Appendix: ReclamationPisces Datastore

2015-08-21

Overview

The ReclamationPisces datastore corresponds to the US Bureau of Reclamation's Pisces database, which stores site and corresponding time series data. Sites are locations where data are collected, such as stream gage stations, reservoirs, and weather stations. The Pisces database was originally designed for use with the Pisces data visualization software (http://www.usbr.gov/pn/hydromet/pisces). More recently, the Pisces database is being used to store Reclamation data extracted from enterprise databases (e.g., HDB and Hydromet) for use with public-facing websites and web services. This allows the public to access a snapshot of Reclamation data without accessing internal Reclamation databases. Pisces uses SQLite for its stand-alone database with the Pisces software and MySQL for the web version. The TSTool software is able to read from the Pisces database to provide data access, visualization, and processing functionality.

ReclamationPisces and Standard Time Series Properties

The standard time series identifier for ReclamationPisces time series is of the form:

```
LocationID.DataSource.DataType.Interval.Scenario[EnsembleTrace]~DataStoreName
```

More specifically, the identifier follows the following convention for Pisces data:

```
SiteID.Server.Parameter.TimeInterval~DataStoreName
```

Time series identifier parts and other time series metadata are described below:

- The TSID location identifier is set to the `seriescatalog.siteid` column value. This is ideally a human-readable identifier for the location, typically a station ID. Because some systems, such as HDB, use numeric identifiers that may conflict with other systems, it may be necessary to use the LocationType: prefix on the LocationID or rely on other parts of the time series identifier to uniquely identify the location.
- The TSID data source is set to the `view_seriescatalog.server` column value, which indicates the data collection network.
- The TSID data type is set to the `view_seriescatalog.parameter` column value, which is a short parameter type matching values in the `ref_parameter` table.
- `Interval` is the data interval using TSTool standards (e.g., `Hour`, `Day`, `Month`, `Year`, `Irregular`). Within TSTool, the precision of date/times for irregular data is typically set when the time series is read and the period of record end dates assigned. The following conversion is made between TSTool and Pisces interval:

Pisces Time Interval	TSTool Interval
`Hourly`	`Hour`
`Daily`	`Day`
`Monthly`	`Month`
`Yearly`	`Year`
`?`	`Irregular`

- The TSID scenario (the optional 5th part of the TSID) currently is not used with Pisces.
- The ensemble trace identifier (optional [identifier] at end of TSID) currently is not used with Pisces.
- `DataStoreName` is the user-defined datastore name from the configuration information.
- The time zone for hourly time series is set to the `sitecatalog.timezone` column value. This is not yet implemented because TSTool only fully supports daily Pisces time series.
- Missing numerical values are internally indicated by `NaN`. Will Pisces data tables ever have a flag but no value, for example for missing values?
- Time series data units are set using `seriescatalog.units`.
- All other column data from `sitecatalog` and `seriescatalog` are set as time series properties and can be accessed in TSTool using `${ts:Property}` notation.

Limitations

ReclamationPisces datastore limitations relative to TSTool standard features are as follows:

- Only daily time interval is currently fully supported. Hour, Month, and Year are enabled similar to Day and additional intricacies are not handled (such as time zone for Hour interval, other than calendar year for Year data).

Datastore Configuration File

A datastore is configured by enabling ReclamationPisces datastores in the main *TSTool.cfg* configuration file, and creating a datastore configuration file for each datastore connection. Configurations are processed at software startup. An example of the TSTool configuration file is shown below. Multiple datastores can be defined using the `[DataStore:DataStoreName]` syntax. Properties for each datastore are specified in an accompanying configuration file described below.

```
# Configuration file for TSTool
[TSTool]
ReclamationPiscesEnabled = true

# Startup datastores (note that datastore name in config file takes precedence)
[DataStore:ReclamationPisces]
ConfigFile = " ReclamationPisces.cfg"
```

TSTool Configuration File with ReclamationPisces Datastore Properties

The following illustrates the ReclamationPisces datastore configuration file format, which in this example is located in the same folder as the TSTool configuration file and configures the "ReclamationPisces" datastore. Authentication for writing data to the database is checked based on the account login and password – typically a read-only account should be used for web access to Pisces.

```
# Configuration properties for Reclamation Pisces database datastore.
#
# Properties are as follows.
# The user will see the following when interacting with the data store:
#
# Name - database identifier for use in applications, for example as the
#     input type/name information for time series identifiers (usually a short string)
# Description - database description for reports and user interfaces (a sentence)
#
# The following are needed to make the low-level data connection:
#
# Enabled - use to enable/disable datastore (default=True)
# Type - must be ReclamationPiscesDataStore
# DatabaseEngine - the database software (MySQL is default)
# DatabaseServer - IP or string address for database server (can be localhost)
# DatabaseName - database name used by the server
# SystemLogin - service account login used for database connection
# SystemPassword - service account password used for database connection
#
# The SystemLogin and SystemPassword can be specified as a literal string or:
# Env:EnvVariable (to specify that environment variable should be used)
# SysProp:JavaRuntimeProperty (to specify that Java runtime property should be used)
#

Enabled = True
Type = "ReclamationPiscesDataStore"
Name = "ReclamationPisces"
Description = "Reclamation Pisces Development database for testing"
DatabaseServer = "localhost"
DatabaseName = "timeseries"
SystemLogin = "XXXX"
SystemPassword = "XXXXXXX"
```

ReclamationPisces Datastore Configuration File

The following table describes the datastore properties.

Datastore Configuration Parameters

Parameter	Description	Default
`Enabled`	Specify whether the datastore should be enabled (`True`) or disabled (`False`). Disabled datastores will not be initialized for use in TSTool.	`True`
`Type`	Must be `ReclamationPiscesDataStore`.	None – must be specified.
`DatabaseEngine`	Indicates database software. For Pisces, it should be `MySQL`. (support for SQLite may be added in the future – this will require packaging a Java JDBC driver with TSTool – have not done this).	`MySQL`
`Name`	The name to use for the datastore, which will be displayed in user interfaces and commands.	None – must be specified.
`Description`	A longer description for the datastore, used in interfaces.	None – must be specified.
`DatabaseServer`	The computer name or IP address for the database server.	None – must be specified.
`DatabaseName`	The database name.	None – must be specified.
`DatabasePort`	The port to use for the database connection.	3306 for MySQL.
`SystemLogin`	The database user account login, which can be one of: • literal string • `Env:EnvVariable` – specify using an environment variable named `EnvVariable` • `SysProp:JavaRuntimeProperty` – specify using a Java Runtime Environment property named `JavaRuntimeProperty`	None – must be specified.
`SystemPassword`	The database user account password. Specify similar to `SystemLogin`.	None – must be specified.

This page is intentionally blank.