

A Direct Readout Technology Suite for Real-Time Applications

Swarvanu Dasgupta, George Mason University

Mentors: Patrick Coronado, Kelvin Brentzel, Direct Readout Laboratory, GSFC, NASA.

I. INTRODUCTION

REMOTE sensing data from the EOS (Earth Observing System) mission satellites is primarily distributed to the broad user community through platforms such as the NASA-DAAC. Although quite an efficient technique for data distribution, it fails to serve the needs of real time applications since in most cases users need to wait for at-least 24 hours before they can acquire the data they are interested in [1]. The ability to acquire and process remote sensing data in real time is imperative for some time-critical applications such as monitoring of forest fires, flash floods, hurricanes volcanoes, oil spills, sandstorms, and smoke and pollution episodes [2]. The idea of Direct Broadcast (DB) was designed to address such real time needs and builds on the fact that the satellites making the measurements would themselves make excellent distribution vehicles [3]. As the Earth is observed by these satellite instruments, data is formatted and broadcast omni-directionally to Earth in real-time. This real-time transmission of satellite data to the ground is termed as Direct Broadcast. Users who have compatible ground receiving equipment and are in direct line of sight to the satellite may receive or directly readout these transmissions. Direct Readout (DR) is thus the process of acquiring this freely transmitted live satellite data [3]. Spatial coverage from DB however is regional and thus can be useful in orienting local policies for the management and protection of natural and human resources.

The direct readout end-user community is confronted with several hurdles ranging from expensive hardware equipments to complex processing requirements. The NASA GSFC (Goddard Space Flight Center), the Direct Readout Lab (DRL) is committed to promote continuity and compatibility among evolving EOS direct broadcast satellite downlink configurations and direct readout acquisition and processing systems. The DRL bridges the EOS missions with the global direct readout community by establishing a clear path and foundation for the continued use of NASA's Earth science DB data [3]. The DRL is also involved in continued efforts to ensure smooth transitions of the Direct Broadcast infrastructure from the EOS mission to the next generation NPP (NPOESS Preparatory Project) and NPOESS (National Polar-orbiting Operational Environmental Satellite System) missions in the future. In an effort to foster global data exchange and to promote scientific collaboration, the DRL with support from other groups, is providing the user community access to Earth remote sensing data technologies and tools that enable the DB community to receive, process, and analyze direct readout data. The Direct Readout Technology Suite (DRTS) described in this document is an effort in this direction. The DRTS is an integrated suite of services, which in its entirety would provide a tool to operate on the geolocated and calibrated Level 1B products to produce subsetted/mosaiced and projected Level 2 scientific data products in contemporary HDF-EOS format, as jpeg images and in GIS ingestible GeoTIFF format. The capability to produce GeoTIFF products that can be easily imported into contemporary GIS platforms would be an important capability and would hopefully serve to bridge the gap between the GIS community and NASA's EOS/NPP/NPOESS missions. As part of this suite we also propose a standard for metadata enhancement within the GeoTIFF format. This standard is aimed at providing the necessary metadata within the GeoTIFF product instead of providing them separately. The newly designed metadata elements within the enhanced GeoTIFF format comply with FGDC's (Federal Geographic Data Committee) metadata standards for digital geospatial data [4]. We propose a technique to encapsulate the essentially hierarchical FGDC metadata structure in the inherently non-hierarchical GeoTIFF format. Metadata enhancement in GeoTIFF would obviously necessitate the development of additional functions on the GIS platforms to be able to interpret these newly incorporated elements. The proposed standard maintains backward compatibility and ensures that these developmental efforts would be minimal.

II. MOTIVATION AND OBJECTIVES

The Direct Readout Technology Suite is aimed at promoting the use of MODIS Direct Broadcast data in regional applications. The primary motivations for the Direct Readout Technology suite

- **One Stop Solution for DB end users:** Technologies for supporting various requirements of DB data processing are available at different stages of maturity. Relevant technologies include scientific algorithms to process the Level 1B products into Level 2 HDF-EOS products such as Vegetation indices (MOD13), Active Fire (MOD14) and Surface Temperature (MOD11); scripts to subset, mosaic and project the data in order to generate browse images and GeoTIFF products for user-defined regions of interest (HDFLook and HEG) and functions to perform metadata enhancements in GeoTIFF. Integrating these independent technologies and using them in tandem is likely to present difficulties for the DB end user, who may need time-consuming developmental and setup efforts. The Direct Readout Technology Suite thus integrates all these relevant technologies under one umbrella, develops upon them, and adds a GUI wrapper in order to ease the processing requirements for the DB end user and reduce the setup time. The inherent complexities of the system are hidden from the end-user reducing the entire processing task into virtually a point and click job. The DRTS in its final form could be available as a downloadable product, which end users can easily install and use to perform an entire gamut of operations for supporting real-time regional applications based on MODIS DB data.
- **Bridging the EOS and GIS Communities:** The ability to produce GeoTIFF products is an important capability. EOS project has produced tremendous amount of data in HDF-EOS format, which unfortunately cannot be imported into most contemporary GIS systems. This has resulted in an alienation of the GIS community from the EOS community. However GIS remains one of the important tools for analyzing NASA's EOS data and promoting research. GeoTIFF is one of the GIS-ingestible formats and therefore the ability to produce GeoTIFF from HDF-EOS will greatly enhance the interoperability and public use of EOS data. The availability of GIS-ready products will significantly increase the accessibility, interoperability, and inter-use of EOS data, improve the data analysis and visualization, promote the use of EOS data not only in the global change research but also in the public concerned issues such as environment and resource management, education, and community planning [5].
- **GeoTIFF Metadata enhancement to promote research.** The Metadata enhancement standard proposed as part of the DRTS is aimed at enriching the GeoTIFF format with the necessary metadata. Geospatial metadata have been recognized for playing four roles [6]: (i) Availability: metadata needed to determine the sets of data that exist for a geographic location. (ii) Fitness for use: metadata needed to understand how the data was acquired and processed (lineage) to determine if a set of data meets a specific need (iii) Access: metadata needed to acquire an identified set of data and (iv) Transfer: Metadata needed to process and use a set of data. Apart from these four obvious uses metadata can be an important component in triggering future research and development by means of a feedback mechanism. As users analyze the lineage metadata associated with a dataset, they may identify processes that may have degraded the data quality and discover alternate techniques that could potentially improve the data quality and make it suitable for a particular application. The DRL encourages the involvement of end-users in this feedback mechanism which would advance scientific research towards earth observing products better adapted to growing needs. The proposed standard for metadata enhancement within the GeoTIFF format is designed to promote this feedback mechanism and encourage end users to understand and contribute to the improvement and applicability of remote sensing data in different fields. The metadata incorporation would also bring GeoTIFF EOS products closer to the legacy HDF-EOS products. Incorporation of standard color tables within the GeoTIFF format would also be useful. Such tables would be helpful in promoting uniformity in visualizations of a particular parameter.
- **Precursor to the NISGS.** The NPP In-situ Ground System (NISGS) is envisaged to be a platform for developing, verifying and demonstrating the requisite hardware and software technologies to receive, process and distribute legacy EOS and future NPP spacecraft Direct Readout Data and derivative products. [7]. One of the NISGS subsystems, namely the NISGS Data System (NISDS) would consist of a coherent infrastructure housing several toolkit modules to enable data

