Utilizing Data Sets from the CReSIS Data Archives to Visualize Greenland Echograms Information in Google Earth

Dr. Linda Hayden, Principal Investigator
Elizabeth City State University
1704 Weeksville Road, CB 672
Elizabeth City, NC

Mr. Jeff Wood, Mentor
Elizabeth City State University
1704 Weeksville Road, CB 672
Elizabeth City, NC

Maya Smith
Winston-Salem State University
601 South Martin Luther King Junior Drive, NC 27110

Shaquia Johnson
Mississippi Valley State University
14000 Hwy 82 West, MS 38941

***Abstract*— Since 1993, the Center for Remote Sensing of Ice Sheets (CReSIS) has been gathering ice thickness data in Greenland. This information was in various formats such as: Postscript Document Format (PDF), Joint Photographer Expert Group (JPEG), Keyhole Markup Language (KML), and Comma Separated Values (CSV). These formats display data in individual visualizations while another format; Matrix Laboratory (MATLAB) will display multiple sources of data, but in the proprietary software application only. The goal of this project was to combine the non-MATLAB visualizations into one window utilizing the PHP Hypertext Preprocessor scripting language and Google Earth. These product files would be simple in their construction, easily adaptable to new data formats, and provide continued display of newly acquired data. The PHP Hypertext Preprocessor language was used to modify the Keyhole Markup Language files to add description tags in order to display data from other formats. The combined files were displayed in the geographical program Google Earth available as a free download to users.**

Keywords — Remote Sensing; Geographic Information Systems; Ice Surface; Google Earth; Keyhole Markup Language; Ice Thickness; Data Visualization; Visualization;

I. Introduction

A. CReSIS

In 2005, the National Science Foundation (NSF) generated the Center of Remote Sensing of Ice Sheets (CReSIS). The CReSIS mission is to develop new technologies and computer models to measure and predict the response of sea level change to the mass balance of ice sheets in Greenland and Antarctica. CReSIS has three main programs, which are: research, education, and knowledge transfer. CReSIS is led by the University of Kansas and partners include: Elizabeth City State University (ECSU), University of Washington (UW), Pennsylvania State University (PSU), Los Alamos National Laboratory, Indiana University (IU), and The Association of Computer/Information Sciences and Engineering Departments at Minority Institutions (ADMI).

B. Types of Data

There are four types of data used in this project: accumulation radar, KU-band radar altimeter, radar depth sounder, and snow radar. These radars use varying frequencies, strengths, and sensitivities to produce specific products. Since 1993 CReSIS used radar depth sound to measure the depth of the ice sheets. In 2002 CReSIS began to use accumulation radars, which are, used to measure the annual measurement of the snow. In 2009 CReSIS began using KU-band radar altimeter and snow radar; Snow radar measures the internal layers of the ice sheets and the KU-band radar altimeter measures the bedrock using the KU-band which has a stronger signal. Examples of the different types of data being used in this project are located in figures 1 through 4.

