDF they do not have dimensions, attributes, and variables in the manner of normal ARM usage.

The pictures are portrayed in a gray scale that is encoded by sky radiance. A bar at the top of the image shows the scale relative to standard units, i.e., milliwatts per meter squared per steradian per nanometer.

3.1.4 Algorithm

The basis for the picture file is a calibrated image, which is extracted from temporary output files from the execution of the IDL program WsiIdpc2_vx, where x stands for the version number of the current release (14 as of this writing in September 2003). The program and its subroutines were written by the instrument mentor and are maintained by him as well. It is coded for execution in a production environment, meaning that it can handle nearly any anomaly in raw file data streams or input data and continue processing to a benign conclusion. WsiIdpc2_vx is parallel code to the program WsiIdpc4_vx which produces the radiance data file (see [Radiance Data File; Chapter 4: Radiance Product](#)) except that it only processes 650 nm and null filter data and does so on a nominal two minute time interval, and that it explicitly stores patch data as a subset of the calibrated radiance array in addition to that array. The time interval is ten minutes for the old style of collection; see [Data Collection Strategy; Chapter 1: Overview](#). The calibration algorithm and mask determination are identical to that of WsiIdpc4_vx as discussed in lengthy detail in [Algorithm; Radiance Data File; Chapter 4: Radiance Product](#), and are not repeated here.

The picture is presented with north to the top and east to the right. This is the same as looking at a road map but the mirror image of looking at a star map. The dynamic range of radiances is so large that a direct rendition into an image would result in mostly black and white values with little useful gray scale information. Therefore the 256 gray values are formed by bounding the radiances and then performing a square root gamma scaling. Specifically, the square root of a pixel's radiance is linearly mapped to the gray scale numbers 0 – 255 by having the gray scale zero set to zero radiance and 255 set to the square root of the 99th percentile value of radiance.

3.1.5 Usage Hints

The sky picture displays well on top of the line calibrated computer monitors. On other monitors the contrast may not appear optimum. The JPEG image may be manipulated in any good image enhancement program such as Adobe Photoshop to augment desired features. If the image is to be used as part of a presentation, the contrast should be checked on screen.

A sample sky picture is shown in Figure 1 for an early afternoon scene of scattered clouds. Note the sun shining through the center of the occulter trolley and the informational banner at the top.

3.1.6 Limitations

The quality control process will cause suspect sky picture files to be deleted from the archive record, therefore there may not be a complete set for any given period. The completeness of the set for a day may be determined at a glance from the file existence plot file discussed in [File Existence Plot File; Chapter 8: Utility Products](#).

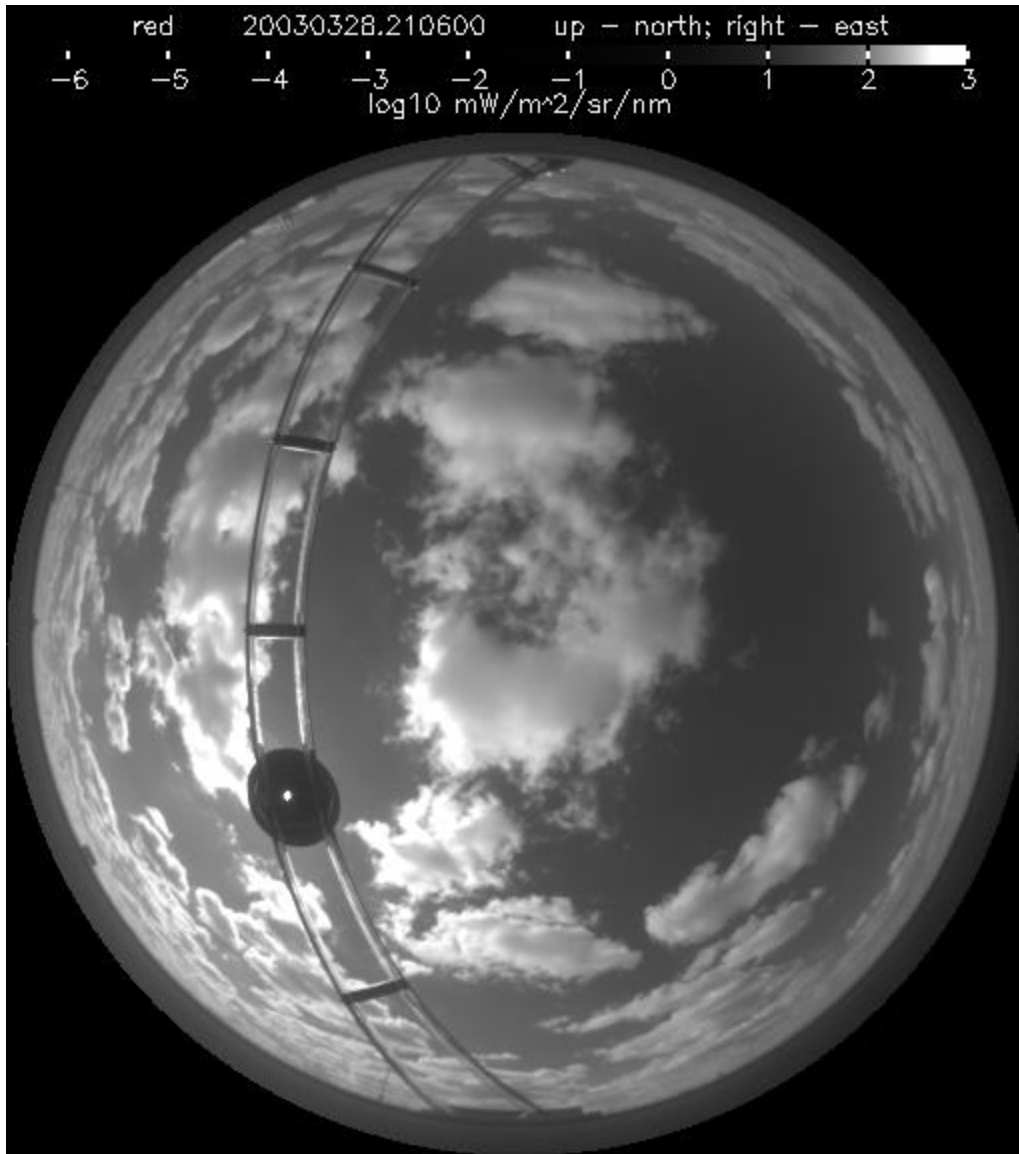


Figure 1. Sample sky picture.

3.2 Movie File

3.2.1 Purpose

The movie file is a MPEG format hemispheric sky view movie based on two-minute (ten-minute old style collection, see [Data Collection Strategy; Chapter 1: Overview](#)) 650 nm filter daytime and null filter nighttime data for a 24-hour Greenwich day. This is grayscale in intensity, with an intensity level bar for interpretation.

The movie is invaluable in understanding the dynamics of a cloud field, and for keying on subtle features only noticeable by their motion. It can be very helpful as well in scanning for days or times within a day having a particular kind of sky feature for detailed study.

3.2.2 Naming Convention

The movie files have the following name based on standard ARM naming practices.

sssiwsipartradmpgff.a1.yyyymmdd.hhmmss.mpg

or

sssiwsipartradmpgff.b1.yyyymmdd.hhmmss.mpg

where the blue characters are literal and

sss ff = the site and facility identifier as listed in Table 1
ii = the time interval of production from the set {02, 10}
yyymmdd.hhmmss = the standard 15 character ARM date time stamp showing year, month, day, hour, minute, and second.

As a result of a change in the ARM data level standard, the “a1” found in earlier files has been replaced by “b1” in later ones. There is no difference in content or structure between the two. Since only one movie file is produced per Greenwich day, the hours, minutes, and seconds are always set to zero.

3.2.3 Dimensions, Variables, and Coding

Since the movie files are MPEG rather than NetCDF they do not have dimensions, attributes, and variables in the manner of normal ARM usage.

The movies are portrayed in a gray scale that is encoded by sky radiance. A bar at the top of each frame shows the scale relative to standard units, i.e., milliwatts per meter squared per steradian per nanometer.

3.2.4 Algorithm

The basis for the movie file is the sky picture file, whose derivation is discussed in [Algorithm: Sky Picture File; Chapter 3: Visualization Products](#). In particular, the movie gray scale coding utilizes the same bounded gamma correction discussed there. The movie is presented with north to the top and east to the right. This is the same as looking at a road map but the mirror image of looking at a star map.

The IDL procedure WsilDpc3m_vx collects all the sky picture files for a day and performs the MPEG encoding. The x stands for the procedure’s version number which as of this writing in September 2003 is 7. The program was written by the instrument mentor and is maintained by him as well.

3.2.5 Usage Hints

The movie displays well on top of the line calibrated computer monitors. On other monitors the contrast may not appear optimum. With primitive MPEG viewers (the kind included with most operating systems) the frame presentation rate may be too fast for analytical viewing; viewers are available which allow user determination of the rate, as well as the very useful freeze frame and frame stepping capabilities.

3.2.6 Limitations

Since the quality control process will cause suspect sky picture files to be deleted, and since the sky pictures underlie this movie, there may be time gaps.

3.3 Sky Feature Spectral Retrieval Picture File

3.3.1 Purpose

The sky feature spectral retrieval picture files are JPEG format hemispheric color coded maps of the sky features detected during spectral retrieval processing. The maps have the same six-minute (ten-minute old style collection, see [Data Collection Strategy; Chapter 1: Overview](#)) time interval as the spectral retrievals, and span the same daylight and twilight period. All files for a single 24-hour Greenwich day are packaged into a UNIX style tar archive.

These cloud maps are useful in determining periods for study based on cloud geometry or distribution. They can be very helpful as well in diagnosing peculiarities in sky feature retrievals.

3.3.2 Naming Convention

The sky feature spectral retrieval picture files have the following name based on standard ARM naming practices.

`ssswsicloudspecff.c1.yyyymmdd.hhmmss.jpg`

where the blue characters are literal and:

`sss ff` = the site and facility identifier as listed in Table 1, and
`yyymmdd.hhmmss` = the standard 15 character ARM date time stamp showing year, month, day, hour, minute, and second.

The UNIX style tar archives associated with these files use the same basic name but have “.tar” appended to the end.

3.3.3 Dimensions, Variables, and Coding

Since the sky feature spectral retrieval files are JPEG rather than NetCDF they do not have dimensions, attributes, and variables in the manner of normal ARM usage.

The sky feature maps are portrayed basic colors that encod the retrieved feature types. The colors used are described in the following table.

Color	Sky Feature
Black	Undefined, e.g., under the occulor
Brick Red	Indeterminate; not matching sky model, e.g., Palm tree
Dark Blue	Near pristine clear sky
White	Sky brighter than selected star
Light Green	Translucent cloud or aerosol
Dark Green	Opaque cloud

3.3.4 Algorithm

The basis for the sky feature spectral retrieval picture file is the spectral retrieval mapping within the sky feature file, whose derivation from radiance data is discussed in [Chapter 5: Sky Feature Product](#). The same IDL procedure discussed in detail in that chapter that produces the sky feature file produces this file as well. The mapping is presented with north to the top and east to the right. This is the same as looking at a road map but the mirror image of looking at a star map. It is presented at the same scale and orientation as the sky picture for easy comparison.

3.3.5 Usage Hints

The sky feature spectral retrieval picture is most useful when placed beside the sky picture of the same time. Often the retrieval near the horizon is suspect, and side by side comparison will quickly indicate if that is the case.

A sample sky feature spectral retrieval picture is shown in Figure 2 for the same early afternoon scene of scattered clouds shown in the sample sky picture of Figure 1. Note the informational banner at the top.

3.3.6 Limitations

The quality control process will cause suspect radiance and sky feature retrieval files to be deleted from the ARM Archive. As these files underlie the sky feature spectral retrieval picture files, there may not be a complete set for any given period. The completeness of the set for a day may be determined at a glance from the file existence plot file discussed in [File Existence Plot File; Chapter 8: Utility Products](#).

3.4 Sky Feature Optical Density Retrieval Picture File

3.4.1 Purpose

The sky feature optical density retrieval picture files are JPEG format hemispheric color coded maps of the sky feature types detected during optical density feature processing. The maps have the same six-minute (ten-minute old style collection, see [Data Collection Strategy; Chapter 1: Overview](#)) time interval as the optical density retrievals, and span the same nighttime period. All files for a single 24-hour Greenwich day are packaged into a UNIX style tar archive.

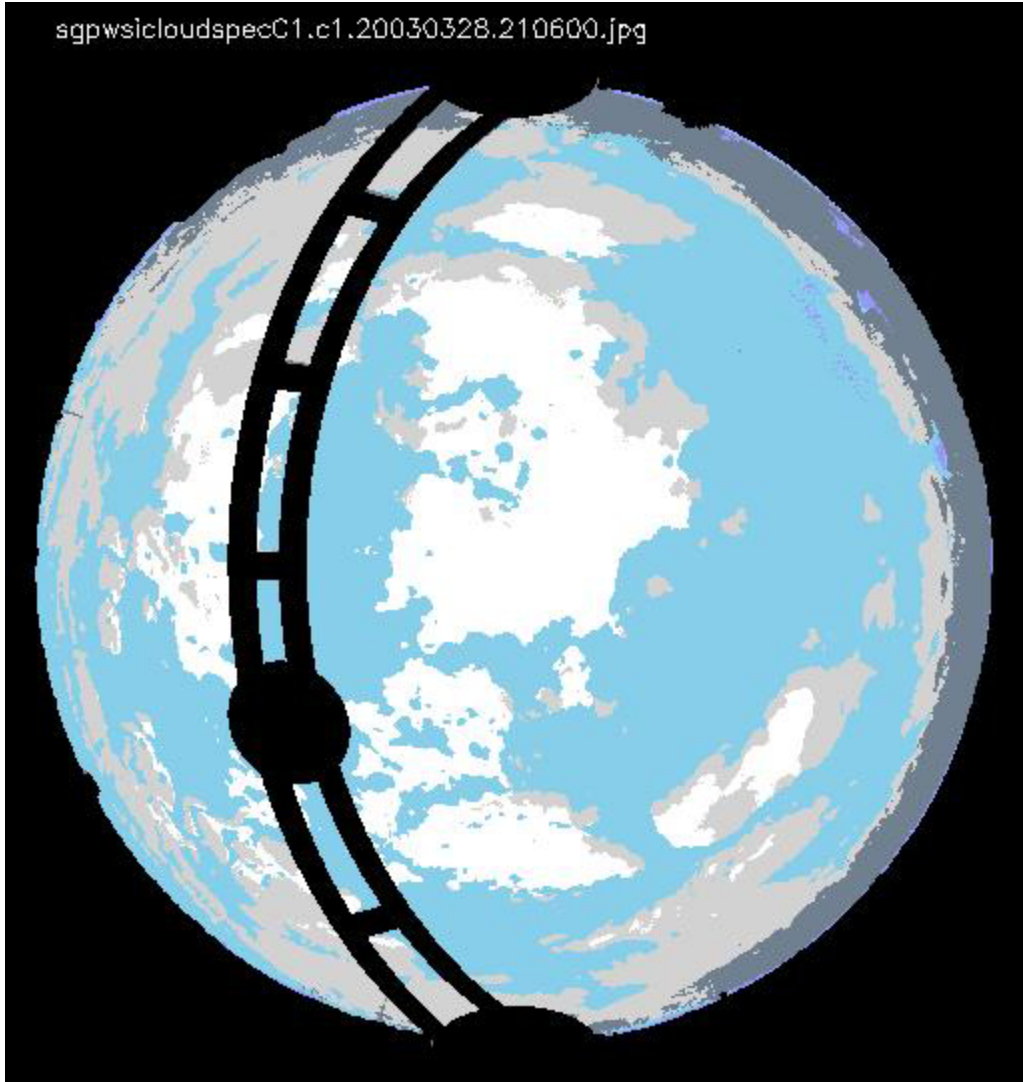


Figure 2. Sample sky feature spectral retrieval picture.

These sky feature maps are useful in determining periods for study based on feature geometry or distribution. They can be very helpful as well in diagnosing peculiarities in optical density retrievals.

3.4.2 Naming Convention

The sky feature optical density retrieval picture files have the following name based on standard ARM naming practices.

[sswsicloudsdenff.c1.yyyymmdd.hhmmss.jpg](#)

where the blue characters are literal and

`sss ff` = the site and facility identifier as listed in Table 1
`yyyymmdd.hhmmss` = the standard 15 character ARM date time stamp showing year, month, day, hour, minute, and second

The UNIX style tar archives associated with these files use the same basic name but have “.tar” appended to the end.

3.4.3 Dimensions, Variables, and Coding

Since the sky feature optical density retrieval files are JPEG rather than NetCDF they do not have dimensions, attributes, and variables in the manner of normal ARM usage.

The sky feature maps are portrayed basic colors that encode the retrieved feature types. The colors used are described in the following table.

Color	Sky Feature
Black	Undefined, e.g., under the occulter
Brick Red	Indeterminate; not matching sky mode, e.g., palm tree
Dark Blue	Near pristine clear sky
White	Sky brighter than selected star
Light Green	Translucent cloud or aerosol
Dark Green	Opaque cloud

3.4.4 Algorithm

The basis for the sky feature optical density retrieval picture file is the optical density retrieval mapping within the sky feature file, whose derivation from radiance data is discussed in [Nighttime Optical Density Retrieval; Algorithms; Sky Feature File; Chapter 5: Sky Feature Product](#). The same IDL procedure discussed in detail in that chapter that produces the sky feature file produces this file as well. The mapping is presented with north to the top and east to the right. This is the same as looking at a road map but the mirror image of looking at a star map. It is presented at the same scale and orientation as the sky picture for easy comparison.

3.4.5 Usage Hints

The optical density spectral retrieval picture is most useful when placed beside the sky picture of the same time. A comparison of the two will immediately determine whether the bright sky areas should be interpreted as aerosol laden clear sky or thin cloudy sky.

The sky feature spectral retrieval picture is most useful when placed beside the sky picture of the same time. Often the retrieval near the horizon is suspect, and side by side comparison will quickly indicate if that is the case.

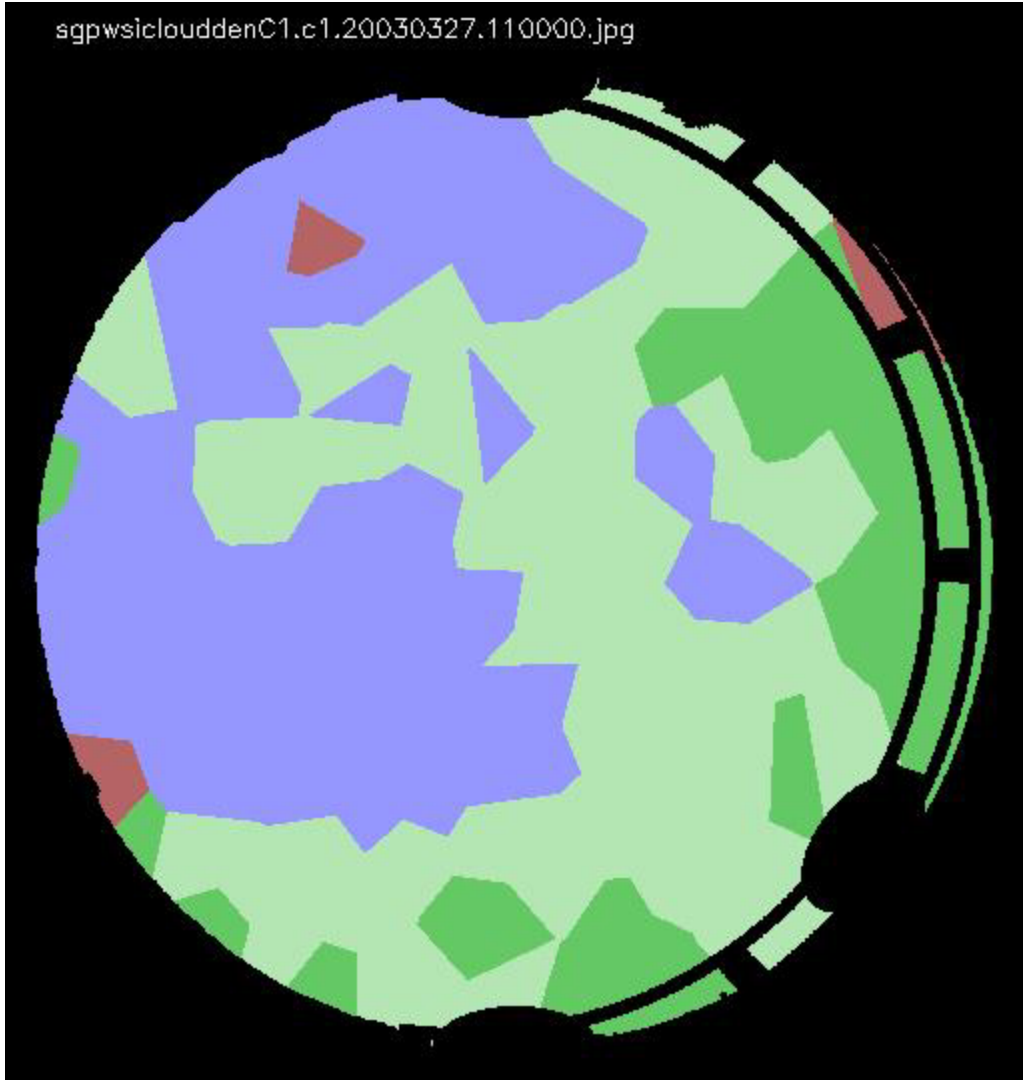


Figure 3. Sample sky feature optical density retrieval picture.

A sample sky feature optical density retrieval picture is shown in Figure 3 late night scene of a sky that is mostly clear to the northwest but has thin clouds stretching from the northeast around to the southwest. Note the informational banner at the top.

3.4.6 Limitations

The quality control process will cause suspect radiance and sky feature retrieval files to be deleted from the ARM Archive. As these files underlie the sky feature optical density retrieval picture files, there may not be a complete set for any given period. The completeness of the set for a day may be determined at a glance from the file existence plot file discussed in [File Existence Plot File; Chapter 8: Utility Products](#).

4. Radiance Product

There is a single radiance product, the NetCDF radiance data file.

4.1 Radiance Data File

4.1.1 Purpose

The primary Whole Sky Imager data product in the ARM Archive is the NetCDF file that contains calibrated radiance data with 30–35 microsteradian spatial resolution for the entire 2π zenith hemisphere. There is one such file for each six-minute collection (ten-minute old style collection, see [Data Collection Strategy; Chapter 1: Overview](#)). The file contains data from all relevant spectral bands – 450 nm, 650 nm, 800 nm, and null filter (see the just referenced section). All such files for one 24-hour Greenwich day are packaged into a UNIX style tar archive that is quite large.

This is a rich data set with many current and potential uses. Certainly it is useful in the study of radiation transport and how it is influenced by clouds. By combining the solar illumination angle and many observation angles of an ice cloud with its radiance in the three spectral bands there is the possibility of retrieving cloud microphysical and optical parameters. Studies have been performed as well to retrieve aerosol parameters. And, of course, the radiance data serves as a basis for the sky feature retrieval discussed in [Chapter 5: Sky Feature Product](#).

4.1.2 Naming Convention

The radiance data file has the following name based on standard ARM naming practices.

`sssfffullradiancea1.yyyymmdd.hhmmss.cdf`

or

`sssfffullradianceb1.yyyymmdd.hhmmss.cdf`

where the blue characters are literal and

`sss ff` = the site and facility identifier as listed in Table 1
`yyyymmdd.hhmmss` = the standard 15 character ARM date time stamp showing year, month, day, hour, minute, and second.

As a result of a change in the ARM data level standard, the “a1” found in earlier files has been replaced by “b1” in later ones. There is no difference in content or structure between the two. The UNIX style tar archives associated with these files use the same basic name but have “.tar” appended to the end.

4.1.3 Dimensions, Variables, and Coding

4.1.3.1 Global Attributes

The global attributes in the radiance data file contain useful background information on the instrument and production environment. The user should always extract them with a suitable NetCDF

dump routine and become familiar with their content. As the names and values are purposely designed to be human readable, they are not listed separately herein.

4.1.3.2 Dimensions

The dimensions associated with the radiance data file are listed in Table 4 and discussed in the paragraphs below. This file is quite unusual in that the unlimited dimension is not associated with the time of the data sample. Recall that each file is for a specific collection time making time a constant.

ID	Name	Size
0	image_count	Unlimited
1	image_cols	512
2	image_rows	512
3	dark_file-types	2
4	char_5	5
5	char-10	10
6	char-25	25
7	char_64	64

The **image_count** dimension can take a value between one and four and is the number of image arrays stored in a file, one image for each spectral band collected; closed shutter images are not stored in this data file. The typical daylight and twilight value is three (450 nm, 650 nm, and 800 nm) and the typical nighttime value is one (null filter). On the night of the full moon the value may be four (450 nm, 650 nm, 800 nm, and null filter). The files are not stored in the orders just mentioned.

The **image_cols** and **image_rows** dimensions always have the value of 512 and are the number of columns and rows in the image arrays.

The other dimensions (**dark_file_types**, **char_5**, **char_10**, **char_25**, and **char_64**) are necessary in the development of the NetCDF file but are of no particular concern to its user.

4.1.3.3 Variables

The variables associated with the radiance data file are listed in Table 5 and discussed in the paragraphs below. The attributes associated with each variable contain valuable information; the user should extract them with a suitable NetCDF dump routine and become familiar with their content. In particular the attributes have null value, range, and allowed delta information.

The **base_time** variable – long name: base time in epoch – of type long is a scalar whose value is the number of seconds between January 1, 1970 at 0000 hours Greenwich time and the data collection event.

ID	Name	Type	Attribute Count	Dimension Count	Dimension Ids	Actual Dimensions
0	base_time	Long	5	0	--	--
1	time_offset	Double	5	0	--	--
2	cal_image	Float	6	3	1, 2, 0	512, 512, x
3	mask	Byte	6	2	1, 2	512, 512
4	image_color	Char	4	2	4, 0	5, x
5	set_quality	Char	4	0	--	--
6	header_date_time	Char	3	1	6	25
7	time_source	Char	5	0	--	--
8	time_status	Char	5	0	--	--
9	file	Char	3	2	7, 0	64, x
10	dark_file	Char	4	2	7, 3	64, 2
11	cal_file	Char	3	1	7	64
12	exposure	Float	6	1	0	x
13	neutral_filter	Short	6	1	0	x
14	spectral_filter	Short	6	1	0	x
15	arc_dest	Float	6	1	0	x
16	arc_pos	Float	6	1	0	x
17	trolley_dest	Float	6	1	0	x
18	trolley_pos	Float	6	1	0	x
19	sun_az	Float	6	1	0	x
20	sun_zen	Float	6	1	0	x
21	moon_az	Float	6	1	0	x
22	moon_zen	Float	6	1	0	x
23	light_source	Char	5	1	4	5
24	red_flags	Char	16	2	6, 0	25, x
25	yellow_flags	Char	26	2	6, 0	25, x
26	housing_temp	Float	6	0	--	--
27	env_temp	Float	6	0	--	--
28	ccd_temp	Float	6	0	--	--
29	humidity	Float	6	0	--	--
30	n2_press	Float	6	0	--	--
31	flow_rate	Float	6	0	--	--
32	hardware_version	Char	3	1	4	5
33	software_version	Char	3	1	4	5
34	camera_offset	Float	6	0	--	--
35	field_offset	Float	6	0	--	--
36	occultor_az_offset	Float	6	0	--	--
37	occultor_zen_offset	Float	6	0	--	--
38	center_column	Float	6	0	--	--
39	center_row	Float	6	0	--	--
40	radius	Float	6	0	--	--
41	field_az_update	Char	4	2	5, 0	10,x
42	site	Char	4	0	--	--
43	latitude	Float	6	0	--	--
44	longitude	Float	6	0	--	--
45	alt	Float	6	0	--	--

The **time_offset variable** – long name: offset from base_time – of type double is a scalar whose value is always zero. It is included for conformity with other ARM NetCDF files whose data is time dependent and that therefore require a time offset value for each data point.

The **cal_image** array – long name: calibrated image – of type float is the data of most interest to the user. It contains the calibrated radiance data of each spectral type collected. The array has three dimensions with the first two being the columns and rows of the image respectively. The third dimension defines layers in the file, one layer for each spectral collection. For the relationship between layers and spectral band refer to the variable image_color.

The **mask** array – long name: source, horizon, poles, occulter, and trolley mask – of type byte is a binary image that aids in the interpretation of each calibrated image. It has the same dimensions as a single layer of the cal_image array and corresponds to such on a pixel by pixel. The mask indicates pixels with valid data (mask = 1) and suspect data (mask = 0). Suspect data may be influenced by light source, horizon, structures or vegetation near the horizon, occulter pole mechanisms, occulter, or trolley.

The **image_color** – long name: image color – vector of type character identifies the spectral band of each layer of the cal_image array. The dimension of this vector corresponds one to one with the third dimension of that array. For example, if the first element (index [0]) in image_color contains the character string “clear”, then the layer image_color[i, j, 0] contains null filter radiance data, where i and j range from 0 to 511. Each element in image_color may have a value from the set {“clear”, “blue”, “red”, “nir”} corresponding to the spectral bands {null filter, 450 nm, 650 nm, and 800 nm} respectively; no value may be repeated.

