データの読み方　その3
HDF5 ライブラリ利用編

奥山　新

JAXA/EORC

2013/02/01 AMSR-2 データ利用講習会

目的

- HDFライブラリを用いたプログラミングの流れを把握する
- AMTKを使う場合との違いを把握する

講義の流れ

- HDF5の紹介
- AMTKを使う場合との違い
- L1Bデータの構成
- プログラミングの流れ
- 便利ツールの紹介（h5dump, hdfview）

HDF5(Hierarchical Data Format 5)とは

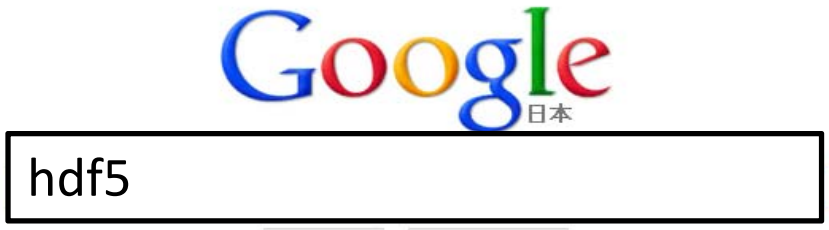
- 米国NCSA(the National Center for Supercomputing Applications)で開発
- 衛星データの一般的な表現形式のひとつ

- 特徴
  - 自己記述型（要素名、次元、サイズ、単位、…）
  - データアクセスはライブラリを通じて行う
  - HDF4とは互換性がない

- ライブラリ
  - C, C++, Fortran, Java で利用可能
  - 要素名を指定すればデータが取得できる

HDFライブラリのインストール

- 取得元
  - http://www.hdfgroup.org/HDF5/
- コンパイル済のバイナリ版は Windows, Linux 用が用意されている。ソースコードも公開。
- szipライブラリ、zlibライブラリが必要。