- processing, scheduling, monitoring, archiving and distribution of the NPP instrument data in a stand-alone configuration. In this perspective the DRTS can serve as a precursor to the NISDS and provide useful inputs to the evolution of NISDS.
- **Immediate relevance.** The DRTS would be immediately useful for two relevant projects: (i) Eastfire and (ii) MAP '05: Project Hurricane. The Eastfire Project at George Mason University is being developed in close collaboration with NASA's Direct Readout Lab at GSFC which has the technological capability of acquiring the freely transmitted EOS-DB data with coverage across the entire eastern United States. Eastfire would be developing a web-portal that would provide near real-time products for monitoring pre-fire, during fire and post-fire conditions. The real time products would range from traditional products such as true color images, vegetation indices, and active fire to indigenous products such as fuel moisture content, fire danger indices, fire scar indices, and fuel load. Remote sensing based air quality monitoring support would also be provided by combining data from MODIS and other sensors such as MOPITT (Measurements of Pollution in the Troposphere) which can measure carbon monoxide and methane. The potential advantages of this portal would be its comprehensiveness in providing an entire gamut of images and products for monitoring eastern USA fire weather, and the availability of these products and images in real time. The second project MAP '05 involves NASA GSFC, NASA Ames, NOAA Tropical Prediction Center and Florida State University and is aimed to showcase major NASA competencies in high performance computing and in atmospheric data assimilation [8]. The DRTS could potentially be useful to the interoperability component of MAP '05 which proposes to provide visualizations of near real-time, high resolution MODIS imagery covering the US Eastern Coast and nearby Atlantic.

The DRTS was developed during the summer program at NASA with these objectives in mind. In the rest of this document we put forward the proposed metadata enhancement standard for GeoTIFF and describe the system design and architecture of the DRTS.

III. PROPOSAL FOR A METADATA ENHANCEMENT STANDARD IN GEOTIFF

Before we describe the proposed standard for metadata enhancement of the GeoTIFF format it would be imperative to introduce the salient features of the Tiff and GeoTIFF format as it pertains to our present discussion. Details of the formats are intentionally overlooked to keep the documentation lucid and to the point. Readers are encouraged to refer to the official documentations of Tiff [9] and GeoTIFF [10] for more comprehensive discussions.

The TIFF format. The Tiff file format description presented here has been taken from [9] [11] [12]. The Tagged Image File Format (TIFF) was designed to address the problems associated with fixed file formats. The most important features of this format are (i) Extendibility: The ability to add new image types and new informational/metadata fields to the format without invalidating older types or impacting the ability of older applications to read the image files; and (ii) Portability: TIFF was designed to be independent of the hardware platform and the operating system on which it executes. The richness of the TIFF format solves many problems, but at the same time creates a few of its own. The TIFF file structure is necessarily complex or more appropriately variable which implies that applications handling the TIFF format would not be straight-forward and would require some intricate coding. The TIFF format is controlled by a specification jointly written by the Aldus Corporation and Microsoft, although many other companies, in addition to these two, contributed to its formulation. Version 6.0 of this specification is dated June 1992. Although the specification document itself is copyrighted by the Aldus Corporation, the TIFF format is in the public domain, which means no fees or royalties are required to use the format.

A TIFF file may contain one or more raster images. Associated with each raster image is a data structure called the Image File Directory or IFD. The IFD is the primary data structure that holds all the informational/metadata fields or Directory Entries associated with the corresponding raster image. The IFD may hold a maximum of 65536 number of such Directory Entries (DE). Each DE or Tag has a unique tag id

which represents the image parameter contained in the DE. For example the Image width DE has a tag id of 256. Tag ids may range from 0 to 65535. However tag ids 0 to 32,767 are reserved for use by TIFF public fields. Tags 32768 to 65535 are private tags that can be used by private agencies to incorporate new metadata elements or proprietary information. It is a good idea to register private tags with Aldus and/or Microsoft to provide visibility to newly defined tags and to prevent different software vendors from using the same tags in different ways.

