Appendix: Data Store Overview

2012-05-15

Overview

Data stores are used to configure a "connection" to allow data access, and can be of the following type:

- Database Data Store – used when making connections to relational databases such as Microsoft SQL Server and Oracle. This type of data store is used when a software API (application programmer interface) library is available for direct database communication, typically in an enterprise computing environment. In this case the data store requires a database server name and database name. Database data stores typically are faster but have the disadvantage of requiring that a database be available.
- Web Service Data Store – used when making connections to web services. This type of data store is used when a web service API is available, such as using SOAP or REST web service technologies. Requests and data retrievals occur by making HTTP requests rather than direct connections to a database. Web service data stores typically are slower than database data stores but have the advantage of not requiring the database to be installed locally. One issue with web services is that it is more difficult to provide data browsing features because lists of stations, etc., must be retrieved with a performance penalty. TSTool compensates for this by caching data locally in memory or on disk; however, this can require more complex code. Internet access is required to use web service data stores and performance is impacted by other internet users. When using web service data stores it may be appropriate to retrieve data to a local format and then use the local data to perform additional processing.

Additional data store types may be added in the future. However, requiring a configuration file is cumbersome for simple file access and consequently data stores will be implemented only in cases where the data store configuration can easily be reused between TSTool sessions.

When opened by TSTool, the data store name is used on the end of time series identifiers (TSID) to indicate the storage location for data. For example, the end of the TSID is as follows:

```
Location.Source.DataType.Interval~DataStoreName
```

It is envisioned that some of the legacy "input types" supported by TSTool will be converted to data stores to take advantage of data store design features, including greater configuration ability.

Data Store Configuration Files

A data store is configured by enabling the data store type in the main *TSTool.cfg* configuration file, and creating a data store configuration file for each connection. Configurations are processed at software startup to enable data stores. An example of the TSTool configuration file for the UsgsNwisDaily data store is shown below. Multiple data stores can be defined using the `[DataStore:DataStoreName]` syntax.

```
# Configuration file for TSTool

[TSTool]

UsgsNwisDailyEnabled = true

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

[DataStore:UsgsNwisDaily]

ConfigFile = "UsgsNwisDaily.cfg"
```

TSTool Configuration File with USGS NWIS Daily Data Store Properties

The following illustrates a data store configuration file format, which in this example is located in the same folder as the TSTool configuration file and configures the "UsgsNwisDaily" data store. Several standard properties are required and additional properties may be used for specific data stores.

```
# Configuration information for "UsgsNwisDaily" data store.
# Properties are:
#
# The user will see the following when interacting with the data store:
#
# Type - UsgsNwisDailyDataStore (required with no changes)
# Name - data store identifier 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)
# Enabled - whether the data store is enabled (default=True)
#
# The following are specific to the USGS NWIS daily data store:
#
# ServiceRootURI - web service root URI, including the server name and root path
# ServiceAPIDocumentationURI - web service API documentation URI, describing
#     the syntax, input, and output
# ServiceOnlineURI - web service interactive page to query data, typically
#     "drill down" or form based

Type = "UsgsNwisDailyDataStore"
Name = "UsgsNwisDaily"
Description = "USGS NWIS Daily Value Web Service"
Enabled = True
ServiceRootURI = "http://waterservices.usgs.gov/nwis/dv"
ServiceAPIDocumentationURI = "http://waterservices.usgs.gov/rest/DV-Service.html"
ServiceOnlineURI = "http://waterservices.usgs.gov/rest/DV-Test-Tool.html"
```

Separate appendices are provided to describe each data store type that is supported by TSTool.

Using Multiple Data Stores to Construct a Complete Time Series

Organizations often provide access to data in different formats, quality, and availability. For example, streamflow may be available as "real-time" on one website and "historical" on another, typically for reasons related to data management, quality control, and other reasons. In many cases there is a concern

about real-time data being used for important decisions and therefore such data are labeled as "provisional" with corresponding disclaimers. There also may be technical issues that complicate data access, such as real-time data using one identifier and historical data using another identifier. The use of maps that include layers for stations also can complicate access because a "streamflow station" symbol on a map may not clearly indicate whether corresponding data are available for real-time, daily, monthly, annual, or other representations. Unfortunately, the different data forms can make it difficult to access and use the data in an integrated analysis.

TSTool does not attempt to automatically merge historical and real-time data unless there is clear guidance from a data provider. Instead, separate data stores are provided where appropriate to allow access to historical and real-time data. TSTool can display different data intervals on graphs and in tabular displays; consequently, it generally is straightforward to query and display time series with different intervals. TSTool also provides commands to merge time series into a single time series. The following table illustrates how multiple data stores can be accessed to create a complete time series – this is a partial list meant for illustration. Data store appendices also reference other data stores as appropriate.

Examples of Data Stores and other Data Sources that can be Used in Conjunction to Create Complete Time Series

Data Type	Data Stores for Historical Data	Data Stores for Real-time Data
Climate (precipitation, temperature, etc.)	ColoradoWaterHBGuest (Colorado+), HydroBase (Colorado+), RccAcis (USA+)	ColoradoWaterSMS (minute, hour, day), HydroBase (limited), RccAcis (day)
Diversions	HydroBase (Colorado), UsgsNwisDaily	ColoradoWaterSMS (minute, hour, day), UsgsNwisDaily (day)
Snow	HydroBase (Colorado), Use TSTool WebGet() command with NRCS website (western USA)	Use TSTool WebGet() command with NRCS website (day)
Streamflow	ColoradoWaterHBGuest (Colorado+), HydroBase (Colorado+), UsgsNwisDaily (USA+)	ColoradoWaterSMS (minute, hour, day), HydroBase, UsgsNwisDaily (day)
Well levels	ColoradoWaterHBGuest (Colorado), HydroBase (Colorado)	UsgsNwisDaily?
Well Pumping	ColoradoWaterHBGuest (Colorado, use diversion records), UsgsNwisDaily (?)	

The above table indicates possible sources of information, as recognized by TSTool; however, the intricacies of determining specific stations, data types, etc., are left to the software user and involve an understanding of specific data offerings.

This page is intentionally blank.