The **set_quality** variable – long name: quality control character for image set – is a scalar of type character that identifies the perceived quality of the entire calibrated image set in this file based on instrument self diagnostic data. The quality control process discussed in [Data Quality Control; Chapter 1: Overview](#) is independent of this variable’s value; therefore the mere existence of a radiance data file in the ARM Archive is indication that the data control process found nothing objectionable with it regardless of the value of this variable. This variable may have a value from the set {“A”, “B”, “C”, “D”, “F”} with the assigned meanings of: A = no problems detected, B = minor problems that do not affect calibrated image validity, C = minor problems that could affect image validity, D = image calibration suspect, and F = bad calibrated images.

The **header_date_time** variable – long name: raw image header time stamp – of type character has the format: yyyy:mm:dd:hh:mm:ss where each letter stands for a numeral representing (in order) year : month : day : hour : minute : second. The value is that common value found in the header of the raw data files upon which this radiance file is built; see [Header Information; Dimensions, Variables, and Coding; Raw Data Files; Chapter 2: Raw Products](#). This date time group is the most accurate available; if the one imbedded in the ARM style file name is in disagreement then header_date_time should be trusted.

The **time_source** variable – long name: source for time – of type character describes the device which provided the time code for this data collection event. It may have a value from the set {“@”, “W”, “B”, “S”, “G”, “F”} with the assigned meaning of @ = null character, W = WWV, B = BIOS, S = system clock, G = GPS, and F = none.

The **time_status** variable – long name: clock diagnostic code – of type character is the code for the clock and should be accessed only by the sophisticated data user. It may have a value from the set {"@", "0", "1", "2", "3", "4", "9", "A"}. The null character is @, but the rest change meanings with the value of the time_source variable as listed below.

- If the time_source is W or G then 1 = first time grab, 2 = match with BIOS, 3 = match with system, 9 = mismatch with BIOS, and A = mismatch with system.
- If the time_source is S then 1 = only system clock valid.
- If the time_source is B then 4 = WWV or GPS invalid.
- If the time_source is F then 0 = no valid clocks.

The **file** vector – long name: raw image file name – of type character contains the raw data file names corresponding to image data in each layer of the cal_image array. The dimension of this vector corresponds one to one with the dimension of the image_color vector, which should be used as a key to interpreting the names in this vector and their relationship to other data in this radiance data file. The file name is stored without any directory information.

The **dark_file** vector – long name: raw dark image file names – of type character contains information concerning the source of data used to calculate dark current offset on a pixel by pixel basis. The vector has two elements, the second being for nighttime spectral images collected on the night of the full moon and the first being for all other images. Dark_file may have either (1) the raw closed shutter file names used in the radiance calibration, (2) the word “average” if a specific raw closed shutter file were not available and generic dark current information from the calibration file were used instead, or (3) the word “null” for the second value for all times other than the night of the full moon and for all times with old style data collection (see [Data Collection Strategy; Chapter 1: Overview](#)). File names are stored without any directory information.

The **cal_file** variable – long name: calibration file name – of type character is a scalar containing the name of the calibration file used to retrieve this radiance file. It is stored without any directory information.

The **exposure** vector – long name: exposure – of type float contains the camera exposure times in units of milliseconds for data in the cal_image array. The dimension of this vector corresponds one to one with the dimension of the image_color vector, which should be used as a key to interpreting the values in this vector and their relationship to other data in this radiance data file. Exposures range from one millisecond to two minutes.

The **neutral_filter** vector – long name: neutral density filter – of type short integer chronicles the neutral density filter used in collecting the data associated with each layer of the cal_image array. Neutral_filter may have a value from the set {0, 1, 2, 3, 4} where 0 = null value and the other numbers indicate approximate filter transmission thus: 1 = 1.0, 2 = 0.01, 3 = 0.001, and 4 = 1.0. The same value will often be repeated for all images in a set. The dimension of this vector corresponds one to one with the dimension of the image_color vector, which should be used as a key to interpreting the values in this vector and their relationship to other data in this radiance data file.

The **spectral_filter** vector – long name: spectral filter – of type short integer chronicles the spectral filter used in collecting the data associated with each layer of the cal_image array. Spectral_filter may have a value from the set {0, 1, 2, 3, 4} where 0 = null value and the other numbers indicate the approximate filter center wavelength and bandpass thus: 1 = 800 nm and 70 nm FWHM, 2 = null filter, 3 = 650 nm and 70 nm FWHM, and 4 = 450 nm and 70 nm FWHM. The same value should not be repeated within a radiance file except for 0. The dimension of this vector corresponds one to one with the dimension of the image_color vector, which should be used as a key to interpreting the values in this vector and their relationship to other data in this radiance data file.

The **arc_dest** vector – long name: occulter arc intended destination – of type float records the commanded position of the occulter arc based on calculations using the sun or moon position. The value is in angular decimal degrees and has a range of 0° to 180° with 0° being to the east, 90° at zenith, and 180° to the west. The dimension of this vector corresponds one to one with the dimension of the image_color vector, which should be used as a key to interpreting the values in this vector and their relationship to other data in this radiance data file. All values in the vector should be similar if not identical.

The **arc_pos** vector – long name: occulter arc sensed destination – of type float records the measured position of the occulter arc. It will be within the positioning mechanism's error of arc_dest. It will be different than the arc's true position by the sensing mechanism's error. See the companion document, [Whole Sky Imager Hardware Guide](#), for a discussion of the hardware involved and issues associated with it. The value is in angular decimal degrees and has a range of 0° to 180° with 0° being to the east, 90° at zenith, and 180° to the west. The dimension of this vector corresponds one to one with the dimension of the image_color vector, which should be used as a key to interpreting the values in this vector and their relationship to other data in this radiance data file.

The **trolley_dest** vector – long name: trolley intended destination – of type float records the commanded position of the trolley on the occulter arc based on calculations using the sun or moon position. The value is in angular decimal degrees and has a range of 0° to 180° with 0° being to the north, 90° at zenith, and 180° to the south. For instruments that are not equipped with a trolley, this vector will be filled with the IEEE value for not a number. The dimension of this vector corresponds one to one with the dimension of the image_color vector, which should be used as a key to interpreting the values in this vector and their relationship to other data in this radiance data file. All values in the vector should be similar if not identical.

The **trolley_pos** vector – long name: trolley sensed destination – of type float records the sensed position of the trolley on the occulter arc. It will be within the positioning mechanism's error of trolley_dest. It will be different than the trolley's true position by the sensing mechanism's error. See the companion document, [Whole Sky Imager Hardware Guide](#), for a discussion of the hardware involved and issues associated with it. The value is in angular decimal degrees and has a range of 0° to 180° with 0° being to the north, 90° at zenith, and 180° to the south. For instruments that are not equipped with a trolley, this vector will be filled with the IEEE value for not a number. The dimension of this vector corresponds one to one with the dimension of the image_color vector, which should be used as a key to interpreting the values in this vector and their relationship to other data in this radiance data file.

The two vectors sun_az and sun_zen – long names: sun azimuth and sun zenith – of type float contain respectively the calculated azimuth and zenith angles for the sun’s position based on the date and time of the underlying raw data file collection and the location of the instrument. The values are in angular decimal degrees. Sun_az has the range 0° to 360° and is measured clockwise from geographic north such that east is 90°. Sun_zen has the range 0° to 180° and is measured from the local vertical such that the horizon is 90° and any value greater than 90° indicates that it is nighttime. The dimension of these vectors corresponds one to one with the dimension of the image_color vector, which should be used as a key to interpreting the values in these vectors and their relationship to other data in this radiance data file. All values in each of these vectors should be similar if not identical.

The two vectors moon_az and moon_zen – long names: moon azimuth and moon zenith – of type float contain respectively the calculated azimuth and zenith angles for the moon’s position based on the date and time of the underlying raw data file collection and the location of the instrument. The values are in angular degrees and have the same range and meaning as for sun_az and sun_zen. The dimension of these vectors corresponds one to one with the dimension of the image_color vector, which should be used as a key to interpreting the values in these vectors and their relationship to other data in this radiance data file. All values in each of these vectors should be similar if not identical.

The light_source variable – long name: light source – of type character is a scalar that specifies the dominant light source in the sky at the time of this radiance data file, e.g., the sun, and thus the light source used to calculate arc_dest and trolley_dest. If neither the sun nor the moon is above the horizon the arc is parked to the east, i.e., arc_dest near 0°. Light_source may have a value from the set {"@@@@@", "sun", "moon", "none"} where @@@@@ is the null value and the rest are self explanatory.

The red_flags vector – long name: serious trouble flags – of type character contains instrument self diagnostic information of a critical nature. Each element has 25 characters that are read right to left. Each of these 25 characters may have the value 0 or 1, where 0 indicates no problem and 1 indicates a problem. The dimension of this vector corresponds one to one with the dimension of the image_color vector, which should be used as a key to interpreting the spectral data collection to which the flag values in these vectors refers and their relationship to other data in this radiance data file. For instruments operating with the new style of collection (see [Data Collection Strategy; Chapter 1: Overview](#)) only the first 11 characters are active; their meanings are shown in Table 6. All deployed instruments are now using this scheme. Red_flags values for old style collection radiance files in the historical record have meanings that are almost identical to these but not quite. Different releases of the Whole Sky Imager control software program had slightly different red_flags. The tracking of these differences is beyond the scope of this document; specific instances may be referred to the instrument mentor for interpretation.

The yellow_flags vector – long name: warning flags – of type character contains instrument self diagnostic information of an informative nature. Each element has 25 characters that are read right to left. Each of these 25 characters may have the value 0 or 1, where 0 indicates no problem and 1 indicates a problem. The dimension of this vector corresponds one to one with the dimension of the image_color vector, which should be used as a key to interpreting the spectral data collection to which the flag values in these vectors refers and their relationship to other data in this radiance data file. For instruments operating with the new style of collection (see [Data Collection Strategy; Chapter 1: Overview](#)) only the first 16 characters are active; their meanings are shown in Table 7. All deployed instruments are now

Table 6. Interpretation of the red_flags characters.

Position	Meaning
1	No response from camera; images are not being snapped
2	Camera CCD chip temperature above 0°C
3	Environmental housing temperature above 49°C
4	Coolant flow rate less than .09 gallons per minute
5	Camera housing temperature above 49°C
6	Arc occulter not responding; arc position > 8° off
7	Trolley occulter not responding; trolley position > 8° off
8	Neutral density filter not in position
9	Spectral filter not in position
10	Flux levels abnormal; too high, at least 5% of pixels are offscale bright
11	Flux levels abnormal; too low, pixel values < 100 in dark region

Table 7. Interpretation of the yellow_flags characters.

Position	Meaning
1	Camera CCD chip temperature above -30°C
2	Environmental housing temperature above 32°C
3	Coolant flow rate less than 0.125 gallons per minute
4	Camera housing temperature above 32°C
5	Nitrogen pressure is less than 2 PSI.
6	Arc occulter not responding; arc position > 2° off
7	Trolley occulter not responding; trolley position > 2° off
8	GPS clock not responding
9	Hard disk is full
10	Unable to resolve IP address
11	Unknown host name
12	Unknown user name
13	Unknown password
14	Can't send files to NT buffer computer
15	Flux levels abnormal, at least 1% of pixels are offscale bright
16	Flux levels abnormal; areas just inside image are < 5% as bright as dark region; shutter may be malfunctioning

using this scheme. Yellow_flags values for old style collection radiance files in the historical record have meanings that differ somewhat from these and differing number of active characters. Also, various historical releases of the Whole Sky Imager control software program had different yellow_flags. The tracking of these is beyond the scope of this document; specific instances may be referred to the instrument mentor for interpretation.

The **housing_temp** variable – long name: camera housing temperature – of type float is a scalar that contains the temperature of the camera housing inside the environmental housing at the time of the data collection. Its units are degrees Celsius. Optimum values are below 32°C.

The **env_temp** variable – long name: environmental housing temperature – of type float is a scalar that contains the temperature of the environmental housing at the time of the data collection. Its units are degrees Celsius. Optimum values are below 32°C.

The **ccd_temp** variable – long name: camera CCD chip temperature – of type float is a scalar that contains the temperature of the camera CCD chip at the time of the data collection. Its units are degrees Celsius. Optimum values are below -30°C.

The **humidity** variable – long name: environmental housing relative humidity – of type float is a scalar that contains the relative humidity inside the environmental housing at the time of the data collection. Its units are percentage.

The **n2_press** variable – long name: camera housing nitrogen pressure – of type float is a scalar that contains the differential pressure of the dry nitrogen fill inside camera housing inside the environmental housing at the time of the data collection. Its units are pounds per square inch. Optimum values are above 2 PSI which is an overpressure.

The **flow_rate** variable – long name: camera coolant flow rate – of type float is a scalar that contains the flow rate of the camera CCD coolant at the time of the data collection. Its units are gallons per minute. Optimum values are above 0.125 GPM.

The **hardware_version** variable – long name: hardware version – of type character is a scalar that reports the instrument hardware revision as indicated in the raw data files that underlie this radiance file. The instrument gets the value to insert in the raw files from a lookup table installed by the instrument operator; therefore, this value is only valid if the operator keeps that table up to date.

The **software_version** variable – long name: runwsi software version – of type character is a scalar that reports the instrument's control software version as indicated in the raw data files that underlie this radiance file. As with the hardware_version, the instrument gets the value to insert from a lookup table installed by the instrument operator; therefore, this value is only valid if the operator keeps that table up to date.

The **camera_offset** variable – long name: camera offset – of type float is a scalar whose value is the angular difference between the line along the camera's pixel columns and the environmental housing's north south alignment fiducial. Its units are angular degrees and its value should be close to 0°. Its value is transcribed from that in the raw data files which underlie this radiance file. The instrument gets the value to insert in the raw files from a lookup table installed by the instrument operator; therefore, this value is only valid if the operator keeps that table up to date. Since camera_offset is not utilized in current radiance and sky feature retrievals, the accuracy of its value is not a matter of highest importance.

The **field_offset** variable – long name: field offset – of type float is a scalar whose value is the angular difference between the environmental housing's north south alignment fiducial and the direction of geographic north. Its units are angular degrees and its value should be close to 0° for fixed base instruments. To date only the SHEBA instrument, located on a ship, is not fixed base. Considering the two cases separately, we have:

- For fixed base instruments, `field_offset` is transcribed from the raw data files which underlie this radiance file. The instrument gets the value to insert in the raw files from a lookup table installed by the instrument operator; therefore, this value is only valid if the operator keeps that table up to date. Since `field_offset` is not utilized in current radiance and sky feature retrievals, the accuracy of its value is not a matter of highest importance.
- For moveable base instruments like on the SHEBA ship, `field_offset` is determined from GPS instrumentation and should be quite accurate.

The **`occultor_az_offset`** and **`occultor_zen_offset`** variables – long names: occultor azimuth offset and occultor zenith offset – of type float are scalars whose values indicate offsets associated with the occultor installation. `Occultor_az_offset` is the angular difference between the occultor’s pivot axis and the direction of geographic north. `Occultor_zen_offset` is the angular difference between the occultor’s as installed zenith position and true zenith. Both variables have the units of angular degrees and both values should be close to 0°. Their values are transcribed from the raw data files which underlie this radiance file. The instrument gets the values to insert in the raw files from a lookup table installed by the instrument operator; therefore, these values are only valid if the operator keeps that table up to date. Since neither `occultor_az_offset` nor `occultor_zen_offset` are utilized in current radiance and sky feature retrievals, the accuracy of their values is not a matter of highest importance.

The **`center_column`** and **`center_row`** variables – long names: image center column and image center row – of type float are scalars whose values indicate the position of the sky zenith point as imaged on the camera CCD chip. Since CCD columns are south north as installed and are numbered 0 to 511 respectively, any deviation of `center_column` from the midpoint value between columns 255 and 256 indicates an east west alignment anomaly. Similarly the CCD rows are west east as installed and are numbered 0 to 511 respectively; any deviation of `center_row` from the midpoint value between rows 255 and 256 indicates a north south alignment anomaly. Their values are transcribed from the raw data files which underlie this radiance file. The instrument gets the values to insert in the raw files from a lookup table installed by the instrument operator; therefore, these values are only valid if the operator keeps that table up to date. While knowledge of the center column and row is critical to the radiance and sky feature retrievals, the values used in those retrievals are derived from data in the calibration file (see [Variables; Dimensions, Variables, and Coding; Calibration File; Chapter 7: Calibration Product](#)) and not from `center_column` and `center_row`. Therefore, the accuracy of their values is not a matter of highest importance.

The **`radius`** variable – long name: image radius – of type float is a scalar whose value indicates averaged radius on the CCD chip from the imaged position of the sky zenith point to the imaged horizon. The latter forms an approximate, but not perfect, circle. The value of the variable `radius` is transcribed from the raw data files which underlie this radiance file. The instrument gets its value to insert in the raw files from a lookup table installed by the instrument operator; therefore, that value is only valid if the operator keeps that table up to date. While knowledge of image size is critical to the radiance and sky feature retrievals, they use geometric values derived from data in the calibration file (see [Variables; Dimensions, Variables, and Coding; Calibration File; Chapter 7: Calibration Product](#)) and not from the variable `radius`. Therefore, the accuracy of its value is not a matter of highest importance.

The **field_az_update** vector – long name: field azimuth update time – of type character indicates the time of the last GPS update of the `field_offset` variable. It is valid only for moveable base instruments such as the Whole Sky Imager deployed on the SHEBA ship; for fixed base instruments it has no meaning. The format is hhmm:ss where each letter stands for a numeral representing (in order) hour : minute : second of the day of image collection. The value is obtained from the headers of the raw data files that underlie this radiance file. The dimension of this vector corresponds one to one with the dimension of the `image_color` vector, which should be used as a key to interpreting the values in these vectors and their relationship to other data in this radiance data file. All values in this vector should be similar if not identical.

The **site** variable – long name: site location – of type character is a scalar that identifies the site location of the instrument involved in the data collection. Its value is one of the set {"sheba", "nsa-1", "nsa-2", "twp-1", "twp-2", "twp-3", "sgp-1", "sgp-2", "other"}. This set corresponds to the designators in Table 1 as follows {shbC1, nsaC1, nsaC2, twpC1, twpC2, twpC3, sgpC1, sgpL3, (none)} respectively. The value of the variable `site` should correspond to the site information imbedded in the radiance data file name; if not clues such as horizon features in the associated sky picture file can be used to determine which is correct. The value of `site` is obtained from the headers of the raw data files that underlie this radiance file, and the instrument gets the values to insert in the raw files from a lookup table installed by the instrument operator. Therefore, these values are only valid if the operator keeps that table up to date. Since this variable serves only as a check on the site designator in the file name, the accuracy of its value is not a matter of highest importance.

The **latitude** and **longitude** variables – long names: latitude and longitude – of type float are scalars that specify the geographical coordinates of the instrument's location. The units for both variables are decimal degrees. Latitude ranges between -90° and $+90^{\circ}$, where -90° is at the Earth's South Pole, 0° is at the equator, and 90° is at the North Pole. Longitude ranges between 180° and $+180^{\circ}$, where 0° is the meridian through Greenwich, England, positive values are in the eastern hemisphere, and negative values are in the western hemisphere. Fixed base (all Whole Sky Imagers located at permanent ARM sites) and moveable base instruments (the one on SHEBA ship) develop the data for these variables a bit differently. Considering the two cases separately, we have:

- For fixed base instruments, latitude and longitude are transcribed from the raw data files which underlie this radiance file. The instrument gets the value to insert in the raw files from a lookup table installed by the instrument operator; therefore, these values are only valid if the operator keeps that table up to date. Since the values for latitude and longitude in the calibration file (see [Variables: Dimensions, Variables, and Coding; Calibration File; Chapter 7: Calibration Product](#)) override those stored in these variables during radiance and sky feature retrievals, the accuracy of their values here is not a matter of highest importance.
- For moveable base instruments like on the SHEBA ship, latitude and longitude is determined from GPS instrumentation and should be quite accurate. They are not overridden by calibration file values as in the fixed base instruments.

The **alt** variable – long name: altitude – of type float is a scalar that specifies the geographical altitude of the instrument's location. It is expressed in meters above mean sea level. Fixed base (all

Whole Sky Imagers located at permanent ARM sites) and moveable base instruments (the one on SHEBA ship) develop the data for these variables a bit differently. Considering the two cases separately, we have:

- For fixed base instruments, altitude is transcribed from the raw data files which underlie this radiance file. The instrument gets the value to insert in the raw files from a lookup table installed by the instrument operator; therefore, the value is only valid if the operator keeps that table up to date. Since the value of altitude in the calibration file (see [Variables; Dimensions, Variables, and Coding; Calibration File; Chapter 7: Calibration Product](#)) overrides those stored in these variables during radiance and sky feature retrievals, the accuracy of their values here is not a matter of highest importance.
- For moveable base instruments like on the SHEBA ship, altitude is determined from GPS instrumentation, but is not very accurate when GPS dither is enabled by the satellite system's controllers as it was during the SHEBA experiment. It is not overridden by calibration file values as in the fixed base instruments. Fortunately, significant errors in altitude do not translate into errors in radiance retrievals and only very minor errors in sky feature retrievals.

4.1.4 Algorithm

4.1.4.1 Introduction

The algorithms that implement the radiance data file retrieval are encoded in an IDL program named `WsiIdpc4_vx`, where `x` represents a version number. The version extant is 13 as of this writing in September 2003. The program and its subroutines were written by the instrument mentor and are maintained by him as well. They are executed at the ARM Archive as part of the Whole Sky Imager data ingest and retrieval process. `WsiIdpc4_vx` is coded for execution in a production environment, meaning that it can handle nearly any anomaly in raw file data streams or input data and continue processing to a benign conclusion.

4.1.4.2 Initialization

Initial blocks in the program open a log file (see [Log Files; Chapter 8: Utility Products](#)), identify the proper raw files to use for the time of the radiance data file to be generated, and access the correct calibration file (see [Calibration File; Chapter 7: Calibration Product](#)). Many checks are performed on these files to insure proper size and content. Next the new radiance data file is opened and dimensions, variables, and attributes are defined.

4.1.4.3 Calibration

To prepare for calibration, the proper raw closed shutter files are identified. These will not normally have the same time stamp as the raw spectral data files, since closed shutter files are acquired only when camera exposure changes. If the selected closed shutter files are not appropriate, then generic dark current data from the calibration file is developed for use instead. Next, each spectral image is transformed into calibrated radiance data by first subtracting the identified dark current data, multiplying by the flat field image, dividing by the exposure, multiplying by the proper roll off image, and finally multiplying by the proper absolute calibration scalar. Other than the raw spectral and (perhaps) dark

current data, all the data items mentioned in the preceding sentence are from the calibration file. Please refer to [Variables, Dimensions, Variables, and Coding; Calibration File; Chapter 7: Calibration Product](#) for a detailed explanation of each.

4.1.4.4 Quality Checking

After calibration, numerous quality and consistency checks are performed, using mainly data from the individual raw file data headers. For a list of the data items within the headers refer to the section [Header Information; Dimensions, Variables, and Coding; Raw Data Files; Chapter 2: Raw Products](#). The result of these checks is a value for `set_quality`, which is set to the lowest quality value for any individual check. The items in the following list are included.

Note that this quality checking is fundamentally different from that discussed in [Data Quality Control; Chapter 1: Overview](#). That quality control process identifies major anomalies in the image capture process that make the image useless, for example dew on the dome, someone standing in the field of view, and sunlight shining on the dome, and causes retrieved products to be stricken from the ARM Archive. The process discussed here looks for more subtle anomalies that down grade the quality of retrieved calibrated radiances but do not render them useless.

- Does a proper shutter closed image exist? If not, then the quality is D. Even though radiance data can be retrieved using generic dark current data from the calibration file that measured under field conditions is much to be preferred.
- If a proper shutter closed image does exist, are its first and second moments across all pixels within a factor of two of those moments for the generic dark current from the calibration file? If not, then the quality is C. While some variation is expected as instrument housing and electrical conditions vary from day to day, excessive dark current variation would indicate an underlying problem that could affect CCD quantum efficiency and hence image data.
- Were all `red_flags` fields set to zero when the closed shutter image was collected? If not, then the quality is C. Since the closed shutter image is suspect in this case, the generic dark current data from the calibration file is used in the radiance retrieval.
- Were all `yellow_flags` fields set to zero when the closed shutter image was collected? If not, then the quality is B. Since the closed shutter image is minimally suspect in this case, it can still be used in the radiance retrieval.
- Does a proper closed shutter image of the second type exist if needed? If not, then the quality is D. Such a file is needed at nighttime on the night of a full moon if the new collection strategy is being used (see the section “Data Collection Strategy” of Chapter 1). Even though radiance data can be retrieved using generic dark current data from the calibration file that measured under field conditions is much to be preferred.
- If a proper closed shutter image of the second type does exist, are its first and second moments across all pixels within a factor of two of those moments for the generic dark current from the calibration file? If not, then the quality is C. While some variation is expected as instrument housing

and electrical conditions vary from day to day, excessive dark current variation would indicate an underlying problem that could affect CCD quantum efficiency and hence image data.

- Were all red_flags fields set to zero when the closed shutter image of the second type was collected? If not, then the quality is C. Since the closed shutter image type 2 is suspect in this case, the generic dark current data from the calibration file is used in lieu of it in the radiance retrieval.
- Were all yellow_flags fields set to zero when the closed shutter image of the second type was collected? If not, then the quality is B. Since the closed shutter image type 2 is minimally suspect in this case, it can still be used in the radiance retrieval.
- If a null filter file is part of the set, does it have the same exposure as the selected closed shutter file? If not, then the quality is C. The dark current is dependent on exposure time, and since the exposures do not match the generic dark current data from the calibration file must be used in lieu of the closed shutter file.
- If a null filter file is part of the set, were all red_flags fields set to zero when the null filter image was collected? If not, then the quality is D. The radiance retrieval will proceed but whatever caused the red flag will be affecting the data.
- If a null filter file is part of the set, were all yellow_flags fields set to zero when the null filter image was collected? If not, then the quality is C. The radiance retrieval will proceed but whatever caused the yellow flag could be affecting the data.
- If a 450 nm file is part of the set, does it have the same exposure as the selected closed shutter file? If not, then the quality is C. The dark current is dependent on exposure time, and since the exposures do not match the generic dark current data from the calibration file must be used in lieu of the closed shutter file.
- If a 450 nm file is part of the set under new style collection (see [Data Collection Strategy: Chapter 1: Overview](#)) on the night of a full moon, does it have the same exposure as the selected closed shutter file type two? If not, then the quality is C. The dark current is dependent on exposure time, and since the exposures do not match the generic dark current data from the calibration file must be used in lieu of the closed shutter file type two.
- If a 450 nm file is part of the set, were all red_flags fields set to zero when the 450 nm image was collected? If not, then the quality is D. The radiance retrieval will proceed but whatever caused the red flag will be affecting the data.
- If a 450 nm file is part of the set, were all yellow_flags fields set to zero when the 450 nm image was collected? If not, then the quality is C. The radiance retrieval will proceed but whatever caused the yellow flag could be affecting the data.
- If a 650 nm file is part of the set, does it have the same exposure as the selected closed shutter file? If not, then the quality is C. The dark current is dependent on exposure time, and since the exposures do not match the generic dark current data from the calibration file must be used in lieu of the closed shutter file.

- If a 650 nm file is part of the set under new style collection (see [Data Collection Strategy: Chapter 1: Overview](#)) on the night of a full moon, does it have the same exposure as the selected closed shutter file type two? If not, then the quality is C. The dark current is dependent on exposure time, and since the exposures do not match the generic dark current data from the calibration file must be used in lieu of the closed shutter file type two.
- If a 650 nm file is part of the set, were all red_flags fields set to zero when the 650 nm image was collected? If not, then the quality is D. The radiance retrieval will proceed but whatever caused the red flag will be affecting the data.
- If a 650 nm file is part of the set, were all yellow_flags fields set to zero when the 650 nm image was collected? If not, then the quality is C. The radiance retrieval will proceed but whatever caused the yellow flag could be affecting the data.
- If an 800 nm file is part of the set, does it have the same exposure as the selected closed shutter file? If not, then the quality is C. The dark current is dependent on exposure time, and since the exposures do not match the generic dark current data from the calibration file must be used in lieu of the closed shutter file.
- If an 800 nm file is part of the set under new style collection (see [Data Collection Strategy: Chapter 1: Overview](#)) on the night of a full moon, does it have the same exposure as the selected closed shutter file type two? If not, then the quality is C. The dark current is dependent on exposure time, and since the exposures do not match the generic dark current data from the calibration file must be used in lieu of the closed shutter file type two.
- If an 800 nm file is part of the set, were all red_flags fields set to zero when the 800 nm image was collected? If not, then the quality is D. The radiance retrieval will proceed but whatever caused the red flag will be affecting the data.
- If an 800 nm file is part of the set, were all yellow_flags fields set to zero when the 800 nm image was collected? If not, then the quality is C. The radiance retrieval will proceed but whatever caused the yellow flag could be affecting the data.
- Does a full spectral set of images exist? If not, then the quality is D. For instruments with 800 nm filters installed, which includes all current instruments, a full set is either {450 nm, 650 nm, and 800 nm} for daylight and twilight or {null filter} for nighttime. For historical data sets for instruments from before the installation of the 800 nm filter, the set {450 nm, 650 nm} is sufficient for no down check during daylight and twilight.
- Were all images in the set collected using the same neutral density filter? If not, then the quality is C. Except for the full moon case with the new style of collection (see [Data Collection Strategy: Chapter 1: Overview](#)) all images should be collected with as little change in instrument configuration as possible to reduce errors in their comparison.
- Did all images in the set have the same year in their raw file headers? If not, then the quality is B. Differing years would indicate some instrument control software problem or a large unexplained gap between individual collections.