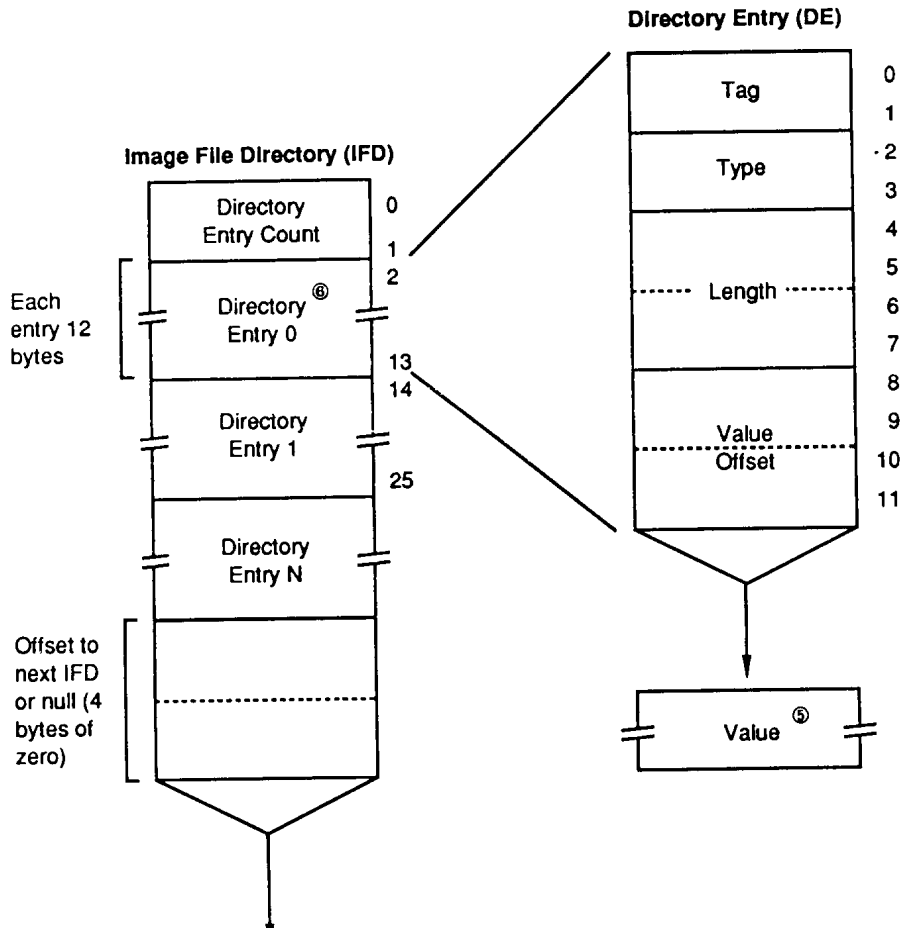


Figure 1: Image File Directory (IFD) and Directory Entry (DE) structure (Source: [10])

Since our standard is primarily based on the utilizing the private tags to incorporate metadata within GeoTIFF, it becomes necessary to discuss the structure of the Tag or the DE in some details. Each DE is exactly 12 bytes in length and is segmented into the four fields shown in Figure 1. The first field of the DE is the tag field which stores the tag id. The second field is the "type" field and it indicates the data type of the image parameter stored in the DE. 12 unique data types are currently defined by the TIFF specification. Each data type is assigned an integer code which is stored in the "type" field of a DE. The data types, their assigned code and description are given in Table 1. The image parameter that the DE describes may have one or multiple values. For example the DE describing the image width (Tag id: 256), needs only one value representing the no. of columns in the image. The DE describing the color map (Tag id: 320) however will need to keep multiple RGB values. The third field called the DE "length" or "count" field is thus designed to keep track of the number of items of the data type as specified in the "type" field that is associated with the DE. The purpose of the final field, the value offset field, is two fold. Firstly if the DE has only one value and it is less than 4 bytes in length, the value offset field directly stores this value. However if multiple values are associated with the DE, or the value is more than four bytes long, the same value offset

TABLE 1
TAG TYPES, CODES AND DESCRIPTIONS

TAG TYPE	CODE	DESCRIPTION
BYTE	1	8 bit unsigned integer
ASCII	2	8 bit ASCII codes terminated with null (hex 0) character
SHORT	3	16 bit (2-byte) unsigned integer
LONG	4	32-bit (4-byte) unsigned integer
RATIONAL	5	Two LONG values. The first represents the numerator of a fraction, the second the denominator
SBYTE	6	An 8 bit signed (2's complement) integer.
UNDEFINED	7	An 8 bit byte that may contain anything, depending on the definition of the field.
SSHORT	8	A 16 bit (2byte) signed (2's complement) integer.
SLONG	9	A 32bit (4byte) signed (2's complement) integer.
SRATIONAL	10	Two SLONG's: the first represents the numerator of a fraction, the second the denominator.
FLOAT	11	Single precision (4byte) IEEE format.
DOUBLE	12	DOUBLE Double precision (8byte) IEEE format.

parameter described in the product. The colormap used for the NDVI product thus is different from the colormap used for the ocean color product, which typically uses a log scale colormap. This implies that it would be beneficial for the user community if the standard colormap is encoded within the EOS GeoTIFF product. This would ensure that the user has access to the standard colorscale used for the product. The user however will have the flexibility to change the color scale within the GIS platform for further analysis. When producing EOS GeoTIFF products we must ensure that the encoded colormap is the standard colormap representative of the parameter. Presently we describe in brief how colormaps are represented in the generic TIFF format.

The actual raster image can belong to either of four different image types. Bilevel images can store only black and white image data utilize only one bit per pixel. Grey scale images are generalizations of bilevel images and can store shades of gray. They utilize 2 to 8 bits per pixel, which results in 4 to 256 levels of gray available in an image. The third type of images are the RGB images which can use up to 24 (8x3) bits per pixel to represent red, blue and green levels associated with the pixel. The fourth class is palette color images and is the class that would be most useful in representing EOS/NPP satellite products. We thus describe this class in a little more detail. Each palette color image is required to define some Tags in their IFD. We describe Tags that are relevant to defining colormaps in GeoTIFF palette color images.

- *SamplesPerPixel*: The number of components or samples per pixel. SamplesPerPixel is usually 1 for bi-level, grayscale, and palette-color images. SamplesPerPixel is usually 3 for RGB images. For EOS products SamplesPerPixel would be 1.
- *BitsPerSample*: Number of bits per component/sample. According to the baseline TIFF 6.0 specification BitsPerSample can range from 1 to 8. The flexibility of the TIFF data structure however poses no difficulties in using any number of bits per pixel, as long as the reading application supports it. EOS data for example would be better represented with 16 bit integers, with scaling and offset information included for conversion to actual parameter values. The HEG (HDFEOS to GIS converter tool) supports the conversion of EOS data to 16 bit TIFF images while ArcGIS, a major GIS tool can read 16 bit TIFF images.
- *PhotometricInterpretation*: The color space of the image data. For color palette images PhotometricInterpretation should be set to 3.
- *ColorMap*: ColorMap must be included in all palette-color images. This field encapsulates a Red-Green-Blue color map (often called a lookup table) for palette-color images. In a palette-color

field instead stores the file offset (a data pointer) to the actual data associated with the DE. In other words, the data associated with a DE does not have to be physically stored with the DE but can reside anywhere in the TIFF file. Application programs handling TIFF formats must consult the "type" and "length" fields to determine if the total amount of storage required by a data item is four or fewer bytes and therefore whether it is stored directly in the "value offset" field. If the data item is more than four bytes, data pointers in the value offset field must be followed to find the actual data.

