Table of Contents

**Chapter 1** - Introduction to HDF

1.1Chapter Overview 1

1.2What is HDF? 1

1.3Why Was HDF Created? 3

1.4High-Level HDF APIs 4

1.5HDF Command-Line Utilities and Visualization Tools 5

1.6Primary HDF Platforms 5

1.7HDF4 versus HDF5 5

**Chapter 2** - HDF Fundamentals

2.1Chapter Overview 7

2.2HDF File Format 7

2.2.1 File Header 8

2.2.2 Data Object 8

2.2.2.1 Data Descriptor 8

2.2.2.2 Data Elements 9

2.2.3 Data Descriptor Block 9

2.2.4 Grouping Data Objects in an HDF File 10

2.3Basic Operations on HDF Files Using the Multifile Interfaces 10

2.3.1 File Identifiers 10

2.3.2 Opening HDF Files: Hopen 11

2.3.3 Closing HDF Files: Hclose 12

2.3.4 Getting the HDF Library and File Versions: Hgetlibversion and Hgetfileversion 12

2.4Determining whether a File Is an HDF File: Hishdf/hishdff 13

2.5Programming Issues 13

2.5.1 Header File Information 13

2.5.2 HDF Definitions 14

2.5.2.1 Standard HDF Data Types 14

2.5.2.2 Native Format Data Types 14

2.5.2.3 Little-Endian Data Types 15

2.5.2.4 Tag Definitions 15

2.5.2.5 Limit Definitions 15

2.5.3 FORTRAN-77 and C Language Issues 16

**Chapter 3** - Scientific Data Sets (SD API)

3.1Chapter Overview 19

3.2The Scientific Data Set Data Model 19

3.2.1 Required SDS Components 20

3.2.2 Optional SDS Components 21

3.2.3 Annotations and the SD Data Model 22

3.3The SD Interface 22

3.3.1 Header Files Required by the SD Interface 22

3.3.2 SD Interface Routines 22

3.3.3 Tags in the SD Interface 26

3.4Programming Model for the SD Interface 26

3.4.1 Establishing Access to Files and Data Sets: SDstart, SDcreate, and SDselect 27

3.4.2 Terminating Access to Files and Data Sets: SDendaccess and SDend 28

3.5Writing Data to an SDS 31

3.5.1 Writing Data to an SDS Array: SDwritedata 31

3.5.1.1 Filling an Entire Array 33

3.5.1.2 Writing Slabs to an SDS Array 36

3.5.1.3 Appending Data to an SDS Array along an Unlimited Dimension 41

3.5.1.4 Determining whether an SDS Array is Appendable: SDisrecord 42

3.5.1.5 Setting the Block Size: SDsetblocksize 42

3.5.1.6 Setting the I/O Access Type of an SDS: SDsetaccesstype 43

3.5.2 Compressing SDS Data: SDsetcompress 47

3.5.3 External File Operations 52

3.5.3.1 Specifying the Directory Search Path of an External File: HXsetdir 52

3.5.3.2 Specifying the Location of the Next External File to be Created: HXsetcreatedir 53

3.5.3.3 Creating a Data Set with Data Stored in an External File: SDsetexternalfile 54

3.5.3.4 Getting External File Information of a Data Set: SDgetexternalinfo 55

3.5.3.5 Moving Existing Data to an External File 56

3.6Reading Data from an SDS Array: SDreaddata 58

3.7Obtaining Information about SD Data Sets 66

3.7.1 Obtaining Information about the Contents of a File: SDfileinfo 66

3.7.2 Obtaining Information about a Specific SDS: SDgetinfo 66

3.7.3 Obtaining Data Set Compression Information: SDgetcompinfo 67

3.7.4 Locating an SDS by Name: SDnametoindex 71

3.7.5 Locating More Than One SDS by the Same Name: SDnametoindices 71

3.7.6 Getting Number of Data Sets Given a Name: SDgetnumvars\_byname 72

3.7.7 Locating an SDS by Reference Number: SDreftoindex 73

3.7.8 Obtaining the Reference Number Assigned to the Specified SDS: SDidtoref 73

3.7.9 Obtaining the Type of an HDF4 Object: SDidtype 73

3.7.10 Determining whether an SDS is empty: SDcheckempty 74

3.7.11 Creating SDS Arrays Containing Non-standard Length Data: SDsetnbitdataset 77

3.8SDS Dimension and Dimension Scale Operations 78