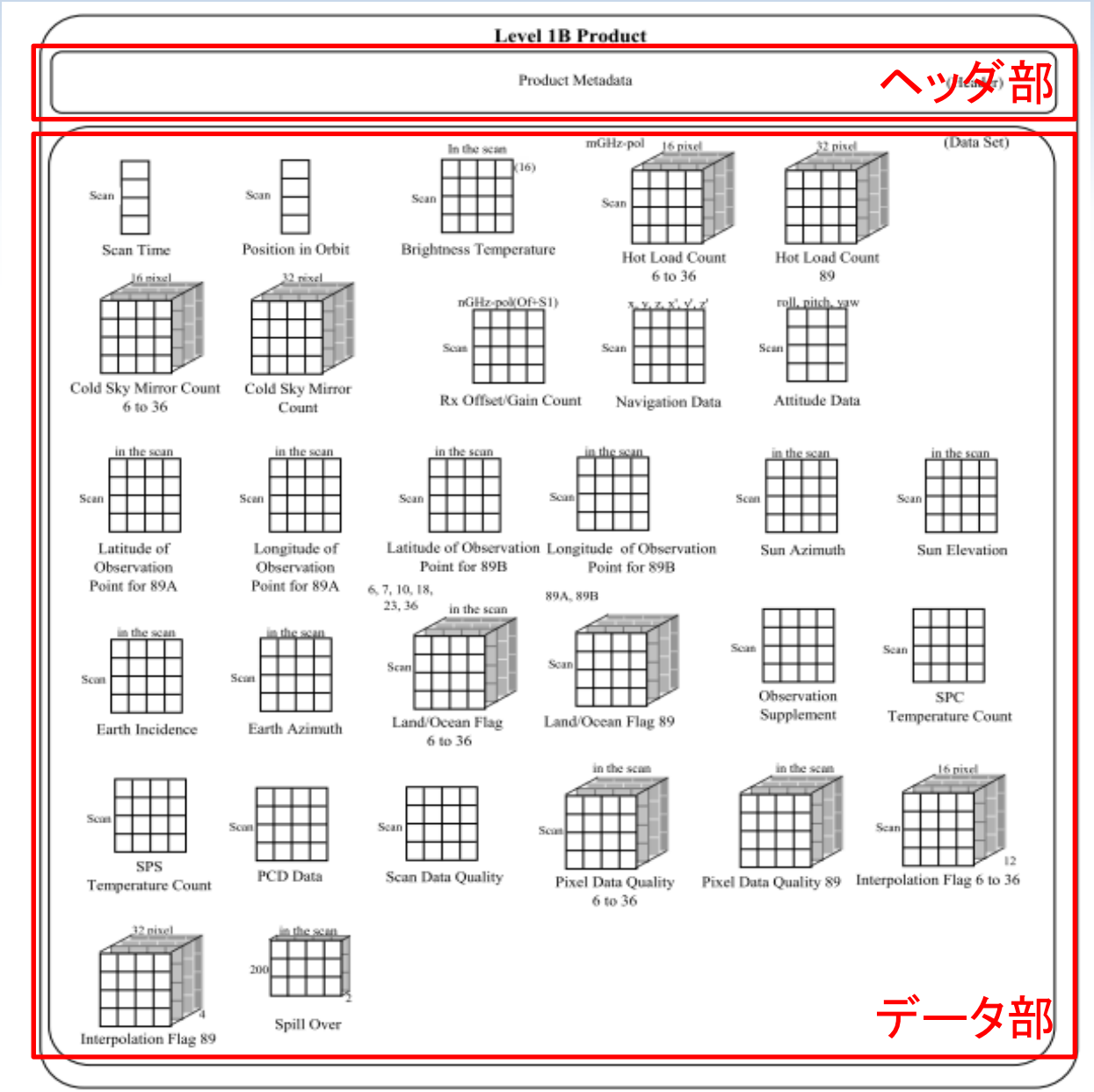


AMTKを使う場合との違い

- データセット取得のための関数はひとつだけ
- いくつかの処理を明示的に行う必要がある
  - スケールの変換
  - 低周波数チャンネルの緯経度算出（モジュール有り）
  - 時刻変換（TAI93→YMDH）（モジュール有り）
- AMTKではなくHDFを選ぶ理由
  - 既にHDFの扱いに長けていて、今さらAMTKを覚える気がない
  - HDFライブラリもAMTKも未経験だが、AMSR-2 以外にも多くの衛星データを扱う可能性がある
  - 単なる好み

L1Bデータの構成

- ヘッダ部
  - 観測開始・終了時刻、スキャンライン数、コレジストレーション係数等
  - テキストデータ
  - メタデータ、File attribute とも。
- データ部
  - 輝度温度、緯度経度、観測時刻、海陸フラグ等
  - 1～3次元のバイナリデータ
  - 各データセットにも個別の付属情報（attribute）が含まれる。



読み取り可能なデータ

メタデータ（ヘッダ部）	格納データ
* GeophysicalName - GranuleID - ObservationStartDateTime - EquatorCrossingDateTime - ObservationEndDateTime * NumberOfScans - OverlapScans - CoRegistrationParameterA1 -CoRegistrationParameterA2 ※HDF5のFORTRAN90ライブラリでは可変長文字列メタデータを読込めません。	* Scan Time * Latitude of Observation Point for 89A - Latitude of Observation Point for 89B - Longitude of Observation Point for 89A - Longitude of Observation Point for 89B * Brightness Temperature (6.9GHz,H) - Brightness Temperature (6.9GHz,V) - Brightness Temperature (7.3GHz,H) - Brightness Temperature (7.3GHz,V) - Brightness Temperature (10.7GHz,H) - Brightness Temperature (10.7GHz,V) - Brightness Temperature (18.7GHz,H) - Brightness Temperature (18.7GHz,V) - Brightness Temperature (23.8GHz,H) - Brightness Temperature (23.8GHz,V) - Brightness Temperature (36.5GHz,H) - Brightness Temperature (36.5GHz,V) - Brightness Temperature (89.0GHz-A,H) - Brightness Temperature (89.0GHz-A,V) - Brightness Temperature (89.0GHz-B,H) - Brightness Temperature (89.0GHz-B,V) * Pixel Data Quality 6 to 36 - Pixel Data Quality 89 * Land_Ocean Flag 6 to 36 - Land_Ocean Flag 89 * Earth Incidence - Earth Azimuth
89G Aホーン緯度経度から算出するデータ	
* 緯度経度(低周波平均) - 緯度経度(6G) - 緯度経度(7G) - 緯度経度(10G) - 緯度経度(18G) - 緯度経度(23G) - 緯度経度(36G)	

