

CERES Software Bulletin 97-10

Revision 1 - January 23, 1998

How to Read Metadata Using the Wrapper

Maria Vallas Mitchum (m.v.mitchum@larc.nasa.gov)

Alice Fan (t.f.fan@larc.nasa.gov)

1.0 Purpose

This bulletin describes the usage of the CERESlib metadata wrapper subroutines which read metadata from a previously-generated CERES output granule and the Process Control File (PCF) requirements for metadata retrieval. Computer Bulletin 97-06 contains a detailed description of the CERES metadata policy, approach, and produced metadata. Computer Bulletin 97-08 describes how to write metadata to a granule.

2.0 The Calling Sequence to the ReadMeta and ReadHeader Subroutines

The metadata wrapper is a Fortran90 module, named meta_util, that has three public subroutines: WriteMeta, ReadMeta, and ReadHeader. Subroutine ReadMeta reads ODL formatted metadata from an HDF file header and the load file metadata (.met file) of all file types. Subroutine ReadHeader reads ASCII formatted metadata from the header of a non-HDF file. The calling sequences for both ReadMeta and ReadHeader subroutines are the same and are written as follows:

```
ReadMeta (LID, Attr_Name, Value, version)
ReadHeader (LID, Attr_Name, Value, version)
```

The four parameters are described below:

- 1) **LID**: the Logical ID for the file to be read, an integer in the input file section of the PCF.
- 2) **Attr_Name**: attribute name, character string. The possible values for this parameter are:
 - (A) a single attribute name, (i.e CERProductionDateTime, CERDataDateYear). The complete list of the CERES selected attribute names are listed in section 4.0. The valid values for Attr_Name are not limited to this list, and should include any Product Specific Attributes (PSA) previously defined for the granule.
 - (B) "Header": to retrieve the entire set of header attributes, excluding the input filename listing
 - (C) "Input": to retrieve the input filenames for a product
- 3) **Value**: This parameter can be one of many data types depending on the attribute or attributes to be retrieved.

One value: If an attribute has a single value, the parameter "Value" should be a single value parameter of the expected data type (integer, real4, real8, or character string). For example, the parameter Value should be a character string when the Attr_Name is entered

as “CERProductionDateTime”, and the parameter Value should be an integer when the Attr_Name is entered as “NumberOfRecords”.

An array of values for one attribute: If the attribute is an array of values, the parameter “Value” should be an array of the expected data type. For example, the parameter Value should be an array of five real numbers, when Attr_Name is entered as “GRingPointLatitude” and the parameter Value should be an array of character string when Attr_Name is entered as “Input”.

An array of values for multiple attributes: When the Attr_Name is entered as “Header”, all but one, “InputPointer”, of the CERES baseline metadata attributes are returned at one time in the array “Value”. This array must contain at least 34 elements of data type “meta_type”, which is described in the Computer Bulletin 97-08 “How to Write Metadata for a Granule using the Wrapper”. For reference, the following is a listing of the definition of the “meta_type” data type.

```
TYPE meta_type
  CHARACTER (LEN=PGSd_MET_NAME_L) :: Name !max length = 30
  INTEGER :: IntVal
  REAL :: RealVal
  REAL(real8) :: DoubleVal
  CHARACTER (LEN= 256) :: StringVal
END TYPE meta_type
```

The order of the attributes is listed in Section 4.6.

- 4) **version:** An optional parameter, integer, to indicate which of the files with the same logical ID is to be read. The default value is 1.

3.0 Examples of Calling ReadMeta Subroutine

3.1 Read Individual Attributes One at a Time

The following are examples of single attribute retrievals.

```
CHARACTER (LEN=2) :: daystr
REAL (real4) :: West
REAL (real4) :: Real4_Value(5)
```

```
CALL ReadMeta(100, Attr_Name= “CERDataDateDay”, Value = daystr)
```

```
CALL ReadMeta(100, Attr_Name= “WestBoundingCoordinate”, Value=West)
```

```
CALL Readmeta(100, Attr_Name = “GRingPointLatitude”, Value = Real4_Value)
```

3.2 Read Header Attributes

When the Attr_Name is entered as “Header”, all of the CERES selected attributes are returned

except the 'InputPointer'. The parameter to receive the returned values (i.e. GranFile_Attr in the example below) should be an array of at least 34 elements. Additional elements are required if PSAs are included. If fewer elements are defined, a full set of header attributes will not be returned because the wrapper only returns parameters to fit the size of the array.