3.8.1 Selecting a Dimension: SDgetdimid 78

3.8.2 Naming a Dimension: SDsetdimname 79

3.8.3 Old and New Dimension Implementations 80

3.8.3.1 Setting the Future Compatibility Mode of a Dimension: SDsetdimval\_comp 80

3.8.3.2 Determining the Current Compatibility Mode of a Dimension: SDisdimval\_bwcomp 81

3.8.4 Dimension Scales 81

3.8.4.1 Writing Dimension Scales: SDsetdimscale 81

3.8.4.2 Obtaining Dimension Scale and Other Dimension Information: SDdiminfo 82

3.8.4.3 Reading Dimension Scales: SDgetdimscale 82

3.8.4.4 Distinguishing SDS Arrays from Dimension Scales: SDiscoordvar 88

3.8.5 Related Data Sets 91

3.9User-defined Attributes 92

3.9.1 Creating or Writing User-defined Attributes: SDsetattr 93

3.9.2 Querying User-defined Attributes: SDfindattr and SDattrinfo 96

3.9.3 Reading User-defined Attributes: SDreadattr 97

3.10Predefined Attributes 103

3.10.1 Accessing Predefined Attributes 104

3.10.2 SDS String Attributes 105

3.10.2.1 Writing String Attributes of an SDS: SDsetdatastrs 105

3.10.2.2 Reading String Attributes of an SDS: SDgetdatastrs 105

3.10.3 String Attributes of Dimensions 106

3.10.3.1 Writing a String Attribute of a Dimension: SDsetdimstrs 106

3.10.3.2 Reading a String Attribute of a Dimension: SDgetdimstrs 107

3.10.4 Range Attributes 107

3.10.4.1 Writing a Range Attribute: SDsetrange 107

3.10.4.2 Reading a Range Attribute: SDgetrange 108

3.10.5 Fill Values and Fill Mode 108

3.10.5.1 Writing a Fill Value Attribute: SDsetfillvalue 109

3.10.5.2 Reading a Fill Value Attribute: SDgetfillvalue 109

3.10.5.3 Setting the Fill Mode for all SDSs in the Specified File: SDsetfillmode 109

3.10.6 Calibration Attributes 110

3.10.6.1 Setting Calibration Information: SDsetcal 110

3.10.6.2 Reading Calibrated Data: SDgetcal 111

3.11Convenient Operations Related to File and Environment 112

3.11.1 Obtaining the Name of a File: SDgetfilename 112

3.11.2 Obtaining the Length of an HDF4 Object’s Name: SDgetnamelen 112

3.11.3 Resetting the Allowed Number of Opened Files: SDreset\_maxopenfiles 112

3.11.4 Obtaining Current Limits on Opened Files: SDget\_maxopenfiles 113

3.11.5 Obtaining Number of Opened Files: SDget\_numopenfiles 113

3.11.6 Creating SDS Arrays Containing Non-standard Length Data: SDsetnbitdataset 113

3.12Chunked (or Tiled) Scientific Data Sets 115

3.12.1 Making an SDS a Chunked SDS: SDsetchunk 115

3.12.2 Setting the Maximum Number of Chunks in the Cache: SDsetchunkcache 117

3.12.3 Writing Data to Chunked SDSs: SDwritechunk and SDwritedata 118

3.12.4 Reading Data from Chunked SDSs: SDreadchunk and SDreaddata 120

3.12.5 Obtaining Information about a Chunked SDS: SDgetchunkinfo 121

3.13Ghost Areas 131

3.14netCDF 132

3.14.1 HDF Interface vs. netCDF Interface 133

3.14.2 ncdump and ncgen 135

3.14.2.1 Using ncdump on HDF Files 135

**Chapter 4** - Vdatas (VS API)

4.1Chapter Overview 137

4.2The Vdata Model 137

4.2.1 Records and Fields 138

4.3The Vdata Interface 138

4.3.1 Header Files Used by the Vdata Interface 138

4.3.2 Vdata Library Routines 138

4.3.3 Identifying Vdatas in the Vdata Interface 142

4.3.4 Programming Model for the Vdata Interface 142

4.3.5 Accessing Files and Vdatas: Vstart and VSattach 143

4.3.6 Terminating Access to Vdatas and Files: VSdetach and Vend 143

4.4Creating and Writing to Single-Field Vdatas: VHstoredata and VHstoredatam 149

4.5Writing to Multi-Field Vdatas 155

4.5.1 Creating Vdatas 155

