CHANGING THE WAY SCS STORES CLIMATIC DATA

By

John M. Huddleston[1], P.E. and James K. Marron[2]

ABSTRACT

The mission of the Soil Conservation Service Climatic Data Access Facility is to access, obtain, evaluate, manage, and disseminate the climatic data needed to support agency programs and activities nationally. The objective is to increase efficiency and effectiveness in climatic data utilization in SCS through the use of a centralized climatic data service center. Historical climatic data obtained in 1985 has been stored online in a custom built data base in Portland. The underlying storage architecture is being replaced by netCDF, an I/O library callable from C or FORTRAN which stores an retrieves scientific data structures in self-describing, machine independent files. Using netCDF will improve the accessibility to the data for scientific data access, management, analysis, and display.

INTRODUCTION

The Soil Conservation Service (SCS), manages over four gigabytes of online climatic data. The data base was established in 1985 to support the SCS water supply forecasting program (Shafer and Huddleston, 1986). The original database was established as ODB, Operational Data Base, under the umbrella of the Centralized Forecast System (Huddleston and Shafer, 1989). The system was enhanced from 1985 through 1989 to further support conservation planning and SCS program activities.

In October, 1989, approval was given for the formation of the Climatic Data Access Facility (CDAF) within SCS. Shortly thereafter, the climatic data portion of the system was re-established under CDAF operations as CDBS, the Centralized Data Base System. The region of influence was extended from twelve western states to all fifty states, as well as the Pacific and Caribbean Islands. In addition, the number of daily sensors was increased from a basic three (precipitation, minimum temperature, and maximum temperature) to a variable number of sensors. Further, under ODB only a specific number of stations data had been loaded, whereas, under CDBS all National Climatic Data Center (NCDC) climatic daily stations were loaded.

Until recently, climatic and hydrologic data was stored by SCS in an online data base built upon a relational model. This paper describes why and how the underlying storage architecture is being replaced by netCDF, network Common Data Format, an interface library callable from C or FORTRAN which stores and retrieves scientific data structures in self-describing, machine independent files.

[1]Computer Engineer, SCS, 2625 Redwing Rd., Fort Collins, CO 80526

[2]Resource Conservationist, SCS, 511 NW Broadway, Portland, OR 97209

CENTRALIZED DATA BASE SYSTEM (CDBS)

CDBS Design

The major instrument for the delivery of climatic data is a national online data base of TD-3200 NCDC data. This includes data from the continental United States sites, as well as sites in Canada, Mexico, Puerto Rico, Hawaii, the Pacific Islands, and the Virgin islands. The principle features of the data base include a data loading function, a data access system to display data reports and perform analysis routines, a data update process, and a site location update process. The load and update commands are designed to be executed as one line commands with arguments by the data base administrator. The data access system is available to everyone with an access id to the computer system. The data access system includes a parsing query language for finding the stations that had been loaded. It also includes specific preprogrammed functions to output the data and/or analyses for the selected stations.

CDBS Warehouse

The data base holds historical hydrometeorological data from a variety of sources. The system contains monthly data for 2505 snow courses (date of measurement, pillow depth, and snow water equivalent), 978 stream gauges (volume), 312 reservoirs (storage volume), and 2022 precipitation (amount) stations.

All of the data loaded into the data base existed in other data bases, both SCS and non-SCS. The design intent was to put portions of the various data bases into a single data base for central accessing. Most of the snow data in the data base has been collected by SCS, however, other agency snow data is also present. The precipitation data is primarily from the National Weather Service (NWS) monthly summaries. Streamflow and reservoir data from state and federal water resource agencies is also contained in the database. While many stations exist, only those required for hydrologic modelling or conservation practice design have been loaded.

The data base contains daily data from 656 SCS SNOTEL (SNOw TELemetry) sites. The SCS meteorburst technology system uses the reflection of VHF signals by ionized meteor trails to enable communications between remote data collection sites and a master station. As many as 64 channels (sensors) of data can be stored in the remote site transceiver. At a minimum, battery, snow water equivalent, total precipitation, minimum daily temperature, maximum daily temperature, and the ambient air temperature are recorded. Some locations also record wind run, soil moisture, soil temperature, solar radiation and humidity.

