# 5.6 THE DATA DISCOVERY SUPPORT REPOSITORY – THE NEXT CHAPTER IN THE SCIENCE OF DATA DISCOVERY

Roland H. Schweitzer*, Travis Stevens[1] and Ted Habermann[1]

NOAA-CIRES Climate Diagnostics Center
NOAA/OAR/CDC
University of Colorado
Boulder, Colorado

National Geophysical Data Center[1]
NOAA/NESDIS
325 Broadway
Boulder, Colorado

## 1. INTRODUCTION

Helping users find data they need is a critical part of NOAA's mission. One of the greatest barriers to providing effective search systems is the lack of accurate metadata. Data providers are obligated to maintain HTML data descriptions that can be indexed by search engines, since many users discover the data they need via these sites. Data providers are also required to maintain FGDC compliant descriptions of their data. (N.B. It is common practice to refer to metadata in this form of metadata as *FGDC metadata*. In keeping with this practice, we will use the term *FGDC metadata* to mean the Content Standard for Digital Geospatial Metadata (CSDGM Version 2 - FGDC-STD-001-1998) as defined by the Federal Geographic Data Committee.) Domain specific search engines such as the GCMD, the FGDC Clearinghouse and NOAAServer use these FGDC metadata. Other practical considerations may force data providers into maintaining still other forms of metadata, including data contained in self-describing netCDF files or relational database tables. The Data Discovery Support System is one approach to unifying metadata into a single system that automatically synchronizes and stays up-to-date.

The Data Discovery Support System is based on a central repository implemented using a back-end tool from Blue Angel Technology. Data are added to the repository by reading in existing FGDC metadata or through a Web-based interface filled in by the provider. Once metadata is added to the repository it can be kept current by configuring a package of Java classes, which can reach across the network and extract bits of metadata from the netCDF files.

## 2. ADDING DATA TO THE DATA DISCOVERY SUPPORT REPOSITORY

The central piece in the system is the Data Discovery Support Repository. The repository supports network connections for importing metadata from and exporting metadata to other parts of the system. The back-end tool is from Blue Angel Technology (www.blueangeltech.com) and is being used for management of metadata at NCDC, NGDC, and NODC.

For those sites that already have a completed set of FGDC metadata records describing their data, populating the repository is simply a mater of reading in the XML version of these files. The “mp” metadata processor from the USGS is a useful tool for reformatting existing metadata records into XML for ingestion into the repository. The “mp” tool will also flag errors in the existing records. Errors can be repaired before continuing, or the incomplete records can be ingested into the system and corrections can be made using the Web interface described below.

### *2.1 Metadata Common to all Data Sets*

The user interface is implemented as Java-servlet that allows users to quickly and easily fill in metadata information. Often metadata from one site contains many elements that are repeated for every data set in the provider’s collection. Data and metadata contact information are an example of information that is repeated and can inserted into the repository via the Web interface one time. This repeated information can then be automatically included in new records. See Figure 1 to see the user interface for editing repeated information.

### *3.2 Metadata Specific to a Data Set*

Even using the Web-interface to fill in repeated elements, there is still a considerable amount of information, which must be added to the metadata record to give a complete description of the data set. The interface is designed to make this process as easy as possible. A “tab” metaphor is used to display links to each section of the FGDC record. Navigating to a particular section can be accomplished with one click. See Figure 2 to see the user interface for editing an FGDC record.

## 3. KEEPING THE REPOSITORY UP-TO-DATE

Unfortunately, after an organization invests the

*Corresponding author address: Roland H. Schweitzer, NOAA/ERL/CDC - (R/E/CD1), 325 Broadway, Boulder, CO 80303. E-mail: rhs@cdc.noaa.gov

resources needed produce quality metadata descriptions often the task of keeping those records current is neglected. The Data Discovery Support Repository automates the process of keeping metadata records up-to-date. This automation relieves the provider from the burden of submitting new metadata for simple changes like time steps being added to the file.