- Did all images in the set have the same month in their raw file headers? If not, then the quality is B. Differing months would indicate some instrument control software problem or a large unexplained gap between individual collections.
- Did all images in the set have the same day in their raw file headers? If not, then the quality is B. Differing days would indicate some instrument control software problem or a large unexplained gap between individual collections.
- Did all images in the set have the same zulu time (Greenwich hours and minutes) in their raw file headers? If not, then the quality is B. Differing times would indicate some instrument control software problem or a large unexplained gap between individual collections. Failure of any of these last four checks would preclude definitively checking the file name time against the imbedded raw file header times.
- Did all images in the set have the same time source in their raw file headers? If not, then the quality is B. Differing time sources could indicate a jump in the header times listed.
- Did all images in the set have the same time status in their raw file headers? If not, then the quality is B. Differing time statuses could indicate a jump in the header times listed.
- Did all images in the set have the same light source in their raw file headers? If not, then the quality is D. Since light source is only calculated by the instrument control software at the beginning of a collection set, this condition would indicate a software problem or a large unexplained gap between individual collections.
- Did the images in the set (including the selected closed shutter images) have camera housing temperatures that differed by less than 10%? If not, then the quality is B. Rapidly changing temperatures in the camera housing could thermally shock the camera and cause a change in the CCD chip quantum efficiency.
- Did the images in the set (including the selected closed shutter images) have environmental housing temperatures that differed by less than 10%? If not, then the quality is B. Rapidly changing temperatures in the environmental housing would be symptomatic of a larger instrument problem, such as a cooler, heater, or thermostat failure.
- Did the images in the set (including the selected closed shutter images) have camera CCD chip temperatures that differed by less than 10%? If not, then the quality is B. Rapidly changing CCD chip temperatures will cause a change in the CCD chip quantum efficiency and thus make calibration suspect.
- Did the images in the set (including the selected closed shutter images) have environmental housing relative humidity percentages that differed by less than 20%? If not, then the quality is B. Rapidly changing relative humidity in the environmental housing would be symptomatic of a larger instrument problem, such as a cooler, heater, or thermostat failure.
- Did the images in the set (including the selected closed shutter images) have camera housing nitrogen pressures that differed by less than 10%? If not, then the quality is B. Rapidly changing

pressure in the camera housing would be symptomatic of a larger instrument problem, such as the ingesting of moist, condensing air into the optics train.

- Did the images in the set (including the selected closed shutter images) have camera coolant flows that differed by less than 10%? If not, then the quality is B. Rapidly changing flow rates could cause CCD chip temperature fluctuations that will cause a change in the CCD chip quantum efficiency and thus make calibration suspect.
- Did all images in the set (including the selected closed shutter images) have the same hardware version in their raw file headers? If not, then the quality is B. Since hardware version is a lookup table item, its variance would indicate an instrument control software problem.
- Did all images in the set (including the selected closed shutter images) have the same software version in their raw file headers? If not, then the quality is B. Since software version is a lookup table item, its variance would indicate an instrument control software problem.
- Did all images in the set (including the selected closed shutter images) have the same camera offset in their raw file headers? If not, then the quality is D. Since camera offset is a lookup table item, its variance would indicate an instrument control software problem.
- Did all images in the set (including the selected closed shutter images) have the same field offset in their raw file headers? If not, then the quality is D. Since field offset is a lookup table item for fixed base instruments, its variance would indicate an instrument control software problem. For moveable base instruments such as the Whole Sky Imager deployed on the SHEBA ship, the platform turning rate is slow compared to the image collection rate. Even so the requirement for sameness is too strict since the dark images may have been taken hours before the rest of the set; a tolerance band would be better. This issue will have to be revisited when the SHEBA data is processed.
- Did all images in the set (including the selected closed shutter images) have the same occulter azimuth offset in their raw file headers? If not, then the quality is C. Since occulter azimuth offset is a lookup table item, its variance would indicate an instrument control software problem.
- Did all images in the set (including the selected closed shutter images) have the same occulter zenith offset in their raw file headers? If not, then the quality is C. Since occulter zenith offset is a lookup table item, its variance would indicate an instrument control software problem.
- Did all images in the set (including the selected closed shutter images) have the same center column in their raw file headers? If not, then the quality is D. Since center column is a lookup table item, its variance would indicate an instrument control software problem. If the center columns really were different, then the images could not be aligned for dark current subtraction or downstream sky feature retrieval. The retrieval software uses a calibration file value for center column.
- Did all images in the set (including the selected closed shutter images) have the same center row in their raw file headers? If not, then the quality is D. Since center row is a lookup table item, its variance would indicate an instrument control software problem. If the center rows really were different, then the images could not be aligned for dark current subtraction or downstream sky feature retrieval. The retrieval software uses a calibration file value for center row.

- Did all images in the set (including the selected closed shutter images) have the same image radius in their raw file headers? If not, then the quality is D. Since image radius is a lookup table item, its variance would indicate an instrument control software problem. If the image radii really were different, then the images could not be aligned for dark current subtraction or downstream sky feature retrieval. The retrieval software uses a calibration file value for image radial geometry.
- Did all images in the set (including the selected closed shutter images) have the same site identifier in their raw file headers? If not, then the quality is B. Since the site identifier is a lookup table item, its variance would indicate an instrument control software problem. This will be of more importance beginning with the upgraded instrument at the Southern Great Plains site in the fall of 2003 since this variable will be used to generate ARM style long file names for the raw data files.
- Did all images in the set (including the selected closed shutter images) have the same latitude in their raw file headers? If not, then the quality is B. Since latitude is a lookup table item, its variance would indicate an instrument control software problem. The retrieval software uses a calibration file value for latitude.
- Did all images in the set (including the selected closed shutter images) have the same longitude in their raw file headers? If not, then the quality is B. Since longitude is a lookup table item, its variance would indicate an instrument control software problem. The retrieval software uses a calibration file value for longitude.
- Were the spectral images collected with the occulter arc edge closest to the horizon more than 10° above it? If not, then the quality is C. The quality is marked down in this case because the image cannot be analyzed to adjust the position of the occulter mask generated (see [Mask Generation; Algorithm; Radiance Data File; Chapter 4: Radiance Product](#)). If the optical encoding on the upgraded unit being deployed at the Southern Great Plains in the fall of 2003 for occulter position proves to be accurate, then this check may be eliminated.
- Did the edge analysis to refine the occulter position (see [Mask Generation; Algorithm; Radiance Data File; Chapter 4: Radiance Product](#)) successfully discover edges? If not, then the quality is C. The quality is marked down because the occulter mask generated has a somewhat inaccurate positioning. If the optical encoding on the upgraded unit being deployed at the Southern Great Plains in the fall of 2003 for occulter position proves to be accurate, then this check may be eliminated.
- Did the edge analysis to refine the occulter position (see [Mask Generation; Algorithm; Radiance Data File; Chapter 4: Radiance Product](#)) find edges within 25% of the occulter's physical width? If not, then the quality is C. The quality is marked down because the occulter mask generated has a somewhat inaccurate positioning. If the optical encoding on the upgraded unit being deployed at the Southern Great Plains in the fall of 2003 for occulter position proves to be accurate, then this check may be eliminated.
- Was the trolley type specified in the calibration file one of the set recognized by the retrieval code? If not, then the quality is D. The mask will not contain features to cover the radiances influenced by the trolley image.

As mentioned before, when these checks are complete the lowest assigned quality figure is used to set the variable `set_quality`.

4.1.4.5 Mask Generation

The final major task is to generate the mask to be loaded into byte variable `mask`. This mask must identify all pixels whose radiances are influenced by features other than sky and which should be eliminated from most analyses. The mask has the following components. The physical constants necessary for the calculations are in the calibration file, as are the pixel coordinate to sky coordinate conversions; please refer to [Variables; Dimensions, Variables, and coding; Calibration file; Chapter 7: Calibration Product](#) for their description.

- It has a horizon band to uniformly block any pixel whose zenith angle is greater than a fixed value found in the calibration file. This angle is often set to 90°.
- It has a semicircular region to block any pixel within a fixed angle of either the south or north occulter pivot where some equipment extends upwards into the field of view. The angle for each pivot is set separately in the calibration file; they are often set to 10°.
- It has an irregular horizon band to block any pixel associated with trees, buildings, etc. The detailed description of this band is in the calibration file.
- It has a ladder shaped structure to block the occulter arc. Since the arc moves from collection to collection, its position must be carefully calculated and then a complex geometric solution obtained to determine which pixels contain its image. Since the reported occulter arc position in the variable `arc_pos` can have several angular degrees of error associated with it, it is only used as a starting position to find the occulter. An edge analysis is performed around that position to find the actual occulter position. If the optical encoding on the upgraded unit being deployed at the Southern Great Plains in the fall of 2003 for occulter position proves to be accurate, then calculation may be considerably simplified.
- It has a disk shaped structure to block the occulter trolley for images taken with an instrument that has a trolley installed. As with the occulter arc, the `trolley_pos` variable may have some error. Images where the error is significant are deleted from the ARM Archive by the process discussed in [Data Quality Procedure; Data Quality Control; Chapter 1: Overview](#) since the sun or moon would be shining on the dome. Therefore the algorithm obtains the geometric solution for affected pixels using the value `trolley_pos` without correction. Again, if the optical encoding on the upgraded unit being deployed at the Southern Great Plains in the fall of 2003 for trolley position proves to be accurate, then this calculation will be improved.
- It has a strip shaped structure to block the occulter shade for images taken with an instrument that does not have a trolley. Since the shade is rigidly fixed to the occulter arc there is no error in knowing its position beyond that of the arc's position. The algorithm obtains the geometric solution for affected pixels using the shade descriptions found in the calibration file.

4.1.4.6 Finalization

The various NetCDF variables are written as the associated data is developed throughout the IDL code. In the final section, this process is completed and files closed as appropriate.

4.1.5 Usage Hints

The careful reader of the above sections in this chapter will have a good idea of how to extract information from the Whole Sky Imager radiance data file and how to use it appropriately. However, a few hints may make the process more enjoyable.

IDL handles character variables in NetCDF files as ASCII encoded byte strings of fixed length, hence the extra dimension variables in Table 4. If one is having trouble matching a search character string to a value retrieved from a NetCDF character variable, it is best to print out the associated character string one time to discover where the blanks and nulls are inserted to pad out to the fixed length. Then the search string can be modified appropriately. Also remember that character variables are case sensitive.

Pixel columns and rows can be related to the angular sky coordinates of zenith and azimuth by using conversion arrays contained in the calibration file; see [Variables; Dimensions, Variables, and Coding; Calibration File; Chapter 7: Calibration Product](#). Arrays for the inverse conversion are also to be found there.

When studying sky radiances, the analyst should always use the mask array to avoid improper data, such as would come from the imaging of structures or the occulter. The mask is designed to cover even marginally affected pixels; however, it is always possible that it is slightly misplaced. Therefore a quick scan of outliers in a data set is in order and might reveal a set of pixels adjacent to the mask – pixels that the mask should have covered but didn't.

4.1.6 Limitations

The Whole Sky Imager radiance data files in the ARM Archive have passed a quality checking process and are believed to present good and reliable data. The quality of that data is further qualified by the `set_quality` value in each file. The determination of that value is discussed in detail above in [Quality Checking; Algorithm; Radiance Data File; Chapter 4: Radiance Product](#). It should be noted that the data is considerably more reliable than `set_quality` would often indicate. If an analyst needs to understand the reason for a low value he should read the comments in the processing log file. These logs are discussed in the [Log Files; Chapter 8: Utility Products](#).

As with any radiance or flux instrument, the accuracy of the current calibration is always an open question. Since Whole Sky Imagers were all initially calibrated when constructed the initial field data for each instrument should be good. Of course, calibrations drift with time, so later data may have slightly larger calibration errors. The usage of the field calibration device starting in 2003 has greatly improved this situation. All calibrations should now be no more than one year old. The analyst should refer to the [Instrument Calibration; Chapter 1: Overview](#) for a detailed discussion.

An analyst should use care in using some of the supporting variables in the radiance NetCDF file. For example `arc_pos` and `trolley_pos` may have errors as large as a few angular degrees each prior to the introduction of the upgraded instrument at the Southern Great Plains site in the fall of 2003. The variable list in [Variables, Dimensions, Variables, and Coding; Radiance Data File; Chapter 4: Radiance Product](#) discusses those dependent on operator input, such as `hardware_version` and `camera_offset`. Their values are only as good as those entered.

5. Sky Feature Product

There is a single detailed sky feature product, the NetCDF sky feature file. This file is summarized in sky feature summary file and the sky feature plot file as discussed in [Chapter 6: Summary Products](#).

5.1 Sky Feature File

5.1.1 Purpose

The primary Whole Sky Imager sky feature product in the ARM Archive is the NetCDF file that contains feature retrieval interpretations with a daytime 30–35 microsteradian spatial resolution for the entire 2π zenith hemisphere. At nighttime, although the data are still presented at this high resolution, in practice the resolution is limited by the existence of stars (see [Nighttime Optical Density Retrieval; Algorithm; Sky Feature File; Chapter 5: Sky Feature Product](#)). There is one such file for each six-minute collection (ten-minute old style collection, see [Data Collection Strategy; Chapter 1: Overview](#)). All such files for one 24-hour Greenwich day are packaged into a UNIX style tar archive that is quite large.

At the elementary level, sky feature retrievals are based on color – clouds are white and clear sky is blue. Thus a simple ratio of radiances in the blue spectral band (for the Whole Sky Imager, the 450 nm band) and any longer wavelength band, perhaps red (in this case 650 nm or even 800 nm) is sufficient to detect clouds on a pixel-by-pixel basis. This works quite well for optically thick clouds when the sun zenith angle is less than $\sim 80^\circ$ and has the advantage of being calibration independent.

When using such data, one quickly perceives the need to progress to a more complex level that can retrieve clouds when they are not white and the sky isn't blue, i.e., when the sun is near or just below the horizon at sunrise and sunset, when it is night, when the sky is heavily loaded with aerosols and thus milky in color, or when the clouds are optically thin and thus have blue showing through.

The sky feature product discussed in this section is not complex enough to handle all these complicating cases. It does retrieve clouds for the elementary case using a modified ratio technique and at nighttime using the star field as a background. It does not currently handle times near sunrise and sunset, nor does it do a sufficiently exact retrieval in atmospheres with heavy aerosol loading or optically thin clouds. The means to retrieve sky feature in these circumstances are well known, but ARM has not yet invested the resource needed to encode the necessary algorithms for use in a production environment.

This sky feature file is useful for the intermediate level user, whose application can benefit from the $\sim 80\%$ diurnal coverage attained and can ameliorate the missing sunrise, sunset, aerosol, and certain thin cloud times. Since the sky feature identification is not just binary, but contains simple cloud brightness designations during the daytime and simple opacity designations during the nighttime the opportunity exists for more advanced usage than reduction to binary averages.

5.1.2 Naming Convention

The sky feature data file has the following name based on standard ARM naming practices.

`ssswsicloudff.c1.yyyymmdd.hhmmss.cdf`

where the blue characters are literal and:

`sssf` = the site and facility identifier as listed in Table 1 and
`yyymmdd.hhmmss` = the standard 15 character ARM date time stamp showing year, month, day, hour, minute, and second.

The UNIX style tar archives associated with these files use the same basic name but have “.tar” appended to the end.

5.1.3 Dimensions, Variables, and Coding

5.1.3.1 Global Attributes

The global attributes in the sky feature file contain useful background information on the instrument and production environment. The user should always extract them with a suitable NetCDF dump routine and become familiar with their content. As the names and values are purposely designed to be human readable, they are not listed separately herein.

5.1.3.2 Dimensions

The dimensions associated with the sky feature file are listed in Table 8 and discussed in the paragraphs below. This file is quite unusual in that the unlimited dimension is not associated with the time of the data sample. Recall that each file is for a specific collection time making time a constant.

The **class_count** dimension can take a value between one and two and is the number of different type of retrievals stored in a file. Typically this dimension has the value of one as either the daylight-twilight spectral retrieval is executed or the nighttime optical density retrieval but not both. At nighttime on the day of the full moon both may be executed.

The **image_cols** and **image_rows** dimensions always have the value of 512, and are the number of columns and rows in the sky feature arrays.

The **colors** dimension refers to the radiance data file image layers used in the retrieval. Since there are four possible radiance images (450 nm, 650 nm, 800 nm, null-filter) its value is 4.

The **areas** dimension allots storage space for each of the ten large sky areas used to collect summary averages of pixels.

The **types** dimension allots storage space for summary statistics for each of the ten cloud and sky types recognized by the sky feature retrieval.

The other dimensions (**char_5**, **char_20**, and **char_64**) are necessary in the development of the NetCDF file but are of no particular concern to its user.

Table 8. Sky feature file dimensions.

ID	Name	Size
0	class_count	Unlimited
1	image_cols	512
2	image_rows	512
3	colors	4
4	areas	10
5	types	10
6	char_5	5
7	char_20	20
8	char_64	64

5.1.3.3 Variables

The variables associated with the sky feature file are listed in Table 9 and discussed in the paragraphs below. The attributes associated with each variable contain valuable information; the user should extract them with a suitable NetCDF dump routine and become familiar with their content. In particular the attributes have range and dimension information.

The **base_time** variable – long name: base time in epoch –of type long is a scalar whose value is the number of seconds since January 1, 1970 at 0000 hours Greenwich Time to the data collection event.

The **time_offset** variable – long name: offset from base_time –of type double is a scalar whose value is always zero. It is included for conformity with other ARM NetCDF files whose data is time dependent and that therefore require a time offset value for each data point.

The **class_image** array – long name: classification image –of type float is the data of most interest to the user. It contains the sky feature data for each of the retrievals performed. The array has three dimensions with the first two being the columns and rows of the image respectively. The third dimension defines layers in the file, one layer for each of the retrievals. For the relationship between layers and retrievals refer to the variable class_type. Each byte has a value between 0 and 9 that encodes the results of retrieval for that pixel. Then encoding is detailed in Table 10.

The **class_type** vector – long name: classification procedure type – of type character identifies the sky feature retrieval type of each layer of the class_image array. The dimension of this vector corresponds one-to-one with the third dimension of that array. For example, if the first element (index [0]) in class_type contains the character string “density”, then the layer class_image [i, j, 0] contains feature data for an optical density retrieval, where i and j range from 0 to 511. Each element in class_type may have a value from the set {“2-color”, “3-color”, “density”}; no value may be repeated. If only one retrieval is performed, which is the usual case, then the second element in this vector has the null value of twenty ampersand characters.

Table 9. Sky feature file variables.

ID	Name	Type	Attribute Count	Dimension Count	Dimension IDs	Actual Dimensions
0	base_time	Long	5	0	--	--
1	time_offset	Double	5	0	--	--
2	class_image	Byte	6	3	1, 2, 0	512, 512, x
3	class_type	Char	5	2	7, 0	20, x
4	class_quality	Char	6	1	0	x
5	class_sums	Float	8	3	4, 5, 0	10, 10, x
6	base_color	Char	6	2	6, 3	5, 4
7	base_quality	Char	5	0	--	--
8	header_date_time	Char	4	1	7	20
9	File	Char	4	1	8	64
10	cal_file	Char	4	1	8	64
11	exposure	Float	7	1	3	4
12	neutral_filter	Short	7	1	3	4
13	Spectral_filter	Short	7	1	3	4
14	sun_calc_az	Float	6	0	--	--
15	sun_calc_zen	Float	6	0	--	--
16	moon_calc_az	Float	6	0	--	--
17	moon_calc_zen	Float	6	0	--	--
18	light_source	Char	5	1	6	5
19	Site	Char	5	1	6	5
20	Latitude	Float	6	0	--	--
21	longitude	Float	6	0	--	--
22	Alt	Float	6	0	--	--

Table 10. Sky feature classification encoding.

Value	Classification	Retrieval Type
0	Clear	Spectral or density
1	Aerosol	Spectral
2	Mixed aerosol and cloud	Spectral
3	Bright cloud or bright sky	Spectral or density
4	Intermediate cloud	Spectral
5	Dark cloud	Spectral
6	Translucent	Density
7	Opaque	Density
8	Indeterminate	Spectral or density
9	Undefined	Spectral or density

The **class_quality** vector – long name: quality control character for classification sets – is a type character that identifies the perceived quality of sky feature retrievals performed. The quality control process discussed in [Data Collection Strategy; Chapter 1: Overview](#) is independent of this vector’s value; therefore the mere existence of a sky feature data file in the ARM Archive is indication that the data control process found nothing objectionable with it regardless of the values in this vector. The dimension

of this vector corresponds one-to-one with the third dimension of the class_image array. For example, if the first element (index [0]) in class_quality contains the character “A”, then the layer class_image [i, j, 0] retrieval had that quality assessment, where i and j range from 0 to 511. This vector may have any value from the set {“A”, “B”, “C”, “D”, “F”} with the assigned meanings of: A = no problems detected, B = minor problems that do not affect classification validity, C = minor problems that could affect classification validity, D = classification suspect, and F = bad classification.

The **class_sums** array – long name: classification tallies by area – of type float contains percentages based on the class_image array that will be of interest to users preparing long term statistics. It contains percentages for each of the sky feature retrievals performed. The array has three dimensions. The first ranges from 0 to 9 and corresponds to different areas of the sky as detailed in Table 11. Note that the lower quadrants do not include the last 10° above the horizon; please refer to [Usage Hints; Sky Feature File; Chapter 5: Sky Feature Product](#) for very important information concerning the use of the data stored in this array. The second dimension also ranges between 0 and 9 but corresponds to the classification type as found in the “Value” column of Table 10. The third dimension defines layers in the file, one layer for each of the retrievals. For the relationship between layers and retrievals refer to the variable class_type. Each element has a value between 0.0 and 100.0 that is the percentage of pixels in the region that corresponds to the particular classification type. For example, the value of class_sums [3, 8, 0] is the percentage of pixels in the upper east quadrant (region 3) that have the value of “indeterminate” (classification 8) that are from the retrieval which is defined by class_type [0]. The percentages in any region add to 100.0; thus the sum from j = 0 to 9 of class_sums [a, j, c] is 100.0 where a and c are any allowable fixed index value.

Table 11. Sky feature summation sky regions.			
Index	Region	Zenith Angles	Azimuth Angles
0	Whole sky	0° – 90°	0° – 360°
1	Upper disk	0° – 10°	0° – 360°
2	Upper north quadrant	0° – 45°	315° – 360° and 0° – 45°
3	Upper east quadrant	0° – 45°	45° – 135°
4	Upper south quadrant	0° – 45°	135° – 225°
5	Upper west quadrant	0° – 45°	225° – 315°
6	Lower north quadrant	45° – 80°	315° – 360° and 0° – 45°
7	Lower east quadrant	45° – 80°	45° – 135°
8	Lower south quadrant	45° – 80°	135° – 225°
9	Lower west quadrant	45° – 80°	225° – 315°

The **base_color** vector – long name: input image color – of type character identifies the calibrated spectral radiance bands available to the sky feature retrieval. Each element may have a value from the set {“clear”, “blue”, “red”, “nir”, “null”} which correspond to the spectral bands {null-filter, 450 nm, 650 nm, 800 nm, and no-radiance-data} respectively. No value may be repeated except for “null”. The availability of radiance retrieval band does not imply that it was used in the retrieval; to determine usage refer to the vector class_type.

The **base_quality** variable – long name: quality control character for input image set – is a scalar of type character that identifies the perceived quality of the entire calibrated image set in the calibrated radiance data file that underlies this sky feature retrieval. It is a pass through of the variable set_quality

from that NetCDF file; please refer to [Variables, Dimensions, Variable, and Coding; Chapter 4: Radiance Product](#) for a complete description of set_quality. This variable may have a value from the set {"A", "B", "C", "D", "F"} with the assigned meanings of: A = no problems detected, B = minor problems that do not affect calibrated image validity, C = minor problems that could affect calibrated image validity, D = image calibration suspect, and F = bad calibrated images.

The **header_date_time** variable – long name: input image time stamp – of type character with the format: yyyy:mm:dd:hh:mm:ss where each letter stands for a numeral representing (in order) year : month : day : hour : minute : second. The value is a pass through of a variable by the same name from the calibrated radiance file that underlies this sky feature retrieval and is the common value found in the header of the raw data files upon which that calibrated radiance file was built; see [Header Information; Dimensions, Variables, and Coding; Raw Data Files; Chapter 2: Raw Products](#). This date-time group is the most accurate available; if the one imbedded in the ARM style file name is in disagreement then header_date_time should be trusted.

The **file** variable – long name: calibrated image file name – of type character is a scalar containing the name of the calibrated radiance file that underlies this sky feature retrieval. The name is stored without any directory information.

The **cal_file** variable – long name: calibration file name – of type character is a scalar containing the name of the calibration file used to retrieve this sky feature file. It is stored without any directory information.

The **exposure** vector – long name: exposure – of type float contains the camera exposure times in units of milliseconds for the data in the calibrated radiance file that underlies this sky feature retrieval. The dimension of this vector corresponds one-to-one with the dimension of the base_color vector, which should be used as a key to interpreting the values in this vector and their relationship to data in the radiance data file and this file. Exposures range from one millisecond to two minutes.

The **neutral_filter** vector – long name: neutral density filter – of type short integer chronicles the neutral density filter used in collecting the data in the calibrated radiance file that underlies this sky feature retrieval. Neutral_filter may have a value from the set {0, 1, 2, 3, 4} where 0 = null value and the other numbers indicate approximate filter transmission thus: 1 = 1.0, 2 = 0.01, 3 = 0.001, and 4 = 1.0. The same value will often be repeated for all images in a set. The dimension of this vector corresponds one-to-one with the dimension of the base_color vector, which should be used as a key to interpreting the values in this vector and their relationship to data in the radiance data file and this file.

The **spectral_filter** vector – long name: spectral filter – of type short integer chronicles the spectral filter used in collecting the data in the calibrated radiance file that underlies this sky feature retrieval. Spectral_filter may have a value from the set {0, 1, 2, 3, 4} where 0 = null value and the other numbers indicate the approximate filter center wavelength and bandpass thus: 1 = 800 nm and 70 nm FWHM, 2 = null filter, 3 = 650 nm and 70 nm FWHM, and 4 = 450 nm and 70 nm FWHM. The same value should not be repeated within this vector except for 0. The dimension of this vector corresponds one-to-one with the dimension of the base_color vector, which should be used as a key to interpreting the values in this vector and their relationship to data in the radiance data file and this file.

The two scalars **sun_calc_az** and **sun_calc_zen** – long names: sun calculated azimuth and sun calculated zenith – of type float contain respectively the calculated azimuth and zenith angles for the sun’s position. They are based on the vectors `sun_az` and `sun_zen` in the calibrated radiance file that underlies this sky feature retrieval. See their description in [Variables; Dimensions, Variables, and Coding; Radiance Data File; Chapter 4: Radiance Product](#) for a discussion of the assumptions in their calculation. `sun_az` and `sun_zen` are vectors with values for each of the raw data images that underlie the calibrated radiance file; the values are very close as the apparent sun position does not change much in the tens of seconds needed for the Whole Sky Image to collect a complete image set. The scalars `sun_calc_az` and `sun_calc_zen` are the averages of `sun_az` and `sun_zen` respectively. The values are in angular degrees. `sun_calc_az` has the range 0° to 360° and is measured clockwise from geographic north such that east is 90°. `sun_calc_zen` has the range 0° to 180° and is measured from the local vertical such that the horizon is 90° and any value greater than 90° indicates that it is nighttime.

The two scalars **moon_calc_az** and **moon_calc_zen** – long names: moon calculated azimuth and moon calculated zenith – of type float contain respectively the calculated azimuth and zenith angles for the moon’s position. They are based on the vectors `moon_az` and `moon_zen` in the calibrated radiance file that underlies this sky feature retrieval. See their description in [Variables; Dimensions, Variables, and Coding; Radiance Data File; Chapter 4: Radiance Product](#) for a discussion of the assumptions in their calculation. `moon_az` and `moon_zen` are vectors with values for each of the raw data images that underlie the calibrated radiance file. The scalars `moon_calc_az` and `moon_calc_zen` are the averages of the vectors `moon_az` and `moon_zen` respectively, for the same reason that `sun_az` and `sun_zen` are averaged in the paragraph above. The values are in angular degrees and have the same range and meaning as for `sun_calc_az` and `sun_calc_zen`.