FortranとCの主な違い

- Fortranでは…
  - 可変長文字列メタデータ(＝ヘッダ部)が読み取れない
    - スキャンライン数
      - メタデータではなくデータセットのアトリビュート情報から読み取る。
    - コレジストレーション係数
      - 低周波数チャンネルの観測緯度経度を求めるための係数
      - 固定値を使う。

プログラミングの流れ

	C	Fortran	
ヘッダファイル等 読み込み	#include “hdf5.h”	use hdf5	
初期化処理	H5open()	H5open_f()	
HDFファイルを開く	H5Fopen()	H5Fopen_f()	ファイルハンドル値 取得
ヘッダ部を開く	H5Aopen()	H5Aopen_f()	アトリビュートハンドル値 取得
（アトリビュート）読む	H5Aread()	H5Aread_f()	
閉じる	H5Aclose()	H5Aclose_f()	
データ部を開く	H5Dopen()	H5Dopen_f()	データセットハンドル値 取得
読む	H5Dread()	H5Dread_f()	
閉じる	H5Dclose()	H5Dclose_f()	
HDFファイル閉じる	H5Fclose()	H5Fclose_f()	
終了処理	H5close()	H5close_f()	

- その他、必要に応じてデータの型・サイズの取得や、スケール変換用のパラメータ等を取得

初期化処理

C

```
ret = H5open();
ret: [戻り値]失敗の場合は負の値
```

Fortran

```
call H5open_f(ret)
ret: [戻り値]失敗の場合は負の値
```

HDFファイルを開く

C

```
fhnd = H5Fopen(fn, label1, label2);
```

fn: オープンするファイル名を指定します
label1: アクセスモードを指定しますH5F_ACC_RDONLYで読込専用になります
label2: H5P_DEFAULTを指定します
fhnd: [戻り値]開いたファイルのハンドル値、失敗の場合は負の値

Fortran

```
call H5Fopen_f(fn,label1,fhnd,ret,label2)
```

fn: オープンするファイル名を指定します
label1: アクセスモードを指定しますH5F_ACC_RDONLY_Fで読込専用になります
fhnd: [戻り値]開いたファイルのハンドル値
ret: [戻り値]失敗の場合は負の値
label2: H5P_DEFAULT_Fを指定します

メタデータ(ヘッダ部)読込み

C

◇アトリビュートのオープン
ahnd = H5Aopen(fhnd, nam, label);
fhnd: ファイルハンドル値を指定します
nam: アトリビュート名を指定します
label: H5P_DEFAULTを指定します
ahnd: [戻り値]アトリビュートのハンドル値
失敗の場合は負の値

◇アトリビュートタイプの取得
atyp = H5Aget_type(ahnd);
ahnd: アトリビュートハンドル値を指定します
atyp: [戻り値]アトリビュートタイプ値
失敗の場合は負の値

◇アトリビュート読込み
ret = H5Aread(ahnd, otyp, buf);
ahnd: アトリビュートハンドル値を指定します
otyp: 出力変数の型を指定します(読込んだデータは、出力変数の型へ自動変換されます)
buf: 出力変数を指定します
ret: [戻り値]失敗の場合は負の値