Describing the color space. Over the years researchers have associated each EOS product with a colormap that best represents the

image, a pixel value is used to index into this RGB lookup table. For example, a palette-color pixel having a value of 0 would be displayed according to the 0th Red, Green, Blue triplet. In a TIFF ColorMap, all the Red values come first, followed by the Green values, then the Blue values. The number of values for each color is $2^{\text{BitsPerSample}}$. Therefore, the ColorMap field for a 16-bit palette-color image would have a total of 3×65536 values. The width of each color value is 16 bits, 0 representing the minimum intensity, and 65535 representing the maximum intensity. Black is represented by (0, 0, 0), and white by (65535, 65535, 65535). The EOS GeoTIFF product should have an appropriate ColorMap tag defined for standard representation.

GeoTIFF Format. The GeoTIFF standard was formulated on top of the existing generic TIFF as a platform interoperable standard for the support of geographic TIFF imagery originating from satellite imaging systems, scanned aerial photography, scanned maps, digital elevation models, or as a result of geographic analyses [9]. Its aim is to allow means for tying a raster image to a known model space or map projection, and for describing those projections. GeoTIFF uses a set of 6 reserved private TIFF tags to encode a large number of geo-referencing information, catering to geographic as well as projected coordinate systems needs [9]. These keys are designed in a manner parallel to standard TIFF tags, and closely follow the TIFF discipline in their structure and layout. New keys may be defined as needs arise, within the current framework, and without requiring the allocation of new tags from Aldus/Adobe [9]. We will adopt the scheme used in the GeoTIFF standard with subtle modifications to propose the metadata enhancement standard in GeoTIFF.

Metadata enhancement. Metadata enhancement techniques in GeoTIFF would be helpful in bringing EOS/NPP GeoTIFF products closer to the legacy HDF-EOS products. Our primary intention presently is to investigate the feasibility and potential advantages of metadata incorporation within GeoTIFF. In our attempt to do so, we will describe a possible technique. We will admit upfront that this is not probably the best technique, although it may be quite close. Exhaustive investigations of what metadata to include within GeoTIFF to best represent the vast array of EOS parameters would be imperative in future. However they will be guided potentially by two resources. The most logical path to selecting a set of metadata would be to fall back on the metadata that supports the current HDF-EOS products (for e.g. the .met files). This set of metadata was described by *Proposed EOSDIS Core System (ECS) Core Metadata Standard* (Hughes Applied Information Systems, 1994). The current ECS Data model is described by Raytheon Information Technology Systems (2000). A second useful resource would be “The Content Standard for Geospatial Metadata: Extensions for Remote Sensing” [4] put forward by the Federal Geospatial Data Committee (FGDC). The ECS standard was used as a guide in the development of these metadata standards for *Remote Sensing data*. Molding the EOS metadata in the FGDC conformant structure is a good way to approach this issue. The purpose of these *Extensions for Remote Sensing* is to provide a common terminology and set of definitions for documenting geospatial data obtained by remote sensing, within the framework of the *FGDC (1998) Content Standard for Digital Geospatial Metadata* (hereafter *FGDC Metadata Content Standard* or simply base standard) [6]. The standard proposes the geospatial metadata structure to be organized in a hierarchy of data elements and compound elements. While data elements are primitive metadata items which cannot be disintegrated further, compound elements are higher level metadata concepts which can be broken down into other compound elements and data elements. All compound elements finally can be described by data elements either directly, or through intermediate compound elements. The highest level compound element is “metadata” which is composed of 9 higher level compound elements: Identification_Information, Data_Quality_Information, Spatial_Data_Organization_Information, Entity_and_Attribute_Information, Distribution_Information and Metadata_reference_Information, Platform_and_Mission_information, Instrument_information.

The potential technique to select a representative set of metadata to include within GeoTIFF would be to consult the FGDC metadata structure to identify possible metadata in the EOS/NPP context. For purposes of exemplifying our approach we illustrate the Data_Quality_Information compound element. The hierarchical structure of the Data_Quality_Information compound element is shown below. The structure is essentially same as the FGDC proposed standard except that a few compound elements i.e. Process_Contact, Time_period_information, Citation_Information, Processing_run_history and Processing_change_history have been simplified to primitive data elements. The primitive data elements

are shown in bold. Detailed descriptions of all these elements can be found in [4] [6]. The structure is exactly same as the FGDC specified document, but for a few simplifications.

```

<1> Data_Quality_Information =
  <10> 0{Attribute_Accuracy}1 +
  <11> Logical_Consistency_Report +
  <12> Completeness_Report +
  <13> 0{Positional_Accuracy}1 +
  <14> Lineage +
  <15> (Image_Quality) +
  <16> (Acquisition_Information) +
  <17> (Cloud_Cover)
    <10> Attribute_Accuracy =
      <100> Attribute_Accuracy_Report +
      <101> (1{Quantitative_Attribute_Accuracy_Assessment}n)
        <101> Quantitative_Attribute_Accuracy_Assessment =
          <1010> Attribute_Accuracy_Value +
          <1011> Attribute_Accuracy_Explanation

    <13> Positional_Accuracy =
      <130> 0{Horizontal_Positional_Accuracy}1 +
      <131> 0{Vertical_Positional_Accuracy}1
        <130> Horizontal_Positional_Accuracy =
          <1300> Horizontal_Positional_Accuracy_Report +
          <1301> (1{Quantitative_Horizontal_Positional_Accuracy_Assessment}n)
            <1301> Quantitative_Horizontal_Positional_Accuracy_Assessment =
              <13010> Horizontal_Positional_Accuracy_Value +
              <13011> Horizontal_Positional_Accuracy_Explanation

        <131> Vertical_Positional_Accuracy =
          <1310> Vertical_Positional_Accuracy_Report +
          <1311> (1{Quantitative_Vertical_Positional_Accuracy_Assessment}n)
            <1311> Quantitative_Vertical_Positional_Accuracy_Assessment =
              <13110> Vertical_Positional_Accuracy_Value +
              <13111> Vertical_Positional_Accuracy_Explanation

    <14> Lineage =
      <140> 0{Source_Information}n +
      <141> 1{Process_Step}n
        <140> Source_Information =
          <1400> Source_Citation +
          <1401> {Source_Scale_Denominator}1 +
          <1402> Type_of_Source_Media +
          <1403> Source_Time_Period_of_Content +
          <1404> Source_Citation_Abbreviation +
          <1405> Source_Contribution
            <1400> Source_Citation =
              <14000> Citation_Information
            <1403> Source_Time_Period_of_Content =
              <14030> Time_Period_Information+
              <14031> Source_Currentness_Reference

        <141> Process_Step =
          <1410> Process_Description +
          <1411> 0{Source_Used_Citation_Abbreviation}n +
          <1412> Process_Date +
          <1413> (Process_Time) +
          <1414> 0{Source_Produced_Citation_Abbreviation}n +
          <1415> (Process_Contact) +
          <1416> (Algorithm_Information) +
          <1417> (Processing_Information)
            <1416> Algorithm_Information =
              <14160> Algorithm_Identifiers +
              <14161> Algorithm_Description +
              <14162> 1{Algorithm_Change_History}n+
              <14163> (1{Algorithm_Peer_Review_Information}n)
                <14160> Algorithm_Identifiers =
                  <141600> Citation_Information
                <14161> Algorithm_Description =
                  <141610> Algorithm_Text_Description