The **light_source** variable – long name: light source – of type character is a scalar that specifies the dominant light source in the sky at the time of this sky feature file, e.g., the sun. The value is a pass through of a variable by the same name from the calibrated radiance file that underlies this feature retrieval. `light_source` may have a value from the set {"@@@@", "sun", "moon", "none"} where @@@@ is the null value and the rest are self explanatory.

The **site** variable – long name: site location – of type character is a scalar that identifies the site location of the instrument involved in the data collection. Its value is one of the set {"sheba", "nsa-1", "nsa-2", "twp-1", "twp-2", "twp-3", "sgp-1", "sgp-2", "other"}. This set corresponds to the designators in Table 1 as follows {shbC1, nsaC1, nsaC2, twpC1, twpC2, twpC3, sgpC1, sgpL3, (none)} respectively. The value is a pass through of a variable by the same name from the calibrated radiance file that underlies this sky feature retrieval. Refer to [Variables; Dimensions, Variables, and Coding; Radiance Data File; Chapter 4: Radiance Product](#) for a complete description of the accuracy associated with this variable and its use.

The **latitude** and **longitude** variables – long names: latitude and longitude – of type float are scalars that specify the geographical coordinates of the instrument’s location. The units for both variables are decimal degrees. Latitude ranges between -90° and +90°, where -90° is at the Earth’s South Pole, 0° is at the equator, and 90° is at the North Pole. Longitude ranges between -180° and +180°, where 0° is the meridian through Greenwich, England, positive values are in the eastern hemisphere, and negative values are in the western hemisphere. These values are a pass through of variables by the same name from the

calibrated radiance file that underlies this file. Refer to [Variables; Dimensions, Variables, and Coding; Radiance Data File; Chapter 4: Radiance Product](#) for a complete description of these variables' derivation.

The **alt** variable – long name: altitude – of type float is a scalar that specifies the geographical altitude of the instrument's location. It is expressed in meters above mean sea level. The value is a pass through of a variable by the same name from the calibrated radiance file that underlies this feature retrieval. Refer to [Variables; Dimensions, Variables, and Coding; Radiance Data File; Chapter 4: Radiance Product](#) for a complete description of this variable's derivation.

5.1.4 Algorithm

5.1.4.1 Introduction

The algorithms that implement the sky feature file retrievals are encoded in an IDL program named WsiIdpc5_vx, where x represents a version number. The version extant is 9 as of this writing in September 2003. The program and its subroutines were written by the instrument mentor and are maintained by him as well. The original work supporting the daytime spectral algorithm was performed by David Soble of Mission Research Corporation, and that supporting the nighttime optical density algorithm by Sean Moore, also of Mission Research. The major subroutines are CloudMap_vx, DayFrac_vx, DefineOrthArea_vx, LoadSpecialColors_vx, NightFrac_vx, Summations_vx, TrueRad3p_vx, WriteJpg_vx, WriteNetCDF_vx, WsiCrackIdpc1, and WsiCrackIdpc4. The program is executed at the ARM Archive as part of the Whole Sky Imager data ingest and retrieval process. The WsiIdpc5_vx tree is coded for execution in a production environment, meaning that it can handle nearly any anomaly in input data and continue processing to a benign conclusion.

5.1.4.2 Initialization

Initial blocks in the program WsiIdpc5_vx open a log file (see [Log Files; Chapter 8: Utility Products](#)), identify the proper calibrated radiance file to use for the time of the sky feature file to be generated, and access the correct calibration file (see [Chapter 7: Calibration Product](#)). Many checks are performed on these files to insure proper size and content. The radiance and calibration files are next read for pertinent data, and closed by the subroutines WsiCrackIdpc4 and WsiCrackIdpc1 respectively. The data is stored in structures accessible to the entire WsiIdpc5 processing tree.

5.1.4.3 Sky Regions

To prepare for later summation of the sky cover findings, the subroutine DefineOrthArea_vx is invoked to divide a 512×512 array matched to the columns and rows of the radiance arrays and cloud map array into regions that correspond to those listed in Table 11. To accomplish this, the routine uses data from the calibration file that relate columns and rows indices to sky azimuth and zenith angles, specifically the variables pixel_to_azimuth and pixel_to_zenith (see [Variables; Dimensions, Variable, and Coding; Chapter 7: Calibration Product](#)).

5.1.4.4 Determine Retrieval Cases

There are four different retrieval conditions, each with their own algorithm and hence their own subroutine. These are:

- Daytime spectral – at least 450 nm and 650 nm radiance arrays are extant, and the sun zenith angle is less than 89.7° ,
- Twilight spectral – at least 450 nm and 650 nm radiance arrays are extant, and the sun zenith angle is greater than 89.7° but less than 95.0° ,
- Nighttime spectral – at least 450 nm and 650 nm radiance arrays are extant, and the sun zenith angle is greater than 95.0° , and
- Nighttime optical density retrieval – a null-filter radiance array is extant, and the sun zenith angle is greater than 95.0° .

In the next stage of the retrieval, the proper case for the retrieval is identified and the corresponding code called. Each of the four cases is described in a separate subsection below.

5.1.4.5 Daytime Spectral Retrieval

The daytime spectral retrieval in the subroutine `DayFrac_vx` first aborts gracefully if the `set_quality` of the underlying calibrated radiance file isn't at least D. Then it determines whether the available calibrated radiance data supports a two-color (450 nm and 650 nm radiances) or three-color (450 nm, 650 nm, and 800 nm radiances).

The algorithm will eventually compare the observed radiances with calculated pristine sky radiances, and then analyze the differences to determine sky condition. To prepare for this, `TrueRad3p_vx` is executed to construct arrays for each relevant spectral band. `TrueRad3p_vx` begins with pristine sky radiances found in the `rad_table` array from the calibration file (see [Variables; Dimensions, Variables, and Coding; Chapter 7: Calibration Product](#)). `rad_table` contains radiances for a sparse set of solar zenith angles, viewing zenith angles, and viewing azimuth angles. The subroutine then rotates the `rad_table` scene to agree in solar azimuth angle with that of the radiance file data, then interpolates to populate a 512×512 array that matches the viewing geometry of the radiance data using the `pixel_to_azimuth` and `pixel_to_zenith` arrays from the calibration file. The three interpolations (solar zenith, viewing azimuth, and viewing zenith angles) are performed using a three point Lagrange technique, which is quite computationally intensive. To avoid excessive execution times `TrueRad3p_vx` has been highly optimized and the coding is therefore a bit difficult to decipher.

In the original work, David Sowle noticed that the `rad_table` radiances (derived from the SBDART model; see [Pristine Sky Radiance Variables; Algorithm; Chapter 7: Calibration Product](#)) did not match those measured by the Whole Sky Imager for exceptionally clear sky scenes. The differences vary with the solar and viewing angles involved, but have a large constant coefficient, different for each spectral band. A quite unsatisfying solution was used to resolve this during algorithm development. That was to multiply the calibrated radiance file data by an empirical constant before comparison with the interpolated

rad_table numbers. The values are listed in Table 12; they are considerable and inconsistent across the bands. Unfortunately, this ad hoc fix has persisted through the current version of the DayFrac_vx and is in use today. Worse, the constants are fixed in the code and not adjustable through the calibration file, a flaw which ought to be fixed with the next version. The new calibrations performed with the field calibration device discussed in [Field Calibration; Instrument Calibration; Chapter 1: Overview](#) will have an impact on this situation.

Spectral Band	Radiance Multiplier
450 nm	1.24
650 nm	0.78
800 nm	0.52
Null-Filter	Not applicable

The next step is the calculation of the ratio of the observed radiances to the modeled pristine sky radiances on a pixel-by-pixel basis and for each spectral band. Of course this ratio is multiplied by the coefficients described in the above paragraph so that values for all viewing angles into an observed clear sky are near 1.0.

The following step is the most difficult; the reinterpretation of the observed-to-pristine ratios just calculated in terms of sky cover condition. First, the generic process will be discussed, then the details of the algorithm. In theory, one could determine sky condition from the ratios for just one spectral band. These vary in practice between 0.2 and 10 with numbers outside this range not uncommon. If the ratio is close to one, then one has clear sky, brighter or darker would indicate clouds. This does not work in practice because of aerosols which tend to brighten clear skies, and would therefore be identified as clouds. By comparing the ratios in two spectral bands the situation is somewhat improved, and in fact the further apart in wavelength the bands are the better the situation as long as absorption is not an issue. The cloud particles are much larger than the wavelengths involved and scatter comparatively more at longer wavelengths than do the much smaller aerosol particles, whose sizes are comparable to the wavelengths. In Figure 4 all pixels from a radiance file are represented on a graph whose abscissa is the ratio of 450 nm observed-to-pristine and whose ordinate is the ratio of 800 nm observed-to-pristine. Each pixel is represented by a dot. Since there are nearly 200,000 pixels, the dots tend to be very dense around some common values; therefore the graph is actually a plot of the density of pixel dots. The color spectrum is black – green – red – purple – white from low to high density respectively. If the underlying scene were of a clear aerosol free sky, then there would be a dense accumulation around the (1.0, 1.0) point, and few elsewhere. The same scene with light aerosol would have comet-like figure with the head around the (1.0, 1.0) point and the tail stretching in some direction toward (4.0, 6.0) depending on the type of aerosol. Scattered cumulus clouds in a clear sky with light aerosols would have the aforementioned “comet” and its twin above and to the left. The two would be nicely separated by a very low density strip. That situation is the basis for the usual red-blue ration cloud determination used in simpler algorithms. The upper “comet” is cloud and the lower clear. The underlying scene in Figure 4 is one of the most difficult to interpret – mostly clear with patches of thin cirrus and a moderate-to-heavy aerosol layer. The pixel dots in the red streak from (1.5, 1.5) and stretching in the direction of (3.5, 6.0) clearly represent clear aerosol portions of the sky; they are the lower “comet” just discussed. Dots around (0.75, 2.0) and (1.5, 4.0) would be part of the upper cloud “comet”, but it doesn’t form. Instead there is a blur of dots

stretching downward and leftward to the aerosol “comet”. These represent portions of the cirrus that become increasingly thin and that eventually thin to being no different than the aerosol sky background. Dividing the groupings of dots in this case is difficult, and as much a matter of interpretation as anything else. Thus when regions are established for the different sky categories, they can be subjected to adjustment or tuning; unfortunately there is not an ideal tuning as that which would optimize cirrus retrieval in one circumstance would identify a heavier aerosol in another scene as cloud.

The algorithm step following the calculation of observed-to-pristine sky ratios is the reinterpretation of these into sky condition following the above generic process. The routine determines whether a two-color or three-color is needed and processes accordingly. Since all Whole Sky Imagers now deployed have all three spectral filters, two-color retrievals are performed only on historical data or data from an instrument with a very peculiar filter wheel anomaly.

The two-color interpretation is based on a graph very similar to Figure 4 except that the ordinate is the ratio of 650 nm observed-to-pristine. The graph’s area is divided into regions, each with an assigned a sky feature. The algorithm scans the sky field pixel by pixel, gets the values of the observed-to-pristine ratios in both spectral bands from the previous calculation, and determines the region for those two values, and assigns the appropriate sky feature to that pixel. The regions used are quadrilaterals; Table 13 lists the corners for each (lower left, upper left, upper right, and lower right). The aerosol region does not include the overlapping clear region. As coded any pixel lying in both the aerosol region and any of the cloud regions is considered mixed; however, as currently defined these regions do not overlap and therefore there is no mixed region. Any pixel whose mask is set (see the mask variable discussion in [Variables; Dimensions, Variables, and Coding; Radiance Data File; Chapter 4: Radiance Product](#)) is considered undefined. Indeterminate is anything that does not lie in one of the other regions.

The three-color interpretation ideally would be based on a three dimensional plot of pixel ratio values, one dimension for each spectral band, 450 nm, 650 nm, and 800 nm. The comet-like areas discussed earlier in this subsection would also be three dimensional, and not symmetric around their axis. Sky feature regions could be defined in such a space, but given the current lack of an ideal adjustment for such regions it was not considered practical. Instead the regional definition is performed for two two-dimensional projections of this region, one similar to Figure 4 with 450 nm ratio and 800 nm ratio axes and the second with 450 nm and 650 nm axes. To be considered a certain feature, the pixel must lie in the same sky feature region on both projections; if it lies in two different regions it is indeterminate. The regions used are quadrilaterals; Table 14 and Table 15 list the corners for each (lower left, upper left, upper right, and lower right). As can be seen from Table 14 the 650 nm versus 450 nm projection is currently being used as little more than a screen. Any pixel with a observed-to-pristine ratio value below the 0.20 (450 nm) or 0.25 (650 nm) will be considered indeterminate; all other pixels will pass in all other sky feature categories leaving the actual category determination to the 800 nm versus 450 nm projection. The sky feature regions for that projection are defined in Table 15 and are somewhat similar to the two-color regions discussed above. The aerosol region does not include the overlapping clear region. As coded any pixel lying in both the aerosol region and any of the cloud regions is considered mixed; however, as currently defined these regions do not overlap and therefore there is no mixed region. Any pixel whose mask is set (see the mask variable discussion in [Variables; Dimensions, Variables, and Coding; Radiance Data File; Chapter 4: Radiance Product](#)) is considered undefined. Indeterminate is any pixel that does not lie in one of the other regions or, as mentioned previously, any pixel that does not lie in the same sky feature region in both projections.

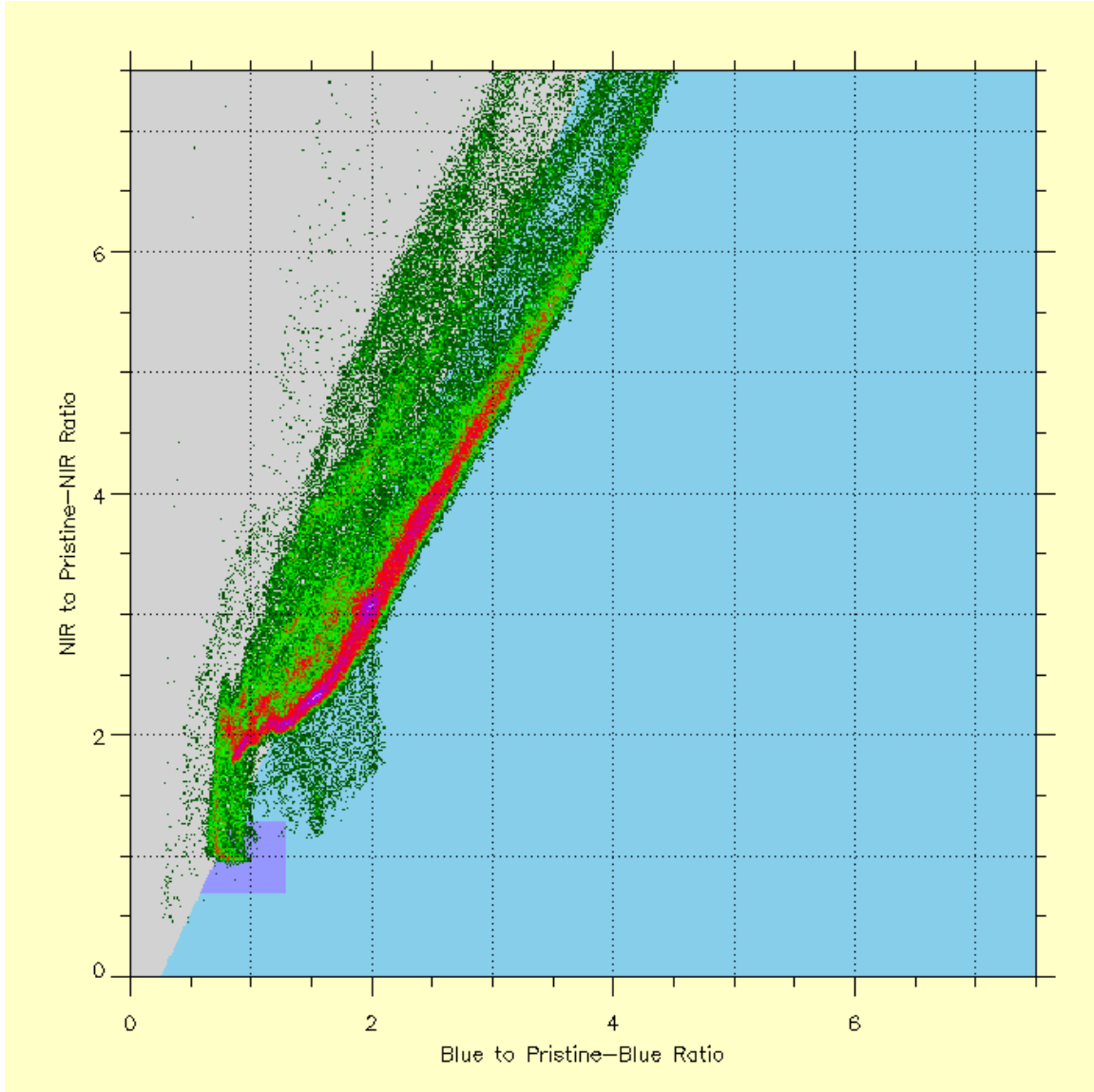


Figure 4. Density of radiance ratio values.

Table 13. Two-color sky interpretation graphical regions.					
Sky Feature	LL Corner	UL Corner	UR Corner	LR Corner	Exclusion Zone
Clear	(0.7, 0.7)	(0.89, 1.38)	(1.3, 1.3)	(1.3, 0.7)	None
Aerosol	(-10.0, -19.4)	(52.8, 100.0)	(100.0, 100.0)	(100.0, -19.4)	Clear region
Mixed	None	None	None	None	None
Dark Cloud	(-10.0, -19.4)	(-10.0, 100.0)	(1.0, 100.0)	(1.0, 1.5)	None
Intermediate Cloud	(1.0, 1.5)	(1.0, 100.0)	(3.0, 100.0)	(3.0, 5.3)	None
Bright Cloud	(3.0, 5.3)	(3.0, 100.0)	(52.8, 100.0)	(3.0, 5.3)	None
Indeterminate	(-∞, -∞)	(-∞, ∞)	(∞, ∞)	(-∞, ∞)	All other regions

Once all pixels in the 512×512 image have an assigned sky feature either by the two- or three-color method, the daylight spectral retrieval is complete.

Table 14. Three-color sky interpretation graphical regions – 800 nm vs. 450 nm.

Sky Feature	LL Corner	UL Corner	UR Corner	LR Corner	Exclusion Zone
Clear	(0.2, 0.25)	(0.2, 100.0)	(100.0, 100.0)	(100.0, 0.25)	None
Aerosol	(0.2, 0.25)	(0.2, 100.0)	(100.0, 100.0)	(100.0, 0.25)	None
Mixed	None	None	None	None	None
Dark Cloud	(0.2, 0.25)	(0.2, 100.0)	(100.0, 100.0)	(100.0, 0.25)	None
Intermediate Cloud	(0.2, 0.25)	(0.2, 100.0)	(100.0, 100.0)	(100.0, 0.25)	None
Bright Cloud	(0.2, 0.25)	(0.2, 100.0)	(100.0, 100.0)	(100.0, 0.25)	None
Indeterminate	$(-\infty, -\infty)$	$(-\infty, \infty)$	(∞, ∞)	$(-\infty, \infty)$	All other regions

Table 15. Three-color sky interpretation graphical regions – 800 nm vs. 450 nm.

Sky Feature	LL Corner	UL Corner	UR Corner	LR Corner	Exclusion Zone
Clear	(0.58, 0.7)	(0.86, 1.3)	(1.3, 1.3)	(1.3, 0.7)	None
Aerosol	(0.25, 0.0)	(47.62, 100.0)	(100.0, 100.0)	(100.0, 0.0)	Clear region
Mixed	None	None	None	None	None
Dark Cloud	(0.0, -0.53)	(0.0, 100.0)	(1.0, 100.0)	(1.0, 1.58)	None
Intermediate Cloud	(1.0, 1.58)	(1.0, 100.0)	(3.0, 100.0)	(3.0, 5.78)	None
Bright Cloud	(3.0, 5.78)	(3.0, 100.0)	(47.6, 100.0)	(3.0, 5.78)	None
Indeterminate	$(-\infty, -\infty)$	$(-\infty, \infty)$	(∞, ∞)	$(-\infty, \infty)$	All other regions

5.1.4.6 Twilight Spectral Retrieval

The twilight spectral retrieval is currently not implemented; when the sun zenith angle is between 89.5° and 95.0° no sky feature retrieval is performed. All the spectral algorithms use a reference clear sky as a base for feature calculations. The current reference sky was developed from the SBDART radiance model which assumes a flat earth. This assumption is adequate for low and moderate solar zenith angles, but at zenith angles associated with the twilight spectral retrieval curved earth effects have significant impact on sky radiance. This algorithm would be trivial to implement given the daylight spectral retrieval code, if the current reference radiances were upgraded for these high sun angles. The needed references could be calculated from a curved earth model or developed empirically.

5.1.4.7 Nighttime Spectral Retrieval

The nighttime spectral retrieval is currently not implemented; when the sun zenith angle is greater than 95° no spectral cloud retrieval is performed, just the nighttime optical density one discussed in the next subsection. There is some historical nighttime data in the archive when only spectral data was acquired during times of bright moonlight. All current instrument configurations however always collect null-filter nighttime data and spectral data only on the night of the full moon, as discussed in [Data Collection Strategy; Chapter 1: Overview](#). Therefore even though this retrieval is not implemented, the investigator does have the optical density retrieval as a source of sky condition information. The real use of the nighttime spectral retrieval would be as a comparison to the optical density retrieval.

The nighttime spectral retrieval will not be as easy to implement as the twilight spectral retrieval. Although it would seem that the reference clear sky needed would be easy to establish – basically zero radiance – such is not the case; there is a significant astronomical background especially in the region of the Milky Way. Also the algorithm must blank regions around each star of magnitude greater than 5.0 or so. Since an opaque unlit cloud would appear black like a pristine clear sky, some means of distinguishing these two situations must be implemented. Finally, the algorithm must account for reflected anthropogenic radiation. Developing and tuning this algorithm would require considerable effort, and without a driving impetus will not be undertaken in the near future.

5.1.4.8 Nighttime Optical Density Retrieval

The nighttime optical density retrieval in the subroutine NightFrac_vx first aborts gracefully if the set_quality of the underlying calibrated radiance file isn't at least D. Then it determines whether the available calibrated radiance data supports optical density retrieval, i.e., has a null-filter radiance array.

The algorithm will eventually compare the observed brightness of stars with their top-of-the-atmosphere brightness calculated from astronomical tables, and then analyze the differences to determine sky condition. To prepare for this, NightFrac_vx accesses the list of all stars with at least the magnitude of 5.0 and their names, magnitudes, right ascensions, and declinations from the relevant calibration file, see [Dimensions, Variables, and Coding, Calibration File, Chapter 7: Calibration Product](#). Then it calculates the column and row in the radiance array where each star should appear by converting the Greenwich Mean Time stamp of the radiance file to Local Sidereal Time, using that to convert each star's right ascension and declination to local zenith and azimuth angles, and then using those and the angle-to-pixel conversions in the calibration file to determine column and row.

Next the algorithm thins the list by selecting for further processing only those stars suitable for the sky feature retrieval by executing the following list.

1. First, eliminate all stars whose magnitude is greater than the cutoff magnitude plus one. That magnitude is fixed in the code, but its value is somewhat arbitrary. A large value would include more stars and hence improve the spatial resolution of the retrieval, but needs high quality radiance data, both in terms of raw data and pixel-to-pixel calibration (see [Calibration Factors; Instrument Calibration; Chapter 1: Overview](#)). A low value would cause only a few stars to be selected and would result in poor spatial resolution for the sky feature map. The current value, as shown in Table 16, leaves 492 stars for consideration (at the cutoff magnitude) in the complete celestial sphere. The magnitude is increased by one in this step to include stars necessary for the third step below; the extra dim stars will be eliminated in step five.
2. Second, all stars with a zenith angle greater than 90° are eliminated, as they are below the horizon.
3. Third, the situation is considered where two stars are close together. The algorithm will eventually consider a box of columns and rows, whose size is as shown in Table 16, around the star's calculated position as its potential locus and will seek to discover a single dominate intensity peak within that box. The box needs to be wide enough to account for Whole Sky Imager positioning error, but not so wide as to unnecessarily encompass multiple stars of the requisite brightness. If two stars of similar magnitude (absolute difference less than 1) are within the box, then the peak finding algorithm will fail; in this case both stars are eliminated from further consideration. If one's

brightness dominates the other, then that star is kept and the dimmer star is eliminated from consideration.

4. Any stars behind the mask from the radiance file (see [Variables; Dimensions, Variables, and Coding; Radiance Data File; Chapter 4: Radiance Product](#)) are eliminated as they will not be observed.
5. Any stars with a magnitude greater than the dim star cutoff (see Table 16) are eliminated. This completes the brightness screen started in step one above.
6. Finally, any stars too near the moon are eliminated. For this purpose, it is any star within a box is drawn around the moon of a size as specified in Table 16.

When the down selection process is finished, on the order of 100 stars will remain in the list for use in the retrieval. Their magnitude is used to calculate the flux expected for each star at the top-of-the-atmosphere in units of milliwatts per meter squared per nanometer (wavelength). This is assumed to be the same as the flux at the instrument’s location on a pristine clear night, to within all other system errors.

The next task is to calculate the flux measured by the Whole Sky Imager for each of the selected stars. I. C. Musat, student of Dr. Robert Ellingson at the University of Maryland, developed a quite sophisticated algorithm for doing this task. The algorithm implemented is simpler and not as accurate; it should be replaced with Musat’s. A constant, *star_width*, is included in the calibration file that specifies the apparent width of stars as viewed by the Whole Sky Imager (see [Dimensions, Variables, and Coding; Calibration File; Chapter 7: Calibration Product](#)). The stars would appear as points, illuminating a single pixel, were it not for the point spread function of the optics train which causes some of the light from the star to illuminate neighboring pixels. This function varies from instrument to instrument. For the purposes of this retrieval it is assumed that the star’s radiance has the form of $e^{-R^2/2*W^2}$ where *R* is the distance measured in pixels from the star image’s center and *W* is the star width constant from the calibration file. Thus the star’s flux is to be calculated by integrating the area under this curve discounting any background radiance. The unknowns are the amplitude of the exponential at the star image’s center and the constant amplitude of the background.

Table 16. NightFrac_vx hardwired constants.

Fixed Constant	Value
Magnitude of dimmest star utilized	4.00
Search box for star	9 pixels full width
Exclusion box for moon	81 pixels full width
Upper brightness limit for acceptability	$2.0 \times$ star chart brightness
Upper brightness limit for translucent	$0.7 \times$ star chart brightness
Upper brightness limit for opaque	$0.1 \times$ star chart brightness

An estimate of the background is obtained by averaging the radiance values around the edge of the star box after discounting the four highest and four lowest values as potential outliers. One hundred twenty-one trials of a least squares fit is next applied using the exponential peak to the radiance data less the background. One trial is for each potential star image center varying by 0.1 pixel in the column

direction and then the same in the row direction starting 0.5 below and extending 0.5 above in column and row. The greatest value found is assumed to correspond to the correct center location and hence be the correct amplitude. Using that value, the integral is performed and the flux determined. This method is considerably more accurate than simply adding the radiance values less the background in the pixels affected by the star's illumination as the latter will result in a flux value dependent on whether the star image's center lies directly on a pixel or somewhere in between two pixels.

Now, finally, the sky classification can commence at each star's location using its observed flux and its calculated pristine sky flux. The classification codes are listed in Table 10. The following sequence is used.

1. First, if the background flux is more than three times that expected from the star, then the classification is "bright". This situation can occur with bright moonlit cirrus, bright moonlit heavy aerosol, and moonlit or artificially lit ground fog.
2. Next, if the star is within 5° of the horizon and the star image's peak location is on the edge of the star box, the classification is "indeterminate". The current algorithm is insensitive to air mass, and this condition arises most often in moderate to heavy aerosol conditions where the horizon, lit either by the moon or from artificial sources, brightens dramatically with increasing zenith angle.
3. Third, if the star image's peak location is on the edge of the star box, the classification is "opaque". In this case, either the star box is improperly located and does not include the star (unlikely with a good position calibration), the background intensity slopes dramatically across the box and has dominated the process of finding the star center (unlikely well above the horizon which was addressed in case two just above), or the star is obscured by cloud and the brightest point left in the box was near the edge (most likely).
4. If the star image's peak is not greater than two-thirds the difference between the highest and lowest radiance around the star box's edge, then the classification is "opaque". Most likely the star is obscured and the peak found just corresponds to variance in the background, i.e., radiance from the cloud.
5. If a star is found, but its observed flux is less than a constant times its pristine sky flux, then it is considered obscured and the classification is "opaque". The constant currently used is shown in Table 16. Its value was selected to give reasonable results for a number of test cases, but could be subjected to further refinement.
6. If a star is within 15° of the horizon and its flux is greater than the opaque cutoff of step five, but less than 80% of the product of the upper limit for translucent features and its pristine sky flux, then the classification is "translucent". The 80% figure is a rough compensation for increased air mass near the horizon. Future versions of this algorithm could improve this retrieval by incorporating a more rigorous treatment. The translucent upper limit currently used is shown in Table 16. Its value was selected to give reasonable results for a number of test cases, but could be subjected to further refinement.
7. If a star is more than 15° above the horizon and its flux is greater than the opaque cutoff of step five, but less than the product of the upper limit for translucent features and its pristine sky flux, then the classification is "translucent". As mentioned in step six, the translucent upper limit currently used is shown in Table 16 with the same caveat.