There are 16,999 climatological stations that hold a variety of daily weather observations loaded from NCDC data tapes encoded in the TD-3200 format. The NCDC is located in Asheville, NC and is the repository for the National Weather Service (NWS) climatological data. A total of 180 6250 bpi variable block, variable length record data tapes were used to load the daily climatic data into CDBS. These tapes contain all the daily weather observations for the period of record through calendar year 1991 for individual states. After these tapes were loaded, annual tapes for 1992 to the present year (1993) have been acquired for appending climatic data to the database files.

There are 309 daily stream gauge stations and two daily reservoir stations that have been loaded into the system. While there is greater capacity for these types of daily data, only those daily stations required for hydrologic modelling have been loaded.

CDBS Structure

All data are stored in a water year format, October 1st through September 30th of the following year. It is conceptually comprised of two major sections that are integrated through a common data base architecture. These sections are defined by the time steps of their data: daily and monthly (See Exhibit #1).

Exhibit 1. - DATA TYPE DESCRIPTIONS

```
0. SNOW   - date of measurement,          (mmm-dd-yr)
          - snow water equivalent         (inches)
          - snow depth                (inches)
1. PREC   - precipitation                 (inches)
2. STRM   - monthly streamflow            (acre-ft)
3. RESV   - monthly reservoir         (acre-ft)
4. MISC   - reserved
5. SNOT   - SNOTEL data                   (multiple)
6. CLIM   - CLIMATE data                  (multiple)
7. STRD   - daily streamflow              (acre-ft)
8. RESD   - daily reservoir                   (acre-ft)
```

Only the snow monthly data type is stored in an ASCII format. All other monthly data have a binary structure (See Exhibit #2).

Exhibit 2. - Monthly Data Structure

```
struct monthlystr {
     char      basin[2],       /* basin or state */
               station[8];          /* 4 bytes for PREC, 6 for
                                ** STRM and RESV. The 7th byte
                                ** is the scale for STRM
                                */
     short          cnt,             /* count of number of years */
               data[400][13]; /* data[i][0] is the ith year
                                ** data[i][j], j = 1,..,12
                                ** are the monthly data
                                */
};
```

The ith year would be stored in data[i][0] and the data are stored in data[i][j], j = 1 to 12. The number of bytes to be written from this structure for a station would be 12+(cnt*26), where cnt is the number of years. Each data item will be 9999 if the field is all blank and negative if the data is estimated, (i.e. the first character in the field is an 'E.'). In the retrieval process, the data must be divided by 100.

Monthly files have an index file associated with them that contains the site id and the byte location of the start of the monthly data for that site. In the non-snow monthly data file, the first 12 bytes are used to identify the basin, the station id, and the count of the number of years of data to follow. Thus, for the monthly data types, the data are stored in variable length yearly records within the structure.

In the daily data types, all data is stored in a binary structure that contains one year of data. (See Exhibit 3.) The index files associated with the daily data have the station id and the byte position in the file for the beginning of the station. While the monthly data is all located in one section, in one structure. The daily data may be located in multiple sections in the data base file. The historical data (typically years 48-84 for NCDC data) will be located in one large chunk and annual section has been appended to the end of the file at the end of each water year. While a dense index would contain the id and byte position for each year of data, this data base index can be categorized as a sparse index since it only identifies the byte position where the id starts.

Exhibit 3. - Daily Data Structure

```
struct dailystr {
     char       station[20],
                type[2];
     short           year,
                data[366];      };
```

CDBS Redesign

During the 1989 restructuring, the system was redesigned to bind to the data at execution time. The program is not "hard-coded" into one set path to the data files. Instead, a file specification argument is passed to the program which establishes file paths and environment parameters. The applications were redesigned and reprogrammed to access data base files through an access library. This redesign was made to facilitate the design and programming of new functions within the data base. The flags within the data for daily data were changed to allow for more homogeneous retrieval from the application programs. After the redesign, the identifying flags were made the same for like sensors. The C structure was changed to shorten the storage size of the NCDC data. While there was a redefinition of the character array in the new structure, the actual data organization did not change for SNOTEL data, and thus, SNOTEL data did not have to be reloaded.