◇アトリビュートのクローズ
ret = H5Aclose(ahnd);
ahnd:アトリビュートハンドル値を指定します
ret: [戻り値]失敗の場合は負の値

データを読み込む

C

◇データセットのオープン

dhnd = H5Dopen(fhnd, nam, label);

fhnd: ファイルハンドル値を指定します

nam: データセット名を指定します

label: H5P_DEFAULTを指定します

dhnd: [戻り値]データセットのハンドル値、失敗の場合は負の値

◇データ読込み

ret = H5Dread(dhnd, otyp, label1, label2, label3, buf);

dhnd: データセットハンドル値を指定します

otyp: 出力変数の型を指定します(読込んだデータは、出力変数の型へ自動変換されます)

label1,label2: 部分配列を読込む場合に使用します

全て読込む場合は、どちらもH5S_ALLを指定します

label3: H5P_DEFAULTを指定します

buf: 出力変数を指定します

ret: [戻り値]失敗の場合は負の値

◇データセットのクローズ

ret = H5Dclose(dhnd);

dhnd: データセットハンドル値を指定します

ret: [戻り値]失敗の場合は負の値

データを読み込む

Fortran

◇データセットのオープン

call H5Dopen_f(fhnd,nam,dhnd,ret)

fhnd: ファイルハンドル値を指定します
nam: データセット名を指定します
dhnd: [戻り値]データセットのハンドル値
ret: [戻り値]失敗の場合は負の値

◇データ読込み

call H5Dread_f(dhnd,dtyp,buf,sz1,ret,label1,label2)

dhnd: データセットハンドル値を指定します
dtyp: データタイプ値を指定します
buf: 読込み用の変数を指定します
sz1: bufの各次元サイズを設定しておきます
ret: [戻り値]失敗の場合は負の値
label1,label2: 部分配列を読込む場合に使用します
全て読込む場合は、どちらもH5S_ALL_Fを指定します

◇データセットのクローズ

call H5Dclose_f(dhnd,ret)

dhnd: データセットハンドル値を指定します
ret: [戻り値]失敗の場合は負の値

データサイズを取得する

Fortran

◇データスペースのオープン
call H5Dget_space_f(dhnd,shnd,ret)
dhnd: データセットハンドル値を指定します
shnd: [戻り値]データスペースのハンドル値
ret: [戻り値]失敗の場合は負の値

◇データセットのサイズ取得
call H5Sget_simple_extent_dims_f(shnd,sz1,sz2,ret)
shnd: データスペースハンドル値を指定します
sz1: [戻り値]データセットの各次元サイズ
sz2: [戻り値]データセットの各次元の最大サイズ
ret: [戻り値]失敗の場合は負の値

◇データスペースのクローズ
call H5Sclose_f(shnd,ret)
shnd: データスペースハンドル値を指定します
ret: [戻り値]失敗の場合は負の値

- Cにも同様の関数は用意されている。

時刻形式の変換

- AMSR2プロダクトの時刻格納形式は、TAI93形式と呼ばれる、うるう秒を含めた1993/01/01からの通算秒です。
- このままでは日時情報として扱い難いので、このサンプルプログラムでは年/月/日/時/分/秒に変換するサブルーチンを用意しています。
- この時刻変換はAMTKを使用する場合は自動的に行われます。

C

```
amsr2time_(num,in,out)
num: スキャン数を指定します
in: TAI93形式の時刻データを指定します
out: [戻り値]AM2_COMMON_SCANTIME構造体で年/月/日/時/分/秒データが返されます
```

Fortran

```
call amsr2time(num,in,out)
num: スキャン数を指定します
in: TAI93形式の時刻データを指定します
out: [戻り値]AM2_COMMON_SCANTIME構造体で年/月/日/時/分/秒データが返されます
```

時刻形式の変換

- AM2_COMMON_SCANTIME 構造体
  - amsr2time.hファイルで定義される、時刻情報格納用の構造体

スキャンタイム			
AM2_COMMON_SCANTIME			
名前	型	サイズ	説明
tai93sec	double	1	1993年1月1日0時0分00秒からの通算秒
year	short	1	年
month	short	1	月
day	short	1	日
hour	short	1	時
minute	short	1	分
second	short	1	秒
ms	short	1	ミリ秒

低周波数チャンネルの緯経度算出

- AMSR2センサには6G/7G/10G/18G/23G/36G/89Gの7つの観測周波数がありますが、それらの観測点緯度経度は正確には異なります。
- AMSR2プロダクトに格納されているのは89G緯度経度だけです。低周波の正確な緯度経度は89G Aホーン緯度経度と相対レジストレーション係数から算出できます。
- このサンプルプログラムでは低周波緯度経度を算出するためのサブルーチンを用意しています。AMTKを使用する場合は、低周波緯度経度は自動的に算出されます。