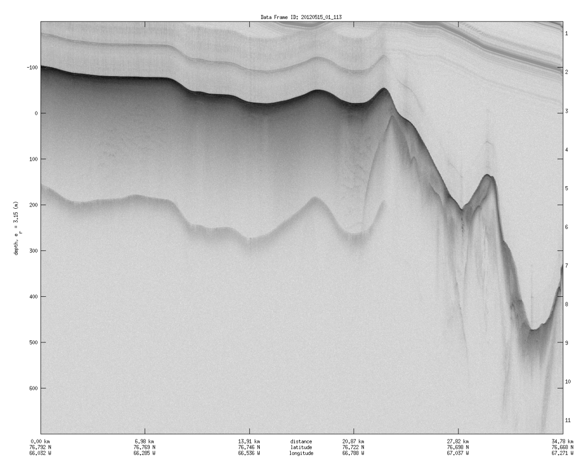

Figure 1- Accumulation Radar

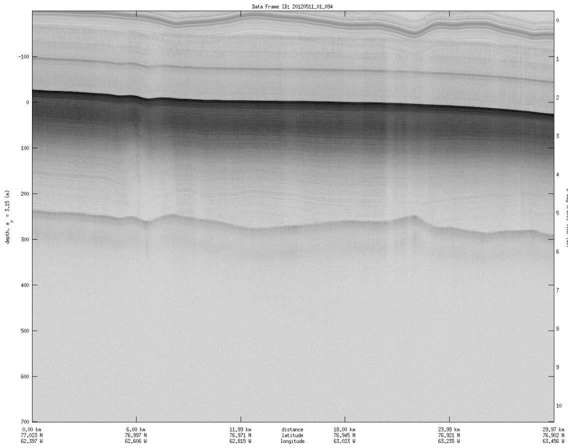

Figure 2 – KU Band Radar

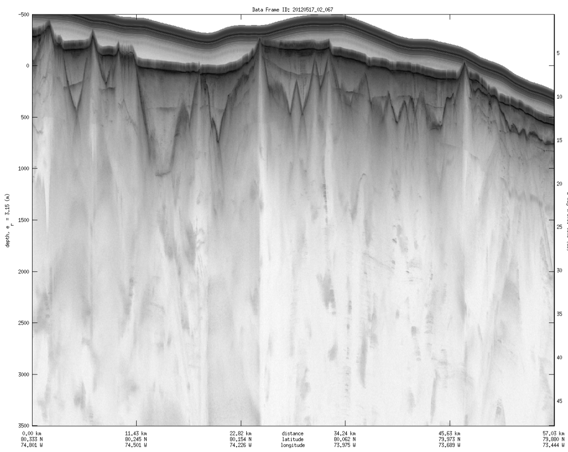

Figure 3 – Radar Depth Sounder

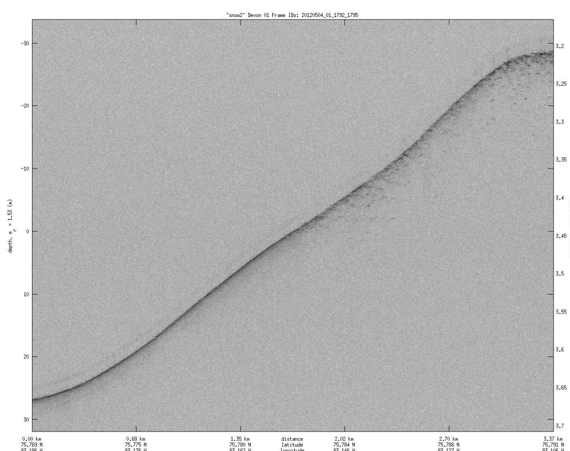

Figure 4 – Snow Radar

C. Data Visualization

On the CReSIS website, the data file types that are currently available are: Keyhole Markup Language (KML), Joint Photographic Experts Group (JPEG), Comma Separated Values (CSV), and Matrix Laboratory (MatLab). The KML file data consists of the flight path in Google Earth format; JPEG files show radar data and visual representation of the flight paths; and the CSV files consists of the exact information.

The files archived at CReSIS (KML, CSV, JPEG) are meant to display parts of the data such as position, depth, or radar images, but not at the same time. MatLab files will display comprehensive data, but rely on understanding the software and having the resource installed.

Through Indiana University, CReSIS has created a visualization tool that works with much of the past data CReSIS has accumulated. This program was written by Jun Wang and provides access to the data through the IU GIS servers. The program can be used online at http://gf2.ucs.indiana.edu/Greenland.html or downloaded to the clients hard-drive. The program is written in Open Layers and is very complex. It is not meant for adding new data or using satellite images to view locations of the data.

This project sought to simplify the visualization of the KU-CReSIS databases to make them more accessible to the public and easier to modify/update. The first goal was to utilize data straight from the CReSIS archive without creating a second archive on the ECSU servers. The second goal was to combine data from a CReSIS KML file with the radar echogram images into a single KML file to be displayed in Google Earth.

II. Scripting Languages

A. Overview

The use of several configurations and languages were attempted during the opening development stages of this project. Beginning with the straight use of PHP only and then the attempt to use JavaScript and cookies passed to PHP, research quickly pointed to the use of PHP: Hypertext Preprocessor Extensible Markup Language Document Object Model (PHP XML DOM) as the configuration to use with modifying the KML files. Strictly PHP failed because the language did not recognize XML tags. JavaScript also failed because it would not write back to the server for execution. This configuration enabled the team to develop quickly and in a minimum number of lines, the code needed to meet the goals set.