The following is an example usage of the ReadMeta subroutine to retrieve the Header information and one PSA.

```
USE meta_util, ONLY : meta_type, ReadMeta
TYPE(meta_type)    :: GranFile_Attr(35)           !* should be at least 34 *

CALL ReadMeta(100,"Header", GranFile_Attr)
DO i = 1, 35
  IF (GranFile_Attr(i)%IntVal .ne. INT4_DFLT) &
    print*, Trim(GranFile_Attr(i)%Name)," = ", GranFile_Attr(i)%IntVal
  IF (GranFile_Attr(i)%RealVal .ne. Real4_DFLT) &
    print*, Trim(GranFile_Attr(i)%Name), " = ", GranFile_Attr(i)%RealVal
  IF (GranFile_Attr(i)%DoubleVal .ne. Real8_DFLT) &
    print*, Trim(GranFile_Attr(i)%Name), " = ", GranFile_Attr(i)%DoubleVal
  IF (GranFile_Attr(i)%StringVal .ne. "") &
    print*, Trim(GranFile_Attr(i)%Name)," = ", Trim(GranFile_Attr(i)%StringVal)
END DO
```

3.3 Read Input Filenames

When the Attr_Name is entered as "Input", all of the input filenames stored in the requested metadata are returned. The parameter Value must be an array of data type "input_type". The array size of the parameter Value should be large enough to store the returned input filenames.

The "input_type" is defined in the meta_util module as follows:

```
TYPE input_type
  CHARACTER(LEN= PGSD_PC_PATH_LENGTH_MAX+PGSD_PC_FILE_NAME_MAX):: filename
END TYPE input_type
```

where PGSD_PC_PATH_LENGTH_MAX = 768 and
 PGSD_PC_FILE_NAME_MAX = 256 are defined in the Toolkit.

The following is an example usage of the ReadMeta subroutine to read the input filenames.

```
USE meta_util, ONLY : input_type, ReadMeta
TYPE(input_type) :: inputfiles(10)

CALL ReadMeta(LID,"input",inputfiles)
DO i=1,10
  IF (inputfiles(i)%filename == "$") EXIT
  print*,"inputfile=", TRIM(inputfiles(i)%filename)
END DO
```

Note: The Toolkit function, used in the subroutine, returns an array of input filenames and places a "\$" to signal the end; the user can check against this character to terminate the list.

4.0 CERES Header Attribute Names

CERES has selected a set of baseline metadata attributes. The attribute names are listed below and should be used when calling the ReadMeta or ReadHeader subroutine to retrieve the information.

4.1 Attributes Having a Character String Value

When the following attribute names are retrieved from the ReadMeta or ReadHeader subroutine, the parameter Value must have the data type of character string.

| | |
|----------------------|---------------------------------|
| ShortName | AutomaticQualityFlag |
| CERPGEName | AutomaticQualityFlagExplanation |
| SamplingStrategy | ExclusionGRingFlag |
| ProductionStrategy | AssociatedPlatformShortName |
| CERDataDateYear | AssociatedInstrumentShortName |
| CERDataDateMonth | ImagerShortName |
| CERDataDateDay | LocalGranuleID |
| CERHrOfMonth | QAGranuleFileName |
| ProductGenerationLOC | ValidationFileName |
| RangeBeginningDate | LocalVersionID |
| RangeBeginningTime | InputPointer |
| RangeEndingDate | PGEVersion |
| RangeEndingTime | CERProductionDateTime |

4.2 Attributes Having a Real4 or Real8 Value

When the following attribute names are retrieved from the ReadMeta or ReadHeader subroutine, the parameter Value must have the data type of either real4 or real8.

| | |
|-------------------------|----------------------------|
| WestBoundingCoordinate | CERWestBoundingCoordinate |
| NorthBoundingCoordinate | CERNorthBoundingCoordinate |
| EastBoundingCoordinate | CEREastBoundingCoordinate |
| SouthBoundingCoordinate | CERSouthBoundingCoordinate |

4.3 Attributes Having an Array of Real4 or Real8 Values

When the following attribute names are retrieved from the ReadMeta or ReadHeader subroutine, the parameter Value must have the data type of either real4 array or real8 array.

| | |
|---------------------|------------------------|
| GRingPointLatitude | CERGringPointLatitude |
| GRingPointLongitude | CERGringPointLongitude |

4.4 Attributes Having an Integer Value

When the following attribute name is retrieved from the ReadMeta or ReadHeader subroutine, the parameter Value must have the data type of an integer number.

| | |
|-----------------|-----------|
| NumberOfRecords | VersionID |
|-----------------|-----------|

4.5 Attributes Having an Array of Integer Values

When the following attribute name is retrieved from the ReadMeta or ReadHeader subroutine, the parameter Value must have the data type of an integer array.

GRingPointSequenceNo

4.6 Ordered Listing of Header Attributes

The Header metadata values are ordered in the following sequence, where the attribute “Input-Pointer” is not included in the listing.

1. ShortName
2. VersionID
3. CERPGEName
4. SamplingStrategy
5. ProductionStrategy
6. CERDataDateYear
7. CERDataDateMonth
8. CERDataDateDay
9. CERHrOfMonth
10. RangeBeginningDate
11. RangeBeginningTime
12. RangeEndingDate
13. RangeEndingTime
14. AssociatedPlatformShortName
15. AssociatedInstrumentShortName
16. LocalGranuleID
17. PGEVersion
18. CERProductionDateTime
19. LocalVersionID
20. ProductGenerationLOC
21. NumberOfRecords

The following parameter locations (22 - 25) will be dependent on which set are present in the metadata record.