4.5.1.1 Assigning a Vdata Name and Class: VSsetname and VSsetclass 156

4.5.1.2 Defining a Field within a Vdata: VSfdefine 156

4.5.1.3 Initializing the Fields for Write Access: VSsetfields 157

4.5.1.4 Specifying the Interlace Mode: VSsetinterlace 157

4.5.1.5 Specifying External Storage Information: VSsetexternalfile 158

4.5.2 Writing Data to Vdatas 159

4.5.2.1 Resetting the Current Position within Vdatas: VSseek 160

4.5.2.2 Writing to a Vdata: VSwrite 161

4.5.2.3 Setting Up Linked Block Vdatas: VSsetblocksize and VSsetnumblocks 166

4.5.2.4 Packing or Unpacking Field Data: VSfpack 167

4.6Reading from Vdatas 173

4.6.1 Initializing the Fields for Read Access: VSsetfields 174

4.6.2 Reading from the Current Vdata: VSread 174

4.7Searching for Vdatas in a File 182

4.7.1 Finding All Vdatas that are Not Members of a Vgroup: VSlone 182

4.7.2 Sequentially Searching for a Vdata: VSgetid 183

4.7.3 Determining a Reference Number from a Vdata Name: VSfind 183

4.7.4 Searching for a Vdata by Field Name: VSfexist 184

4.7.5 Retrieving Vdatas in a File or in a Vgroup: VSgetvdatas 187

4.7.6 Determining Internal Vdata: VSisinternal 188

4.7.7 Retrieving Vdatas in a File or in a Vgroup: VSofclass 189

4.8Vdata Attributes 190

4.8.1 Querying the Index of a Vdata Field Given the Field Name: VSfindex 190

4.8.2 Setting the Attribute of a Vdata or Vdata Field: VSsetattr 191

4.8.3 Querying the Values of a Vdata or Vdata Field Attribute: VSgetattr 191

4.8.4 Querying the Total Number of Vdata and Vdata Field Attributes: VSnattrs 192

4.8.5 Querying the Number of Attributes of a Vdata or a Vdata Field: VSfnattrs 192

4.8.6 Retrieving the Index of a Vdata or Vdata Field Attribute Given the Attribute Name: VSfindattr 193

4.8.7 Querying Information on a Vdata or Vdata Field Attribute: VSattrinfo 193

4.8.8 Determining whether a Vdata Is an Attribute: VSisattr 194

4.9Obtaining Information about a Specific Vdata 199

4.9.1 Obtaining Vdata Information: VSinquire 199

4.9.2 Obtaining Linked Block Information: VSgetblockinfo 203

4.9.3 Obtaining Linked Block Information: VSgetblockinfo 203

4.9.4 VSQuery Vdata Information Retrieval Routines 204

4.9.5 Other Vdata Information Retrieval Routines 205

4.9.6 VF Field Information Retrieval Routines 206

**Chapter 5** - Vgroups (V API)

5.1Chapter Overview 209

5.2The Vgroup Data Model 209

5.2.1 Vgroup Names and Classes 209

5.2.2 Vgroup Organization 210

5.2.3 An Example Using Vgroups 211

5.3The Vgroup Interface 211

5.3.1 Vgroup Interface Routines 211

5.3.2 Identifying Vgroups in the Vgroup Interface 214

5.4Programming Model for the Vgroup Interface 214

5.4.1 Accessing Files and Vgroups: Vstart and Vattach 215

5.4.2 Terminating Access to Vgroups and Files: Vdetach and Vend 216

5.5Creating and Writing to a Vgroup 216

5.5.1 Assigning a Vgroup Name and Class: Vsetname and Vsetclass 217

5.5.2 Inserting Any HDF Data Object into a Vgroup: Vaddtagref 218

5.5.3 Inserting a Vdata or Vgroup Into a Vgroup: Vinsert 218

5.5.4 Building a Vgroup with or without Elements: VHmakegroup 218

5.6Reading from Vgroups 230

5.6.1 Locating Vgroups and Obtaining Vgroup Information 231

5.6.1.1 Locating Lone Vgroups: Vlone 231

5.6.1.2 Sequentially Searching for a Vgroup: Vgetid 231

5.6.1.3 Retrieving vgroups in a file or in a vgroup: Vgetvgroups 231

5.6.1.4 Determining Internal Vgroup: Vgisinternal 233

5.6.1.5 Obtaining the Name of a Vgroup: Vgetname 234

5.6.1.6 Obtaining the Length of a Vgroup’s Name: Vgetnamelen 234

