Scientific Datasets

The scientific dataset defines a scientific data group, which can contain a variable number of scientific data descriptors. Current specifications only support floating-point scientific data organized in rectangular grid arrays. Rank and dimension sizes may vary. A user interface exists for storing and retrieving scientific datasets. See the NCSA HDF manual for details.

Required Tags

HDF scientific datasets must always contain at least the data objects listed in Table 3.4.

Table 3.4 Required Tags for Scientific Datasets

Tag	Contents of Data Element
SDG	scientific data group
SDD	scientific data dimension record for array-stored data. It includes the rank (number of dimensions) and the size of each dimension.
SD	scientific data

Optional Tags

The data objects presented in Table 3.5 are optional. Inclusion of these objects is supported by NCSA's user interface for scientific datasets.

Table 3.5 Optional Tags for Scientific Datasets

Tag	Contents of Data Element
SDS	scales along the different dimensions to be used when interpreting or displaying the data
SDL	labels for all dimensions and for the data. Each of the dimension labels can be interpreted as an independent variable, and the data label as the dependent variable.
SDU	units for all dimensions and for the data
SDF	format specifications to be used when displaying values of the data
SDM	maximum and minimum values of the data
SDC	coordinate system to be used when interpreting or displaying the data

specialized video production or film color correction parameters.

NCSA currently supports a calling interface, documented in the NCSA HDF manual, defined for the easy storage and retrieval of 8-bit raster images using RIGs. The interface is called *RIS8* and it handles 8-bit color tables and different compression capabilities. Calling interfaces for storing 24-bit images and other special cases are under development.

Tags for Raster Image Sets

The tags presented in Table 3.2 must be fully supported by any raster image set implementation.

Table 3.2 Tags for Raster Image Sets

Tag	Contents of Data Element
RIG	raster image group
ID	image dimension record
RI	raster image data

With full support for the above tags, images can be stored and read from HDF files at any bit depth, with several different component ordering schemes. Colors would be up to the calling program to assign if the image is not in true color. In addition to the required tags that just define an image dataset, the tags listed in Table 3.3 define color properties and other image features. These tags are described fully in Appendix A.

Table 3.3 Additional Tags for Raster Image Sets

Tag	Contents of Data Element
XYP	XY position of image
LD	look-up table dimension record
LUT	color look-up table for non true-color images
MD	matte channel dimension record
MA	matte channel data
CCN	color correction factors
CFM	color format designation
AR	aspect ratio
MTO	machine-type override

Compression of Raster Images

Tags for two types of compression have been defined for raster images. They are run-length encoding (RLE) and IMCOMP aerial averaging (IMC). Each encoding tag is documented under its specific tag type (see Appendix A). Support for RIG and RI does not require that all of the compression tag types be supported. If you find an unknown compression type, provide a suitable error message to the user.

Compatibility between Raster-8 and Raster Image Sets

In order to maintain backward compatibility with raster-8 sets, raster image set interfaces store tag/refs for both types of sets in HDF raster image files. For example, if an image is stored as part of a raster image set, there will be one copy each of the image dimension data, image data, and palette data stored, but there will be two sets of tag/refs pointing to each data element, one from each set. The image data, for instance, will be associated with tag RI8 and RI.

Raster Image Sets

There are currently two sets for handling raster images: the raster-8 set and the raster image set. The raster-8 set was the first raster set to be implemented. It includes a small set of data objects for efficiently storing and retrieving color raster images. The raster image set is more powerful, but it is also more complex. Raster image sets can include more detailed information about a raster image than a raster-8 set can. Both sets are supported by HDF software.

The Raster-8 Set

The *raster-8 set* is a set of tags that provide the basic information necessary to store 8-bit raster images in a data file and display them accurately without prompting the user to supply dimensions or color information. The raster-8 set consists of the tags presented in Table 3.1.

Table 3.1 Tags for Raster-8 Sets

Tag	Contents of Data Element
RI8	eight-bit raster image data
CI8	eight-bit raster image data compressed with run-length encoding
II8	IMCOMP compressed image data
ID8	image dimension record
IP8	image palette data

Even though the raster-8 set is minimal and is superseded in most applications by the raster image set, some implementations may want to deal with this type of data in a rough form for prototyping or simple applications. The data stored is compatible with the raster image set. For more information, refer to the section entitled “Compatibility Between Raster-8 and Raster Images Sets.”

If you develop software for processing raster-8 sets, it must support RI8, ID8, and IP8. If you do not implement CI8 or II8, then be sure to provide appropriate error indicators to higher layers that might expect to find these tags.

The Raster Image Set

The raster image set provides a framework for storing images and any number of optional image descriptors. It provides for a description of the image data layout, with the optional presence of color look-up tables, aspect ratio, color correction, associated matte or other overlay information, or any other data related to the display of the image.

Tying everything together is the raster image group (RIG), examples of which were given earlier (Figure 3.1). A RIG contains a list of data identifiers that point in turn to the data objects that describe and make up the image.

The number of entries in a RIG is variable and the presence of most of the description information is optional. Complex applications can store data identifiers of image-modifying data, such as the color table and aspect ratio, in the RIG along with the reference to the image data itself. Simple applications can use simple application level calls and ignore

```
  94    DD    (RIG  1   480951  12)

 106    DD    (RIG  2   480963  12)

 118    DD    (empty)
130     "sw3"
134     "solar wind simulation:  third try. 8/8/88"
175     <data for image palette>
943     <data for 1st image dimension rec>: 400, 600
947     <data for 1st raster image>
240947  <data for 2nd image dimension rec>: 400, 600
240951  <data for 2nd raster image>
480951  tag/refs for 1st RIG:  IP8/1, ID/1, RI/1
480963  tag/refs for 2nd RIG:  IP8/1, ID/2, RI/2
```

The HDF file structure depicted in Figure 3.1 reflects the grouping of raster image sets. This file contains the same raster image information as the file in Figure 1.5, but the information is organized into two sets and groups. Note that there is only one palette (IP8/1) and it is included in both groups.