Set A:

22. WestBoundingCoordinate
23. NorthBoundingCoordinate
24. EastBoundingCoordinate
25. SouthBoundingCoordinate

Set B:

22. GRingPointLatitude
23. GRingPointLongitude
24. GRingPointSequenceNo
25. ExclusionGRingFlag

The following parameter locations (26 - 29) will be dependent on which set are present in the

metadata record.

Set A:

- 26. CERWestBoundingCoordinate
- 27. CERNorthBoundingCoordinate
- 28. CEREastBoundingCoordinate
- 29. CERSouthBoundingCoordinate

Set B:

- 26. CERGringPointLatitude
- 27. CERGringPointLongitude

Note: Items 28 and 29 here are duplicates of items 24 and 25, placeholders for consistency.

- 28. GRingPointSequenceNo
- 29. ExclusionGRingFlag

- 30. AutomaticQualityFlag
- 31. AutomaticQualityFlagExplanation
- 32. QAGranuleFileName
- 33. ValidationFileName
- 34. ImagerShortName

5.0 PCF Requirements

To read metadata from an input file, the input file must be an entry in the INPUT FILE section of the PCF.

5.1 Read from an HDF file

The following entry in the PCF is used to read metadata from an HDF or HDF-EOS file. The filenames in the second and sixth fields must be the same. Note here that the .met filename extension must not be used for HDF or HDF-EOS files in the sixth field of the PCF record entry.

```
<LID>| <input filename>| <input file path> |||<input filename>|<version>
```

5.2 Read from a Direct Access Binary File

Subroutine ReadHeader reads metadata on a direct access binary file. Either one of the following entries can be used in the PCF.

```
<LID>| <nonHDF input filename>| <input file path> ||||<version> or
```

```
<LID>| <nonHDF input filename>| <input file path> ||||<nonHDF input filename>.met|<version>
```

The subroutine ReadHeader accesses the 2nd field of the PCF entry for the filename. There are no Metadata Toolkit functions for reading metadata on a non-HDF file, therefore the filename in the 6th field of the PCF is ignored.

5.3 Read from the Load File (.met file)

Normal usage of the metadata read routines would read directly from the data file itself. But it is possible to read metadata from the .met load file. The ReadMeta subroutine uses the Metadata Toolkit functions to read the metadata from the .met file. The following entry in the PCF is required to identify the .met filename. Please note that the file '<input filename>.met' must

exist in the file system.

The PCF entry for a .met file of an HDF file is:

<LID>| <input filename>| <input file path> |||<input filename>.met|<version>

The PCF entry for a .met file of a nonHDF file is:

<LID>| dummy| <input file path> |||<input filename>.met|<version>

6.0 Read an Attribute with Multiplicity

Eleven of the CERES selected attributes are multiplicity attributes. These attributes are defined with class = "M" in the MCF template, which indicates multiple classes of these attributes are allowed. They are:

| | |
|--------------------------|---|
| GRingPointLatitude.1 | AutomaticQualityFlag.1 |
| GRingPointLongitude.1 | AutomaticQualityFlagExplanation.1 |
| GRingPointSequenceNo.1 | (ParameterName.1 - FixedNameForGranule) |
| ExclusionGRingFlag.1 | |
| CERGRingPointLatitude.1 | AssociatedPlatformShortname.1 |
| CERGRingPointLongitude.1 | AssociatedInstrumentShortName.1 |

For example, if more than one class of polygons were to be used to describe the spatial domain of a granule, multiple sets of GRingPointLatitude, GRingPointLongitude, GRingPointSequenceNo, and ExclusionGRingFlag would be required. The official name for the first GRingPointLatitude is GRingPointLatitude.1.

Since the CERES selected attributes have only one instance for each attribute at the time of this writing, the wrapper allows the user to enter GRingPointLatitude.1 or its default GRingPointLatitude to the wrapper. If, in the future, the CERES metadata evolves into the need of multiple sets of these attributes, a number will be appended to the attribute name in order to retrieve the desired value.

A near term example of this feature will be required when the CERES Multi-Satellite Processing begins. The AssociatedPlatformShortname will be expanded to store each CERES platform and AssociatedInstrumentShortName will be expanded to store each CERES instrument processed in an executable.

Another example of this feature is the AutomaticQualityFlag. CERES has one set of flags for the entire granule. The ECS data model allows QA flags for each individual parameter or for each set of parameters. Therefore, at a later time, if CERES requires parameter level quality flags then multiple sets of the attribute 'ParameterName' with the associated AutomaticQualityFlag and AutomaticQualityFlagExplanation can be introduced in the metadata summary. Currently, the ParameterName is a fixed value set to "GranuleParameters" in the wrapper, reflecting the evaluation of the granule.

7.0 How to Document Input Files from a Previous PGE Into the Next PGE

If the user wishes to extract the input files that were used to create the present input file being used in the current process, then the following steps should be used. Call the ReadMeta or ReadHeader subroutine to retrieve the input files which were used to create the input file you are using. Save them into a Product Specific Array with the “name” field as “InputPointer”. Pass them through the parameter SpecificAttr of the WriteMeta subroutine. The wrapper takes the information and concatenates with the current input files and writes the output to the InputPointer attribute of the output granule. The Product Specific Array (SpecificAttr) is described in Software bulletin 97-08.