C

amsr2latlon_(num,prm1,prm2,lat89a,lon89a,latlow,lonlow)
num: スキャン数を指定します
prm1: 相対レジストレーション係数A1を指定します(6G/7G/10G/18G/23G/36G/平均)
prm2: 相対レジストレーション係数A2を指定します(6G/7G/10G/18G/23G/36G/平均)
lat89a: 89G Aホーン緯度データを指定します
lon89a: 89G Aホーン経度データを指定します
latlow: [戻り値]指定した低周波の緯度データが返されます
lonlow: [戻り値]指定した低周波の経度データが返されます

Fortran

call amsr2latlon(num,prm1,prm2,lat89a,lon89a,latlow,lonlow)
num: スキャン数を指定します
prm1: 相対レジストレーション係数A1を指定します(6G/7G/10G/18G/23G/36G/平均)
prm2: 相対レジストレーション係数A2を指定します(6G/7G/10G/18G/23G/36G/平均)
lat89a: 89G Aホーン緯度データを指定します
lon89a: 89G Aホーン経度データを指定します
latlow: [戻り値]指定した低周波の緯度データが返されます
lonlow: [戻り値]指定した低周波の経度データが返されます

データセットのアトリビュート読込み

C

◇アトリビュートのオープン
ahnd = H5Aopen(dhnd, nam, label);
dhnd: データセットハンドル値を指定します
nam: アトリビュート名を指定します
label: H5P_DEFAULTを指定します
ahnd: [戻り値]アトリビュートのハンドル値
失敗の場合は負の値

◇アトリビュート読込み
ret = H5Aread(ahnd, atyp, buf);
ahnd: アトリビュートハンドル値を指定します
atyp: 出力変数の型を指定します(読込んだデータは、出力変数の型へ自動変換されます)
buf: 出力変数を指定します
ret: [戻り値]失敗の場合は負の値

◇アトリビュートのクローズ
ret = H5Aclose(ahnd);
ahnd:アトリビュートハンドル値を指定します
ret: [戻り値]失敗の場合は負の値

- メタデータ読み取りで使ったのと同じ関数

データセットのアトリビュート読込み

Fortran

◇アトリビュートのオープン
call H5Aopen_f(dhnd,nam,ahnd,ret)
dhnd: データセットハンドル値を指定します
nam: アトリビュート名を指定します
ahnd: [戻り値]アトリビュートのハンドル値
ret: [戻り値]失敗の場合は負の値