B. *PHP Preprocessor Hypertext*

PHP: Hypertext Preprocessor (PHP) is a general-purpose scripting language that is a free software program released under the PHP License. This scripting language is used to create dynamic web pages and is similar in syntax to the PERL or C languages.

The most used function of PHP is its server-side scripting capabilities. PHP is connected to the clients with a PHP parser (server module), a web server, and a browser. These components allow the user to request a PHP page from the server where the PHP parser processes the code and returns Hyper Text Markup Language (HTML) to the client. This allows the developer to hide calculations, data calls, and other code implementations from the user. PHP is platform independent and allows developers a wide range of servers with which to operate from. It not only outputs HTML, but also has the ability to create and output PDF files, images, and

Flash movies. PHP also supports a wide variety of databases and is useful for completing text processing including the ability to parse and access XML documents.

Adobe Dreamweaver, an application utilized in developing web sites, was used to construct the PHP files and incorporate them with supporting files. Dreamweaver provided code hints and warnings during the development period before uploading to a server.

C. Extensible Markup Language Document Object Model

Extensible Markup Language Document Object Model (XML DOM) defines the objects and properties of XML elements, and the methods to access them. XML DOM is a standard for how to get, change, add, or delete XML elements, which are called nodes. The PHP core provides access to XML DOM parser functions removing the need for any additional software installations.

The XML DOM views an XML document as a tree-structure. There are several different types of nodes, such as: element nodes, attribute nodes, comment nodes, and node-trees. Their contents can be modified or deleted, and new elements can be created. The node tree shows the set of nodes, and the connections between them. The nodes in the node tree have a hierarchical relationship to each other. The terms parent, child, and sibling are used to describe the relationships. The top node is called the root and any node (Parent) can have an infinite number of children. A leaf is node with no children whereas sibling nodes all have the same parent. An example of these relationships is shown in figure 5.

```
Root Node                          <bookstore>
Parent Node and Attribute Node        <book category="Sports">
Child Node and Attribute Node            <title lang="en">Tennis</title>
Child Node                               <author>James</author>
Child Node                               <year>2012</year>
Child Node                               <price>180.00</price>
End of Parent Node                    </book>
End of Root Node                   </bookstore>
```

Figure 5

III. Google Earth

A. Overview

Google Earth is a free virtual globe, map and geographic information programing application provided by Google. It must be downloaded to the user's computer, as the full package is not web-based. Limited plug-in capability does exist for several browsers. Google Earth supports managing three-dimensional Geospatial data through Keyhole Markup Language (KML). It maps the Earth by the superimposition of images obtained from satellite imagery, aerial photography, and GIS 3D globe. Google Earth is based on 3D maps and it has the capability to show 3D buildings and structures (such as bridges), which consist of users' submissions using SketchUp, a 3D modeling program software. This project used Google Earth to visualize the CReSIS flight paths while matching the echogram pictures with the segments of a flight.

B. Extensible Markup Language

Extensible Markup Language (XML) defines a set of rules for encoding documents in a format that is both human-readable and machine-readable. XML is designed to simply carry data, not to display data. XML tags are not predefined the user must define them. XML was created to structure, store, and transport information. By definition, an XML document is a string of characters. The characters, which make up an XML document, are divided into markup and content. All strings which establish markup either begin with the left angel bracket character (<) and end with aright angel bracket character (>), or begin with the and character (&) and end with a semicolon (;). Strings of characters, which are not markup, are content. There are three types of markup tags: start-tags <section>; end-tags </section>; empty-element tags <line-break />. An example of an element is <Greeting>Hello, world.</Greeting>. An attribute is a markup construct consisting of a name/value pair that exists within a start-tag or empty-element tag. An example of an attribute <step number="3">Connect A to B. </step> where the name of the attribute is "number" and the value is "3". Schema is the structure or format and refers to the organization of data to create a blueprint of how a database will be constructed.

