Appendix: RCC ACIS Data Store

2011-01-09

Overview

The RCC ACIS data store corresponds to the Regional Climate Center (RCC) Applied Climate Information System (ACIS), which stores real-time and historical climate data (see overview: http://data.rcc-acis.org/doc/VariableTable.html). These data are provided via a REST web service (see http://en.wikipedia.org/wiki/Representational_State_Transfer) application programmer interface (API) described here:

http://data.rcc-acis.org

The following document lists available variables (data types):

http://data.rcc-acis.org/doc/VariableTable.html

Support of ACIS in TSTool is being implemented. Additional effort is needed due to the large number of data types and data reductions. The initial effort has focused on basic integration between ACIS and TSTool conventions; however, technical issues remain, as described below.

RCC ACIS and Standard Time Series Properties

The standard time series identifier for RCC ACIS time series is of the form:

```
Location.DataSource.DataType.Interval~DataStoreName
```

More specifically, the identifier follows the following convention:

```
IDType:StaID.VariableSource.VariableName.Interval~DataStoreName
```

where identifier parts are described as follows:

- The location is indicated by an identifier type and the identifier. Because ACIS cross-references identifiers used in multiple data systems, the identifier cannot be guaranteed to be unique without knowing the identifier type. For the above identifier, the identifier type is selected in the following order (these abbreviations are taken from the ACIS REST API documentation): NWSLI, FAA, WMO, WBAN, COOP, ICAO, ThreadEx, AWDN, ACIS. This order may in the future be added as an RCC ACIS data store configuration property. Is this a good order?
- `VariableSource` is taken from the ACIS variable list.
- `VariableName` is taken from the ACIS variable list.
- Interval is determined from the ACIS variable list (e.g., “daily” is translated to “Day”) for consistency with TSTool conventions. Intervals that are not explicitly specified (e.g., “sub-hrly”) or are not directly supported by TSTool (e.g., “weekly”) are converted to “Irregular”.
- `DataStoreName` is the user-defined data store name from the configuration information.
- Data units are taken from the ACIS variable list.
- Missing numerical values from ACIS are indicated by a data value of “M” and are converted to NaN internally, with an “M”flag on the data value.

- Trace numerical values for precipitation and snow from ACIS are indicated by a data value of “T” and are converted to .001 internally, with a “T” flag on the data value. Is this trace value appropriate? A configurable parameter may be implemented for the data store.

Limitations

RCC ACIS data store limitations relative to TSTool standard features are as follows (need to understand ACIS better and perhaps some ACIS enhancements are needed):

- Limited testing of combinations of query parameters has been implemented. Some technical issues have been identified, as documented below, and if resolved, may clarify other issues.
- The TSTool input filter feature for querying data has been developed with database interaction in mind and needs to be further constrained. For example, all criteria are currently treated as “Matches” even though other constraints may be available. Constraints will be clarified in a future release of the software.
- The ACIS general API does not provide the variable list – this would be useful.
- The ACIS variable list is unique according to the “major” (and variable name?). Using a unique string for variable is more human readable than the numeric “major” and consequently the name is used in TSTool. Although abbreviations are mentioned in the ACIS documentation, a universal list has not been published (this would be useful).
- The ACIS general web services provide area information that could be presented as choices – this has not been used – performance would need to be evaluated in populating application choices.
- Although a variety of query parameters can be specified (e.g., for location), not all of the information is returned as metadata in results. Consequently, when displaying lists of station-time series in TSTool, some attributes are omitted or would need to be determine from the query criterial (but this is not as robust). A requested design change is therefore to allow returned metadata for all query parameters (such as clim_div, cwa).
- Query parameters are case specific. This is not a huge issue but can lead to troubleshooting difficulty (e.g., sID vs. sId).
- Minor issue: the bbox query parameter is documented with an example as being surrounded by [ ] when in fact these characters should not be included.
- Area parameters can include multiple values (e.g, postal=CO,WY), but this is not documented in text (only in an example). Some clarification may be appropriate, in particular if this behavior only applies to postal.
- Would it be possible and is it desirable to overlap the area parameters so that subsets of joins could be queried (e.g., HUC that intersects a climate division)? If not, TSTool may need to implement post-processing to do additional filtering (if this is needed by users).
- Area type “basin” does not seem to work.
- Would it be possible to allow “meta” with CSV time series output and provide the results in the first line separated by commas (with double quotes as appropriate to surround commas)? This would allow CSV to provide more context about the data.
- How are start and end dates with year 9999 to be interpreted?
- Major issue: When a list of stations is provided based on query parameters… are the time series guaranteed to have data? Limited testing appears to show that attempting to get data for time series results in no data, even when the full period is requested. Users should be able to get data or clearly see why there is none.
- Although daily interval data seem to work via the API, limited experimentation with variables that have other “Reporting Interval” values was unsuccessful. In particular, clarification is needed on whether sDate/eDate need to be specified to a different precision and will returned data records have date/times with different precision.

- No attempt has been made to use reduced data available using JSON input. Some anticipated issues are:
    - Would it be possible to implement the reductions via a URI rather than having to provide data behind the scenes? This would make the API more transparent for troubleshooting.
    - Need to understand how to map the reduced data time series identifiers to TSTool time series identifiers? For example, can the variable list names be used or will they be inappropriate? In particular, if the interval changes, then some variable names will be inappropriate (e.g., “3-hour precipitation”).
    - Does the reduction result in a time series or a statistic (table of 1+ numbers)? TSTool has some conventions for dealing with time series and statistics and additional evaluation is needed.
    - Should the API for reduced data fundamentally be the same as getting other time series (e.g., specify an interval of monthly instead of daily) or should the user need to specify additional input parameters to control the reduction. This depends on how “acceptable” the defaults are. It would seem that some standard reductions could simply result in more time series choices… but how does the major/variable name change?

Data Store Configuration File

A data store is configured by enabling RCC ACIS HDB data stores in the main *TSTool.cfg* configuration file, and creating a data store configuration file for each data store connection. Configurations are processed at software startup. An example of the TSTool configuration file is shown below. Multiple data stores can be defined using the `[DataStore:DataStoreName]` syntax. For ACIS, this would allow, for example, accessing systems at different RCC or different versions of the web services.

```
# Configuration file for TSTool

[TSTool]

RCCACISEnabled = true

# Startup data stores (note that data store name in config file takes precedence)

[DataStore:RCC-ACIS]

ConfigFile = "RCC-ACIS.cfg"
```

TSTool Configuration File with RCC ACIS Data Store Properties

Properties for each data store are specified in an accompanying configuration file described below.

The following illustrates the RCC ACIS data store configuration file format, which in this example is located in the same folder as the TSTool configuration file and configures a data store named “RCC-ACIS”.

```
# Configuration information for "RCC-ACIS" data store.
# Properties are:
#
# The user will see the following when interacting with the data store:
#
# Name - data store name used in applications, for example as the
#     input type information for time series identifiers (usually a short string)
# Description - data store description for reports and user interfaces (short phrase)
#
# The following are specific to the RCC ACIS data store:
#
# ServiceRootURI - web service root URI, including the server name and root path

Type = "RccAcisDataStore"
Name = "RCC-ACIS"
Description = "RCC ACIS Web Services"
ServiceRootURI = "http://data.rcc-acis.org"
```

RCC ACIS Data Store Configuration File