5.6.1.7 Obtaining the Class Name of a Vgroup: Vgetclass 234

5.6.1.8 Obtaining the Length of a Vgroup’s Class Name: Vgetclassnamelen 235

5.6.1.9 Locating a Vgroup Given Its Name: Vfind 235

5.6.1.10 Locating a Vgroup Given Its Class Name: Vfindclass 235

5.6.2 Obtaining Information about the Contents of a Vgroup 239

5.6.2.1 Obtaining the Number of Objects in a Vgroup: Vntagrefs 239

5.6.2.2 Obtaining the Tag/Reference Number Pair of a Data Object within a Vgroup : Vgettagref 239

5.6.2.3 Obtaining the Tag/Reference Number Pairs of Data Objects in a Vgroup: Vgettagrefs 240

5.6.2.4 Testing Whether a Data Object Belongs to a Vgroup: Vinqtagref 241

5.6.2.5 Testing Whether a Data Object within a Vgroup is a Vgroup: Visvg 241

5.6.2.6 Testing Whether an HDF Object within a Vgroup is a Vdata: Visvs 241

5.6.2.7 Locating a Vdata in a Vgroup Given Vdata Fields: Vflocate 242

5.6.2.8 Retrieving the Number of Tags of a Given Type in a Vgroup: Vnrefs 242

5.6.2.9 Retrieving the Reference Number of a Vgroup: VQueryref 243

5.6.2.10 Retrieving the Tag of a Vgroup: VQuerytag 243

5.7Deleting Vgroups and Data Objects within a Vgroup 247

5.7.1 Deleting a Vgroup from a File: Vdelete 247

5.7.2 Deleting a Data Object from a Vgroup: Vdeletetagref 247

5.8Vgroup Attributes 248

5.8.1 Obtaining the Vgroup Version Number of a Given Vgroup: Vgetversion 248

5.8.2 Setting the Attribute of a Vgroup: Vsetattr 249

5.8.3 Retrieving the Index of a Vgroup Attribute Given the Attribute Name: Vfindattr 249

5.8.4 Obtaining the Total Number of Vgroup Attributes: Vnattrs and Vnattrs2 250

5.8.5 Obtaining Information on a Given Vgroup Attribute: Vattrinfo 251

5.8.6 Obtaining Information on a Given Vgroup Attribute: Vattrinfo2 251

5.8.7 Retrieving the Values of a Given Vgroup Attribute: Vgetattr 252

5.8.8 Retrieving the Values of a Given Vgroup Attribute: Vgetattr2 253

5.9Obsolete Vgroup Interface Routines 257

5.9.1 Determining the Next Vgroup or Vdata Identifier: Vgetnext 257

5.9.2 Determining the Number of Members and Vgroup Name: Vinquire 257

**Chapter 6** - 8-Bit Raster Images (DFR8 API)

6.1Chapter Overview 259

6.1.1 Required 8-Bit Raster Image Data Set Objects 259

6.1.1.1 8-Bit Raster Image Data Representation 259

6.1.1.2 8-Bit Raster Image Dimension 260

6.1.2 Optional 8-Bit Raster Image Data Set Objects 260

6.1.2.1 Palettes 260

6.1.3 Compression Method 260

6.2The 8-Bit Raster Image Interface 262

6.2.1 8-Bit Raster Image Library Routines 262

6.3Writing 8-Bit Raster Images 262

6.3.1 Storing a Raster Image: DFR8putimage and DFR8addimage 263

6.3.2 Adding a Palette to an RIS8 Object: DFR8setpalette 264

6.3.3 Compressing 8-Bit Raster Image Data: DFR8setcompress 266

6.3.4 Specifying the Reference Number of an RIS8: DFR8writeref 270

6.4Reading 8-Bit Raster Images 270

6.4.1 Reading a Raster Image: DFR8getimage 270

6.4.2 Querying the Dimensions of an 8-Bit Raster Image: DFR8getdims 271

6.4.3 Reading an Image with a Given Reference Number: DFR8readref 273

6.4.4 Specifying the Next 8-Bit Raster Image to be Read: DFR8restart 273

6.58-Bit Raster Image Information Retrieval Routines 273

6.5.1 Querying the Total Number of 8-Bit Raster Images: DFR8nimages 273

6.5.2 Determining the Reference Number of the Most-Recently-Accessed   
 8-Bit Raster Image: DFR8lastref 274

6.5.3 Determining the Reference Number of the Palette of the   
 Most-Recently-Accessed 8-Bit Raster Image: DFR8getpalref 274