8. If the star's observed flux is considerably greater than the calculated pristine sky flux, then some anomaly has occurred in the flux integration process, such as the selection of an aircraft light in lieu of the star. Any flux above the acceptability limit shown in Table 16 will cause the classification to be fixed as "indeterminate".
9. Finally, any situation not considered in the above circumstances results in a classification of "clear".

At this point, approximately 100 pixels – those at the calculated star positions – have been assigned sky feature codes. To assign codes to the rest a simple nearest neighbor procedure is implemented in pixel space. The image is scanned pixel-by-pixel and if the pixel is not behind the radiance file mask (see [Variables; Dimension, Variables, and Coding; Radiance Data File; Chapter 4: Radiance Product](#)), then the nearest star location pixel found, and the value associated with that star assigned. Points behind the mask are assigned the undefined feature. This results in a sky map that has a patchwork appearance; the size of an individual patch depends on the density of usable stars in the associated sky region.

5.1.4.9 Summations

The next task is the preparation of the summation statistics that will populate the three dimensional array `class_sums` in the resulting NetCDF file (see [Variables; Dimension, Variables, and Coding; Sky Feature File; Chapter 5: Sky Feature Product](#) for a detailed description of the data fields involved). This is a straight forward but tedious calculation of the necessary sky classification versus sky region percentages needed for either the spectral retrieval, optical density retrieval, or both. The sky regions were defined earlier in the processing as discussed in [Sky Regions; Algorithm; Sky Feature File; Chapter 5: Sky Feature Product](#).

5.1.4.10 Quality Checking

After the sky feature retrievals have been completed as described above, a quality figure is generated for the result, which will be the value for `base_quality` in the resultant NetCDF file. Sky feature retrieval quality, defined in terms of the values A, B, C, D, and F, is set initially to the value of `set_quality` from the radiance file (see [Variables; Dimensions, Variables, and Coding; Radiance Data File; Chapter 4: Radiance Product](#)), i.e., sky feature quality can be no greater than the underlying quality of the radiance data on which it is based. This figure is then reduced one value for each 5% of indeterminate in the retrieval, whether spectral or optical density to arrive at the final quality value. A large number of pixels assigned the indeterminate feature indicates the retrieval was being performed on data outside normal expectations.

5.1.4.11 Extra Information

Since the algorithms involve a number of tunable constants, considerable work has been undertaken to find the best values for these. Of course, there is considerable personal bias involved in the process and so it is best subjected to review. To facilitate in this `WsiIdpc5_vx` has an extensive imbedded graphics package to visualize all the intermediate steps in processing. The user may wish to consult these graphics on a case by case basis to understand a particular situation, develop his own "tuning", or to aid in a related study. These are not produced as a normal product, but require a special run of `WsiIdpc5_vx` at the ARM Archive for a specified radiance file; the graphics are invoked by the setting of an operating system environmental variable.

The graphs are produced as a set of PNG files that can be displayed in most common photo programs such as Adobe PhotoShop. The following are included in the set. Note that the file names are not keyed to the ARM standard naming structure but are generic in nature; therefore, the files would be overwritten if not renamed or moved between the special WsiIdpc5_vx runs. These may be changed to a standardized form in the next retrieval release. In the list below, the character ° represents a space in the file name.

1. The pristine sky radiances as interpolated by TrueRad3p_vx are pictured in the three files Blue°Pristine.png, Red°Pristine.png, and NIR°Pristine.png for the 450 nm, 650 nm, and 800 nm bands respectively. These are produced only if a daylight spectral retrieval is being performed; additionally, NIR°Pristine.png is produced only if a three-color retrieval is being performed. The pictures are scaled in the same manner as the sky picture file as discussed in [Algorithm; Sky Picture File; Chapter 3: Visualization Products](#) except that a black-green-red-purple-white color spectrum is substituted for the gray scale of the sky picture.
2. The observed sky radiances from the radiance data file are pictured in the three files Blue°Image.png, Red°Image.png, and NIR°Image.png for the 450 nm, 650 nm, and 800 nm bands respectively. These are produced only if a daylight spectral retrieval is being performed; additionally, NIR°Image.png is produced only if a three-color retrieval is being performed. The pictures are scaled in the same manner as the sky picture file as discussed in the first item in this list except that a black-green-red-purple-white color spectrum is substituted for the gray scale of the sky picture.
3. Next the ratios of the first two list items are plotted; specifically the ratios of observed-to-pristine sky radiances multiplied by the ad hoc constants. These are pictured in the three files Blue°Ratio°to°Pristine.png, Red°Ratio°to°Pristine.png, and NIR°Ratio°to°Pristine.png for the 450 nm, 650 nm, and 800 nm bands respectively. These are produced only if the first two list items are generated; additionally, NIR°Ratio°to°Pristine.png is produced only if a three-color retrieval is being performed. Known bad pixels (e.g., those in the occulter or a building image) are depicted as black by an application of the array mask from the radiance data file; see [Variables; Dimensional Variables, and Coding; Radiance Data File; Chapter 4: Radiance Product](#) for a description of this array. The pictures are scaled in the same manner as the sky picture file as discussed in the first item in this list except that a black-green-red-purple-white color spectrum is substituted for the gray scale of the sky picture.
4. Ratios of the ratios from list item three above are developed next. The 450 nm sky ratio of blue-observed-to-blue-pristine is considered the basis for these ratios of ratios. The first is for the 650 nm spectral band and is the ratio of red-observed-to-red-pristine to this 450 nm basis; the second is for the 800 nm band and is the ratio of NIR-observed-to-NIR-pristine to the 450 nm basis. The file names are Red-Pristine°to°Blue-Pristine.png and NIR-Pristine°to°Blue-Pristine.png respectively. These are produced only if the pictures in the third list item above are generated; additionally the latter is produced only if a three-color retrieval is being performed. As in the third list item above the mask from the radiance data file is applied to blacken known bad pixels. The pictures are scaled in the same manner as the sky picture file as discussed in the first item in this list except that a black-green-red-purple-white color spectrum is substituted for the gray scale of the sky picture.
5. The fifth product is currently only displayed to the consul when the special run is made, but in the next release of the WsiIdpc5_vx tree should be available as a PNG file as well. It is a cumulative graphical representation of the data in list item 3, the ratios of observed-to-pristine radiances. The abscissa is the cumulative pixel count as a fraction of the total number of pixels and the ordinate is a

ratio value. The three plotted lines show the number of pixels with a value equal to or less than the ratio value for each spectral band (450 nm, 650 nm, and 800 nm). Of course, this plot is only produced if the third list item is, and the 800 nm line is produced only if a three-color retrieval is being performed. As in the third item above the mask from the radiance data file is applied to identify known bad pixels, but in this case they are ignored in the accumulation process.

6. The sixth product is a pair of graphs produced when the daylight spectral retrieval is executed. It is a pixel density plot versus ratios displayed in the third product above; a sample is shown in Figure 4 above. The abscissa for both is the 450 nm observer-to-pristine ratio, and the either the similar ratio for 650 nm or 800 nm. The file names are Histogram^oin^oBlue-Red^oPlane.png and Histogram^oin^oBlue-NIR^oPlane.png respectively. In these graphs all pixels from a radiance file are represented by dots, one per pixel. Since there are nearly 200,000 pixels, the dots tend to be very dense around some common values; therefore the graph is actually a plot of the density of pixel dots. The color spectrum is black–green–red–purple–white from low to high density respectively. The Histogram^oin^oBlue-NIR^oPlane.png graph is produced only if a three-color retrieval is being performed.
7. Next the results of the daylight spectral retrieval are displayed as an image of the same size and orientation as the sky radiances in item two above. The file name is Decision^oSpectral.png; it is only produced when the relevant retrieval is performed. The picture is color coded according to the scheme of Table 2.
8. The observed sky radiances from the radiance data file are pictured in the file Clear^oImage.png for the null-filter band. It is produced only if a nighttime optical density retrieval is being performed. The pictures are scaled in the same manner as the sky picture file as discussed in [Algorithm; Sky Picture File; Chapter 3: Visualization Products](#) using the same gray scale.
9. A detailed graph for the nighttime optical density retrieval is in the file named Star^oFlux.png; this log-log graph displays observed star fluxes on the ordinate versus calculated pristine sky star fluxes on the abscissa. Both axes are in the units of milliwatts per meter squared per nanometer of wavelength. A color coded symbol is plotted for each star that had a sky classification assigned strictly based on flux considerations, i.e., items 5 to 9 in the list found in the subsection [Nighttime Optical Density Retrieval; Algorhthm; Sky Feature File; Chapter 5: Sky Feature Product](#) but not items 1 to 4. The color code is red for opaque determination, green for translucent, blue for clear, and orange for indeterminate. Various lines are also plotted showing the values for different magnitude stars and the various relevant cutoff points of Table 16. The graph is produced only when the relevant retrieval is performed.
10. A sky radiance image is plotted next that is a modification of that discussed in 8 above. The name for this file is Star^oSelection.png and it is produced only when the nighttime optical density retrieval is performed. A small square is overlaid on the gray scale radiance image at the location of each selected star. The square has a color coded outline with a background subtracted image of the star inside. The color code is white for bright determination, red for opaque, green for translucent, blue for clear, and orange for indeterminate. The user can readily tell if the stars are being located (if there is an image inside the box), and then how accurately (how centered the image is in the box). Inaccurate location is most often due to instrument having physically shifted after the last calibration file was released.
11. The results of the nighttime optical density retrieval are displayed as an image of the same size and orientation as the sky radiances in item two above. The file name is Decision^oDensity.png; it is only

produced when the relevant retrieval is performed. The picture is color coded according to the scheme of Table 3.

5.1.4.12 Finalization

The main work of the algorithm is now complete; only two tasks remain. The first is to write the Sky feature NetCDF file to disk using the results of all the calculations just discussed. The file is opened, the dimensions, variables, and attributes defined, the data written, and the file closed. The second is to write the two visualization products (as appropriate for the retrieval kinds performed) discussed in [Sky Feature Spectral Retrieval Picture File; Chapter 3: Visualization Products](#) and [Sky Feature Optical Density Retrieval Picture File; Chapter 3: Visualization Products](#). The two IDL sub-procedures `WriteNetCDF_vx` and `WriteJpg_vx` perform these chores.

5.1.5 Usage Hints

The careful reader of the above sections in this chapter will have a good idea of how to extract information from the Whole Sky Imager sky feature data file and how to use it appropriately. However, a few hints may make the process more enjoyable.

IDL handles character variables in NetCDF files as ASCII encoded byte strings of fixed length, hence the extra dimension variables in Table 8. If one is having trouble matching a search character string to a value retrieved from a NetCDF character variable, it is best to print out the associated character string one time to discover where the blanks and nulls are inserted to pad out to the fixed length. Then the search string can be modified appropriately. Also remember that character variables are case sensitive.

Pixel columns and rows in the classification image can be related to the angular sky coordinates of zenith and azimuth by using conversion arrays contained in the calibration file discussed in [Variables; Dimensions, Variables, and Coding; Calibration File; Chapter 7: Calibration Product](#). Arrays for the inverse conversion are also to be found there.

The percentages found in the variable `class_sums` can be a bit tricky to implement in a study. The user must use the indexing scheme properly (see [Variables; Dimensions, Variables, and Coding; Sky Feature File; Chapter 5: Sky Feature Product](#)); it is straightforward but easy to invert on the first try. Also the user must decide how the categories of indeterminate and undefined will be handled in his analysis. If they can be handled explicitly, then the data can be used as written and the percentages by sky feature will sum to 100% in any given region. However, in most analyses they should be normalized away – *a very important step*. Take the simple example of an actual daytime sky half covered with bright clouds and half clear. The percentages in `class_sums` would be something like 45% bright clouds, 45% clear, and 10% undefined. The undefined pixels would be mostly near the horizon and under the occulter. Obviously, if the bright cloud figure were used to represent cloud fraction the result would be 5% too low.

The correct figure renormalizes for the undefined by dividing the bright cloud percentage by the sum of all percentages not associated with indeterminate or undefined pixels. Thus $45\% \div (45\% + 45\%) = 50\%$, the correct answer.

The user of `class_sums` with optical density retrievals must not only be aware of the renormalization just discussed, but also must decide how to handle the bright sky feature. This situation can occur with bright moonlit cirrus, bright moonlit heavy aerosol, and moonlit or artificially lit ground fog. Most frequently, the bulk of bright pixels in any given scene will be the result of just one of these causes. Thus if the analyst is calculating a single sky feature percentage, he must decide for each scene whether to include bright pixels as clouds or clear. The surest way is to study the sky picture file, but this does not lend itself to automation. From the study of many such images, it has been discovered that a reasonable estimate can be made in the following manner. First, if the moon is not in the sky and with a sufficient portion of the disk illuminated, then the number of bright pixels will be small but difficult to automatically reclassify. With sufficient moonlight, if the bright are interspersed with clear, then most likely aerosol is causing that classification and those pixels can be considered clear. On the other hand and again with sufficient moonlight, if the bright are interspersed with opaque, then they can be reasonably considered as brightly lit clouds.

The sky regions in Table 11 used in the array `class_sums` are defined in a special manner to accommodate the user who does not desire to include sky feature retrievals made near the horizon in his study. While radiances near the horizon are extremely useful, sky feature retrievals are problematic do to the air masses involved. Since the lower quadrants in regions 6 to 9 only extend downwards to 80° in zenith angle, a proper summation of regions 1 to 9 would give sky feature percentages that do not include the last 10° near the horizon. Of course region 0 includes the entire hemisphere.

If the user is having difficulty in interpreting a particular sky feature retrieval, and that retrieval is of special importance, then consulting the graphs and pictures in the [Extra Information; Algorithm; Sky Feature File; Chapter 5: Sky Feature Product](#) section above will clarify the situation. These products can be *very* helpful, but are only available through special production runs by the ARM Archive or instrument mentor.

5.1.6 Limitations

An analyst should be very careful in how the sky feature data is interpolated. In the section above ([Usage Hints; Algorithm; Sky Feature File; Chapter 5: Sky Feature Product](#)), a method is suggested for the effective spatial interpolation of sky feature percentages as found in `class_sums` to bridge regions of indeterminate or undefined. That will introduce little bias. However, temporal interpolation of those same percentages to span times when the Sky Feature Files are not present in the ARM Archive *will* introduce bias. One major cause of the files not being present is a quality down check for improper daytime retrieval of thin clouds (see [Potential Biases; Data Quality Control; Chapter 1: Overview](#)). Thus the Sky Feature Files present in the ARM Archive on either side of a time gap may not represent the situation in the time between them.

Within the preceding sections of this chapter several limitations in the usage of the sky feature retrieval have been mentioned. They are mentioned below again is a summary but with considerably less detail. Please refer above for a complete description.

The calculated pristine sky radiance values must be adjusted to match those actually observed on very clear days; see [Daytime Spectral Retrieval; Algorithm; Sky Feature File; Chapter 5: Sky Feature Product](#).

The three spectral band daytime spectral retrieval does not utilize a full three dimensional analysis of the observer-to-pristine ratios; see [Daytime Spectral Retrieval; Algorithm; Sky Feature File; Chapter 5: Sky Feature Product](#).

The twilight sky feature retrieval is not implemented; see [Twilight Spectral Retrieval; Algorithm; Sky Feature File; Chapter 5: Sky Feature Product](#).

The nighttime spectral sky feature retrieval is not implemented; see [Nighttime Spectral Retrieval; Algorithm; Sky Feature File; Chapter 5: Sky Feature Product](#).

The additional graphical products do not have unique names at different retrieval time steps; see [Extra Information; Algorithm; Sky Feature File; Chapter 5: Sky Feature Product](#).

A better means exists to calculate star flux, but it is not implemented; see [Nighttime Optical Density Retrieval; Algorithm; Sky Feature File; Chapter 5: Sky Feature Product](#).

The effect of the air mass near the horizon should be considered in a more elegant manner; see [Nighttime Optical Density Retrieval; Algorithm; Sky Feature File; Chapter 5: Sky Feature Product](#).

6. Summary Products

There are three Whole Sky Imager summary products available from the ARM Archive. The first is a NetCDF file with 650 nm radiance values from selected patches of the sky at a nominal two-minute interval for a 24-hour period. The second is also a NetCDF file containing 24 hours of data, but for sky feature codes by sky region at a nominal six-minute interval. The last is a graphic depicting the data just mentioned but only for the whole sky hemisphere and not for any of the component sectors.

6.1 Patch Radiance File

6.1.1 Purpose

The patch radiance file is in NetCDF format and contains a 24-hour Greenwich day's worth of 650 nm radiance data at two-minute time intervals (ten-minute old style, see [Data Collection Strategy; Chapter 1: Overview](#)) for three $5^\circ \times 5^\circ$ sky patches at zenith and 30° off-zenith to the east and west. The radiance data is extracted from temporary output files from the execution of the IDL program `WsiIdpc2_vx`, where x stands for the version number of the current release (14 as of this writing). `WsiIdpc2_vx` is parallel code to the program `WsiIdpc4_vx` which produces the radiance data file (see [Radiance Data File; Chapter 4: Radiance Product](#)) except that it only processes 650 nm and null-filter data on a nominal two-minute time interval, and that it explicitly stores patch data as a subset of the calibrated radiance array in addition to that array. The calibration algorithm and mask determination are identical to that of `WsiIdpc4_vx` as discussed in [Algorithm; Radiance Data File; Chapter 4: Radiance Product](#), therefore the radiances reported in this patch radiance file can be considered a time superset (two versus six-minute) and spatial subset (patches versus sky hemisphere) of the radiance data file.

This file is useful to the experimenter desiring to compare Whole Sky Imager data with narrow field-of-view zenith pointing instruments, or to the analyst developing cloud type (e.g., cumulus) information based on radiance distributions.

6.1.2 Naming Convention

The patch radiance file has the following name based on standard ARM naming practices.

ssswsipatchsummaryff.a1.yyyymmdd.hhmmss.cdf

or

ssswsipatchsummaryff.b1.yyyymmdd.hhmmss.cdf

where the blue characters are literal and

sss ff = the site and facility identifier as listed in Table 1

yyyymmdd.hhmmss = the standard 15 character ARM date time stamp showing year, month, day, hour, minute, and second.

As a result of a change in the ARM data level standard, the “a1” found in earlier files has been replaced by “b1” in later ones. There is no difference in content or structure between the two. Since one file is produced per Greenwich day, there is no corresponding UNIX style tar archive. Also the hours, minutes, and seconds are always set to zero.

6.1.3 Dimensions, Variables, and Coding

6.1.3.1 Global Attributes

The global attributes in the patch summary file contain useful background information on the instrument and production environment. The user should always extract them with a suitable NetCDF dump routine and become familiar with their content. As the names and values are purposely designed to be human readable, they are not listed separately herein.

6.1.3.2 Dimensions

The dimensions associated with the patch summary file are listed in Table 17 and discussed in the paragraphs below.

ID	Name	Size
0	patch_index	Unlimited
1	patch_count	3
2	patch_cols	15
3	patch_rows	15
4	char_5	5
5	char_25	25
6	char_64	64

Patch_index is the unlimited dimension and, as is typical in NetCDF files, represents the time steps for the data.

The **patch_count** dimension always has the value of 3, and is the number of patches recorded at each time step.

The **patch_cols** and **patch_rows** dimensions always have the value of 15, and are the number of columns and rows in each of the extracted patch arrays.

The other dimensions (**char_5**, **char_25**, and **char_64**) are necessary in the development of the NetCDF file but are of no particular concern to its user.

6.1.3.3 Variables

The variables associated with the patch summary file are listed in the next table and discussed in the paragraphs below. The attributes associated with each variable contain valuable information; the user should extract them with a suitable NetCDF dump routine and become familiar with their content. In particular the attributes have null value, range, and allowed delta information.

The **base_time** variable – long name: base time in epoch – of type long is a scalar whose value is the number of seconds since January 1, 1970 at 0000 hours Greenwich time to the time of the first included data collection.

The **time_offset** variable – long name: offset from base time – of type double is a vector whose values are the time to be added to the **base_time** to calculate the moment of each time step.

The **patches** array – long name: calibrated patch values – of type float is the data of most interest to the user. It contains the calibrated radiance data for each patch extracted in units of milliwatts per meter squared per steradian of source per nanometer of wavelength. Its null value, IEEE floating point not-a-number, is inserted for all suspect values as indicated by the radiance file mask. The array has four dimensions with the first two being the columns and rows of the patch respectively. The third dimension defines layers in the file, one layer for each each of the three patches reported at every time step. The three elements in this dimension correspond in order to a patch centered 30° due east of zenith, a patch at zenith, and a patch centered 30° due west of zenith. The fourth dimension is the time step.

The **patch_avg** array – long name: patch averages – of type float contains the average value of each patch in the **patches** array in units of milliwatts per meter squared per steradian of source per nanometer of wavelength. Its null value, IEEE floating point not-a-number, is inserted whenever more than 75% of the values in the averaged patch are themselves null. The array has two dimensions. The first dimension defines layers in the file, one layer for each of the three patches reported at every time step. The three elements in this dimension correspond in order to a patch centered 30° due east of zenith, a patch at zenith, and a patch centered 30° due west of zenith. The second dimension is the time step.

Table 18. Patch summary file variables.

ID	Name	Type	Attribute Count	Dimension Count	Dimension IDs	Actual Dimensions
0	base_time	Long	5	0	--	--
1	time_offset	Double	5	1	0	x
2	patches	Float	6	4	2, 3, 1, 0	15, 15, 3, x
3	patch_avg	Float	6	2	1, 0	3, x
4	patch_sd	Float	6	2	1, 0	3, x
5	image_quality	Char	4	1	0	x
6	header_date_time	Char	3	2	5, 0	25, x
7	cal_file	Char	3	1	6	64
8	sun_az	Float	6	1	0	x
9	sun_zen	Float	6	1	0	x
10	site	Char	4	1	4	5
11	latitude	Float	6	0	--	--
12	longitude	Float	6	0	--	--
13	alt	Float	6	0	--	--

The **patch_sd** array – long name: patch standard deviations – of type float has exactly the same form as **patch_avg** and has the same null value logic, but contains the standard deviation for each patch.

The **image_quality** vector – long name: quality control character for image – of type character identifies the perceived quality of the underlying calibrated image sets for this file. The single dimension corresponds to the time steps of the data. For a discussion of this quality value see [Quality Checking; Algorithm; Radiance Data File; Chapter 4: Radiance Product](#). The quality control process discussed in [Data Quality Control; Chapter 1: Overview](#) is independent of this variable’s values; therefore the mere existence of a time step in this file is indication that the data control process found nothing objectionable with the radiance data file that underlies it, regardless of the value of this variable. This variable may have a value from the set {"A", "B", "C", "D", "F"} with the assigned meanings of: A = no problems detected, B = minor problems that do not affect calibrated image validity, C = minor problems that could affect image validity, D = image calibration suspect, and F = bad calibrated images.

The **header_date_time** vector – long name: raw image header time stamp – of type character has the format: yyyy:mm:dd:hh:mm:ss where each letter stands for a numeral representing (in order) year : month : day : hour : minute : second. The single dimension corresponds to the time steps of the data. The value at a time step is that common value found in the header of the raw data files upon which the underlying radiance file was built; see [Header Information; Dimensions, Variables, and Coding; Raw Data Files; Chapter 2: Raw Products](#). This date-time group is the most accurate available; if the value of **base_time** plus **time_offset** is in disagreement then **header_date_time** should be trusted.

The **cal_file** variable – long name: calibration file name – of type character is a scalar containing the name of the calibration file used to retrieve the underlying radiance file with the lowest time in the set.

One can assume with considerable assurance that this same calibration file was used for the entire set. It is stored without any directory information.

The two vectors **sun_az** and **sun_zen** – long names: sun azimuth and sun zenith – of type float contain respectively the calculated azimuth and zenith angles for the sun’s position extracted from the underlying radiance data files. The single dimension for each corresponds to the time steps of the data. The values are in angular degrees. Sun_az has the range 0° to 360° and is measured clockwise from geographic north such that east is 90°. Sun_zen has the range 0° to 180° and is measured from the local vertical such that the horizon is 90°; value greater than 90° indicates that it is nighttime and should not be present.

The **site** variable – long name: site location – of type character is a scalar that identifies the site location of the instrument involved in the data collection. Its value is one of the set {"sheba", "nsa-1", "nsa-2", "twp-1", "twp-2", "twp-3", "sgp-1", "spg-2", "other"}. This set corresponds to the designators in Table 1 as follows {shbC1, nsaC1, nsaC2, twpC1, twpC2, twpC3, sgpC1, sgpL3, (none)} respectively. The value of the variable site should correspond to the site information imbedded in the patch summary file name; if not clues such as horizon features in the associated sky picture file can be used to determine which is correct. However, the value of site is ultimately obtained from the headers of the raw data files that underlie the radiance file with the lowest time that underlies this patch summary file, and the instrument gets the values to insert in the raw files from a lookup table installed by the instrument operator. Therefore, these values are only valid if the operator keeps that table up to date. Since this variable serves only as a check on the site designator in the file name, the accuracy of its value is not a matter of highest importance.

The **latitude** and **longitude** variables – long names: latitude and longitude– of type float are scalars that specify the geographical coordinates of the instrument’s location at the lowest time step. The units for both variables are decimal degrees. Latitude ranges between -90° and +90°, where -90° is at the Earth’s South Pole, 0° is at the equator, and 90° is at the North Pole. Longitude ranges between -180° and +180°, where 0° is the meridian through Greenwich, England, positive values are in the eastern hemisphere, and negative values are in the western hemisphere. Fixed base (all Whole Sky Imagers located at permanent ARM sites) and moveable base instruments (the one on SHEBA ship) develop the data for these variables a bit differently. Considering the two cases separately, we have:

- For fixed base instruments, latitude and longitude are transcribed from the raw data files which underlie the lowest time radiance file underlying this patch summary file. The instrument gets the value to insert in the raw files from a lookup table installed by the instrument operator; therefore, these values are only valid if the operator keeps that table up to date. Since the values for latitude and longitude in the calibration file (see [Variables; Dimensions, Variables, and Coding; Calibration File; Chapter 7: Calibration Product](#)) override those stored in these variables during radiance and sky feature retrievals, the accuracy of their values here is not a matter of highest importance.
- For moveable base instruments like on the SHEBA ship, latitude and longitude is determined from GPS instrumentation and should be quite accurate. They are not overridden by calibration file values as in the fixed base instruments.

The **alt** variable – long name: altitude – of type float is a scalar that specifies the geographical altitude of the instrument’s location at the lowest time step. It is expressed in meters above mean sea level. Fixed base (all Whole Sky Imagers located at permanent ARM sites) and moveable base instruments (the one on SHEBA ship) develop the data for these variables a bit differently. Considering the two cases separately, we have:

- For fixed base instruments, altitude is transcribed from the raw data files which underlie the lowest time radiance file underlying this patch summary file. The instrument gets the value to insert in the raw files from a lookup table installed by the instrument operator; therefore, the value is only valid if the operator keeps that table up to date. Since the value of altitude in the calibration file (see [Variables; Dimensions, Variables, and Coding; Calibration File; Chapter 7: Calibration Product](#)) overrides those stored in these variables during radiance and sky feature retrievals, the accuracy of their values here is not a matter of highest importance.
- For moveable base instruments like on the SHEBA ship, altitude is determined from GPS instrumentation, but is not very accurate when GPS dither is enabled by the satellite system’s controllers as it was during the SHEBA experiment. It is not overridden by calibration file values as in the fixed base instruments. Fortunately, significant errors in altitude do not translate into errors in radiance retrievals and only very minor errors in cloud retrievals.

6.1.4 Algorithm

The processing associated with WsiIdpc3_vx is rather simple. The first step is to define the patch summary NetCDF file’s dimensions, variables, and attributes. Then the program accesses all of the temporary products of the WsiIdpc2_vx associated with the day of interest, extracts the appropriate data, and populates the patch summary file. Refer to the section [Purpose; Patch Radiance File; Chapter 6: Summary Products](#) above for a discussion of the WsiIdpc2_vx calibrated radiance products.

6.1.5 Usage Hints

The data in this file are straightforward to use. A useful plot is the average and standard deviation versus time. Such plots show unique signatures for clear, aerosol, and a variety of cloudy conditions. For example, cirrus clouds are very bright and have large deviations, the amount of aerosol in a clear sky is proportional to the brightness at zenith, and status clouds have low brightness with a deviation considerably smaller than cirrus.

If the analyst is comparing Whole Sky Imager brightness with that of another instrument, he should use the pixel to angle conversions in the calibration file (see [Variables; Dimensions, Variables, and Coding; Calibration File; Chapter 7: Calibration Product](#)) to match fields of view.

IDL handles character variables in NetCDF files as ASCII encoded byte strings of fixed length, hence the extra dimension variables in Table 17. If one is having trouble matching a search character string to a value retrieved from a NetCDF character variable, it is best to print out the associated character string one time to discover where the blanks and nulls are inserted to pad out to the fixed length. Then the search string can be modified appropriately. Also remember that character variables are case sensitive.