C. Keyhole Markup Language

Keyhole Markup Language (KML), based on XML language schema, is a file format used to display geographic data in an Earth browser, such as Google Earth. KML files can pinpoint locations, add image overlays, and expose data in new ways.

The first line structure of a KML file is a XML header. No spaces or other characters can appear before this line. The second line is a KML namespace declaration. Namespace declaration identifies and assigns a unique name to a user and provides a method to avoid element name conflicts. The namespace for KML file can be found at: http://earth.google.com/kml/2.1 and identifies element names used by Google Earth.

The next major element is a placemark object that contains the following elements: name, description, and coordinates. The name is used as the label of for the placemark. The description is what appears in the bubble attached to the Placemark inside of Google Earth. The coordinates are the specifics; it has the longitude and latitude of a line.

Please refer to the text below for describing the CReSIS KML file:

```
<?xml version="1.0" encoding="utf-8"?>
<kml xmlns="http://earth.google.com/kml/2.1">
  <Document>
    <name>20120504_02</name>
   <Placemark>
      <name>20120504_02_001</name>
       <coordinates> -
82.5152270000000,75.6800980000000, -9999.000
         -82.5129010000000,75.6747480000000,-9999.000
         -82.5105970000000,75.6693350000000, -9999.000
```

```
        -82.5080110000000,75.6638810000000, -9999.000
        </coordinates>
    </Placemark>
```

IV. Project Phases

A. Data Location

In order to confirm what data was available from CReSIS and in what format the files were in it was imperative to organize and sort the CReSIS data file names and locations. Collecting KML and echogram data file names from the CReSIS 2012 Greenland archive directories did this. These file names, along with their locations, were placed into an Excel file to determine what was available, what was required, and what was missing. This enabled the team to target which flights paths were available to use. While beginning this process the team began to find many CReSIS files of one particular flight in various unrelated locations. This required a more thorough search to find the files and document their location in the CReSIS archives. Once these file names and locations were documented in Excel, their correlation to a particular KML flight path was established. This chart can be seen in appendix 1. The first step in finding the CReSIS data is to locate the data file in the 2012 Greenland archives. Once the KML flight path has been chosen then the echogram files must be located. Included in the directory of the echogram files are map files. These file names are copied into Microsoft Word and the map file names are eliminated. Then the complete URL path is added to the echogram file names. This text is then saved as a CSV file.

B. Excel

Excel is a software package used to create and format spreadsheets. Excel uses a grid of cells arranged in numbered rows and lettered columns to organize data and perform mathematical computations. The team used Excel to collect, sort, and organize date files such as KML, CSV, and JPEG maintained by CReSIS in their online data archive. It was used to take inventory of what data files CReSIS had available and where they were located.

C. Data Manipulation

The PHP/KML modification file starts with the definition of the function "output_file." This function reads the modified file and supplies the mime type to the browser so that the KML file is opened in the Google Earth application. This function was copied from the website "How To Force File Download With PHP "[2] and modified to include the KML+XML mime type.

The first step of the actual modification process was to import the URLs to the echogram images on the CReSIS web site. These URLs were packaged into a CSV file due to their storage directory also containing other files. These URLs were then placed into an array to be called later in the file.

The next step in the modification file was to import the contents of the current KML file from the CReSIS site. This involved the use of the PHP command "file_get_contents" to place the contents into a variable.

Once the data from the CReSIS KML file is downloaded and the CReSIS KML file is closed, a local temporary file: tempKMLflight.kml, is opened. The data from the CReSIS KML file is then loaded into the temporary file and the file is closed.

Once the reading section of the code is complete, the modification section begins. First a new DOM document is created to hold the structure of the temporary XML document previously created. The temporary file is loaded into a variable and the first node is located. This node is the root node and all other elements within the file are contained within this node. The next step is to locate and count each of the "Placemark" elements, as these are the elements, which separate each of the strings of coordinates, which are linked to a particular echogram image.

Now that the data has been loaded and the "Placemarks" located, a loop is begun to perform the actions of building the value for the new "description" element for each "Placemark" element. This value includes PHP variables and HTML code for displaying a reduced version of the echogram image and a link to the actual image. Once this value is assembled, a new node titled "description" is created. This is repeated for each "Placemark" within the temporary KML file.