The value 9999 represents blank or missing data. Each SNOTEL data element is multiplied by 10 before being converted to a short integer for storage. The data must be divided by 10 when retrieved. The NCDC climatic data is multiplied by 100 before storage as a short integer. Careful examination of the units of the NCDC data, per the TD-3200 report, is made to ensure that all data is stored in the same manner.

CDBS Data Access

Most users are primarily interested in data retrieval and report generation to address a specific problem. This task is started by invoking the query language that locates and retrieves stations matching key attributes (see Exhibit 4.). Key attributes are defined as cross reference identifiers and geographic location information that are used to locate data.

Exhibit 4. - 12 Key Attributes of XREF Records

1. state name or FIPS id (40 char.)
2. latitude (deg min secs)
3. longitude (deg min secs)
4. elevation (feet msl.)
5. site name (30 char.)
6. station ID (8 char.)
7. data type (4 char.)
8. region (2 char.)
9. subregion (2 char.)
10. accounting unit (2 char.)
11. cataloging unit (2 char.)
12. county fips code (3 char.)

Depending upon the retrieval key selected, a location process is started that uses the various sets of location indices based on station names, station identifiers, hydrologic unit code (8-11), or county FIPS number. This process coupled with the sparse data location indices for the data files provides the mechanism for fast data retrieval.

For example, the following command 'find state colorado and datatype snow' will locate all snow course stations in Colorado. Station location information is available to list to screen or to a file.

Once the station list has been generated, the data can be retrieved in a user selectable report format. The user enters a command that corresponds to a predefined application. For each data type two possible output formats are automatically made available. One report displays the data in a format easily comprehended by a human being. The other creates a compact format optimized for machine readability. Once a report format is chosen the data can be displayed on a terminal or routed to a file for later use. This latter option is particularly advantageous because it provides a readily accessible input file for other applications programs.

DATA LOAD FACILITY

Data Loading

When data is to be loaded, the station id must exist in an attributes file called 'MAINSITE'. The procedure checks for the existence of a station id using the MAINSITE indices. If the station is not found, the data is discarded, and the next station's data is retrieved. If the station is found in MAINSITE, a check is made of another attribute file called 'NEWSITE'. If the attribute record is found in NEWSITE, then the data has already been loaded and the load procedure skips to the next station. If the attribute record is not in NEWSITE, then the attribute record is added to NEWSITE and the data is loaded into the appropriate data file. The data load procedure is a manually intensive procedure that has been automated with the use of Data General computer AOS/VS CLI macros and UNIX shell scripts. First, there is a separate load program for each data type. Next, when the data load program has completely read all the data, another program must be run against the data file to create an ASCII file containing the starting byte position for each station in the data file. These ASCII files can be described as sparse data location index files. They contain the station id (or the station name in the case of SNOTEL data only), a tab character, and the byte location number to the beginning of the station's data. All of these files end with the three characters CNT. When all of the data types have been loaded, an index process must be run against the NEWSITE file to create binary indices used for station retrieval by the data access system.

Update Facility

A custom data base update mechanism written in C is designed to manage insertion, updating, and appending of the data described. Data base editing is performed in three stages: 1) by using the data access system to output a data table to disk; 2) using an editor to update the table; and 3) using the update facility to reload the data. The first lines in all the data tables identify the path to the data. If the user elects to delete data records a minus sign is placed in the first column. The user would then only include those lines to be deleted. If there is a plus sign in the first column, then the data records that follow will be appended. If there is neither a minus nor plus sign, then the records that follow are assumed to be edited records for updating. The only exception to this is the use of an asterisk to identify an older format, or 80 column card format, of the data in the table. If an asterisk is used to append or delete records, it must follow the plus or minus sign. In an edit process, there will be no change in the byte count of the file and the sparse data location indices will not have to be computed. The data is first assigned into the appropriate data structure, the position location in the data base file is determined, and the C 'write' function is used to overlay the edited data over the old data. If the append or deletion process is used, a new file will be created for the data. First the data up to the station to be modified is written to the file. The modified station's data is written to the file. The remaining data from the original file is added to the new file. The new file is moved over the original file with a system call using the MV/UX 'mv' command. A byte count must be computed for each station's position in the file to create new sparse data location indices.

NCDC Error Correction Update

An error identification procedure was performed by NCDC which produced an ASCII list of nearly 500,000 errors in the one billion value database. A data correction procedure was implemented which used the ASCII list to insert the corrected values into the online data. This effort made CDBS the only online repository of corrected climatic data in the country.