```

```

    <141611> Algorithm_Reference
        <141611>Algorithm_Reference =
            <1416110> Citation_Information
    <14162> Algorithm_Change_History =
        <141620> Algorithm_Change_Reference]
        <141620>Algorithm_Change_Reference =
            <1416200> Citation_Information
    <14163> Algorithm_Peer_Review_Information =
        <141630> Algorithm_Peer_Review_Reference
        <141630>Algorithm_Peer_Review_Reference =
            <1416300> Citation_Information
<1417> Processing_Information =
    <14170> Processing_Identifiers +
    <14171>1 {Processing_Input_Dataset}n +
    <14172> (1 {Ancillary_Dataset}n) +
    <14173> Processing_Software +
    <14174> Processing_Procedure +
    <14175> 0{Processing_Change_History}1 +
    <14176> (1 {Processing_Documentation}n)
        <14170> Processing_Identifiers =
            <141700> Citation_Information
    <14171> Processing_Input_Dataset =
        <141710> Input_Dataset_Identifier +
        <141711>[Input_Description]
        <141712> Input_Reference] +
        <141713> 0 {Input_Level}1
            <141710>Input_Dataset_Identifier =
                <1417100> Dataset_Identifier
            <141712>Input_Reference =
                <1417120> Citation_Information
            <141713> Input_Level =
                <1417130> Processing_Level
    <14172> Ancillary_Dataset =
        <141720> 0 {Ancillary_Dataset_Identifier} 1 +
        <141721> [Ancillary_Description]
        <141722> Ancillary_Reference]
            <141720> Ancillary_Dataset_Identifier =
                <1417200> Dataset_Identifier
            <141721> Ancillary_Description =
                <1417210>Ancillary_Dataset_Description +
                <1417211> 0{Command_Line_Processing_Parameter}n
            <141722> Ancillary_Reference =
                <1417220> Citation_Information
    <14173>Processing_Software =
        <141730> [Processing_Software_Description]
        <141731> Processing_Software_Reference]
        <141732> Processing_Software_Reference =
            <1417320> Citation_Information
    <14174> Processing_Procedure =
        <141740> Processing_Run_History +
        <141741> Processing_Environment +
        <141742> (Processing_Procedure_Description)
        <141741> Processing_Environment =
            <1417410> Native_Data_Set_Environment
    <14176> Processing_Documentation =
        <14160> Citation_Information
<16> Acquisition_Information =
    <160> Satellite/Local_Zenith_Angle +
    <161> Satellite/Local_Azimuth_Angle +
    <162> Solar_Zenith_Angle +
    <163> Solar_Azimuth_Angle +
    <164> 0{Relative_Azimuth}1 +
    <165> Clock_Time/Drift +
    <166> Ascending/Descending_Indicator +
    <167>(Nadir)
        <167> Nadir =
            <1670> Nadir_Latitude +
            <1671> Nadir_Longitude

```

We now embark upon a technique to effect the metadata enhancement. The metadata enhancement scheme is exactly similar to the approach taken in GeoTIFF which uses just 6 TIFF private tags to encode numerous EOS/NPP metadata elements. MetaTags or Keys are identified by a unique KeyID and are virtually identical in function to a TIFF Tag, but has one level of abstraction above TIFF Tags. The keys would correspond to primitive data elements. We will be defining 3 tags to include all the Data quality primitive metadata elements presented above. The tag numbers are just temporary numbers for the time being. In case there is a consensus for using this technique, it will be a good idea to register these tags with Adobe. The first of the 3 private Tags used for EOS-metadata enhancement is called the EOSKeyDirectoryTag which is defined as follows:

EOSKeyDirectoryTag:
Tag = 40000
Type = LONG (4-byte unsigned integer)
N = variable, >= 4

This tag which serves as a MetaTag/Key directory and lists all the "Keys" defined. The tag is an array of unsigned SHORT values, which are primarily grouped into blocks of 4. The first 4 values are special, and contain EOSKey directory header information. The header values consist of the following information, in order:

Header={KeyDirectoryVersion, KeyRevision, MinorRevision, NumberOfKeys}

where

- "KeyDirectoryVersion" indicates the current version of Key implementation, and will only change if this Tag's Key structure is changed. The current KeyDirectoryVersion number is 0. This will change as this specification matures
- "KeyRevision" indicates what revision of Key-Sets are used.
- "MinorRevision" indicates what set of Key-codes are used. The complete revision number is denoted <KeyRevision>.<MinorRevision>
- "NumberOfKeys" indicates how many Keys are defined by the rest of this Tag.

This header is immediately followed by a collection of <NumberOfKeys> KeyEntry sets, each of which is also 4-SHORTS long. Each KeyEntry is modeled on the TIFF Tag format and is of the form:

KeyEntry = { KeyID, TIFFTagLocation, Count, Value_Offset }

where

- "KeyID" gives the key-ID value of the Key (identical in function to TIFF tag ID, but completely independent of TIFF tag-space),
- "TIFFTagLocation" indicates which TIFF tag contains the value(s) of the Key: if TIFFTagLocation is 0, then the value is LONG, and is contained in the "Value_Offset" entry. Otherwise, the type (format) of the value is implied by the TIFF-Type of the tag containing the value.
- "Count" indicates the number of values in this key.
- "Value_Offset" Value_Offset indicates the index-offset into the TagArray indicated by TIFFTagLocation, if it is nonzero. If TIFFTagLocation=0, then Value_Offset contains the actual (LONG) value of the Key, and Count=1 is implied. Note that the offset is not a byte-offset, but rather an index based on the natural data type of the specified tag array.