Once the loop is completed, the tempKMLflight.kml file is saved. A call is then made to the function "output_file" with the parameters "name", "path", and "mime-type" which downloads the tempKMLflight.kml file and attempts to open the application Google Earth. This function will not override mime a setting on a local computer set by a user and behaves in different manners with different browsers. Some browsers will simply download the file (Chrome) while others (Safari, Firefox, Explorer) will download and prompt you as to whether you would like to save or open the file.

Examples of modification files at work can be found at: http://nia.ecsu.edu/reuomps2012/teams/data/research.html The files listed are representative of those flights that had corresponding data available in the correct format (png).

V. Results

The final PHP files developed were able to import data from two separate sources and combine them into one visualization. This visualization enables users to view both the flight path and the radar echogram images in one window with a minimal amount of coding taking place. The final script, including comments, was 52 lines with only two inputs needing change if a new flight is desired (CReSIS KML File location, Echogram image URL text file). Though various other methods and scripting languages were attempted, the use of the XML DOM application-programming interface (API) allowed the most versatile and minimal amount of programming to accomplish the goal of data integration. Figure 6 displays the overall results of this project.

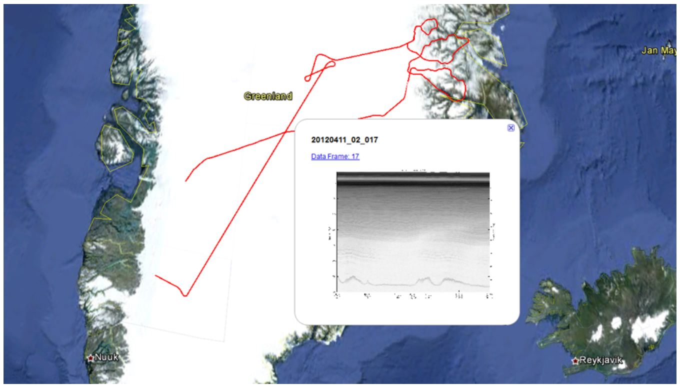

Figure 6

VI. Conclusion

PHP XML DOM scripting is a valid tool to be used in modifying and visualizing KML data files from CReSIS. More work needs to be done in order to increase the amount of data that can be viewed in this manner. Better organization, scripting reviews and rewriting, and the provision of more presentation software applications will enable developers to quickly present information to users as needed. Developers should continue to focus on Google Earth as a main presentation software, but move to Google Maps or other mapping software when able. The use of the PHP XML DOM configuration of development enabled quick development of the commands needed and produced a small file to complete the task. This visualization was produced for K-12 and small universities who do not have access to MATLAB.

VII. Future Work

In this research the information that was utilized was from the 2012 Greenland expedition using the P3 aircraft. The data type that is displayed within this research is JPEGs images of echograms. Future projects should look at adding other data types (CSV, PDF) to description tags for display will be needed.

To overcome of the obstacles with the research project, the team had to create a text file for each location of echograms because the echograms and maps are located in the same file. In the future, creating a code that will locate the echogram file on the CReSIS site will be more useful. Creating code that passes the flight selection and images into variables instead of creating individual pages for each flight will also be work for the future teams.

To view the results of research, Google Earth has to be downloaded to the user's workstation. Implementing the code in Google Maps for online access without downloading the Google Earth application or giving the user the options to open with Google Maps or Google Earth would assist users who do not have the permission to install software.

At the current time the years from 1993 to 2011 data are not being used. In the future researchers should be able to work with older KML files, which display points instead of segments and also utilize the older data. The team also had a difficult time with matching the echograms to the flight path files. Researchers should work with CReSIS to better organize their data files and provide better consistency as to how the data is presented. An organization chart of the current file structure for Greenland 2011 can be found in Appendix 2.

VIII. Acknowledgments

The 2012 Database Team would like to thank Principal Investigator, Dr. Linda B. Hayden, for making the Research Experience for Undergraduate in Ocean, Marine, and Polar Science. We would also like to thank CReSIS for making the data available. In addition we would like to thank Mr. Jeffrey Wood for being an exceptional mentor.