6.1.6 Limitations

The uses of this product involve the studies of daylight radiances. Since raw 650 nm spectral data is generally collected only during daylight and twilight hours for the new style of collection (see [Data Collection Strategy; Chapter 1: Overview](#)), the patch summary data rarely contains any nighttime radiances. More nighttime data exists in those files associated with the old collection style.

6.2 Sky Feature Summary File

6.2.1 Purpose

The sky feature summary file is in NetCDF format and contains sky feature data at six-minute time intervals (ten-minute old style, see [Data Collection Strategy; Chapter 1: Overview](#)) for ten sky regions for an entire 24-hour Greenwich day. The feature data is extracted from the appropriate set of sky feature files as described in [Chapter 5: Sky Feature Product](#). This file is useful to the experimenter desiring to compute long term averages of sky cover but who isn't concerned with the high resolution coverage details within broad sky regions.

6.2.2 Naming Convention

The sky feature summary file has the following name based on standard ARM naming practices.

`ssswsiccloudsummaryff.c1.yyyymmdd.hhmmss.cdf`

where the blue characters are literal and

`sss ff` = the site and facility identifier as listed in Table 1

`yyyymmdd.hhmmss` = the standard 15 character ARM date time stamp showing year, month, day, hour, minute, and second.

Since one file is produced per Greenwich day, there is no corresponding UNIX style tar archive. Also the hours, minutes, and seconds are always set to zero.

6.2.3 Dimensions, Variables, and Coding

6.2.3.1 Global Attributes

The global attributes in the patch summary file contain useful background information on the instrument and production environment. The user should always extract them with a suitable NetCDF dump routine and become familiar with their content. As the names and values are purposely designed to be human readable, they are not listed separately herein.

6.2.3.2 Dimensions

The dimensions associated with the sky feature summary file are listed in the next table and discussed in the paragraphs below.

Table 19. Sky feature summary file dimensions.

ID	Name	Size
0	time	Unlimited
1	retrieval_types	2
2	areas	10
3	types	10
4	char_5	5
5	char_20	20
6	char_64	64

Time is the unlimited dimension and, as is typical in NetCDF files, represents the time steps for the data.

The **retrieval_types** dimension always has the value of 2, and is the maximum number of sky feature retrieval that may be performed at each time step, specifically spectral and optical density (see [Daytime Spectral Retrieval; Sky Feature File; Chapter 5: Sky Feature Product](#) and [Nighttime Optical Density Retrieval; Sky Feature File; Chapter 5: Sky Feature Product](#)).

The **areas** dimension always has the value of 10, and is the number of reporting regions into which the sky is divided.

The **types** dimension always has the value of 10, and is the number of different sky feature types.

The other dimensions (**char_5**, **char_25**, and **char_64**) are necessary in the development of the NetCDF file but are of no particular concern to its user.

6.2.3.3 Variables

The variables associated with the sky feature summary file are listed in the next table and discussed in the paragraphs below. The attributes associated with each variable contain valuable information; the user should extract them with a suitable NetCDF dump routine and become familiar with their content. In particular the attributes have null value, range, and allowed delta information.

The **base_time** variable – long name: base time in epoch – of type long is a scalar whose value is the number of seconds since January 1, 1970 at 0000 hours Greenwich time to the time of the first included data collection.

The **time_offset** variable – long name: offset from base time – of type double is a vector whose values are the time to be added to the **base_time** to calculate the moment of each time step.

Table 20. Sky feature summary file variables.

ID	Name	Type	Attribute Count	Dimension Count	Dimension IDs	Actual Dimensions
0	base_time	Long	5	0	--	--
1	time_offset	Double	5	1	0	x
2	header_date_time	Char	3	2	5, 0	20, x
3	class_exist	Byte	5	2	1, 0	2, x
4	class_sums	Float	9	4	2, 3, 1, 0	10, 10, 2, x
5	class_quality	Char	7	2	1, 0	2, x
6	c1_file	Char	3	2	6, 0	64, x
7	cal_file	Char	3	1	6	64
8	sun_az	Float	6	1	0	x
9	sun_zen	Float	6	1	0	x
10	moon_az	Float	6	1	0	x
11	moon_zen	Float	6	1	0	x
12	site	Char	5	1	4	5
13	latitude	Float	6	0	--	--
14	longitude	Float	6	0	--	--
15	alt	Float	6	0	--	--

The **header_date_time** vector – long name: input image time stamp – of type character has the format: yyyy:mm:dd:hh:mm:ss where each letter stands for a numeral representing (in order) year : month : day : hour : minute : second. The single dimension corresponds to the time steps of the data. The value at a time step is that common value found in the header of the raw data files upon which the underlying radiance file and therefore sky feature file was built; see [Header Information; Dimensions, Variables, and Coding; Raw Data Files; Chapter 2: Raw Products](#). This date-time group is the most accurate available; if the value of base_time plus time_offset is in disagreement then header_date_time should be trusted.

The **class_exist** array – long name: classification type existence – of type byte informs the user of the type of sky feature retrievals available at any time step. Since the first dimension has the value of two, there are two elements for each time step. The first of these elements is related to the spectral density retrieval, and assumes the value of one if the retrieval was performed and zero if it was not. The second element assumes the same values but for the optical density retrieval. The second dimension is the time step.

The **class_sums** array – long name: classification tallies by area – of type float is the data of most interest to the user. It contains the percentages extracted from the variable of the same name in the underlying sky feature file for each of the sky feature retrievals performed, see [Variables; Dimensions, Variables, and Coding; Sky Feature File; Chapter 5: Sky Feature Product](#). The array has four dimensions. The first ranges from 0 to 9 and corresponds to different areas of the sky as detailed in Table 11. The second dimension also ranges between 0 and 9 but corresponds to the classification type as found in the “Value” column of Table 10. The third dimension defines layers in the file, one layer for each of the

cloud-cover retrievals. Index zero in this dimension (first layer) corresponds to the spectral retrieval and index one (second layer) to the optical density retrieval. The fourth dimension is for the time step of the retrievals. Each element has a value between 0.0 and 100.0 that is the percentage of sky in the region that corresponds to the particular classification type. For example, the value of class_sums [3, 8, 0, 124] is the percentage of pixels in the upper east quadrant (region 3) that have the value of “indeterminate” (classification 8) that are from the spectral retrieval (0) for the 125th time step (124, with zero based counting). The percentages in any region add to 100.0; thus the sum from $j = 0$ to 9 of class_sums [a, j, c, d] is 100.0 where a, c, and d are any allowable fixed index value.

The **class_quality** array – long name: quality control character for classification sets – is a type character that identifies the perceived quality of sky feature retrievals upon which the array class_sums is based. The quality control process discussed in the section [Data Quality Control; Chapter 1: Overview](#) is independent of this vector’s value; therefore the mere existence of sky feature summary data for a given time step is indication that the data control process found nothing objectionable with the underlying radiance and sky feature files regardless of the values in this vector. The first dimension defines two elements, one element for each of the cloud-cover retrievals. Index zero in this dimension (first layer) corresponds to the spectral retrieval and index one (second layer) to the optical density retrieval. The second dimension is for the time step of the retrievals. The values of this array are those of the vectors of the same name in the underlying sky feature files (see [Variables; Dimensions, Variables, and Coding; Sky Feature File; Chapter 5: Sky Feature Product](#)). The elements of this array may have any value from the set {“A”, “B”, “C”, “D”, “F”} with the assigned meanings of: A = no problems detected, B = minor problems that do not affect classification validity, C = minor problems that could affect classification validity, D = classification suspect, and F = bad classification.

The **c1_file** vector – long name: input c1 file name – of type character contains the names of the sky feature files from which the data in this file was extracted. The dimension of this vector corresponds to the time step of the retrievals and hence the time associated with each sky feature file. The names are stored without any directory information.

The **cal_file** variable – long name: calibration file name – of type character is a scalar containing the name of the calibration file used to retrieve the underlying sky feature file and its radiance file. The name is extracted from the sky feature file with the lowest time in the set. One can assume with considerable assurance that this same calibration file was used for the entire set. It is stored without any directory information.

The two vectors **sun_az** and **sun_zen** – long names: calculated sun azimuth and calculated sun zenith – of type float contain respectively the calculated azimuth and zenith angles for the sun’s position as extracted from the variables sun_calc_az and sun_calc_zen in the underlying sky feature files (see [Variables; Dimensions, Variables, and Coding; Sky Feature File; Chapter 5: Sky Feature Product](#)). The single dimension for each corresponds to the time steps of the data. The values are in angular degrees. Sun_az has the range 0° to 360° and is measured clockwise from geographic north such that east is 90°. Sun_zen has the range 0° to 180° and is measured from the local vertical such that the horizon is 90°; value greater than 90° indicates that it is nighttime.

The two vectors **moon_az** and **moon_zen** – long names: calculated moon azimuth and calculated moon zenith – of type float contain respectively the calculated azimuth and zenith angles for the moon’s

position. They are extracted from the scalars `moon_calc_az` and `moon_calc_zen` in the sky feature files that underlie this sky feature summary file (see [Variables; Dimensions, Variables, and Coding; Sky Feature File; Chapter 5: Sky Feature Product](#)). The single dimension for each corresponds to the time steps of the data. The values are in angular degrees and have the same range and meaning as for `sun_az` and `sun_zen`.

The **site** variable – long name: site location – of type character is a scalar that identifies the site location of the instrument involved in the data collection. Its value is one of the set {"sheba", "nsa-1", "nsa-2", "twp-1", "twp-2", "twp-3", "sgp-1", "sgp-2", "other"}. This set corresponds to the designators in Table 1 as follows {shbC1, nsaC1, nsaC2, twpC1, twpC2, twpC3, sgpC1, sgpL3, (none)} respectively. The value of the variable site should correspond to the site information imbedded in the patch summary file name; if not clues such as horizon features in the associated sky picture file can be used to determine which is correct. However, the value of site is ultimately obtained from the headers of the raw data files that underlie the radiance file with the lowest time that underlies this patch summary file, and the instrument gets the values to insert in the raw files from a lookup table installed by the instrument operator. Therefore, these values are only valid if the operator keeps that table up to date. Since this variable serves only as a check on the site designator in the file name, the accuracy of its value is not a matter of highest importance.

The **latitude** and **longitude** variables – long name: latitude and longitude – of type float are scalars that specify the geographical coordinates of the instrument's location at the lowest time step. The units for both variables are decimal degrees. Latitude ranges between -90° and $+90^{\circ}$, where -90° is at the Earth's South Pole, 0° is at the equator, and 90° is at the North Pole. Longitude ranges between -180° and $+180^{\circ}$, where 0° is the meridian through Greenwich, England, positive values are in the eastern hemisphere, and negative values are in the western hemisphere. Fixed base (all Whole Sky Imagers located at permanent ARM sites) and moveable base instruments (the one on SHEBA ship) develop the data for these variables a bit differently. Considering the two cases separately, we have:

- For fixed base instruments, latitude and longitude are transcribed from the raw data files which underlie the lowest time radiance file underlying this patch summary file. The instrument gets the value to insert in the raw files from a lookup table installed by the instrument operator; therefore, these values are only valid if the operator keeps that table up to date. Since the values for latitude and longitude in the calibration file (see [Variables; Dimensions, Variables, and Coding; Calibration File; Chapter 7: Calibration Product](#)) override those stored in these variables during radiance and sky feature retrievals, the accuracy of their values here is not a matter of highest importance.
- For moveable base instruments like on the SHEBA ship, latitude and longitude is determined from GPS instrumentation and should be quite accurate. They are not overridden by calibration file values as in the fixed base instruments.

The **alt** variable – long name: altitude – of type float is a scalar that specifies the geographical altitude of the instrument's location at the lowest time step. It is expressed in meters above mean sea level. Fixed base (all Whole Sky Imagers located at permanent ARM sites) and moveable base instruments (the one on SHEBA ship) develop the data for these variables a bit differently. Considering the two cases separately, we have:

- For fixed base instruments, altitude is transcribed from the raw data files which underlie the lowest time radiance file underlying this patch summary file. The instrument gets the value to insert in the raw files from a lookup table installed by the instrument operator; therefore, the value is only valid if the operator keeps that table up to date. Since the value of altitude in the calibration file (see [Variables; Dimensions, Variables, and Coding; Calibration File; Chapter 7: Calibration Product](#)) overrides those stored in these variables during radiance and sky feature retrievals, the accuracy of their values here is not a matter of highest importance.
- For moveable base instruments like on the SHEBA ship, altitude is determined from GPS instrumentation, but is not very accurate when GPS dither is enabled by the satellite system's controllers as it was during the SHEBA experiment. It is not overridden by calibration file values as in the fixed base instruments. Fortunately, significant errors in altitude do not translate into errors in radiance retrievals and only very minor errors in cloud retrievals.

6.2.4 Algorithm

The processing associated with `WsiIdpc6_vx` is rather simple. The first step is to define the sky feature summary NetCDF file's dimensions, variables, and attributes. Then the program accesses the information needed from all the sky feature files associated with the day of interest and populates the sky feature summary file. An additional section of this program creates the sky feature plot file discussed below in [Sky feature Plot File; Chapter 6: Summary Products](#).

6.2.5 Usage Hints

The percentages found in the array `class_sums` can be a bit tricky to implement in a study. The user must use the indexing scheme properly (see [Variables; Dimension, Variables, and Coding; Sky Feature Summary File; Chapter 6: Summary Products](#)); it is straightforward but easy to invert on the first try. Also the user must decide how the categories of indeterminate and undefined will be handled in his analysis. If they can be handled explicitly, then the data can be used as written and the percentages by sky feature will sum to 100% in any given region. However, in most analyses they should be normalized away – *a very important step*. Take the simple example of an actual daytime sky half covered with bright clouds and half clear. The percentages in `class_sums` would be something like 45% bright clouds, 45% clear, and 10% undefined. The undefined pixels would be mostly near the horizon and under the occulter. Obviously, if the bright cloud figure were used to represent feature fraction the result would be 5% too low. The correct figure renormalizes for the undefined by dividing the bright cloud percentage by the sum of all percentages not associated with indeterminate or undefined pixels. Thus $45\% \div (45\% + 45\%) = 50\%$, the correct answer.

The user of `class_sums` with optical density retrievals must not only be aware of the renormalization just discussed, but also must decide how to handle the bright sky feature. This situation can occur with bright moonlit cirrus, bright moonlit heavy aerosol, and moonlit or artificially lit ground fog. Most frequently, the bulk of bright pixels in any given scene will be the result of just one of these causes. Thus if the analyst is calculating a single sky feature percentage, he must decide for each scene whether to include bright pixels as clouds or clear. The surest way is to study the sky picture file, but this does not lend itself to automation. From the study of many such images, it has been discovered that a reasonable estimate can be made in the following manner. First, if the moon is not in the sky and with a sufficient

portion of the disk illuminated, then the number of bright pixels will be small but difficult to automatically reclassify. With sufficient moonlight, if the bright are interspersed with clear, then most likely aerosol is causing that classification and those pixels can be considered clear. On the other hand and again with sufficient moonlight, if the bright are interspersed with opaque, then they can be reasonably considered as brightly lit clouds.

The sky regions in Table 11 used in the array `class_sums` are defined in a special manner to accommodate the user who does not desire to include feature retrievals made near the horizon in his study. While radiances near the horizon are extremely useful, feature retrievals are problematic do to the air masses involved. Since the lower quadrants in regions 6 to 9 only extend downwards to 80° in zenith angle, a proper summation of regions 1 to 9 would give sky feature percentages that do not include the last 10° near the horizon. Of course region 0 includes the entire hemisphere. The data in this file are straightforward to use. A useful plot is the average and standard deviation versus time. Such plots show unique signatures for clear, aerosol, and a variety of cloudy conditions. For example, cirrus clouds are very bright and have large deviations, the amount of aerosol in a clear sky is proportional to the brightness at zenith, and status clouds have low brightness with a deviation considerably smaller than cirrus.

IDL handles character variables in NetCDF files as ASCII encoded byte strings of fixed length, hence the extra dimension variables in Table 19. If one is having trouble matching a search character string to a value retrieved from a NetCDF character variable, it is best to print out the associated character string one time to discover where the blanks and nulls are inserted to pad out to the fixed length. Then the search string can be modified appropriately. Also remember that character variables are case sensitive.

6.2.6 Limitations

An analyst should be very careful in how the sky feature data is interpolated. In the section above ([Usage Hints](#); [Sky feature Summary File](#); [Chapter 6: Summary Products](#)), a method is suggested for the effective spatial interpolation of sky feature percentages as found in `class_sums` to bridge regions of indeterminate or undefined. That will introduce little bias. However, temporal interpolation of those same percentages to span times when the Sky Feature Files are not present in the ARM Archive *will* introduce bias. One major cause of the files not being present is a quality down check for improper daytime retrieval of thin clouds (see [Potential Biases; Data Quality Control](#); [Chapter 1: Overview](#)). Thus the `class_sums` data on either side of a time gap may not represent the situation in the time between them.

6.3 Sky Feature Plot File

6.3.1 Purpose

The sky feature summary file is the PNG or GIF graphics and is a visualization of whole-sky-dome data from the sky feature summary file described in [Sky feature Summary File](#); [Chapter 6: Summary Products](#). It is at six-minute time resolution (ten-minute old style, see [Data Collection Strategy](#); [Chapter 1: Overview](#)) for an entire 24-hour Greenwich day. This visualization is useful to the experimenter desiring to understand the cloud dynamics for a day but who isn't concerned with any spatial resolution in the coverage details.

6.3.2 Naming Convention

The sky feature summary file has the following name based on standard ARM naming practices.

`sssffwsicloudsummarypngff.c1.yyyymmdd.hhmmss.png`

or

`sssffwsicloudsummarygiffff.c1.yyyymmdd.hhmmss.gif`

where the blue characters are literal and

`sss ff` = the site and facility identifier as listed in Table 1

`yyymmdd.hhmmss` = the standard 15 character ARM date time stamp showing year, month, day, hour, minute, and second.

Historically, the first data processed was into the GIF format, but all recent data is in the PNG format; the file name indicates the format. Since one file is produced per Greenwich day, there is no corresponding UNIX style tar archive. Also the hours, minutes, and seconds are always set to zero.

6.3.3 Dimensions, Variables, and Coding

Since the sky feature plot files are PNG or GIF rather than NetCDF they do not have dimensions, attributes, and variables in the manner of normal ARM usage.

The plots are portrayed as a bar chart, one bar for each time interval. The abscissa is time and the ordinate percentage. Each bar is subdivided into color coded stacked segments that indicate the various sky feature types; the values of all the segments in one bar always add to 100%. The color coding is listed in Table 21.

Color	Sky Feature
Black	Undefined, e.g., Under the occulter
Brick Red	Indeterminate; not matching sky model, e.g., palm tree
Dark Blue	Near pristine clear sky
Light Blue	Aerosol laden clear sky
Magenta	Mixed aerosol and cloud features
White	Bright cloud with blue component more than 3× clear sky for spectral retrieval or bright sky with background more than 3× star flux for optical density retrieval
Light Gray	Intermediate cloud with blue component between 1× and 3× clear sky
Dark Gray	Dark cloud with blue component less than clear sky
Light Green	Translucent cloud or aerosol
Dark Green	Opaque cloud

6.3.4 Algorithm

The basis for the sky feature plot file is the sky feature summary file that is discussed in [Sky Feature Summary File; Chapter 6: Summary Products](#). It is created as a final step in the WsiIdpc6_vx program that creates that summary file (x represents the version number which is 8 as of this writing). Data from the hemispheric sky dome is used – region 0 of Table 11 – in the plotting to a z-buffer that is later transformed into the PNG file complete with color tables.

6.3.5 Usage Hints

The sky feature plot is most useful as a quick means of understanding the dynamics of the sky during an entire day. It is also useful as an image to include in presentations.

A sample is shown in Figure 5 for the Southern Great Plains site in Oklahoma on January 15, 2003. It is to be interpreted as a series of 240 vertical bars, each color coded with sky features for its particular time. The segments associated with the different sky features are always stacked in the same order – from bottom to top: clear, aerosol, mixed aerosol and cloud, bright and bright cloud, intermediate cloud, dark cloud, translucent, opaque, indeterminate, and finally undefined. For example, the vertical bar at 0500 hours indicates that the sky is approximately 70% clear, $85 - 70 = 15\%$ bright, $87 - 85 = 2\%$ opaque, $88 - 87 = 1\%$ indeterminate, and $100 - 88 = 12\%$ undefined. The Oklahoma time zone is GMT-6, so the dark bands at 0000 and 2400 hours occur during sunset and the one centered at 1330 hours occurs at sunrise.

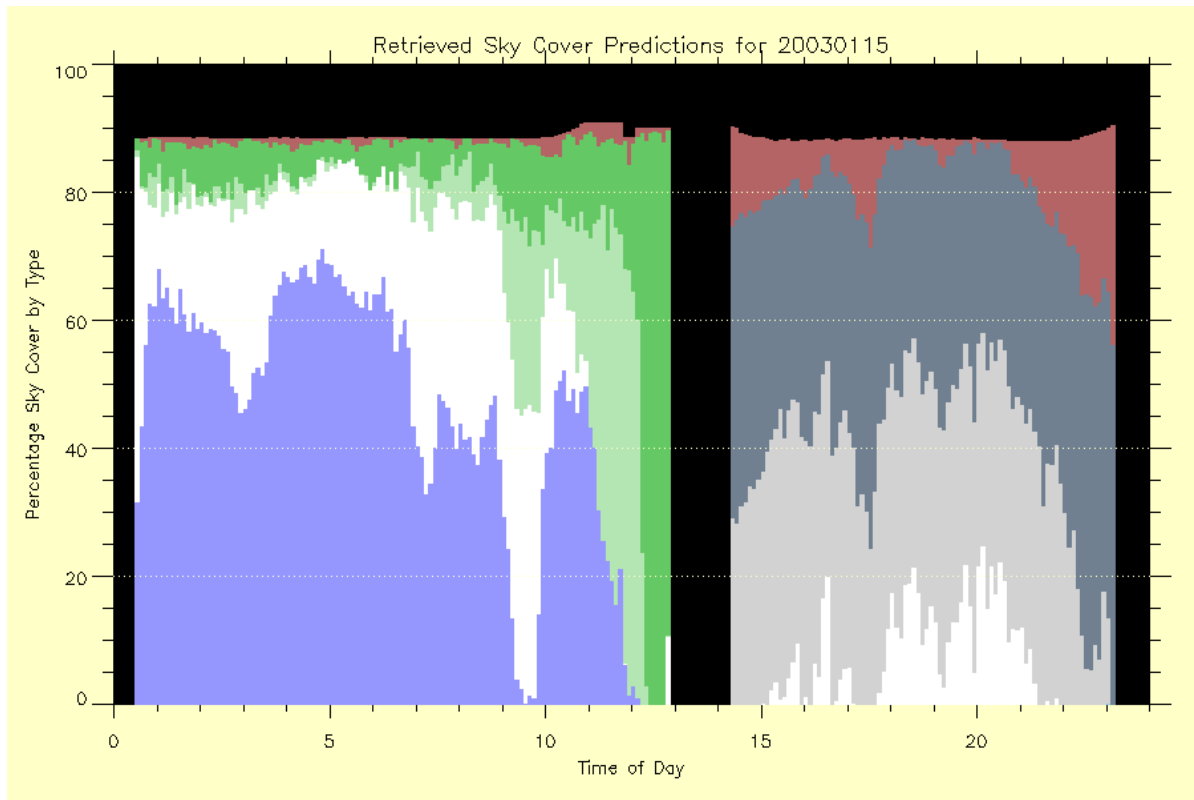


Figure 5. Sample sky feature plot file.

The retrievals at these times are not performed as the twilight spectral retrieval is not implemented (see [Twilight Spectral Retrieval; Algorithm; Sky Feature File; Chapter 5: Sky Feature Product](#)). Between 0030 and 1256 hours the results from nighttime optical density retrievals are presented. On this particular night there was a waxing gibbous moon with 92% visible disk illumination; it rose during the previous day and set at 1120 hours. The bright moonlight illuminates atmospheric aerosol enough in some cases to cause a pixel to be identified as bright. Using the technique discussed in [Usage Hints; Sky Feature File; Chapter 5: Sky Feature Product](#), the bright percentages between 0030 and 0900 hours should be identified as part of the clear sky. An optically thin band moves across the sky between 0900 and 1000 hours, followed by a similarly thin precursor beginning at 1100 hours to the heavy overcast that completely covers the sky for the rest of the day. The considerable percentage of indeterminate toward the end of the day is a result of the extreme darkness of the clouds. The constants in Table 14 could be adjusted to cause these to be identified as dark cloud; see the discussion in [Daytime Spectral Retrieval; Algorithm; Sky featureFile; Chapter 5: Sky Feature Product](#). As can be seen from the top black-colored segment at the top of each bar, a bit more than 10% of each sky scene is undefined. As discussed in detail in the section just referenced, it may be important to renormalize the other percentages to eliminate such undefined, and perhaps even the indeterminate.

6.3.6 Limitations

The quality control process will cause suspect radiance and sky feature retrieval files to be deleted from the archive record upon which this sky feature plot file is built. Therefore there may not be a complete set for any given period, and the vertical bar for any missing time will be completely black.

7. Calibration Product

There is a single calibration product, the NetCDF calibration file.

7.1 Calibration File

7.1.1 Purpose

The calibration file provides critical calibration and instrument configuration information to the various retrievals described in this document; examples are the absolute calibration coefficients and the occulter width. This file also provides extremely useful information to the user of Whole Sky Imager data; an example is the pixel-to-angle coordinate transformation. A separate “current” file is maintained for each deployed instrument; it will be replaced aperiodically as instrument calibration, mounting, or hardware features change.

7.1.2 Naming Convention

The calibration file has the following name based on standard ARM naming practices.

`sssffwsicalff.c1.yyyyymmdd.hhmmss.cdf.yyyyymmdd.hhmmss.vv`

where the blue characters are literal and

`sss ff` = the site and facility identifier as listed in Table 1,

yyyymmdd.hhmmss = the standard 15 character ARM date-time stamp showing year, month, day, hour, minute, and second

vv = the version number of the file.

The first date-time group is the first-valid date, and the second is the last-valid date. The most current calibration file will have fourteen nines for the last-valid data; when that file is superseded then it will be renamed with a last-valid at the date of its being supplanted. The set of calibration files for a given instrument have non-overlapping time-date information, i.e., any date will fall within the first-valid to last-valid interval of only one calibration file, excluding versions of the same file. It is possible for a date to fall outside the range of any calibration file, but that happens only when the date overlaps an extended period of inoperability. The version numbers are counted from one and are incremented as the file for a given period is updated for any reason; only the latest numbered file should be used unless the user is matching to a file used for a particular historical retrieval.

7.1.3 Dimensions, Variables, and Coding

7.1.3.1 Global Attributes

The global attributes in the calibration file contain useful background information on the instrument and production environment. The user should always extract them with a suitable NetCDF dump routine and become familiar with their content. As the names and values are purposely designed to be human readable, they are not listed separately herein.

7.1.3.2 Dimensions

The dimensions associated with the sky feature summary file are listed in Table 22 and discussed in the paragraphs below.

Number_stars is the unlimited dimension but does not represent time steps in the data as is typical in NetCDF files; it is the number of stars for which this file contains data.

Number_image_cols is the number of columns in a raw data collection. It has the value of 512 which matches the number of columns in the camera CCD array.

Number_image_rows is the number of rows in a raw data collection. It has the value of 512 which matches the number of rows in the camera CCD array.

Number_dark_fit_constants is the order of the polynomial plus one that is used to approximate the CCD dark current for each pixel.

Number_spectral_filters is the number of holes in the spectral filter changer wheel; there are four holes in the filter changer currently deployed.

Number_neutral_filters is the number of holes in the neutral density filter changer wheel; there are four holes in the filter changer currently deployed.

Table 22. Calibration file dimensions.		
ID	Name	Size
0	Number_stars	Unlimited
1	number_image_cols	512
2	number_image_rows	512
3	number_dark_fit_constants	2
4	number_spectral_filters	4
5	number_neutral_filters	4
6	number_zenith_angles	91
7	number_azimuth_angles	361
8	number_rad_table_solar_zeniths	21
9	number_rad_table_solar_azimuths	37
10	number_rad_table_zeniths	19
11	number_rad_table_colors	3
12	number_occ_rungs	5
13	number_poles	2
14	char_2	2
15	char_8	8
16	char_20	20

Number_zenith_angles is the dimension of the arrays in the zenith direction that will contain the pixel-coordinate to sky-coordinate conversions. The current value is 91, which corresponds to a angular resolution of 1° throughout the range 0°–90°.

Number_azimuth_angles is the dimension of the arrays in the the azimuth direction that will contain the pixel-coordinate to sky-coordinate conversions. The current value is 361, which corresponds to an angular resolution of 1° throughout the range 0°–360°.

Number_rad_table_solar_zeniths is the size of the four-dimensional pristine sky radiation table in the solar zenith angle dimension. The current value is 21, indicating that radiation data is stored for 21 different solar zenith angles.