Following the KeyEntry definitions, the KeyDirectory tag may also contain additional values. For example, if a Key requires multiple LONG values, they shall be placed at the end of this tag, and the KeyEntry will set TIFFTagLocation=EOSKeyDirectoryTag, with the Value_Offset pointing to the location of the value(s).

All key-values which are not of type LONG are to be stored in one of the following two tags, based on their format:

EOSDoubleParamsTag:

Tag = 40001

Type = DOUBLE (IEEE Double precision)

N = variable

This tag is used to store all of the DOUBLE valued metadata Keys, referenced by the EOSKeyDirectoryTag. The meaning of any value of this double array is determined from the EOSKeyDirectoryTag reference pointing to it. FLOAT values should first be converted to DOUBLE and stored here.

EOSAsciiParamsTag:

Tag = 40002

Type = ASCII

N = variable

These tags would store only the primitive data elements described in Table 2. However there needs to be some mechanism so that the reading application can construct the metadata tree of the FGDC structure (i.e. place a primitive data element at its appropriate location in the FGDC hierarchy). We propose to encode this capability within the key id scheme. The proposed key ids are shown in Fig 2. The key Ids are specially designed to reflect the hierarchical metadata structure. The underlying idea is that an application reading these keys would be able to use the key id to find the correct position of the key value in the FGDC metadata tree structure. For e.g. a key value of 1301 would imply that the key is located in the 4th, 1st and 2nd branches at succeeding tree depth levels.

IV. DRTS SYSTEM ARCHITECTURE AND DESIGN

The system architecture for the current implementation is shown in figure 2. It may undergo some revisions to refine the architecture or retrieve better products. The input includes the MODIS Terra/Aqua Calibrated and Geolocated MODIS L1B swath granules (MOD02) and the corresponding MODIS Terra/Aqua Geolocation Fields (MOD03) swath granules. Apart from this the user would need to initialize some parameter files based on his preferences on processing the data. Important specifications in the parameter files would be the region of interest (specified by latitude and longitude of the corners), filenames of the input granules, output file formats and file names, etc. A GUI would be developed to aid in the process of creating these parameter files in later versions. Once the parameter files are set, the controller assumes responsibility of scheduling and controlling the entire process. In the first step, MODIS Rapid Response versions (stand-alone versions) of MOD13 and MOD14 algorithms are applied on the Level 1B granules to get the corresponding MOD13 (Vegetation Indices, NDVI and EVI) and MOD14 (Thermal anomalies) HDF products. These products are not geolocation. A add geolocation module uses the HDF and HDF-EOS C libraries to incorporate geolocation information from MOD01 geolocation granules into the MOD13 and MOD14 products. The controller now uses HEG (HDF-EOS data format converter tool) based scripts on the geolocated MOD13 and MOD14 HDF-EOS products to subset/mosaic the input granules in order to extract the user- defined region of interest. The HEG tool was developed under the Synergy Program under the auspices of NASA's EOS and provides a tool for conversion of HDF-EOS formatted granules to GIS compatible formats, namely GeoTIFF [13]. These are baseline GeoTIFF products which then undergo a metadata enhancement process via the metadata enhancer. The metadata enhancer is a C code built on the libtiff [14] and libgeotiff [15] libraries, inputs proper standard color maps within the GeoTIFF product as well as other FGDC compliant EOS metadata as discussed before. These metadata enhanced GeoTIFF products can then be used in most GIS's. The newly added information in the form of color-tables and metadata may not be accessible in many GIS currently. However with minimal modifications within these GIS platforms, they would be access these information in future. In order to stimulate this augmentation, we could develop and distribute libraries in C and other languages for handling this extra information. Apart from these GeoTIFF products, the DRTS would be using HEG to get subsetted/mosaiced HDF-EOS

products for the user-defined region. Another tool, HDFLook [16] will be used to generate browse images of the EOS parameters within the defined region.

The current output products are in HDF-EOS, GeoTIFF and jpeg formats for the following parameters:

True color images: True color images are built using MODIS band 1 (250m), band 4 (500m) and band 3 (500m). Standard atmospheric corrections are performed for generating the true color image.

Vegetation Indices: We are currently using the MODIS Rapid Response version of the Vegetation Index algorithm which can be used to generate Vegetation indices for a single swath. Vegetation indices include the Normalized Difference Index and the Enhanced Vegetation Index. The algorithms are simple and are applied on a corrected reflectance product. The corrected reflectance product is generated by a simple atmospheric correction using MODIS visible and near-infrared bands (bands 1 to 7). Corrections for molecular (Rayleigh) scattering and gaseous absorption (water vapor, ozone) are also made using climatological values for gas contents. Thus no real time input or ancillary data is necessary. However no aerosol correction is made. Since the reflectance product is not corrected for aerosols the Rapid Response version of the products may be of poorer quality in presence of aerosols compared to the official MOD13 16 day composite vegetation indices product. Moreover the Rapid response version of NDVI/EVI currently performs no correction for directional effect and can be generated for a single swath as opposed to MOD13 16 day composite product.

Active Fires: The Active Fire Detection Product alternatively called the Thermal anomaly product is generated using the algorithm used by the MODIS Rapid Response program. The algorithm uses brightness temperatures derived from the MODIS 4 and 11 micrometer channels. The fire detection strategy is based on absolute detection of the fire, if the fire is strong enough i.e radiating at a brightness temperature of 360K. Temperatures less than that could be natural to the surface, for eg in deserts. To filter out naturally hot surfaces and detect weaker fires, the algorithm uses a contextual test. This latter test identifies pixels with values elevated above a background thermal emission obtained from the surrounding pixels. This