6.6RIS8 Backward Compatibility Issues 274

6.6.1 Attribute "long\_name" Included in HDF for netCDF Compatibility 274

6.6.2 Raster Image Group Implementation with New RIS8 Tags 275

**Chapter 7** - 24-bit Raster Images (DF24 API)

7.1Chapter Overview 277

7.2The 24-Bit Raster Data Model 277

7.2.1 Required 24-Bit Raster Image Data Set Objects 277

7.2.1.1 24-Bit Raster Image Data Representation 277

7.2.1.2 24-Bit Raster Image Dimension 278

7.2.2 Optional 24-Bit Raster Image Data Set Objects 278

7.2.2.1 Compression Method 278

7.2.2.2 Interlace Modes 279

7.3The 24-Bit Raster Interface 280

7.3.1 24-Bit Raster Image Library Routines 280

7.4Writing 24-Bit Raster Images 281

7.4.1 Writing a 24-Bit Raster Image: DF24putimage and DF24addimage 281

7.4.2 Setting the Interlace Format: DF24setil 283

7.4.3 Compressing Image Data: DF24setcompress and d2sjpeg 284

7.5Reading 24-Bit Raster Images 286

7.5.1 Reading a Raster Image: DF24getimage 286

7.5.2 Determining the Dimensions of an Image: DF24getdims 287

7.5.3 Modifying the Interlacing of an Image: DF24reqil 287

7.5.4 Reading a 24-Bit Raster Image with a Given Reference Number: DF24readref 289

7.5.5 Specifying that the Next Image Read to be the First 24-Bit Raster  
 Image in the File: DF24restart 289

7.624-Bit Raster Image Information Retrieval Routines 290

7.6.1 Querying the Total Number of Images in a File: DF24nimages 290

7.6.2 Querying the Reference Number of the Most Recently Accessed 24-Bit  
 Raster Image: DF24lastref 290

**Chapter 8** - General Raster Images (GR API)

8.1Chapter Overview 293

8.2The GR Data Model 293

8.2.1 Required GR Data Set Components 294

8.2.2 Optional GR Data Set Components 295

8.3The GR Interface 295

8.3.1 GR Interface Routines 295

8.4Header Files Required by the GR Interface 297

8.5Programming Model for the GR Interface 297

8.5.1 Accessing Images and Files: GRstart, GRselect, and GRcreate 298

8.5.2 Terminating Access to Images and Files: GRendaccess and GRend 299

8.6Writing Raster Images 300

8.6.1 Writing Raster Images: GRwriteimage 300

8.6.2 Compressing Raster Images: GRsetcompress 309

8.6.3 Setting I/O Access Type for a Raster Image: GRsetaccesstype 309

8.6.4 External File Operations Using the GR Interface 310

8.6.4.1 Creating a Raster Image in an External File: GRsetexternalfile 310

8.6.4.2 Moving Raster Images to an External File 311

8.7Reading Raster Images 312

8.7.1 Reading Data from an Image: GRreadimage 312

8.7.2 Setting the Interlace Mode for an Image Read: GRreqimageil 312

8.8Difference between the SD and GR Interfaces 318

8.9Obtaining Information about Files and Raster Images 318

8.9.1 Obtaining Information about the Contents of a File: GRfileinfo 318

8.9.2 Obtaining Information about an Image: GRgetiminfo 319

8.9.3 Obtaining the Reference Number of a Raster Image from Its Identifier: GRidtoref 319

8.9.4 Obtaining the Index of a Raster Image from Its Reference Number: GRreftoindex 320

8.9.5 Obtaining the Index of a Raster Image from Its Name: GRnametoindex 320

8.9.6 Obtaining Compression Information for a Raster Image: GRgetcompinfo 320

8.9.7 Checking Whether a Raster Image Is To Be Mapped: GR2bmapped 321

8.10GR Data Set Attributes 325

8.10.1 Predefined GR Attributes 325

8.10.2 Setting User-defined Attributes: GRsetattr 325

8.10.3 Querying User-Defined Attributes: GRfindattr and GRattrinfo 329

8.10.4 Reading User-defined Attributes: GRgetattr 329

8.11Reading and Writing Palette Data Using the GR Interface 334

8.11.1 Obtaining a Palette Identifier: GRgetlutid 334

8.11.2 Obtaining the Number of Palettes Associated with an Image: GRgetnluts 335

8.11.3 Obtaining the Reference Number of a Specified Palette: GRluttoref 335

8.11.4 Obtaining Palette Information: GRgetlutinfo 335

8.11.5 Writing Palette Data: GRwritelut 336

8.11.6 Setting the Interlace Mode for a Palette: GRreqlutil 336

8.11.7 Reading Palette Data: GRreadlut 337

8.12Chunked Raster Images 344

8.12.1 Difference between a Chunked Raster Image and a Chunked SDS 345

8.12.2 Making a Raster Image a Chunked Raster Image: GRsetchunk 345