Number_rad_table_azimuths is the size of the four-dimensional pristine sky radiation table in the viewer azimuth angle dimension. The current value is 37, indicating that radiation data is stored for 37 different azimuth angles from the viewer’s point of view.

Number_rad_table_zeniths is the size of the four-dimensional pristine sky radiation table in the viewer zenith angle dimension. The current value is 19, indicating that radiation data is stored for 19 different zenith angles from the viewer’s point of view.

Number_rad_table_colors is the size of the four-dimensional pristine sky radiation table in the spectral band dimension. The current value is 3, indicating that radiation data is stored for 3 different spectral bands.

Number_occ_rungs is the number of visible rungs on the occulter “ladder” as pictured in a raw data image. This value is currently five; the new occulter to be first deployed in the Southern Great Plains in October 2003 does not have occulter rungs, so this value will be irrelevant in files for instruments equipped with it (see [Evolution; Features of the Instrument; Chapter 1: Overview](#)).

Number_poles is the number of places near the horizon where instrument hardware protrudes into the camera’s field-of-view. The current value is two corresponding to the occulter pivot points to the north and south. The new occulter to be first deployed in the Southern Great Plains in October 2003 has pivot points below the camera’s horizon, so this value will be irrelevant in files for instruments equipped with it (see [Evolution; Features of the Instrument; Chapter 1: Overview](#)).

The other dimensions (**char_2**, **char_8**, and **char_20**) are necessary in the development of the NetCDF file but are of no particular concern to its user.

7.1.3.3 Variables

The variables associated with the calibration file are listed in the next table and discussed in the paragraphs below. The attributes associated with each variable contain valuable information; the user should extract them with a suitable NetCDF dump routine and become familiar with their content. In particular the attributes may have null value, range, and allowed delta information.

The **generic_dark** variable – long name: **generic dark image** – is a three-dimensional array of type float that contains coefficients for a polynomial that predicts dark counts for each pixel in an image based on exposure time. Since the dark counts are different for each pixel of the camera’s CCD array, there are $512 \times 512 = 262144$ sets of polynomial coefficients, one for each pixel. The first two dimensions of **generic_dark** correspond respectively to the column and row of the pixel of interest using zero based counting. The third dimension is the order of the polynomial minus one. To generate dark counts for a pixel the following equation should be used:

$$D[c,r] = g[c,r,0] + E \times g[c,r,1] + E^2 \times g[c,r,2] + \dots$$

- where D is the calculated dark counts
- g is the generic_dark array
- E is the exposure time in milliseconds
- c is the zero based column number of the pixel of interest
- r is the zero based row number of the pixel of interest.

The dark count values generated by these means should only be used if a suitable raw dark image is not available; such an image would differ from this calculation as a result of minor temporal variations in the camera’s physical environment and by aging of the CCD array and associated electronics. See [Calibration; Algorithm; Radiance Data File; Chapter 4: Radiance Product](#) for a description of the application of dark counts. The source data for this variable derive from averages of raw dark data taken at the instrument’s deployed site.

Table 23. Calibration file variables.

ID	Name	Type	Attribute Count	Dimension Count	Dimension IDs	Actual Dimensions
0	generic_dark	Float	9	3	1, 2, 3	512, 512, 2
1	flat_field	Float	8	2	1, 2	512, 512
2	roll_off	Float	9	3	1, 2, 4	512, 512, 4
3	cal_constant	Float	10	2	4, 5	4, 4
4	pixel_to_zenith	Float	9	2	1, 2	512, 512
5	pixel_to_azimuth	Float	9	2	1, 2	512, 512
6	angle_to_column	Float	9	2	6, 7	91, 361
7	angle_to_row	Float	9	2	6, 7	91, 361
8	center_col	Float	7	0	--	--
9	center_row	Float	7	0	--	--
10	rad_table_solar_zeniths	Float	6	1	8	21
11	rad_table_azimuths	Float	6	1	9	37
12	rad_table_zeniths	Float	6	1	10	19
13	rad_table	Float	9	4	11, 8, 9, 10	3, 21, 37, 19
14	occ_radius	Float	7	0	--	--
15	occ_rail_distance	Float	7	0	--	--
16	occ_rail_width	Float	7	0	--	--
17	occ_rotation	Float	7	0	--	--
18	occ_rung_angles	Float	7	1	12	5
19	occ_rung_width	Float	7	0	—	—
20	trolley_type	Float	6	1	15	8
21	trolley_width	Float	8	0	--	--
22	trolley_center	Float	8	0	--	--
23	trolley_length	Float	8	0	--	--
24	trolley_radius	Float	8	0	--	--
25	pole_cutoff	Float	7	1	13	2
26	source_cutoff	Float	7	0	--	--
27	horizon_cutoff	Float	7	0	--	--
28	horizon_mask	Byte	9	2	1, 2	512, 512
29	star_name	Char	5	2	16, 0	20, x
30	star_type	Char	6	2	15, 0	8, x
31	star_spectrum	Char	5	2	14, 0	2, x
32	star_ra	Float	7	1	0	x
33	star_dec	Float	7	1	0	x
34	star_mag	Float	7	1	0	x
35	star_width	Float	8	0	--	--
36	lat	Float	7	0	--	--
37	lon	Float	7	0	--	--
38	alt	Float	6	0	--	--

The **flat_field** variable – long name: normalized flat field image – is a two-dimensional array of type float that contains a calibration correction for CCD and fiber optic irregularities. The two dimensions correspond respectively to the column and row of the pixel of interest using zero based counting. The same flat_field correction is multiplicatively applied to all images regardless of spectral band, attenuation, or exposure time. See [Calibration; Algorithm; Radiance Data File; Chapter 4: Radiance Product](#) for a description of the application of flat field correction. The source data for this variable derive from laboratory or field calibrations.

The **roll_off** variable – long name: normalized intensity roll off with zenith angle – is a three-dimensional array of type float that contains a calibration correction for optic and filter efficiencies that are zenith angle dependent, for example the darkening of a fisheye image near the horizon due to lowered optical efficiency. The first two dimensions correspond respectively to the column and row of the pixel of interest using zero based counting. The third dimension corresponds to spectral band indexed in the same order as the raw file header variable “spectral filter” (see [Header Information; Dimensions, Variables, and Coding; Raw Data Files, Chapter 2: Raw Products](#)) except the range is (0, 3) for this dimension and (1, 4) for the raw header. The most used, but not guaranteed, order is [800 nm, null-filter, 650 nm, 450 nm] for [0, 1, 2, 3] respectively. The same spectral band roll_off correction is multiplicatively applied to all images regardless of attenuation or exposure time. See [Calibration; Algorithm; Radiance Data File; Chapter 4: Radiance Product](#) for a description of the application of roll off correction. The source data for this variable derive from laboratory or field calibrations.

The **cal_constant** variable – long name: calibration constant – is a two-dimensional array of type float that contains scalar absolute calibration constants. The constants change as different combinations of spectral and neutral density filters are inserted in the optical train, hence there is one constant for each such combination. The first dimension corresponds to spectral band indexed in the same order as the raw file header variable “spectral filter” (see [Header Information; Dimensions, Variables, and Coding; Raw Data Files, Chapter 2: Raw Products](#)) except the range is (0, 3) for this dimension and (1, 4) for the raw header. The most used, but not guaranteed, order is [800 nm, null-filter, 650 nm, 450 nm] for [0, 1, 2, 3] respectively. The second dimension corresponds to neutral density attenuation indexed in the same order as the raw file header variable “neutral filter” except, like the situation above, the range is (0, 3) for this dimension and (1, 4) for the raw header. The most used, but not guaranteed, order is [null-filter, log 2, log 3, null-filter] for [0, 1, 2, 3] respectively. The same spectral and neutral density filter dependent calibration constant is multiplicatively applied to all images regardless of exposure time. See [Calibration; Algorithm; Radiance Data File; Chapter 4: Radiance Product](#) for a description of the application of roll off correction. The source data for this variable derive from laboratory or field calibrations.

The **pixel_to_zenith** variable – long name: pixel to zenith angle conversion – is a two-dimensional array of type float that contains the zenith sky angle in decimal degrees imaged by each pixel as identified by column and row. The two dimensions correspond respectively to the column and row of the pixel of interest using zero based counting. Pixel_to_zenith has the range 0° to 90° and is measured from the point straight above the instrument so that 90° is the horizon. The source data for this variable derive from viewing a star field at a known time.

The **pixel_to_azimuth** variable – long name: pixel to azimuth angle conversion – is a two-dimensional array of type float that contains the azimuth sky angle in decimal degrees imaged by each pixel as identified by column and row. The two dimensions correspond respectively to the column and row of the pixel of interest using zero based counting. Pixel_to_azimuth has the range 0° to 360° and

is measured clockwise from geographic north such that east is 90° . The source data for this variable derive from viewing a star field at a known time.

The **angle_to_column** variable – long name: point zenith and azimuth angle to pixel column conversion – is a two-dimensional array of type float that contains the fractional column number upon which the image of a point in the sky identified by zenith and azimuth angles will fall. The first dimension corresponds to the whole number zenith angle in degrees, and the second to the whole number azimuth angle in degrees. The ranges are (0, 90) and (0, 360) respectively. Zenith angle is measured from the vertical point downward to the horizon and azimuth is measured from geographic north in a clockwise direction, i.e., toward east. The column number is zero based and has a fractional component since, in general, the image of point in the sky that has whole numbers for its zenith and azimuth angles will not be directly centered on a CCD chip column of pixels. The source data for this variable derive from viewing a star field at a known time.

The **angle_to_row** variable – long name: point zenith and azimuth angle to pixel row conversion – is a two-dimensional array of type float that is the row analogue of the **angle_to_column** variable; the various points in the discussion of that variable apply to this one as well. The source data for this variable derive from viewing a star field at a known time.

The **center_col** variable – long name: center column – is a scalar of type float that contains the fractional column number upon which the image of the zenith point in the sky will fall. The column number is zero based and has a fractional component since, in general, the image of the zenith point will not be directly centered on a CCD chip column of pixels. The source datum for this variable derives from viewing a star field at a known time.

The **center_row** variable – long name: center row – is a scalar of type float that is the row analogue of the **center_col** variable; the various points in the discussion of that variable apply to this one as well. The source datum for this variable derives from viewing a star field at a known time.

The **rad_table_solar_zeniths** variable – long name: clear sky radiance table solar zenith angles – is a vector of type float that contains the key to the second dimension of the variable **rad_table**. The values represent zenith angles of the sun in angular degrees and have the range of -10° to 90° , where the negative value indicates and extension on the arc past zenith. See the discussion of **rad_table** for a more complete description. The source data for this variable derive from arbitrary definition within the specified range.

The **rad_table_azimuths** variable – long name: clear sky radiance table azimuth angles – is a vector of type float that contains the key to the third dimension of the variable **rad_table**. The values represent azimuth angles relative to the sun of the viewed point in the sky in angular degrees. The range is (0° , 180°) and is to be interpreted as both clockwise and counterclockwise due to the symmetry involved. See the discussion of **rad_table** for a more complete description. The source data for this variable derive from arbitrary definition within the specified range.

The **rad_table_zeniths** variable – long name: clear sky radiance table zenith angles – is a vector of type float that contains the key to the fourth dimension of the variable **rad_table**. The values represent zenith angles relative to the sun of the viewed point in the sky in angular degrees. The range is (0° , 90°).

See the discussion of `rad_table` for a more complete description. The source data for this variable derive from arbitrary definition within the specified range.

The **rad_table** variable – long name: clear sky radiance table – is a four-dimensional array of type float that contains the calculated clear sky radiance values for different spectral bands in a sparse grid that spans the sky and relevant solar positions. The first dimension identifies layers in the array containing radiances by spectral band. The bands are fixed as the set {450 nm, 650 nm, 800 nm} each with a 70 nm full width corresponding in order to the index set {0, 1, 2}. The second dimension identifies layers in the array containing radiances associated with various solar angles. The particular solar angle for this type of layer is found in the vector `rad_table_solar_zeniths` at the index that matches the index of this second dimension. The third dimension identifies layers in the array containing radiances associated with the azimuth angles of various viewed elements of the sky. The particular viewed azimuth angle for this type of layer is found in the vector `rad_table_azimuths` at the index that matches the index of this third dimension. The fourth dimension identifies layers in the array containing radiances associated with the zenith angles of various viewed elements of the sky. The particular viewed zenith angle for this type of layer is found in the vector `rad_table_zeniths` at the index that matches the index of this fourth dimension. The array values have the units milliwatts per meter squared (sensor area) per steradian (sky area) per nanometer (bandwidth). This array is used in the spectral sky cover routine for reference radiances; see [Daytime Spectral Retrieval; Algorithm; Sky Feature File; Chapter 5: Sky Feature Product](#). The source data for this variable derive from the SBDART radiation model.

The **occ_radius** variable – long name: occulter radius – is a scalar of type float that contains the radius in centimeters from the optical center of the lens assembly to the inner edge of the occulter arc. It is used in calculating the portion of the mask that is the image of the occulter; see [Mask Generation; Algorithm; Radiance Data File; Chapter 4: Radiance Product](#). The source datum for this variable derives from a physical measurement.

The **occ_rail_distance** variable – long name: occulter spacing between rails – is a scalar of type float that contains the spacing in centimeters of the two occulter rails from rail center to rail center for the original occulter design. In calibration files for instruments outfitted with the new occulter (see [Evolution; Features of the Instrument; Chapter 1: Overview](#)) this scalar is set to 0.0. It is used in calculating the portion of the mask that is the image of the occulter; see [Mask Generation; Algorithm; Radiance Data File; Chapter 4: Radiance Product](#). The source datum for this variable derives from a physical measurement.

The **occ_rail_width** variable – long name: occulter rail width – is a scalar of type float that contains the width in centimeters of a single occulter rail. It is used in calculating the portion of the mask that is the image of the occulter; see [Mask Generation; Algorithm; Radiance Data File; Chapter 4: Radiance Product](#) and will generally be wider than the physical width of the rail to allow for all affected pixels in the raw data near the edge of the image of the rail. The source datum for this variable derives from a physical measurement as modified by data analysis.

The **occ_rotation** variable – long name: occulter axis rotation – is a scalar of type float that contains the angle in decimal degrees between the occulter axis as imaged on the camera's CCD array and the alignment of columns in that array. `Occ_rotation` has the range of -90° to 90° with positive values indicating that the northern end of the occulter axis is appears east of a vector aligned with the image

column from the axis' southern end. It is used in calculating the portion of the mask that is the image of the occulter; see [Mask Generation; Algorithm; Radiance Data File; Chapter 4: Radiance Product](#). The source datum for this variable derives from raw data analysis.

The **occ_rung_angles** variable – long name: occulter rung angles – is a vector of type float that contains the position in angular degrees of the five occulter rungs for the original occulter design. The value is the angle between a vector from the optical center of the lens assembly toward the north and parallel to the occulter pivot axis and a vector from the optical center to the physical center of a rung. In calibration files for instruments outfitted with the new occulter which does not have rungs (see [Evolution; Features of the Instrument; Chapter 1: Overview](#)) this vector is zero filled. It is used in calculating the portion of the mask that is the image of the occulter; see [Mask Generation; Algorithm; Radiance Data File; Chapter 4: Radiance Product](#). The source datum for this variable derives from a physical measurement.

The **occ_rung_width** variable – long name: occulter rung width – is a scalar of type float that contains the largest width in centimeters of the five occulter rungs for the original occulter design. In calibration files for instruments outfitted with the new occulter which does not have rungs (see [Evolution; Features of the Instrument; Chapter 1: Overview](#)) its value is zero. It is used in calculating the portion of the mask that is the image of the occulter; see [Mask Generation; Algorithm; Radiance Data File; Chapter 4: Radiance Product](#) and will generally be wider than the physical width of any rung to allow for all affected pixels in the raw data near the edge of the image of the rung. The source datum for this variable derives from a physical measurement as modified by data analysis.

The **trolley_type** variable – long name: trolley type – is a scalar of type character that indicates the type of trolley installed on an instrument. It has a value from the set {fixed, moveable}. The moveable trolley is a small disk that maintains its proper optical dome shading position by rolling along the occulter arc; the fixed trolley is a strip affixed to the arc of sufficient length to shade the dome for all possible sun and moon positions at the instrument's location. Both apply to the old and new occulter designs. It is used in calculating the portion of the mask that is the image of the trolley; see [Mask Generation; Algorithm; Radiance Data File; Chapter 4: Radiance Product](#). The source datum for this variable derives from a physical observation.

The **trolley_width** variable – long name: fixed trolley width – is a scalar of type float that contains the full width in centimeters of a fixed trolley strip, i.e., the strip dimension perpendicular to the occulter plane. In calibration files for instruments outfitted with a moveable trolley its value is -9999.0. It is used in calculating the portion of the mask that is the image of the trolley; see [Mask Generation; Algorithm; Radiance Data File; Chapter 4: Radiance Product](#) and will generally be wider than the physical width of the strip to allow for all affected pixels in the raw data near the edge of the image of the strip. The source datum for this variable derives from a physical measurement as modified by data analysis.

The **trolley_center** variable – long name: fixed trolley center – is a scalar of type float that locates the center of a fixed trolley strip in angular degrees. The value is the angle between a vector from the optical center of the lens assembly toward the north and parallel to the occulter pivot axis and a vector from the optical center to the physical center of fixed trolley strip which lies in the occulter arc plane. In calibration files for instruments outfitted with a moveable trolley its value is -9999.0. It is used in

calculating the portion of the mask that is the image of the trolley; see [Mask Generation; Algorithm; Radiance Data File; Chapter 4: Radiance Product](#). The source datum for this variable derives from a physical measurement.

The **trolley_length** variable – long name: fixed trolley length – is a scalar of type float that contains the full length in angular degrees of a fixed trolley strip, i.e., the strip dimension parallel to the occulter plane. The value is the angle between a vector in the occulter plane from the optical center of the lens to one end of the occulter strip and a similar vector to the other end. In calibration files for instruments outfitted with a moveable trolley its value is -9999.0. It is used in calculating the portion of the mask that is the image of the trolley; see [Mask Generation; Algorithm; Radiance Data File; Chapter 4: Radiance Product](#) and will generally be wider than the physical length of the strip to allow for all affected pixels in the raw data near the end of the image of the strip. The source datum for this variable derives from a physical measurement as modified by data analysis.

The **trolley_radius** variable – long name: moveable trolley radius – is a scalar of type float that contains the radius in angular degrees of a moveable trolley disk. The value is the angle between a vector from the optical center of the lens to the center of the trolley disk and a similar vector any point on the edge of the disk. In calibration files for instruments outfitted with a fixed trolley its value is -9999.0. It is used in calculating the portion of the mask that is the image of the trolley; see [Mask Generation; Algorithm; Radiance Data File; Chapter 4: Radiance Product](#) and will generally be wider than the physical radius of the trolley disk to allow for all affected pixels in the raw data near the edge of its image. The source datum for this variable derives from a physical measurement as modified by data analysis.

The **pole_cutoff** variable – long name: pole cutoff – is a vector of type float that indicates the size of the two regions in angular decimal degrees near the poles of the image which may contain the image of occulter pivot hardware. The value of element zero for the north pole is the angle between a vector from the optical center of the lens to the sky point with zenith angle of 90° and azimuth angle of 0° and a similar vector to a sky point on the edge of the semicircular region circumscribing the corrupting image influences. The value of element one for the south pole is similarly defined but for a sky point with zenith angle of 90° and azimuth angle of 180° . In calibration files for instruments outfitted with the new occulter whose occulter pivot hardware is below the camera's field-of-view (see [Evolution; Features of the Instrument; Chapter 1: Overview](#)) their values are zero. Pole_cutoff is used in calculating the portion of the mask associated with horizon features; see [Mask Generation; Algorithm; Radiance Data File; Chapter 4: Radiance Product](#). The source datum for this variable derives from raw data analysis.

The **source_cutoff** variable – long name: sun or moon cutoff – is a scalar of type float that indicates the size of the region in angular decimal degrees near the identified light source (sun or moon) in the image which is corrupted by that source. The value is the angle between a vector from the optical center of the lens to the location of the sun or moon and a similar vector to a sky point on the edge of the circular region circumscribing affected region. Source_cutoff is used in calculating the portion of the mask associated with light source features; see [Mask Generation; Algorithm; Radiance Data File; Chapter 4: Radiance Product](#). The source datum for this variable derives from raw data analysis.

The **horizon_cutoff** variable – long name: horizon cutoff – is a scalar of type float that indicates the size of the band in angular decimal degrees near the horizon of the image which is to be ignored in radiance and sky feature retrievals. The value is the zenith angle at the top of the band. If a detailed **horizon_mask** (see the variable defined just below) is provided, then this variable often is set to 90° indicating no cutoff. **horizon_cutoff** is used in calculating the portion of the mask associated with horizon features; see [Mask Generation; Algorithm; Radiance Data File; Chapter 4: Radiance Product](#). The source datum for this variable derives from raw data analysis and personal preference.

The **horizon_mask** variable – long name: horizon mask – is a two-dimensional array of type byte that contains a mask for the screening of non-sky near horizon features such as buildings and foliage. The two dimensions correspond respectively to the column and row of any raw image data using zero based counting. **horizon_mask** is either zero indicating that the data for that pixel will be affected by horizon features or one indicating a clear view of the sky. The same **horizon_mask** is used in building the image mask regardless of spectral band, attenuation, or exposure time of the raw data involved; see [Mask Generation; Algorithm; Radiance Data File; Chapter 4: Radiance Product](#). The source data for this variable derive from raw data analysis.

The **star_name** variable – long name: star name – is a vector of type character that contains the names of the stars to be considered in the optical density retrieval, e.g., Rigil Kentaurus, Arcturus, and Canopus; see [Nighttime Optical Density Retrieval; Algorithm; Sky Feature File; Chapter 5: Sky Feature Product](#). The vector length corresponds to the number of stars. Most calibration files as of this writing contain information for 1377 stars of magnitude 5.0 or less, i.e., brighter. The source data for this variable is the Yale Bright Star Catalog.

The **star_type** variable – long name: star type – is a vector of type character that contains the types of the stars to be considered in the optical density retrieval; see [Nighttime Optical Density Retrieval; Algorithm; Sky Feature File; Chapter 5: Sky Feature Product](#). This variable will assume one of the values of the set {multiple, double, single, binary, variable, starlike, unknown}. The vector length corresponds to the number of stars and is in the same order as **star_name**. The source data for this variable is the Yale Bright Star Catalog.

The **star_spectrum** variable – long name: star spectra – is a vector of type character that contains the spectral codes of the stars to be considered in the optical density retrieval, e.g., G2, K1, and F0; see [Nighttime Optical Density Retrieval; Algorithm; Sky Feature File; Chapter 5: Sky Feature Product](#). The vector length corresponds to the number of stars and is in the same order as **star_name**. The source data for this variable is the Yale Bright Star Catalog.

The **star_ra** variable – long name: star right ascension – is a vector of type float that contains the right ascension values in decimal hours of the stars to be considered in the optical density retrieval; see [Nighttime Optical Density Retrieval; Algorithm; Sky Feature File; Chapter 5: Sky Feature Product](#). The vector length corresponds to the number of stars and is in the same order as **star_name**. The source data for this variable is the Yale Bright Star Catalog.

The **star_dec** variable – long name: star declination – is a vector of type float that contains the declination values in decimal degrees of the stars to be considered in the optical density retrieval; see [Nighttime Optical Density Retrieval; Algorithm; Sky Feature File; Chapter 5: Sky Feature Product](#). The

vector length corresponds to the number of stars and is in the same order as `star_name`. The source data for this variable is the Yale Bright Star Catalog.

The **star_mag** variable – long name: star magnitude – is a vector of type float that contains the magnitudes of the stars to be considered in the optical density retrieval; see [Nighttime Optical Density Retrieval; Algorithm; Sky Feature File; Chapter 5: Sky Feature Product](#). The vector length corresponds to the number of stars and is in the same order as `star_name`. The 1377 star magnitudes in most calibration files as of this writing range from -1.46 for the brightest star Sirius to 5.00. The source data for this variable is the Yale Bright Star Catalog.

The **star_width** variable – long name: star image width – is a scalar of type float that contains the average standard deviation in column and row for many two-dimensional Gaussian distributions fit to the images of stars in raw null-filter image data. The blurring that this variable represents is a result of atmospheric and optical phenomena. `star_width` is used in the optical density retrieval; see [Nighttime Optical Density Retrieval; Algorithm; Sky Feature File; Chapter 5: Sky Feature Product](#). The source data for this variable is image analysis.

The **lat** variable – long name: north latitude – is a scalar of type float that contains the geographical latitude in decimal degrees of a fixed base instrument's location. Latitudes range between -90° and $+90^\circ$, where -90° is at the Earth's South Pole, 0° is at the equator, and 90° is at the North Pole. For moveable base instruments (the one on SHEBA ship) this variable is set to its null value of -9999.0. `lat` is used in calculating sun and moon positions for the radiance and sky feature retrievals. The source data for this variable is a physical measurement.

The **lon** variable – long name: east longitude – is a scalar of type float that contains the geographical longitude in decimal degrees of a fixed base instrument's location. Longitudes range between -180° and $+180^\circ$, where 0° is the meridian through Greenwich, England, positive values are in the eastern hemisphere, and negative values are in the western hemisphere. For moveable base instruments (the one on SHEBA ship) this variable is set to its null value of -9999.0. `lon` is used in calculating sun and moon positions for the radiance and sky feature retrievals. The source data for this variable is a physical measurement.

The **alt** variable – long name: altitude – of type float is a scalar that specifies the geographical altitude of the instrument's location. It is expressed in meters above mean sea level. The source data for this variable is a physical measurement.

7.1.4 Algorithm

7.1.4.1 Overview

There are several computer codes that contribute to the formation of a calibration file that are written and maintained variously by the Marine Physical Laboratory (the Whole Sky Imager developer) and the instrument mentor; none are generally available, but special requests for access will be considered by the instrument mentor. The data produced by these codes is collected and formed into an actual calibration file by the IDL procedure `WsiIdpc1_x`, where the `x` represents a multi-character ad hoc identifier for the variant of this procedure used to produce a given file; there is one variant for each file with new ones

being produced continually. In this section, an overview will be presented of the manipulations necessary to produce data ready for WsiIdpc1_x assembly into the calibration file; see [Instrument Calibration; Chapter 1: Overview](#) for background information.

7.1.4.2 Radiance Variables

After an instrument is deployed and collecting data, all dark raw files from a typical day (or days) are processed to produce generic_dark values; for a description of this variable see [Variables; Dimensions, Variables, and Coding; Calibration File; Chapter 7: Calibration Product](#). Care is taken that the files represent as wide a range of exposure times as possible, since the resulting values will be coefficients of a polynomial with exposure as the dependent variable. After the dark raw files are culled for any anomalies that would skew the results those remaining, on the order of 30, are used as the basis of 262144 linear fits to find the generic_dark coefficients for each pixel. The IDL function “linfit” is used which fits the paired data {exposure-time_i, raw-dark-counts_i} = {x_i, y_i} for i from 1 to about 30 to the linear model $y = A + Bx$ by minimizing the chi-square error statistic where A and B are the generic_dark values for a pixel.

The source of data for variables cal_constant, flat_field, and roll_off (see [Variables; Dimensions, Variables, and Coding; Calibration File; Chapter 7: Calibration Product](#) for a description of these variables) is a physical calibration either in the calibration facility at the Marine Physical Laboratory or by the new Whole Sky Imager Field Calibration Device as reported in [Instrument Calibration; Chapter 1: Overview](#). The details of these processes and the reformatting of the data collected to a form suitable for input to WsiIdpc1_x are discussed elsewhere and are beyond the scope of this document; information can be obtained from the website at <http://www-mpl.ucsd.edu/people/jshields/> and the two documents:

- Shields, J. E., R. W. Johnson, M. E. Karr, D. R. Sauer, and R. A. Weymouth, “Theory of Operations and Interactive Optimization”, University of California San Diego, Scripps Institution of Oceanography, Marine Physical Lab, Technical Note No. 243, September 1996.
- Shields, J. E., R. W. Johnson, M. E. Karr, A. R. Burden, and J. G. Baker, “WSI Field Calibration System Operations Manual”, University of California San Diego, Scripps Institution of Oceanography, Marine Physical Lab, Technical Note No. 252, February 2003.

7.1.4.3 Star Field Variables

Deriving the star field variables involves a complicated multi-step iterative process. These are star_width, center_col, center_row, pixel_to_zenith, pixel_to_azimuth, angle_to_column, and angle_to_row; see [Variables; Dimensions, Variables, and Coding; Calibration File; Chapter 7: Calibration Product](#) for a description of these variables.