### *3.1 Keeping The Repository Up-to-date*

In order to keep the metadata up-to-date by reading information from the original data files, a connection must be established between the two. The connection between the repository and netCDF files was accomplished via a set of Java classes. These classes read a simple XML configuration file. The configuration file associates particular netCDF file attributes or netCDF data values with FGDC metadata elements. The input also specifies a Java class, which implements the logic needed to transform the content extracted form the netCDF file into content suitable for use in an FGDC description. When that transformation is accomplished the metadata content can be written out as an XML file or it can be ingested directly into the repository using a Java class library provided by Blue Angel Technologies.

### *3.2 The Java Classes (A Detailed Example)*

One of the most basic pieces of metadata that can easily go out-of-date is the time range covered by the data set. Often data collections have new data automatically added as it becomes available. Giving users that are depending on the FGDC descriptions complete and accurate metadata about the time extents of the data is difficult without the automation features that have been added to the Data Discovery Support System.

To look at how this is accomplished in the system, consider the example of the National Centers for Environmental Prediction, Optimal-Interpolation Sea Surface Temperature data set maintained in netCDF form at the Climate Diagnostics Center. New data are appended to the netCDF file weekly. To keep the repository current, the maximum time can be extracted directly from the netCDF file. We use a simple XML configuration file to tell the software which pieces of the netCDF file contain the information we need and how to transform the information into metadata the meets the FGDC standard.

```
<file name="sst.wkmean.1990-present.nc">
  <outputElement
    name="timeperd/timeinfo/rngdates/enddate">
   <inputVariable name="time">
    <inputAttribute name="units"/>
    <inputAttribute name="actual_range"
                    index="1"/>
   </inputVariable>
   <transform
     class="netcdftofgdc.timeTransform"/>
  </outputElement>
</file>
```

**Figure 3. Excerpt from the XML configuration file.**

The <outputElement> configuration parameter in Figure 3 is the XML XPath name of the FGDC element whose value will be updated. In this case, the *End Date* of the FGDC *Time Period* element will be updated.

The configuration parameters tell the Java classes to look at the netCDF variable named "time" and to extract the *units* attribute and element 1 of the array attribute *actual_range*. As they are stored in the netCDF files, the values of these two attributes (units and actual_range) are not suitable for inclusion in an FGDC file. The units attribute is a string with a value of *days since 1-1-1 00:00:0.0* and the actual_range element is a large double containing the number of days from the time origin of the beginning and end of the time period covered by the data set. These two attributes must be transformed into a date string formatted according to FGDC rules.

The transformation class listed in the <transform> configuration element is the code responsible to for converting these two attributes into the FGDC formatted date string. The code for this class is available with the Java package. The beauty of supplying the class responsible for making the transformation along with the attributes to be extracted is that for the cost of writing a little bit of Java code this Java package can be used to extract *any* information from the netCDF file that is useful for populating the FGDC record. The transform class can depend on *any* netCDF attribute the author wishes. By listing the netCDF attributes explicitly in the configuration, the code can be generalized. For example, but changing the <outputElement> to begdate and extracting the 0 element of the actual_range this same transform class can be used to update the *Beginning Date* of the FGDC *Time Period* element.

### *3.3 Connecting the Repository to the Java Classes*

As originally implemented, the results of all the transformations were written out in as an XML FGDC file fragment containing only the changed elements. This provides a simple way to extract the information from the netCDF file and incorporate the results into the FGDC. However, since the repository is implemented using a powerful back-end tool, it is possible to use Java classes supplied by the software vendor to insert the results directly into the repository. The update process can be totally automated by using the supplied classes.

## 4. CONCLUSIONS

Undeniably, creating good metadata is an expensive and laborious task, but many of the features of the Data Discovery Support System can reduce the cost for data providers. In addition to reducing the initial cost, the system can protect the provider's investment by making it easy to keep the metadata current.

The Data Discovery Support System has many advantages. First of all metadata can be entered using an easy to use Web-based user interface. Instead of requiring data providers to decipher the complexity of the FGDC standard they can rely on the user interface to guide them through the process of specifying their

metadata. The interface also makes it easy to reuse information to create records for many datasets.

Secondly, once the metadata record is included in the repository details in the record can be kept current automatically.