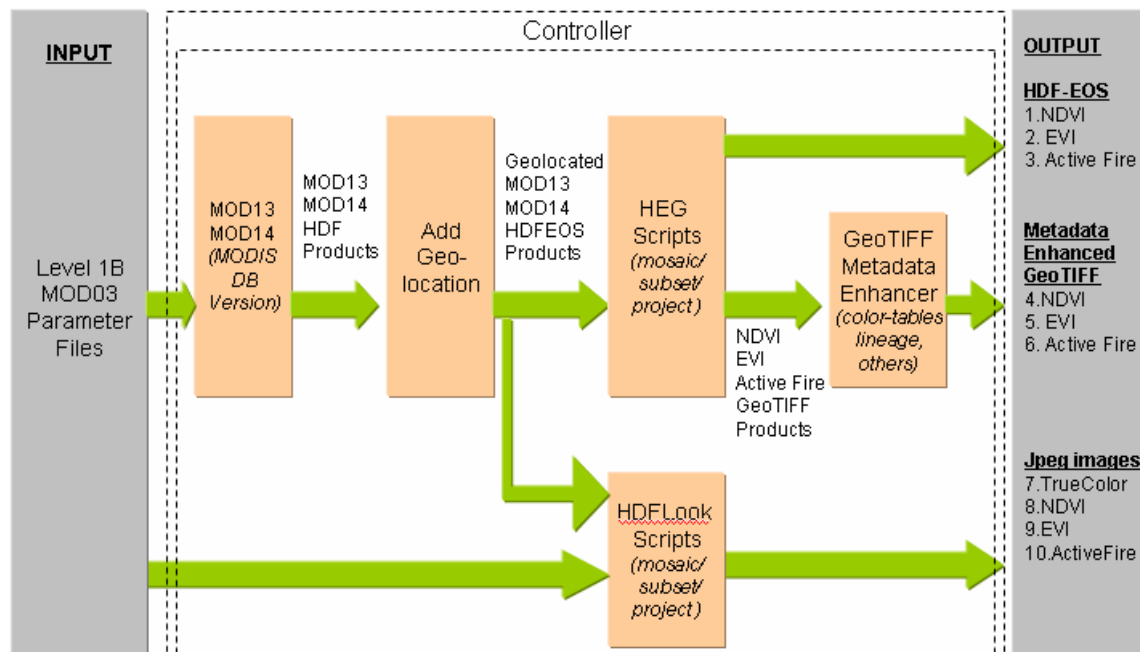


Fig 2: DRTS System Design

method accounts for variability of the surface temperature and reflection by sunlight. The contextual approach was a significant improvement in the original algorithm and results in less false detection than

traditional threshold-based algorithms. The algorithm is also sensitive to small fires, which makes it even more suited to wildfire applications.

Fig 3. illustrates the colormap and metadata enhancement in the baseline GeoTIFF product. Fig 3(a) shows the Tag structure before the augmentation using the tiffdump tool. Fig 3(b) shows the Tag structure after the colormap and metadata tag additions. Note the added tags. GIS platforms however need to add support facilities within their system to properly interpret these new tags. We haven't added any meaningful metadata, but just demonstrated the feasibility of this enhancement. Fig 4 shows the true color jpeg image generated using HDFLook, while Fig 5 shows the import of the produced GeoTIFF product in ArcGIS. It also shows the overlaying of state boundaries, and river features over MODIS GeoTIFF products.

Future improvements in the DRTS would be as follows:

- A GUI will be added to allow the user to interactively construct the input parameter files containing all user specifications.
- A database component could be added to DRTS in line with design of NISDS, to allow automatic storing of metadata elements of the generated products. Such a database would be useful in searching the product archive.
- The public domain GRASS GIS system could also be integrated within DRTS to allow analysis of the GeoTIFF products.
- We will need to develop a EOSMetadataTIFF library to stimulate the process of augmentation on contemporary GIS platforms, so that they can make full use of the color table and metadata enhanced EOS GeoTIFF products.
- Currently the system has been developed on an Unix platform with associated libraries, such as hdf, hdfs, libtiff and libgeotiff. It would be important to build a stand alone version that can be easily installed on other system.

V. SUMMARY

A Direct Readout Technology Suite was developed by integrating EOS MOD13, MOD14 modules, tools such as HDFLook and HEG, and indigenously developed modules for adding geolocation, enhancing metadata within Geotiff in order to streamline the product development process in potential regional real time monitoring agencies. The tool will support the generation of real time products using MODIS DB data in both HDF-EOS and GeoTIFF formats along with browse images in jpeg format. The capability to produce GeoTIFF products would be immensely helpful in real-time monitoring of natural and human resources using NASA's EOS platforms. As part of this suite we investigated the feasibility of incorporating standard color tables and EOS-specific metadata within GeoTIFF. While colormaps can be incorporated using already defined TIFF structures, incorporation of metadata would need the definition of new tags or metadata elements in TIFF. Incorporation of such new "tags" would be backward compatible, in a sense that current applications would still be read the rest of the product. However minimal augmentations would be required to access the newly added color tables and metadata information and make the best use of it. We propose the development of a new library to support these new "tags" and stimulate this process of augmentation in GIS platforms. These new library would not be difficult to build, given that it could be built based on the GeoTIFF libraries with possibly minimal modifications.

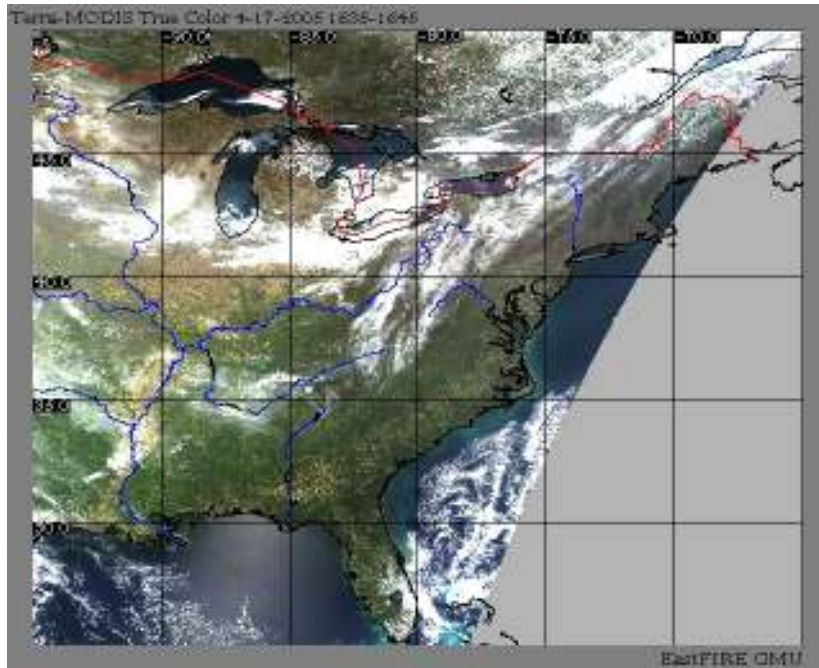


Fig 4 True Color Browse images in jpeg generated using mosaicing/subsetting capabilities in HDFLook integrated in DRTS