1. At its root is the detailed analysis of a single null-filter raw image captured with the instrument in its final position on an extremely clear night with no significant moonlight. A second such image from a different night is used as well to verify the analysis results.
2. The next step is the hand identification by name of about 100 stars in the image, and the determination of the zenith and azimuth angle of each from ephemeris data, the location of the instrument, and the time of image acquisition. The analyst is assisted in this by two IDL programs. The first calculates the sky coordinates from ephemeris data and then plots the position and names of

bright stars on a sky-picture-file-like image using crude estimations for the center position and angle-to-pixel conversions; see [Algorithm; Sky Picture File; Chapter 3: Visualization Products](#) for a description of the image size and orientation. With this plot side by side with a sky picture of null-filter raw data, the analyst then identifies points of light in the raw data with specific stars. Using the second program, he then obtains column and row numbers for each point of light by mouse clicking as close to the center as hand coordination allows. This process is repeated in full for the second, verification raw image. The final products for this step are two spreadsheets, one for each raw image, showing for each bright star its identification, magnitude, calculated zenith and azimuth angles, and observed nearest column and row position.

3. The third step is to refine the whole number column and row position of each star to the nearest 0.01 pixel. The star image is broadened by the atmosphere and optics so that some of its radiance will fall typically into nine to sixteen pixels. If one considers just a row through the image one can imagine the following situations. If one pixel is much brighter than the adjacent pixels then the star's position will be very close to the center of that pixel. However, if two adjacent pixels are equally bright, then the star's position will be half way between them. By mathematically fitting a peaked function to the observed raw counts, the center position for the star can be very accurately determined. This is not as easy, as the raw data being used has background and pixel-to-pixel noise as well as signal shot noise; additionally, some of the bright stars have near neighbors whose peak will intertwine. (For retrieved radiance data, the calibration process reduces these irregularities; see [Calibration; Algorithm; Radiance Data File; Chapter 4: Radiance Product.](#)) An IDL procedure helps automate this process; it loops through all the stars attempting a non-linear least squares fit of a two-dimensional peak function to each by attempting in sequence until success is achieved: an elliptical Gaussian fit within a 9×9 pixel area, an elliptical Gaussian fit within a 7×7 pixel area, an elliptical Gaussian fit within a 5×5 pixel area, and two one-dimensional Gaussian fits of 9 pixel column and row vectors. If none of these four are successful the star is eliminated from the roster. When complete the spreadsheets from step 2 above are updated with fractional column and row positions. Also the mean Gaussian standard deviation for all the surviving stars is `star_width`.
4. Next the image location of the zenith point of the sky must be determined. This is done by brute force and the power of Mathematica using the somewhat valid assumption that the optical elements have perfect cylindrical symmetry. The process is this: (1) assume a center column and row number, (2) calculate the radial distance to each star from that center in pixel units, (3) using all stars as the data set find the coefficients for a least squares fit of a fifth order polynomial in radial distance for zenith angle, (4) calculate the residuals between this fit and the known star zenith angles, (5) find residuals that fall well outside the pattern of the rest and eliminate the associated star from further consideration, and (6) repeat operations 2–5 until no stars need to be eliminated. The stars culled in operation 5 most likely have been misidentified in step 2 above and thereby misrepresented in the spreadsheet. When plotted as a function of azimuth angle the residuals will have an exploitable pattern; those on the side of the image to which the assumed center is offset from the true center will have smaller residuals than those on the other side. In fact the plot will have the appearance of a noisy sine wave of some phase and amplitude with the amplitude being proportional to the offset between the true zenith point and the assumed center. Therefore the operations continue: (7) perform a least squares fit a sine wave of wavelength 360° to the plot of residuals versus true azimuth angle, (8) mark the amplitude of this sine wave as a measure of error for the assumed center point, (9) repeat all of operations 1–8 forty-eight more times with all possible combinations of assumed centers within ± 3 columns and rows of the start point to complete an error grid, and (10) identify the lowest error value in the grid and its associated center point. Now the image location of the zenith point is located to the nearest column and row but tenth column and row accuracy is needed. The process is continued: (11) repeat all of operations 1–8 one hundred

twenty-one more times with all possible combinations of assumed centers within ± 0.5 columns and rows of the point identified in operation 10 but varying by increments 0.1 in each direction, and (12) identify the lowest error value in this new grid and its associated center point. This point now becomes center_col and center_row and is accurate to the nearest 0.1 pixel.

5. Determining an equation to predict sky azimuth angle is the next task. If the columns of the camera's CCD chip were aligned perfectly with geographical north if all the CCD pixels were on a perfectly square grid and if all the optics elements had perfect cylindrical symmetry then the azimuth angle would be simply the arctangent of the column offset from the center point found in step 4 divided by the row offset with consideration for quadrant and range. In practice, the alignment is offset but by a constant additive constant. The grid is reasonably square and requires no correction. The optical train taken as a whole has an elliptical distortion that requires correction; a sine wave with a wavelength of 180° (of azimuth angle) and specified phase and amplitude is sufficient. Note that the equivalent sine wave with 360° wavelength resulted in corrections for center position in step 4. Taking all these factors into consideration, an appropriate equation for azimuth angle is:

$$\phi = \phi_0 + a + b\rho \cos(2\phi_0) + c\rho \sin(2\phi_0)$$

$$\phi_0 = \tan^{-1} \left(\frac{C - C_c}{R - R_c} \right)$$

$$\rho = +\sqrt{(C - C_c)^2 + (R - R_c)^2}$$

where ϕ is the desired azimuth angle,

C and R are the known fractional column and row numbers,

C_c and R_c are the known zenith image fractional column and row, and

a, b, and c are constants.

Considerable effort must be undertaken in the implementing computer code to handle the various singularity, principle range, quadrant, and unit conversion issues that arise in these equations. A Mathematica notebook has been programmed to solve these equations for a, b, and c using a least squares technique. In general, the solution must be iterated after studying the residuals. Any star whose position results in an anomalous residual can have an over powerful affect on the solution, so such are eliminated from consideration and a new solution found. When finished, the azimuth angle residuals are on the order of 0.10° to 0.25° across the entire sky.

6. The obvious next step is to determine an equation to predict sky zenith angle. As with the azimuth angle, if the situation were perfect then this equation would be quite simple – proportional to the distance from the center to the horizon. However, the lens exhibits cylindrically symmetric nonlinear behavior away from the center and the optical train has the elliptical distortion mentioned in step 5 above. A third order polynomial is sufficient to handle the nonlinearity and the same sine form elliptical correction used in azimuth angle correction will work here as well. Taking these into consideration, an appropriate equation for the zenith angle is:

$$\theta = a\rho + b\rho^2 + c\rho^3 + d\rho \cos(2\phi) + e\rho \sin(2\phi)$$

$$\rho = +\sqrt{(C - C_c)^2 + (R - R_c)^2}$$

where θ is the desired zenith angle

ϕ is the azimuth angle calculated in step 5

C and R are the known fractional column and row numbers

C_c and R_c are the known zenith image fractional column and row

a, b, c, d, and e are constants.

As with the azimuth equation, care must be undertaken in the implementing computer code to handle the various principle range, quadrant, and unit conversion issues that arise in these equations. The same Mathematica notebook mentioned in step 5 also solves these equations for a, b, c, d, and e using a least squares technique. As before, any star whose position results in an anomalous residual can have an over powerful affect on the solution, so such are eliminated from consideration and a new solution found by iteration of the least squares fit. When finished, the zenith angle residuals are on the order of 0.10° to 0.25° across the entire sky.

7. At this point half of the conversion problem is solved, i.e., expressing azimuth and zenith angles as functions of column and row numbers. Next the inverse transformation is needed. The usual process would be to solve the pair of equations just found for column and row numbers as expressions of azimuth and zenith angles. This is not easy for transcendental equations; it is easier to derive new equations from further least squares fits. The first step in this process is to derive an expression for radial distance in pixel space from the zenith point image location – equivalent to the ρ in steps 5 and 6. The same considerations must be applied as mentioned in those steps to accommodate the lens's cylindrically symmetric nonlinear behavior away from the center and the optical train's elliptical distortion. Again, a third order polynomial is sufficient to handle the nonlinearity and a sine form of wavelength 180° in azimuth angle is sufficient for the elliptical correction. Taking these into consideration, an appropriate equation for the radial distance angle is:

$$\rho = a\theta + b\theta^2 + c\theta^3 + d\theta \cos(2\phi) + e\theta \sin(2\phi)$$

where ρ is the desired radial distance

ϕ and θ are the known azimuth and zenith angles

a, b, c, d, and e are constants.

Note that the coefficient of the trigonometric functions related to the distance from the zenith image point is the zenith angle itself instead of the more natural but unknown distance in pixel space. As with the above equations, care must be undertaken in the implementing computer code to handle the various principle range, quadrant, and unit conversion issues that arise in these equations. The next section of the Mathematica notebook mentioned earlier solves these equations for a, b, c, d, and e using a least squares technique. As before, any star whose position results in an anomalous residual can have an over powerful affect on the solution, so such are eliminated from consideration and a new solution found by iteration of the least squares fit. When finished, the radial angle residuals average across the entire sky around 0.3 pixels but can be as large as 1.0 pixel.

8. Now that the radial distance to a point is known all that is necessary is the radial's angle in pixel space to calculate desired column and row numbers using simple trigonometric relationships. Unfortunately, this is a bit tricky. The desired answer will be close to the known azimuth angle of the point, as corrected for rotational offset and the optical train's elliptical distortion. The equation will thus be similar in form to that found in step 5; the difficulty is that the argument of the sine and cosine functions contains the desired answer – the radial's angle in pixel space. The equation is:

$$\phi_p = \phi + a + b\rho \cos(2\phi_p) + c\rho \sin(2\phi_p)$$

where ϕ_p is the desired radial angle in pixel space

ϕ is the known azimuth angle

ρ is the radial distance calculated in step 7

a, b, and c are constants.

As always, considerable effort must be undertaken in the implementing computer code to handle the various principle range, quadrant, and unit conversion issues that arise in these equations. The Mathematica notebook mentioned above continues with the solution of this equation for a, b, and c using a least squares technique. This is actually straightforward since the known values of ϕ_p can be used on the right hand side of the equation. As above, any star whose position results in an anomalous residual can have an over powerful affect on the solution, so such are eliminated from consideration and a new solution found by iteration of the least squares fit. When finished, the radial pixel angle residuals are on the order of 0.10° to 0.25° across the entire sky. The use of this equation is not straightforward, but requires iteration in the following manner:

$$\begin{aligned}\phi_1 &= \phi + a + b\rho \cos(2\phi) + c\rho \sin(2\phi) \\ \phi_2 &= \phi + a + b\rho \cos(2\phi_1) + c\rho \sin(2\phi_1) \\ \phi_3 &= \phi + a + b\rho \cos(2\phi_2) + c\rho \sin(2\phi_2) \\ &\dots\end{aligned}$$

where ϕ_n is the increasingly improved radial angle approximation in pixel space

ϕ is the known azimuth angle

ρ is the radial distance calculated in step 7

a, b, and c are the just calculated constants.

This method works because the correction to ϕ that results in ϕ_p is quite small; in fact three iterations reduces the difference between ϕ_3 and ϕ_2 to about one part in a million.

9. The final step in deriving equations for column and row numbers as expressions of azimuth and zenith angles is to use the radial distance and radial angle just derived to calculate column and row. The equations are:

$$\begin{aligned}C &= C_c + \rho \sin \phi_p \\ R &= R_c + \rho \cos \phi_p\end{aligned}$$

where C and R are the desired fractional column and row numbers

C_c and R_c are the known zenith image fractional column and row

ϕ_p is the radial angle in pixel space calculated in step 8, and

ρ is the radial distance calculated in step 7.

Typical residuals for both column and row are 0.35 pixels, with the largest being about 1.0 pixel.

10. The next step is to verify that steps 4–9 are correct, since the transformation equations at this point are based on a single collection on a single night. The second spreadsheet developed in step 3 is used for this verification, which has the actual observed star positions on a different night as well as each star’s zenith and azimuth position from ephemeris data. All the transformations are applied to this data and residuals calculated. If the residuals are significantly different than those observed during derivation, then the cause is analyzed and ameliorated.
11. Finally, the calibration file variables `pixel_to_zenith`, `pixel_to_azimuth`, `angle_to_column`, and `angle_to_row` are populated by solving the transformation equations from steps 5, 6, and 9 for variety of whole number inputs necessary. See [Variables; Dimensions, Variables, and Coding; Calibration File; Chapter 7: Calibration Product](#) for a discussion of these variables.

7.1.4.4 Pristine Sky Radiance Variables

The variable `rad_table` contains a four-dimensional array with pristine sky radiance information necessary for the spectral sky cover retrieval as explained in the section [Daytime Spectral Retrieval; Algorithm; Sky Feature File; Chapter 5: Sky Feature Product](#). There are three associated variables that define the various axes in this array; `rad_table_solar_zeniths`, `rad_table_azimuths`, and `rad_table_zeniths`. All four of these variables are discussed in detail above; see [Variables; Dimensions, Variables, and Coding; Calibration File; Chapter 7: Calibration Product](#).

The values for the axis definition variables are somewhat arbitrary, but were selected with the following three principles in mind. First, `rad_table` should be of reasonable size, with no dimension being excessively extensive. Second, the axes should not only completely span the range of interest but in some cases extend beyond to avoid edge effects during interpolation within the spectral retrieval. And finally, the interval between values does not need to be constant but rather should be more closely spaced where the radiance values change more rapidly.

The values in `rad_table` were calculated with the widely used model SBDART, or Santa Barbara DISORT Atmospheric Radiative Transfer. This model has been well documented elsewhere, for example see [Introduction to SBDART](#), and the interested reader should refer to that documentation for a complete understanding of its strengths and limitations. All model runs were made with a standard mid-latitude aerosol component consistent with 100 km visibility for the various combinations of spectral band, sun angle, and viewing angles.

7.1.5 Usage Hints

Most users of Whole Sky Imager retrieved data products will not need to use the calibration file. Those that do should insure that the file they are using is valid for their study dates; this can be determined by reviewing the from- and to- dates imbedded in the file name as explained in [Naming Convention; Calibration File; Chapter 7: Calibration Product](#).

The data of most use to the user will be the pixel conversion arrays. These present accurate floating point transforms but for only whole number inputs of pixel position or sky angle. The user should write a two-dimensional linear or three-point Lagrange interpolation for the general case.

The pedigree of a particular calibration file must be tracked by the user. Several things can precipitate the issuance of a new calibration file in the series associated with a particular site. The most common is the physical movement of the environmental housing surrounding the camera requiring an update of the transformation arrays. Another cause is the replacement of a spectral filter. On rare occasions the reason is the replacement of the entire instrument. With the deployment of the field calibration device (see [Field Calibration; Instrument Calibration; Chapter 1: Overview](#)) every instrument should be recalibrated yearly, and so should have a new calibration file at least that frequently.

If the user is accessing the rad_table radiances, he should insure that he understands how the solar and viewing angles are defined (see [Variables; Dimensions, Variables, and Coding; Calibration File; Chapter 7: Calibration Product](#)). In particular, the solar azimuth angle is suppressed by defining the viewing azimuth angles relative to the solar plane; therefore it is always necessary to rotate the radiances to develop a geographically meaningful solution. Since the array is somewhat sparse, the user is strongly urged to adopt at least a three-point Lagrange interpolation.

IDL handles character variables in NetCDF files as ASCII encoded byte strings of fixed length, hence the extra dimension variables in Table 17. If one is having trouble matching a search character string to a value retrieved from a NetCDF character variable, it is best to print out the associated character string one time to discover where the blanks and nulls are inserted to pad out to the fixed length. Then the search string can be modified appropriately. Also remember that character variables are case sensitive.

7.1.6 Limitations

The calibration data presented in each file is quite complete. However, its accuracy is subject to the usual errors of any calibration process. In particular the user should read the literature referenced in [Radiance Variables; Algorithm; Calibration File; Chapter 7: Calibration Product](#) concerning the accuracy of the radiance calibrations to understand any limitations that accuracy could impose on his study.

8. Utility Products

There are two Whole Sky Imager utility products available from the ARM Archive. The first is a pictorial representation of the files available for a 24-hour period. The second is an archive filled with the run-time log files from the retrieval execution. Both are intended only for the more sophisticated users of Whole Sky Imager data and retrieval products.

8.1 File Existence Plot File

8.1.1 Purpose

The file existence plot file is a PNG format pictograph showing which of the various possible Whole Sky Imager files for a given day actually are in the archive. In a perfect situation all would be available but instrument down times, quality down checks, and the not-yet-implemented retrievals all cause some to be missing.

Often an experimenter will have a particular time frame in mind for an analysis and will want to know what Whole Sky Imager data or retrieval products are available to support his study. Since all the

non-summary products are stored in the ARM Archive in UNIX style tar archives it is not possible to get the names of available files directly without first downloading the extremely large tar archives. Even if a list of existing file names were available, searching it for the ones of interest would be a daunting task as some 3000–3500 raw and retrieved files are potentially produced for each Whole Sky Imager for each day. However, the experimenter can discover data and retrieval gaps very quickly by scanning the file existence plot.

8.1.2 Naming Convention

The file existence plot files have the following name based on standard ARM naming practices.

`sssfilessummarygiff.c1.yyyymmdd.hhmmss.gif`

or

`sssfilessummarypngff.c1.yyyymmdd.hhmmss.png`

where the blue characters are literal and:

`sss ff` = the site and facility identifier as listed in Table 1

`yyymmdd.hhmmss` = the standard 15 character ARM date time stamp showing year, month, day, hour, minute, and second.

Historically, the first data processed was into the GIF format, but all recent data is in the PNG format; the file name indicates the format. Since only one file is produced for each 24-hour period the hours, minutes, and seconds are always set to zero.

8.1.3 Dimensions, Variables, and Coding

Since the file existence plot files are GIF or PNG rather than NetCDF they do not have dimensions, attributes, and variables in the manner of normal ARM usage. The vertical axis of the plot does have some obscure abbreviations indicating file type as well as a line below the plot; these are expanded in Table 24.

Code	Associated File
drk	Raw file; shutter closed
dr2	Raw file; shutter closed; full moon
clr	Raw file; null-filter
blu	Raw file; 450 nm filter
red	Raw file; 650 nm filter
nir	Raw file; 800 nm filter
jpg	Sky picture file
rad	Radiance data file
cld	Sky feature file
spc	Cloud spectral retrieval picture file
den	Cloud optical density retrieval picture file
patchsummary	Patch radiance file
partradmpg	Movie file
cloudsummary	Sky feature summary file

8.1.4 Algorithm

This file is produced by an IDL plot procedure run on a regular basis at the ARM Archive. The procedure is named WsiIdpc8_vx where the x stands for the current version number. As of this writing version 4 is current. The procedure runs at the end of the retrieval process when all the files for a day are still on disk; it accesses the directories in which the relevant files are maintained and plots a symbol for each.

8.1.5 Usage Hints

A sample file existence plot file is shown in Figure 6. The existence of files which are produced one per day is indicated in a text line near the bottom. The existence of all other files is indicated by colored dots on the three bars – one for each eight hour period of the Greenwich day of interest. The width of a file's dot corresponds to about two minutes re the time scale; thus files that are produced every two minutes will form a solid colored line. Files that are produced on some other longer interval will form a dotted line; in the example plot the radiation data file is produced on six-minute intervals. The user should become familiar with the Whole Sky Image data collection scheme discussed in [Data Collection Strategy; Chapter 1: Overview](#) to best use this plot.

In the example, the day to night transition occurred between 0100 and 0200 hours resulting in the production of several raw closed-shutter files and an abrupt change from three kinds of raw spectral files to raw null-filter files. Sometime after 0000 hours, the sun's zenith angle became too large for daytime spectral retrievals. Since the twilight spectral retrieval is not yet implemented there is a gap in the sky feature file line until nighttime optical density retrievals begin just before 0200 hours. The situation was stable until after 100 hours with all files meeting quality expectations; otherwise there would be gaps indicating missing down check files. Between 1100 and 1400 hours the night to day transition occurred with the inverse of the changes just discussed for the day to night transition. Around and just after 2300 hours there is a gap in the sky feature file line. Since the radiance data files exist for this period, it is most likely caused by quality down check of the six possible retrievals. The cloud spectral retrieval picture files of course do not exist either as their underlying sky feature files don't exist.

8.1.6 Limitations

If not Whole Sky Imager files exist for a certain day, then the file existence plot file will not exist either. This situation can occur when an instrument is non-operational for an extended period.

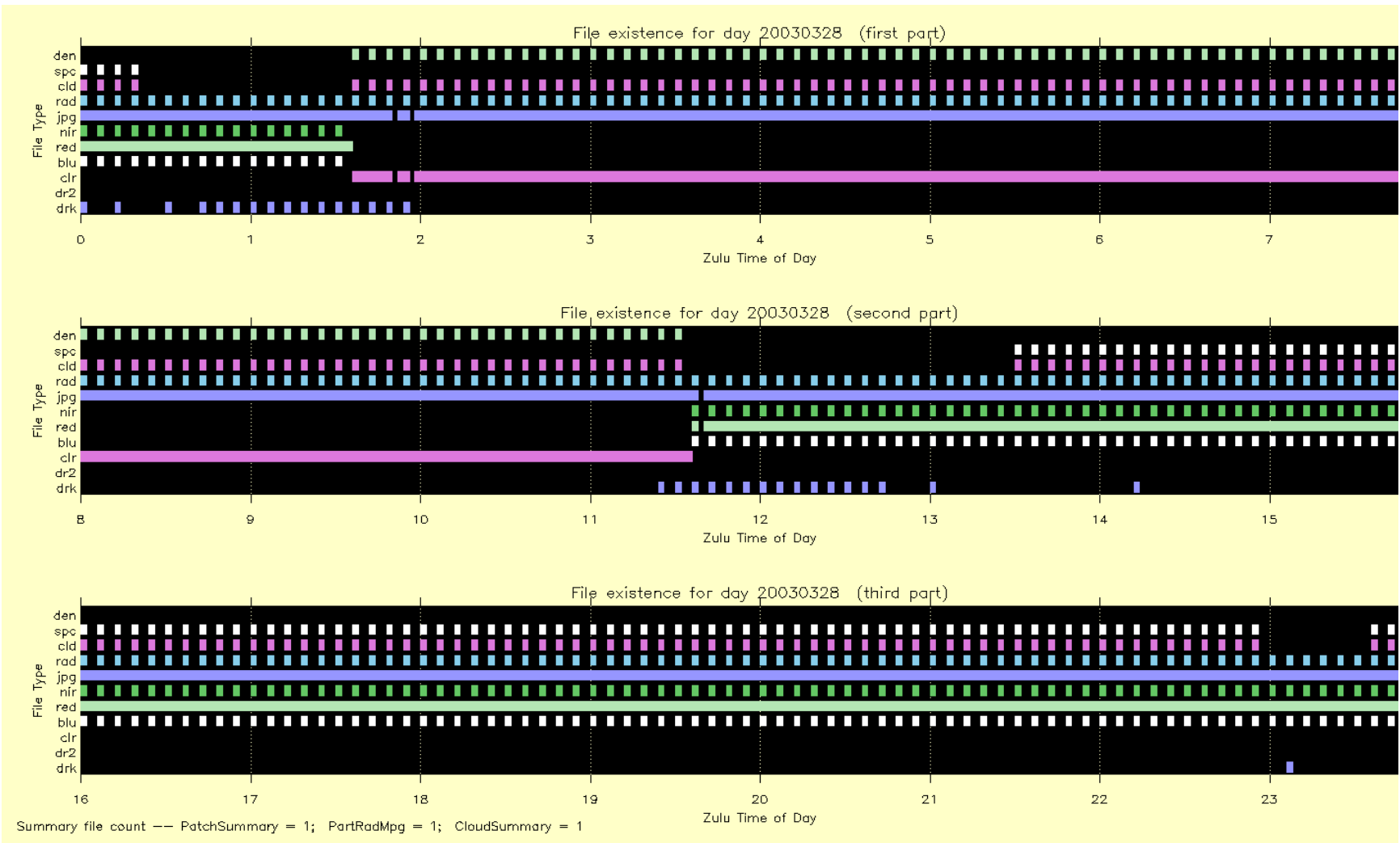


Figure 6. Sample file existence plot.

8.2 Log Files

8.2.1 Purpose

There are a set of seven ASCII log files produced during normal Whole Sky Imager single Greenwich day retrieval processing at the ARM Archive, one by each of the major IDL and quality control procedures that is run. These text logs contain human readable statements about conditions during execution. They are of little use to the general Whole Sky Imager data and retrieval user. However, they can help in understanding such things as the inner workings of the quality processes discussed in [Quality Checking; Algorithm; Radiance Data File; Chapter 4: Radiance Product](#) and [Quality Checking; Algorithm; Sky Feature File; Chapter 5: Sky Feature Product](#) to diagnose seemingly low marks. Also they are of use to those responsible for maintaining the retrieval code and processes in diagnosing anomalous behavior.

8.2.2 Naming Convention

The WsiIdpc2_vx IDL procedure produces one log file while processing an entire Greenwich day's worth of raw data. It has the following name based on standard ARM naming practices.

[sssiwsipartradianceqclgff.00.yyyymmdd.hhmmss.asc](#)

The WsiIdpc3_vx IDL procedure produces one log file while processing for a Greenwich day. It has the following name based on standard ARM naming practices.

[ssswsipatchsummaryqclgff.00.yyyymmdd.hhmmss.asc](#)

The WsiIdpc4_vx IDL procedure produces one log file while processing an entire Greenwich day's worth of raw data. It has the following name based on standard ARM naming practices.

[ssswsifullradianceqclgff.00.yyyymmdd.hhmmss.asc](#)

The WsiIdpc5_vx IDL procedure produces one log file while processing an entire Greenwich day's worth of radiance data. It has the following name based on standard ARM naming practices.

[ssswsicloudqclgff.00.yyyymmdd.hhmmss.asc](#)

The manual quality checking process produces lists of down checked files to be deleted from the archiving process for a Greenwich day. These lists are assembled into one file and preserved with the following name based on standard ARM naming practices.

[ssswsideletionlogff.00.yyyymmdd.hhmmss.asc](#)

The WsiIdpc6_vx IDL procedure produces one log file while processing for a Greenwich day. It has the following name based on standard ARM naming practices.

[ssswsicloudsummaryqclgff.00.yyyymmdd.hhmmss.asc](#)

Finally, the WsiIdpc8_vx IDL procedure produces one log file while processing for a Greenwich day. It has the following name based on standard ARM naming practices.

[ssswsifilesummaryqclgff.00.yyyymmdd.hhmmss.asc](#)

These seven log files are saved in the ARM Archive in a single UNIX style tar archive. That archive file has the following name based on standard ARM naming practices.

`sss``wsiprologs``ff.00.``yyymmdd``.hhmmss``.asc.tar`

In each of these names, the blue characters are literal and

- `sss` `ff` = the site and facility identifier as listed in Table 1
- `ii` = the time interval of production from the set {02, 10}
- `yyymmdd``.hhmmss` = the standard 15 character ARM date time stamp showing year, month, day, hour, minute, and second.

Since only one file is produced for each 24-hour period the hours, minutes, and seconds are always set to zero.

8.2.3 Dimensions, Variables, and Coding

Since the log files are ASCII text rather than NetCDF they do not have dimensions, attributes, and variables in the manner of normal ARM usage; they are intended to be human readable. Most included messages begin with the name of the subroutine or function from which it arises and include a severity level from the following set: {information, warning, error}. The normal user of Whole Sky Imager data should never encounter any error messages as these result in process termination and operator intervention.

8.2.4 Algorithm

The `WsiIdpc2_vx` IDL procedure which produces the “`wsipartradianceqclog`” log file is discussed in [Algorithm; Sky Picture File; Chapter 3: Visualization Products](#).

The `WsiIdpc3_vx` IDL procedure which produces the “`wsipatchsummaryqclog`” log file is discussed in [Algorithm; Patch Radiance File; Chapter 6: Summary Products](#).

The `WsiIdpc4_vx` IDL procedure which produces the “`wsifullradianceqclog`” log file is discussed in detail in [Algorithm; Radiance Data File; Chapter 4: Radiance Product](#).

The `WsiIdpc5_vx` IDL procedure which produces the “`wsicloudqclog`” log file is discussed in detail in [Algorithm; Sky Feature File; Chapter 5: Sky Feature Product](#).

The manual quality checking process that results in “`wsideletionlog`” log file is discussed in detail in [Data Quality Procedure; Data Quality Control; Chapter 1: Overview](#). The log originates in an IDL procedure maintained by the instrument mentor.

The `WsiIdpc6_vx` IDL procedure which produces the “`wsicloudsummaryqclog`” log file is discussed in [Algorithm; Sky feature Summary File; Chapter 6: Summary Products](#).

The `WsiIdpc8_vx` IDL procedure which produces the “`wsifilesummaryqclog`” log file is discussed in [Algorithm; File Existence Plot File; Chapter 8: Utility Products](#).

8.2.5 Usage Hints

Some of these logs are very long, in particular those produced by WsiIdpc2_vx, WsiIdpc4_vx, and WsiIdpc5_vx which process hundreds of input files. One technique useful in scanning such files is to invoke two copies in a multifunction word processor such as MS Word. Next sort the first copy by paragraph alphabetically to place similar messages in blocks which can then be deleted wholesale if of no interest. Interesting messages can be back traced to their context by using a find command on the second copy.

8.2.6 Limitations

These routines represent tens of thousands of lines of computer code; considerable effort has been made to keep all the message formats unique, readable, consistent, and informative. However, since maintaining log messages is not a high priority item when funds are disbursed, it is possible that a few do not meet these high standards.