IX. References

[1] (2012, June 29) PHP: strpos - Manual [Online]. Available: http://www.w3schools.com/php/func_string_strpos.asp

[2] (2007, August 12) How To Force File Download With PHP [Online]. Available:

[3] http://w-shadow.com/blog/2007/08/12/how-to-force-file-download-with-php/

[4] (2012, February 24) KML Tutorial [Online]. Available: https://developers.google.com/kml/documentation/kml_tut

[5] (2012, June 29) PHP Document Object Model [Online]. Available: http://php.net/manual/en/book.dom.php

[6] W3C Tutorials and Reference [Online]. Available: http://www.w3schools.com/dom/default.asp

[7] (2012, June 29) PHP: fgetcsv - Manual [Online]. Available: http://us3.php.net/manual/en/function.fgetcsv.php

[8] PHP Tutorial - File Write [Online]. Available: http://www.tizag.com/phpT/filewrite.php

[9] L. Welling and L. Thompson, PHP and MySQL Web Development, (4th ed.). Addison Wesley, 2009.

[10] PHP XML DOM [Online]. Available: http://www.coursesweb.net/php-mysql/php-xml-dom

[11] CReSIS Data [Online]. Available: https://www.cresis.ku.edu/data

[12] PolarGrid GIS Server [Online]. Available: http://gf2.ucs.indiana.edu/

[13] (2000, November 13) P. Le Hégaret, L. Wood, and J. Robie. "What is the Document Object Model?" [Online]. http://www.w3.org/TR/DOM-Level-2-Core/introduction.html

Appendix 1

2012_Greenland_P3

KML https://data.cresis.ku.edu/data/accum/2012_Greenland_P3/kml/	Accumulation https://data.cresis.ku.edu/data/accum/2012_Greenland_P3/images/	KU-Band https://data.cresis.ku.edu/data/kuband/2012_Greenland_P3/images/	Radar Depth Sounder https://data.cresis.ku.edu/data/rds/2012_Greenland_P3/images/
		20120321_02/	
Browse_Data_20120326_01.kml	20120326_01.pdf		
Browse_Data_20120326_02.kml	20120326_02.pdf		
Browse_Data_20120327_01.kml	20120327_02.pdf		
Browse_Data_20120327_02.kml	20120327_02_deeper.pdf		
Browse_Data_20120327_03.kml			
Browse_Data_20120329_01.kml		20120329_01/	
Browse_Data_20120402_01.kml			
Browse_Data_20120404_01.kml	20120404_01.pdf		
Browse_Data_20120410_01.kml	20120410_01.pdf		
Browse_Data_20120410_02.kml	20120410_02.pdf		
Browse_Data_20120410_03.kml	20120410_03.pdf		
Browse_Data_20120410_04.kml	20120410_04.pdf		
Browse_Data_20120410_05.kml	20120410_05.pdf		
Browse_Data_20120411_01.kml	20120411_01.pdf		
Browse_Data_20120412_01.kml	20120412_01.pdf		
Browse_Data_20120413_01.kml	20120413_01.pdf		
Browse_Data_20120414_01.kml	20120414_01.pdf		
Browse_Data_20120416_01.kml	20120416_01.pdf		
Browse_Data_20120416_02.kml	20120416_02.pdf		
Browse_Data_20120417_01.kml	20120417_01.pdf		
Browse_Data_20120418_01.kml	20120418_01.pdf		
Browse_Data_20120419_01.kml	20120419_01.pdf		
Browse_Data_20120420_01.kml	20120420_01.pdf		20120420_01/
Browse_Data_20120421_01.kml	20120421_01.pdf		20120421_01/
Browse_Data_20120423_01.kml	20120423_01.pdf		20120423_01/
Browse_Data_20120425_01.kml	20120425_01.pdf		20120425_01/
Browse_Data_20120428_01.kml	20120428_01.pdf		20120428_01/
Browse_Data_20120429_01.kml	20120429_01.pdf		20120429_01/
Browse_Data_20120502_01.kml			
Browse_Data_20120504_00.kml	20120504_00.pdf		20120504_00/
Browse_Data_20120504_01.kml	20120504_01.pdf		20120504_01/
Browse_Data_20120504_02.kml	20120504_02/(PNG) 20120504_02.pdf (PDF)		20120504_02/
		20120509_01/	20120509_01/
	20120511_01/	20120511_01/	20120511_01/
	20120514_01/		20120514_01/
	20120514_02/		20120514_02/
	20120515_01/		20120515_01/
			20120515_02/
			20120515_04/
			20120516_01/
			20120517_02/