8.12.3 Writing a Chunked Raster Image: GRwritechunk 347

8.12.4 Reading a Chunked Raster Image: GRreadchunk 353

8.12.5 Obtaining Information about a Chunked Raster Image: GRgetchunkinfo 353

8.12.6 Setting the Maximum Number of Chunks in the Cache: GRsetchunkcache 354

**Chapter 9** - Palettes (DFP API)

9.1Chapter Overview 357

9.2The Palette Data Model 357

9.3The Palette API 358

9.3.1 Palette Library Routines 358

9.4Writing Palettes 358

9.4.1 Writing a Palette: DFPaddpal and DFPputpal 358

9.4.2 Specifying the Reference Number of a Palette: DFPwriteref 360

9.5Reading a Palette 360

9.5.1 Reading a Palette: DFPgetpal 361

9.5.2 Reading a Palette with a Given Reference Number: DFPreadref 362

9.5.3 Specifying the Next Palette to be Accessed to be the First Palette: DFPrestart 362

9.6Other Palette Routines 362

9.6.1 Querying the Number of Palettes in a File: DFPnpals 362

9.6.2 Obtaining the Reference Number of the Most Recently Accessed Palette: DFPlastref 363

9.7Backward Compatibility Issues 363

**Chapter 10** - Annotations (AN API)

10.1Chapter Overview 365

10.2The Annotation Data Model 365

10.2.1 Labels and Descriptions 365

10.2.2 File Annotations 366

10.2.3 Object Annotations 366

10.2.4 Terminology 367

10.3The AN interface 367

10.3.1 AN Library Routines 367

10.3.2 Type and Tag Definitions Used in the AN Interface 368

10.3.3 Programming Model for the AN Interface 368

10.3.4 Accessing Files and Annotations: ANstart, ANcreatef, and ANcreate 369

10.3.5 Terminating Access to Annotations and Files: ANendaccess and ANend 369

10.4Writing an Annotation: ANwriteann 370

10.5Reading Annotations Using the AN Interface 375

10.5.1 Selecting an Annotation: ANselect 375

10.5.2 Reading an Annotation: ANreadann 375

10.6Obtaining Annotation Information Using the AN Interface 379

10.6.1 Obtaining the Number of Annotations: ANfileinfo 379

10.6.2 Getting the Length of an Annotation: ANannlen 379

10.6.3 Obtaining the Number of Specifically-typed Annotations of a Data Object: ANnumann 380

10.6.4 Obtaining the List of Specifically-typed Annotation Identifiers of a Data Object: ANannlist 380

10.6.5 Obtaining the Tag/Reference Number Pair of the Specified  
 Annotation Index and Type: ANget\_tagref 381

10.6.6 Obtaining the Tag/Reference Number Pair from a Specified Annotation Identifier: ANid2tagref 381

10.6.7 Obtaining the Annotation Identifier from a Specified Tag/Reference Number Pair: ANtagref2id 382

10.6.8 Obtaining an Annotation Tag from a Specified Annotation Type: ANatype2tag 382

10.6.9 Obtaining an Annotation Type from a Specified Object Tag: ANtag2atype 382

**Chapter 11** - Single-file Annotations (DFAN API)

11.1Chapter Overview 389

11.2The Single-file Annotation Interface 389

11.2.1 DFAN Library Routines 389

11.2.2 Tags in the Annotation Interface 390

11.3Programming Model for the DFAN Interface 391

11.4Writing Annotations 392

11.4.1 Assigning a File Label: DFANaddfid 392

11.4.2 Assigning a File Description: DFANaddfds 392

11.4.3 Assigning an Object Label: DFANputlabel 393

11.4.4 Assigning an Object Description: DFANputdesc 394

11.5Reading Annotations 396

11.5.1 Reading a File Label: DFANgetfidlen and DFANgetfid 396

11.5.2 Reading a File Description: DFANgetfdslen and DFANgetfds 396

11.5.3 Reading an Object Label: DFANgetlablen and DFANgetlabel 398

11.5.4 Reading an Object Description: DFANgetdesclen and DFANgetdesc 399

11.6Maintenance Routines 401

11.6.1 Clearing the DFAN Interface Internal Structures and Settings: DFANclear 401

11.7Determining Reference Numbers 401

11.7.1 Determining a Reference Number for the Last Object Accessed: DF\*lastref and DF\*writeref 402

11.7.2 Querying a List of Reference Numbers for a Given Tag: DFANlablist 403

11.7.3 Locate an Object by Its Tag and Reference Number: Hfind 405

**Chapter 12** - Single-File Scientific Data Sets (DFSD API)

12.1Chapter Overview 407

12.2The DFSD Scientific Data Set Data Model 407

12.2.1 Required DFSD SDS Objects 407

12.2.1.1 Dimensions 408

12.2.2 Optional DFSD SDS Objects 408

12.2.2.1 Dimension Scales 408