Finally, the information in the repository can be easily extracted (on-demand if desired) as HTML, XML or as conventional FGDC documents. Being able to extract the data in this way closes the loop back to the data provider by offering a single solution to the practical considerations that drove the provider toward many different forms of metadata in the first place

**5. REFERENCES**

The Federal Geographic Data Committee, The Content Standard for Digital Geospatial Metadata (CSDGM Version 2 - FGDC-STD-001-1998) http://www.fgdc.gov/metadata/contstan.html

Schweitzer, Peter N., 1995, MP: A compiler for formal metadata: U.S. Geological Survey, Reston, Virginia


**Figure 1 The user interface for creating default values.**

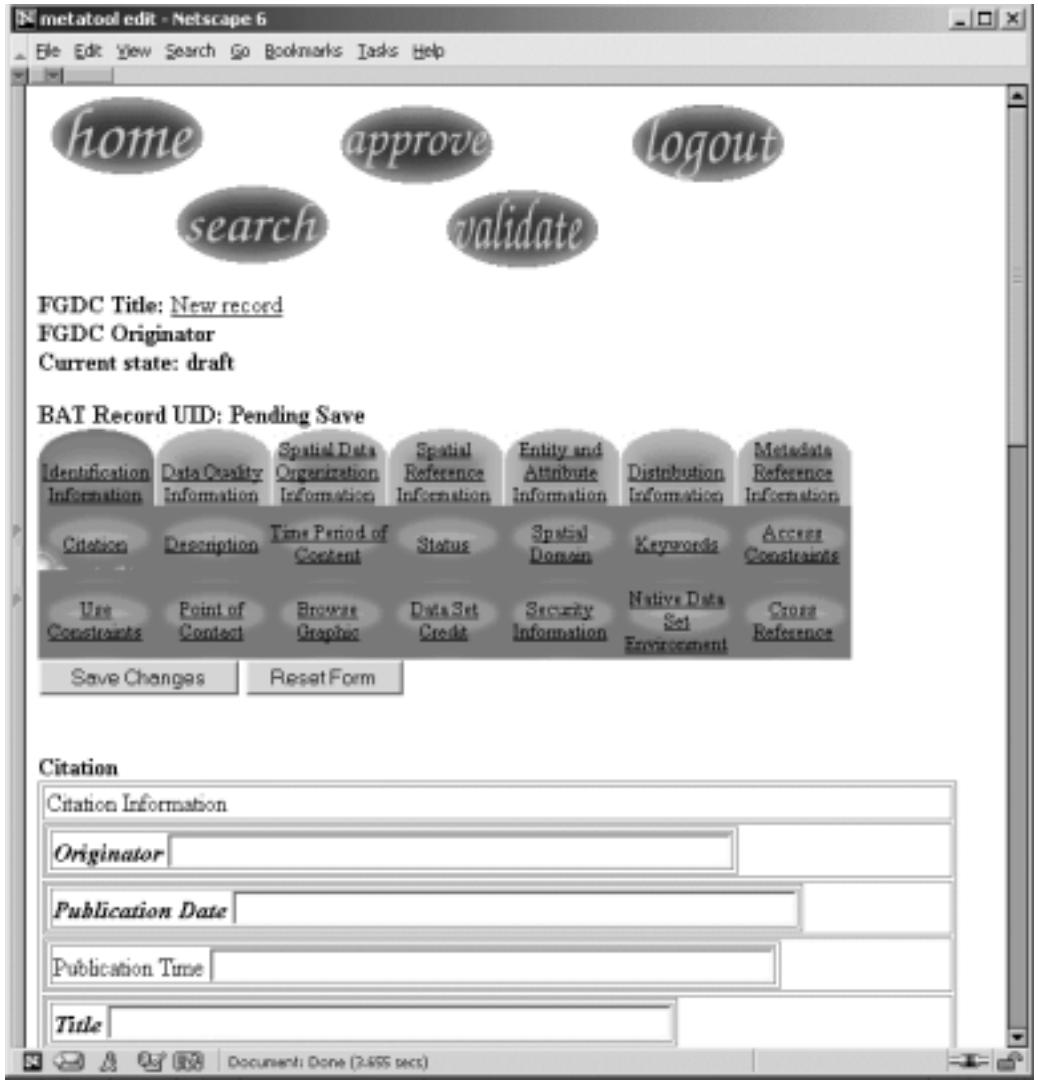


**Figure 2 User interface for creating an FGDC record.**