Appendix 2

2011_Greenland_P3

KML	Accumulation	KU-Band	Radar Depth Sounder
https://data.cresis.ku.edu/data/accum/2011_Greenland_P3/kml/ (points not segments)	https://data.cresis.ku.edu/data/accum/2011_Greenland_P3/images/	https://data.cresis.ku.edu/data/rds/2011_Greenland_P3/images/	https://data.cresis.ku.edu/data/kuband/2011_Greenland_P3/images/
Browse_Data_20110516_01.kml	20110516_01/	20110516_01/	20110516_01/
Browse_Data_20110513_01.kml	20110513_01/	20110513_01/	20110513_01/
Browse_Data_20110512_01.kml	20110512_01/	20110512_01/	20110512_01/
Browse_Data_20110511_01.kml	20110511_01/	20110511_01/	20110511_01/
Browse_Data_20110510_03.kml	20110510_03/	20110510_03/	20110510_03/
Browse_Data_20110510_02.kml	20110510_02/	20110510_02/	20110510_02/
Browse_Data_20110510_01.kml	20110510_01/	20110510_01/	20110510_01/
Browse_Data_20110509_01.kml	20110509_01/	20110509_01/	20110509_01/
Browse_Data_20110507_01.kml	20110507_01/	20110507_01/	20110507_01/
Browse_Data_20110506_01.kml	20110506_01/	20110506_01/	20110506_01/
Browse_Data_20110505_01.kml	20110505_01/	20110505_01/	20110505_01/
Browse_Data_20110502_01.kml	20110502_01/	20110502_01/	20110502_01/
Browse_Data_20110429_01.kml	20110429_01/	20110429_01/	20110429_01/
Browse_Data_20110428_01.kml	20110428_01/	20110428_01/	20110428_01/
Browse_Data_20110426_01.kml	20110426_01/	20110426_01/	20110426_01/
Browse_Data_20110425_01.kml	20110425_01/	20110425_01/	20110425_01/
Browse_Data_20110423_01.kml	20110423_01/	20110423_01/	20110423_01/
Browse_Data_20110422_02.kml	20110422_02/	20110422_02/	20110422_02/
Browse_Data_20110422_01.kml	20110422_01/	20110422_01/	20110422_01/
Browse_Data_20110419_01.kml	20110419_01/	20110419_01/	20110419_01/
Browse_Data_20110418_01.kml	20110418_01/	20110418_01/	20110418_01/
Browse_Data_20110416_01.kml	20110416_01/	20110416_01/	20110416_01/
Browse_Data_20110414_01.kml	20110414_01/	20110414_01/	20110414_01/
Browse_Data_20110413_01.kml	20110413_01/	20110413_01/	20110413_01/
Browse_Data_20110412_01.kml	20110412_01/	20110412_01/	20110412_01/
Browse_Data_20110411_02.kml	20110411_02/	20110411_02/	20110411_02/
Browse_Data_20110411_01.kml	20110411_01/	20110411_01/	20110411_01/
Browse_Data_20110408_01.kml	20110408_01/	20110408_01/	20110408_01/
Browse_Data_20110407_01.kml	20110407_01/		
Browse_Data_20110406_01.kml	20110406_01/	20110406_01/	20110406_01/
Browse_Data_20110331_01.kml	20110331_01/	20110331_01/	20110331_01/
Browse_Data_20110329_05.kml	20110329_05/		
Browse_Data_20110329_04.kml	20110329_04/		
Browse_Data_20110329_03.kml	20110329_03/		
Browse_Data_20110329_02.kml	20110329_02/	20110329_02/	20110329_02/
Browse_Data_20110329_01.kml	20110329_01/	20110329_01/	20110329_01/
		20110328_01/	20110328_01/
		20110326_01/	20110326_01/
		20110325_01/	20110325_01/
		20110323_01/	20110323_01/
		20110322_01/	20110322_01/