12.2.2.2 Predefined Attributes 408

12.3The Single-File Scientific Data Set Interface 408

12.3.1 DFSD Library Routines 408

12.3.2 File Identifiers in the DFSD Interface 409

12.4Writing DFSD Scientific Data Sets 410

12.4.1 Creating a DFSD Scientific Data Set: DFSDadddata and DFSDputdata 410

12.4.2 Specifying the Data Type of a DFSD SDS: DFSDsetNT 411

12.4.3 Overwriting Data for a Given Reference Number: DFSDwriteref 412

12.4.4 Writing Several Data Sets: DFSDsetdims and DFSDclear 412

12.4.5 Preventing the Reassignment of DFSD Data Set Attributes: DFSDsetdims 413

12.4.6 Resetting the Default DFSD Interface Settings: DFSDclear 413

12.5Reading DFSD Scientific Data Sets 414

12.5.1 Reading a DFSD SDS: DFSDgetdata 414

12.5.2 Specifying the Dimensions and Data Type of an SDS: DFSDgetdims and DFSDgetNT 415

12.5.3 Determining the Number of DFSD Data Sets: DFSDndatasets and DFSDrestart 416

12.5.4 Obtaining Reference Numbers of DFSD Data Sets: DFSDreadref and DFSDlastref 417

12.6Slabs in the DFSD Interface 417

12.6.1 Accessing Slabs: DFSDstartslab and DFSDendslab 417

12.6.2 Writing Slabs: DFSDwriteslab 418

12.6.3 Reading Slabs: DFSDreadslab 419

12.7Predefined Attributes and the DFSD Interface 420

12.7.1 Writing Data Set Attributes 420

12.7.1.1 Assigning String Attributes to a DFSD SDS: DFSDsetlengths and DFSDsetdatastrs 420

12.7.1.2 Assigning Value Attributes to a DFSD SDS: DFSDsetfillvalue, DFSDsetrange, and  
 DFSDsetcal 421

12.7.2 Reading DFSD Data Set Attributes 424

12.7.2.1 Reading Data Set Attributes: DFSDgetdatalen and DFSDgetdatastrs 424

12.7.2.2 Reading the Value Attributes of a DFSD Data Set: DFSDgetfillvalue and DFSDgetcal 426

12.7.3 Writing the Dimension Attributes of a DFSD SDS 427

12.7.3.1 Writing the String Attributes of a Dimension: DFSDsetlengths and DFSDsetdimstrs 427

12.7.3.2 Writing a Dimension Scale of a DFSD SDS: DFSDsetdimscale 428

12.7.4 Reading the Dimension Attributes of a DFSD SDS 428

**Chapter 13** - Error Reporting

13.1Chapter Overview 429

13.2The HDF Error Reporting API 429

13.3Error Reporting in HDF 429

13.3.1 Writing Errors to a File: HEprint 430

13.3.2 Returning the Code of the Nth Most Recent Error: HEvalue 430

13.3.3 Returning the Description of an Error Code: HEstring/hestringf 430

13.3.4 Clearing the error stack: HEclear 431

**Chapter 14** - HDF Performance Issues

14.1Chapter Overview and Introduction 435

14.2Examples of HDF Performance Enhancement 435

14.2.1 One Large SDS versus Several Smaller SDSs 436

14.2.2 Sharing Dimensions between Scientific Data Sets 437

14.2.3 Setting the Fill Mode 439

14.2.4 Disabling Fake Dimension Scale Values in Large One-dimensional Scientific Data Sets 440

14.3Data Chunking 442

14.3.1 What Is Data Chunking? 442

14.3.2 Writing Concerns and Reading Concerns in Chunking 443

14.3.3 Chunking without Compression 443

14.3.4 Chunking with Compression 446

14.3.5 Effect of Chunk Size on Performance 447

14.3.6 Insufficient Chunk Cache Space Can Impair Chunking Performance 447

14.4Block Size Tuning Issues 451

14.4.1 Tuning Data Descriptor Block Size to Enhance Performance 451

14.4.2 Tuning Linked Block Size to Enhance Performance 452

14.4.3 Unlimited Dimension Data Sets (SDSs and Vdatas) and Performance 453

**Chapter 15** - HDF Command-line Utilities

15.1Chapter Overview 455

15.2Displaying the Contents of an HDF File: hdp (or HDF Dumper) 456

15.2.1 General Description 456

15.2.2 Command-line Syntax 456

15.3Comparing two HDF Files: hdiff 460

15.3.1 General Description 460

15.3.2 Command-line Syntax 460

15.3.3 Examples 461

15.4Displaying Vdata Information: vshow 461

15.4.1 General Description 461

15.4.2 Command-line Syntax 461

15.4.3 Examples 461

15.5Converting Floating-point or Integer Data to SDS or RIS8: hdfimport 461

15.5.1 General Description 461

15.5.2 Command-line Syntax 462

15.5.3 Structure of Data in non-HDF Input Files 464

15.6Converting 8-Bit Raster Images to the HDF Format: r8tohdf 465

15.6.1 General Description 465

15.6.2 Command-line Syntax 465

15.6.3 Examples 466

15.7Converting 24-Bit Raw Raster Images to RIS8 Images: r24hdf8 466