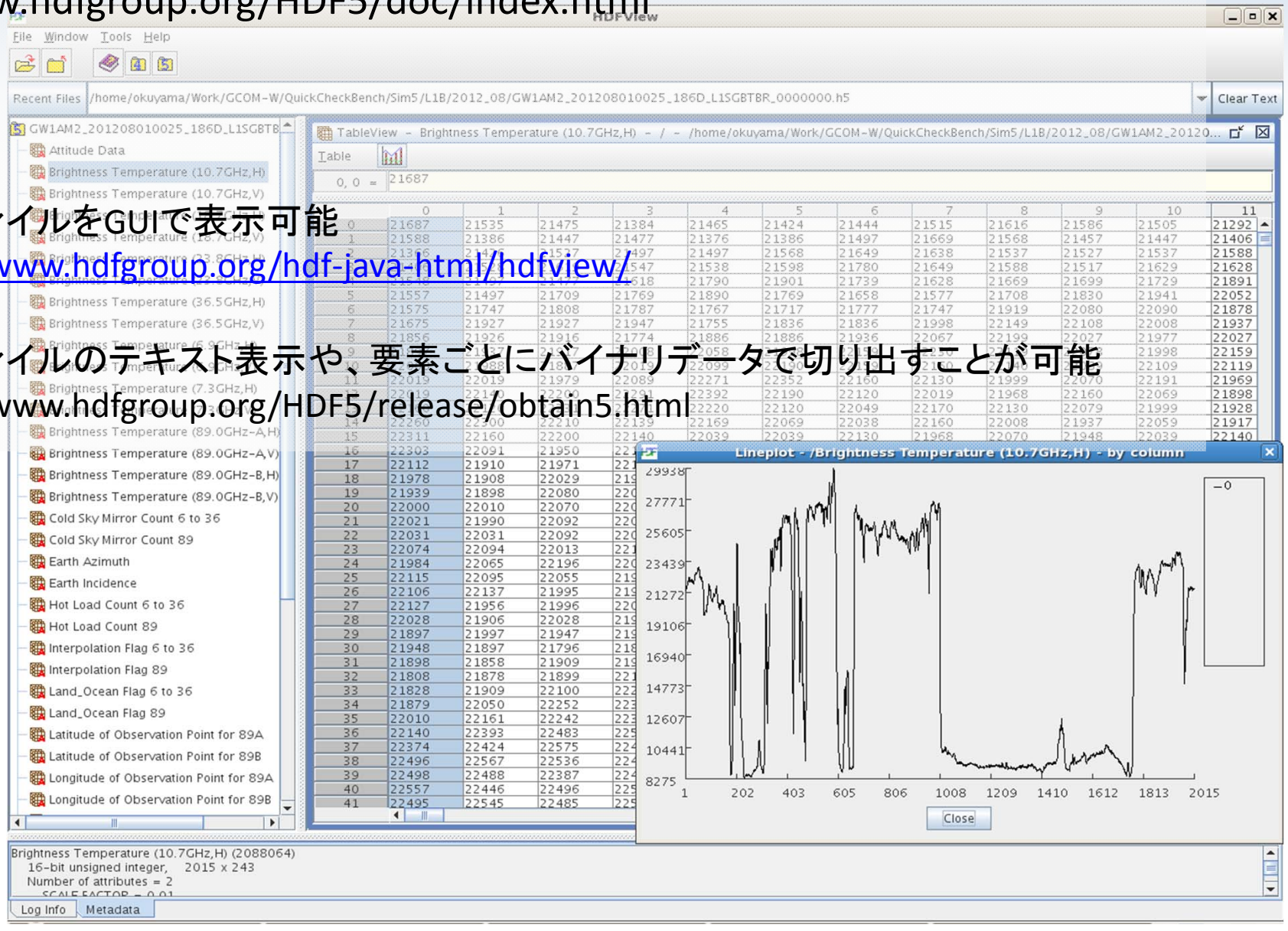
◇アトリビュート読込み
call H5Aread_f(ahnd,atyp,buf,sz1,ret)
ahnd: アトリビュートハンドル値を指定します
otyp: 出力変数の型を指定します
buf: 出力変数を指定します
sz1: bufの各次元サイズを設定しておきます(スカラ読込みでは無視されます)
ret: [戻り値]失敗の場合は負の値

◇アトリビュートのクローズ
call H5Aclose_f(ahnd,ret)
ahnd:アトリビュートハンドル値を指定します
ret: [戻り値]失敗の場合は負の値

- ここではスケール値の読み取りに使っている。

参考

- HDFのマニュアル
  - http://www.hdfgroup.org/HDF5/doc/index.html
- 便利ツール
  - hdfview
    - HDFファイルをGUIで表示可能
    - http://www.hdfgroup.org/hdf-java-html/hdfview/
  - H5dump
    - HDFファイルのテキスト表示や、要素ごとにバイナリデータで切り出すことが可能
    - http://www.hdfgroup.org/HDF5/release/obtain5.html



JAXA 宇宙航空研究開発機構

GCOM Global Change Observation Mission

まとめ

- HDFライブラリだけでもAMTKと（ほぼ）同等の処理が可能。
- AMTKと比べると、いくつかの処理は明示的に行う必要がある
  - スケールの変換（attribute情報を取得する）
  - 低周波数チャンネルの緯経度算出（モジュール有り）
  - 時刻変換（TAI93→YMDH）（モジュール有り）

- HDF5のFORTRAN90ライブラリでは可変長文字列メタデータを読込めない。