ANALYSIS OF PRESENT SITUATION

It was important to discuss the level of detail for the creation and management of the attribute and data records, the data loading, the data update, and query process to demonstrate the increased level of system administration required to maintain the system. Whereas, the 1985 ODB system had less than 2,000 climatic stations with only three elements per station, the CDBS system has nearly 17,000 stations with an average of 7 elements per station. The update process under ODB was managed adequately by the Portland water supply staff, water supply specialists, and snow survey supervisors. However, in its present configuration, the CDBS system has outgrown the current computer's capacity both in terms of online storage, data retrieval performance, and increasing user activity.

The SCS is in the process of migrating all of its offices to a relational data base management system (RDMS) using Informix. CDAF was faced with the task of storing nearly a billion data values in this RDMS. Initial schema designs indicated that several terrabytes of disk space would be required, with potentially excessive seek times. What was needed was to provide SCS with a tool to marry RDMS technology with an efficient structure for storing large amounts of time series data.

In 1993, the decision was made to distribute climatic data sets to support SCS modeling. In May, 1993 the newly formed (1992) SCS National Information Systems Division in Fort Collins, Colorado, met with CDAF to discuss data needs for these systems. As a follow-up to that meeting, an Ad Hoc group was formed to discuss issues related to the storage and retrieval of the CDAF data. While the current CDBS system stores daily duration data, the proposed system would store hourly and 15 minute observed and synthesized data. The current proposed RDMS at the SCS field office will be Informix SE. On a 386 AT&T UNIX system, thirty years of data for five elements require 15 MegaBytes of storage and five minutes for retrieval. This imposed too large an amount of disk space and too long a period for data access. The Ad Hoc team examined RIFF tagged file formats, the NOAA NCAR enhanced BUFR format, and a product developed by UCAR named netCDF. The Ad Hoc team adopted the UCAR netCDF.

NetCDF is an interface for scientific data access and a freely-distributed software library that provides an implementation of the interface (Rew and Davis, 1990). It was developed by the University Corporation for Atmospheric Research/Unidata in Boulder, Colorado. The netCDF library also defines a machine-independent format for representing scientific data (Brown, et al, 1993). Together, the interface, library, and format support the creation, access, and sharing of scientific data. NetCDF data is: 1) self-describing, a netCDF file includes information about the data it contains; 2) network-transparent, a netCDF file is represented in a form that can be accessed by computers with different ways of storing integers, characters, and floating-point numbers; 3) direct-access, a small subset of a large data set may be accessed efficiently, without first reading through all the preceding data; 4) appendable, data can be appended to a netCDF data set along one dimension without copying the data set or redefining its structure; and 5) sharable, one writer and multiple readers may simultaneously access the same netCDF file. There is no cost for licensing; it has product maturity; there is good documentation and user community acceptance; and there is good technology transfer capabilities among users who use InterNet to discuss issues, problems, etc.

On the technical side, the netCDF excels over the relational data performance and disk space requirements. On the same UNIX 386, thirty years of five elements of data store on disk in 1.5 MegaBytes and can be retrieved in 17 seconds. For Local Area Network access, the data can be loaded onto a server of one CPU type and can be accessed from a variety of other hardware platforms including DOS. For distribution to the field, a directory and file naming convention was established for simple and reliable distribution to SCS State, Area, and Field Offices. One netCDF file will contain data elements for a specific duration for a station. Both Fortran and C language libraries are available for developers to access the data as well as a high level Common Data Language (CDL) interface for loading, updating, and data retrieval.

TRANSITION TO netCDF

What is the impact on the current CDBS software in order to transition to the netCDF file storage structure? What are the implications of using this storage structure? First, by storing the data in netCDF, CDAF establishes a format that can be distributed and used nationally by SCS applications. Secondly, the CDBS interface library will be rewritten to access the netCDF data files and the CDBS data access (query) system will still be operational. The load and update functions for CDBS will be rewritten to use CDL and the netCDF library.

The CDAF netCDF design for the storage of climatic data specifies the elements separately (See Exhibit 5).