General Features of Groups

Figure 3.1 also illustrates a number of important general features of groups:

- The contents of each set are consistent with one another. Since the palette (IP8) is designed for use with 8-bit images, the image must be an 8-bit image, rather than a 24-bit, 12-bit, or other image.

- An application program can easily process all of the images in the file by accessing the groups in the file. The non-RIG information contained in the file can be either used or ignored, depending on the needs and capabilities of the application program.

- There is usually more than one way to group sets. For example, an extra copy of the image palette (IP8) could have been stored in the file, so that each grouping would have its own image palette. But in this instance that is not necessary because the same palette is to be used with both images. On the other hand, in this example there are two image dimension records (one per group), even though one would suffice.

- Group status does not alter the fundamental role of HDF objects. They are still accessible as individual data objects, despite the fact that they also belong to raster image sets. In a very real sense, the individual data elements are in the file, whether or not there are groups that contain them.

What each of the RIGs does is provide an index showing what sets exist and what their members are. There is nothing to prevent the imposition of other groupings (indexes) that provide a different view of the same collection of data objects. In fact, HDF is designed to encourage the addition of alternate views, when appropriate.

In addition to the libraries, a growing number of command-line utility routines are available for working with sets. For example, a utility called r8tohdf is an HDF command that converts one or more raw raster images to HDF 8-bit raster image set format.

NCSA supports calling interfaces for the following machines: Cray (UNICOS), Alliant (Concentrix), Silicon Graphics (UNIX), Sun (UNIX), Macintosh (MacOS), and IBM PC/AT (MS-DOS). The calling interfaces that are currently available are described in the NCSA HDF manual.

Groups

An *HDF set* is a collection of HDF data objects in a file. Unless some mechanism is used to identify explicitly those objects that belong to a set, there is often no way to tie them together. This problem is solved in HDF by means of groups. A *group* is a data object that explicitly identifies all of the data objects in a set.

Since a group is a type of data object, its structure is like that of any other data object. A *group data identifier* (tag/ref) points to a data element that consists of the collection of data identifiers that make up the corresponding set. A *group tag* can be defined for any set. For instance, *raster image group (RIG)* is the group tag used to group members of raster image sets; RIG data consists of a list of all data identifiers that belong to a particular raster image set.

Groups provide a convenient mechanism for application programs to locate all of the information that they need about a set. Application programs that deal with RIGs, for instance, read all of the elements in a RIG group, using only those that they need for their application and ignoring the others.

Sample Groups

Suppose that the two images shown in Figure 1.5 are organized into two sets with group tags. Since they are images, they may be stored as RIG groups. Figure 3.1 illustrates the type of organization that incorporates RIG groupings of these images.

Figure 3.1 Physical Organization of Sample RIG Groupings

Offset	Contents				
0	FH				
4	DDH	(5	0L)		
10	DD	(FID	1	130	4)
22	DD	(FD	1	134	41)
34	DD	(IP8	1	175	768)
46	DD	(ID	1	943	4)
58	DD	(RI	1	947	240000)
70	DD	(ID	2	240947	4)
82	DD	(RI	2	240951	240,000)

Chapter Overview

This chapter describes raster image sets and scientific datasets, explains the role of sets and groups in an HDF file, and discusses the routines that are available for storing and retrieving scientific datasets.

Sets

Sometimes tags are grouped into *sets*, where each set is designed to serve a particular user requirement. For example, the raster image set, which is described in the following sections, contains several tags that are used for storing information about 8-bit raster images.

One advantage of using sets is that it allows different sites to implement different parts of the entire HDF specification. When you take an HDF file to a new site, you obtain a list of *supported sets* and your file is perfectly valid for operations dealing with those sets. Other data which you may have stored in your file is ignored at that site but is still there when you transfer the file back.

Each HDF set is defined in terms of a minimum collection of data objects that must be present when the set is used. For instance, every raster image set must contain at least the following three data objects: an *image dimension record*, which gives the width and height of the corresponding image; *raster image data*, which consists of the pixel values that make up the image; and a *raster image group*, which lists of all of the members in the set.

In addition, there are optional data objects that may be included in a set. A raster image set, for instance, often contains a palette, or lookup table, which gives the red, green, and blue values to be associated with each pixel in the raster image data.

Types of Sets

At this time, three sets have been implemented—two sets for storing 8-bit raster images, and a *scientific dataset* for storing sets of floating-point numbers. Specifications are given for these sets in following sections.

Some important sets that may be added later include:

Utility set	basic utilities
Utility II	suggested utilities
Composite set	combinations of tags and movie organization
Vector	vector and polygon images
Raster image set	extension of raster set to 24-bit images

Calling Interfaces for Sets

NCSA provides calling interfaces for all the HDF sets that it supports. The primary purpose of these calling interfaces is to provide libraries of routines for reading and writing the data that is associated with each set. The libraries currently supported at NCSA are callable from either C or Fortran programs.

Chapter 3 Sets and Groups

Chapter Overview

Sets

- Types of Sets
- Calling Interfaces for Sets

Groups

- Sample Groups
- General Features of Groups

Raster Image Sets

- Raster-8 Set
- Raster Image Set
- Tags for Raster Image Sets
- Compression of Raster Images
- Compatibility between Raster-8 and Raster Image Sets

Scientific Datasets

- Required Tags
- Optional Tags