15.7.1 General Description 466

15.7.2 Command-line Syntax 466

15.7.3 Examples 467

15.8Converting Raw Palette Data to the HDF Palette Format: paltohdf 467

15.8.1 General Description 467

15.8.2 Command-line Syntax 467

15.9Extracting 8-Bit Raster Images and Palettes from HDF Files: hdftor8 467

15.9.1 General Description 467

15.9.2 Command-line Syntax 467

15.9.3 Examples 468

15.10Extracting Palette Data from an HDF File: hdftopal 468

15.10.1 General Description 468

15.10.2 Command-line Syntax 468

15.11Converting Several RIS8 Images to One 3D SDS: ristosds 468

15.11.1 General Description 468

15.11.2 Command-line Syntax 468

15.11.3 Examples 469

15.12Converting an HDF RIS24 Image to an HDF RIS8 Image: hdf24hdf8 469

15.12.1 General Description 469

15.12.2 Command-line Syntax 469

15.13Compressing RIS8 Images in an HDF File: hdfcomp 469

15.13.1 General Description 469

15.13.2 Command-line Syntax 469

15.13.3 Examples 469

15.14Compressing an HDF File: hdfpack 470

15.14.1 General Description 470

15.14.2 Command-line Syntax 470

15.14.3 Examples 470

15.15Reformatting an HDF File: hrepack 470

15.15.1 General Description 470

15.15.2 Command-line Syntax 471

15.16Creating Vgroups and Vdatas: vmake 472

15.16.1 General Description 472

15.16.2 Command-line Syntax 472

15.16.3 Examples 472

15.17Listing Basic Information about Data Objects in an HDF File: hdfls 473

15.17.1 General Description 473

15.17.2 Command-line Syntax 473

15.17.3 Examples 474

15.18Editing the Contents of an HDF File: hdfed 475

15.18.1 General Description 475

15.18.2 Command-line Syntax 476

15.19Working with Both HDF4 and HDF5 File Formats 485

15.20Converting an HDF File to a GIF File: hdf2gif 485

15.20.1 General Description 485

15.20.2 Command-line Syntax and Requirements: 485

15.20.3 Structure of the GIF File 486

15.20.4 Building the Utility 486

15.21Converting a GIF File to an HDF File: gif2hdf 486

15.21.1 General Description 486

15.21.2 Command-line Syntax and Requirements 486

15.21.3 Structure of the GIF and HDF Files and the Mapping between Them 487

15.21.4 Building the Utility 488

15.22Compiling C applications that Use HDF4: h4cc 488

15.22.1 General Description 488

15.22.2 Command-line Syntax 488

15.22.3 Examples 489

15.23Compiling Fortran applications that Use HDF4: h4fc 489

15.23.1 General Description 489

15.23.2 Command-line Syntax 489

15.23.3 Example 490

15.24Updating HDF4 Compiler Tools after an Installation in a New Location: h4redeploy 490

15.24.1 General Description 490

15.24.2 Command-line Syntax 490

**Chapter 16** - Raw Data Information

16.1Chapter Overview 493

16.2The Data Information Retrieval Routines 493

16.3Addition to the AN Interface 494

16.3.1 Retrieving Data Information of an Annotation: ANgetdatainfo 494

16.4Addition to the SD Interface 495

16.4.1 Retrieving Data Information of an SDS: SDgetdatainfo 495

16.4.2 Retrieving Data Information of an Attribute: SDgetattdatainfo 496

16.4.3 Retrieving Data Information of a DFSD API Attribute: SDgetoldattdatainfo 496

16.4.4 Retrieving Data Information of an SDS Annotation: SDgetanndatainfo 498

16.5Addition to the GR Interface 499

16.5.1 Retrieving Data Information of a Raster Image: GRgetdatainfo 499

16.5.2 Retrieving Data Information of a GR API Attribute: GRgetattdatainfo 500

16.6Addition to the V Interface 500

16.6.1 Retrieving Data Information of a V API Attribute: Vgetattdatainfo 500

16.7Addition to the VS Interface 501

16.7.1 Retrieving Data Information of a Vdata: VSgetdatainfo 501

16.7.2 Retrieving Data Information of a VS API Attribute: VSgetattdatainfo 502

- Appendices

Appendix A Reserved HDF Tags

A.1 Overview 505

A.2 Tag Types and Descriptions 505

Appendix B HDF Installation Overview

B.1 General HDF Installation Overview 509

B.1.1 Acquiring the HDF Library Source 509

B.1.2 Building the HDF Library Source 509

Appendix C Attributes in HDF

C.1 Attribute Overview 510

C.2 Underlaying storage issues 510

C.2.1 Predefined Attributes in DFSD API 511

C.2.2 Vgroup Attribute Without Vsetattr 512