Exhibit 5. - a sample netCDF CDL definition of daily pillow data

```
            // anything after a double slash is a CDL comment
netcdf dailystr {              // daily duration
dimensions:
     Y = unlimited;
     D = 366;
variables:
     short PILL(Y, D);         // what follows is the self descriptions
     PILL: long_name = "pillow water content of accumulated snow depth";
     PILL: units = "tenths of inches";
     PILL: signedness = "unsigned";
     PILL: valid_min = 0s;
     PILL: valid_max = 1000s;
     PILL: _FillValue = 65535s;
     PILL: scale_factor = 0.100000;
     PILL: C_format = "%3d";
     PILL: scaled_C_format = "%5.1f";
// global attributes
     : sta_id = "sta_id";      // dummy id to be replaced upon load
     : sta_nm = "sta_nm";      // dummy name to be replaced upon load
     : data_tp = "O";          // observed
     : dur = "D";         // daily
     : base_Y = 1961;          // come 1996 this will be 1966
} // end of the CDL definition
```

Conceptually, the storage of climatic data in a binary format is not new to the SCS. Data are presently stored in CDBS as a type short and the scale factor varies between elements. The 1989 conversion of ODB to CDBS removed the scale factors and the display format from within the C source code and placed them in a separate file for program usage. Similarly, the netCDF form is also going to be of type short and the data will have a scale factor depending upon the element. In exhibit 5. above, the scale factor for PILL is 0.10. While not shown here, the scale factor for PRCP (precipitation) will be 0.01. However, the user having access to the data does not have to have the knowledge of the storage format; there is a scale factor definition for all the elements. This is part of the self-describing feature of netCDF.

While netCDF can allow storage of data in a multidimensional way (e.g. sensors, x years, x days), the storage of data by individual element allows for discrete segmentation of the data. The attribute information that is used in the retrieval and identification process can be associated with each element rather than a matrix of data. Using this structure as the basis for the storage, there will initially be a partitioning of the netCDF files by state directories. Within each state there will be a separate netCDF file for each station. This will result in a heavy inode structure for the states with many stations (e.g. 1000+ stations in California), however, the individual files allow for a simple distribution scheme to SCS field offices.

SUMMARY

The SCS embarked on the use of climatic data in 1985 with the design and construction of an online data base of daily data. While NCDC's TD-3200 format daily was available to users with a 9 track tape drive and the resources to read the variable block variable record format, only SCS put a system together to read the data, store it in a fashion that allowed for quick retrieval in an online manner, and produce tabular data as well as probabilistic and statistical analyses of the data. Cheaper and faster computers allowed simulation models to be moved from the mainframe environment to the desktop thus driving a need for more climatic data. As a result of this demand, the CDAF was formed and immediately expanded the custom built ODB data base to go national as the CDBS. Increased system administration activities, and the need to distribute climatic data in a format that can be used in models at the 2500+ SCS field offices, drove the CDAF team to explore alternate methods for the storage of the data.

The analysis of several formats for storing climatic data resulted in the acceptance of netCDF. Performance analysis using Informix SE and netCDF showed retrieval times being reduced by a factor of 12 and storage space being reduced by a factor of 10. NetCDF is supported by UCAR and it has wide acceptance from the scientific community as the means to store time series data. Both high level (CDL) and programming language (Fortran, C, and C++) interfaces exist for netCDF. The definition for the storage of all the elements for daily, hourly, and 15 minute data have been proposed and approved. It remains to complete the transition to use the data at two levels. Functions will be written to extract data for use of climatic at field office. Secondly, the CDBS library will be rewritten to access the netCDF data files and provide a smooth transition for CDBS users.

REFERENCES

Brown, S.A., Folk, M., Goucher, G., and Rew, R., May/June, 1993, "Software for Portable Scientific Data Management". Computers in Physics. V.7, N.3, pp. 304-308.

Huddleston, J.M. and Shafer, B.A., 1989. "The Operational Database in the Centralized Forecasting System". Presentation to the North Atlantic Data General User's Group. 1989.

Rew R. and Davis, G., July, 1990. "NetCDF: An Interface for Scientific Data Access". IEEE Computer Graphics and Applications. pp. 76-82.

Shafer, B.A. and J. Huddleston, 1986. "A Centralized Forecast System for the Western United States". Proceedings of the 54th Western Snow Conference. pp. 61-70.