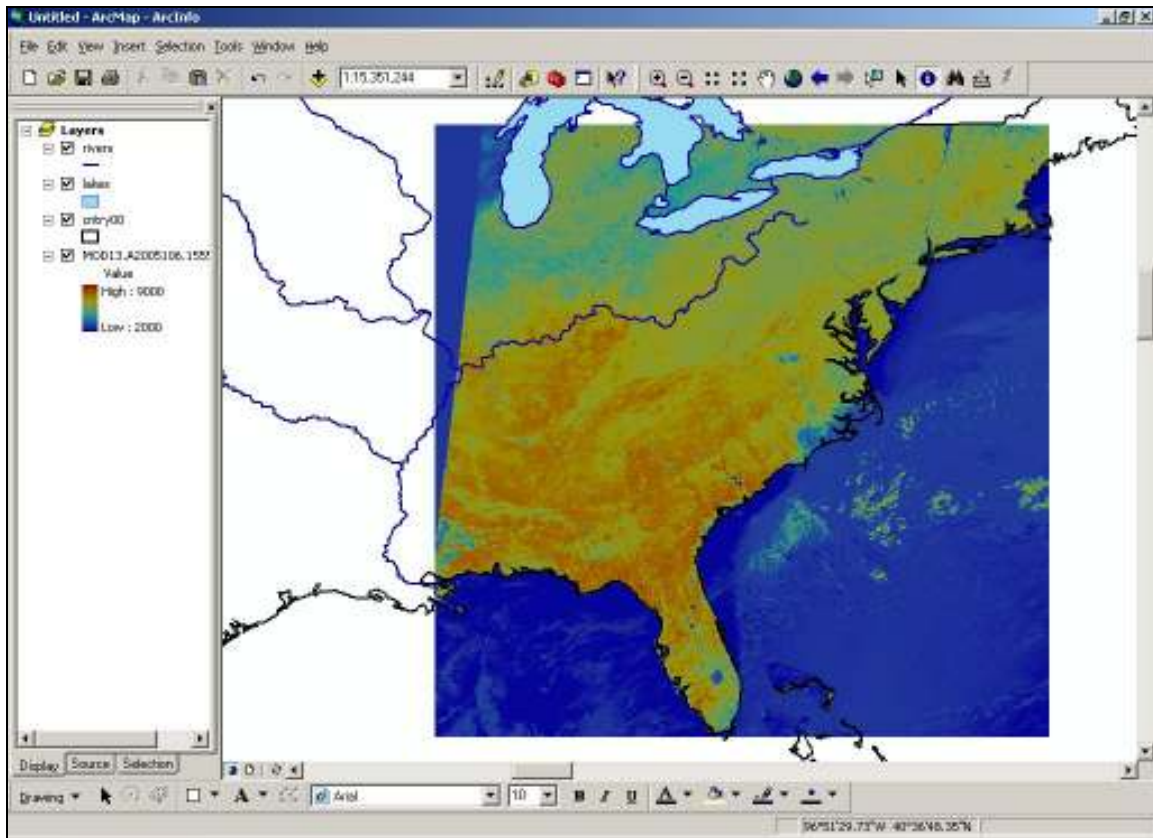


Fig 5. Using EOS GeoTIFF NDVI product in ArcGIS. The product was generated using subsetting/mosaicing and GeoTIFF capabilities in HEG and color table/metadata augmentation by the metadata enhancer.

VI. REFERENCES

- [1] James C. Dodge, "The Earth Observing System Direct Broadcast System", Earth Observation Magazine (EOM), April 1999, Volume 8, Number 3.
- [2] James C. Dodge, "Real-time environmental applications of EOS satellite direct broadcast data (Abstract)", The 2002 EUMETSAT Meteorological Satellite Conference, Dublin, Ireland, 2-6 September 2000.
- [3] Patrick Coronado, "Direct Broadcast to Direct Readout - Providing a bridge between NASA Direct Broadcast Earth Science Missions and the Broader User Community", Book Chapter (in press).
- [4] "Content Standard for Digital Geospatial Metadata: Extensions for Remote Sensing" Metadata Ad Hoc Working Group, Federal Geographic Data Committee, 2002
- [5] Liping Di, Wenli Yang, Meixia Deng, D. Deng and K. McDonald, "Interoperable Access of Remote Sensing Data through NWGISS", In Proceedings of IGARSS 2002. Toronto, Canada. Jun. 2002.
- [6] "Content Standard for Digital Geospatial Metadata" Metadata Ad Hoc Working Group, Federal Geographic Data Committee, 1998.
- [7] "NISGS documentation, Draft Version 5", Internal Document, Direct Readout Lab, 2005
- [8] Jeff de La Beaujardiere, "Interoperable Access to Hurricane Observations and Measurements, A proposal to NASA's Geospatial Interoperability Office", 2005.
- [9] "Tiff 6.0 Revision, document", Aldus Corporation, Seattle, June 1992
- [10] Niles Ritter and Mike Ruth, "GeoTIFF Specification Version: 1.8.2", <http://www.remotesensing.org/geotiff/spec/geotiffhome.html>, December 2000
- [11] http://www.ee.cooper.edu/courses/course_pages/past_courses/EE458/TIFF/
- [12] <http://www.awaresystems.be/imaging/tiff/tiffflags/baseline.html>
- [13] Cid Praderas, Abe Taaheri, Date Sharon Ni, and Tony Fu, "HDF-EOS to GeoTIFF Conversion Tool (HEG) Stand-alone Users Guide, Version 2.0" December 2000.
- [14] "Libtiff, TIFF Libraries and Utilities", <http://www.remotesensing.org/libtiff/>
- [15] "Libgeotiff", <http://www.remotesensing.org/geotiff/api/index.html>
- [16] Louis Gonzalez - Christine Deroo, "HDFLook/HDFLook MODIS Handbook", http://www-loa.univ-lille1.fr/Hdflook/E_HDF.html