TxHIS Central Registry Design Document (Final)

# Description:

The TxHIS Central web service will provide a parameter mapping service that can be used by other applications to access a controlled set of parameters from multiple WaterML 1.0 compliant web services. The TxHIS Central web service will be hosted by TWDB (on a linux server) and will be publicly available. Figure 1 shows where TxHIS Central fits in the overall vision for the Texas HIS architecture.

The proposed mechanism for the use of TxHIS Central for applications to be able to query the service for a list of available WaterML 1.0 services (networks), their WSDL addresses, parameters provided by those services and a mapping of the VariableCode required to retrieve those parameters from each individual service. This information can then be used to retrieve data from the individual sources.

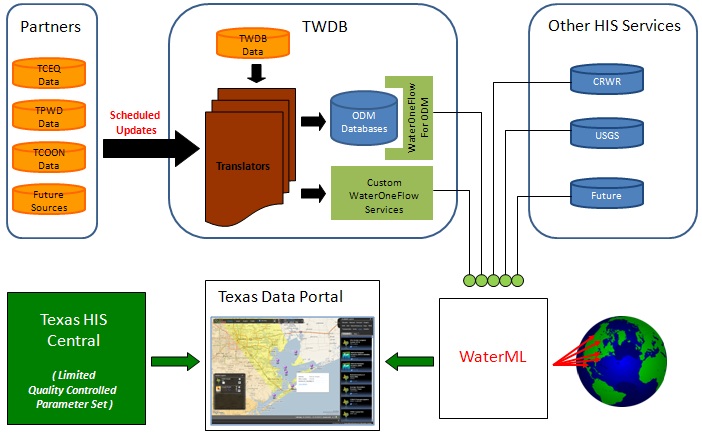


Figure 1. The Texas Water Development Board’s vision for Texas HIS architecture.

# Structure of Backend Database:

The Variables, Units and Sources tables shown in Figure 2 are taken from the ODM 1.1 Schema. The only modifications are the introduction of WSDLLink, UpdateFrequencyType, UpdateFrequencyValue, LastUpdatedOn, RemoteSourceSummerizedDescription, RemoteSourceDetailedDescription, LogoLink, and DescriptionLink 7 fields in the Sources table. The VariableMaps table is a custom table that enables mapping of the variable codes and units across services. The Variables table will contain standardized parameter names probably based on Bryan Enslein’s work (<https://webspace.utexas.edu/bje356/TheODM.mht>)

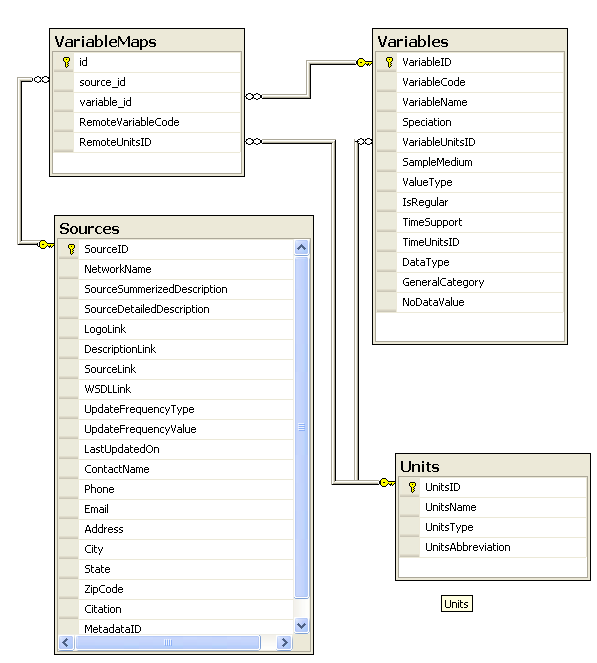


Figure 2. TxHIS Central Backend Schema

## Webservice Operations:

1. **GetSourcesGEM:**

Calling this service function gives details about the list of sources and the parameters that are mapped for TWDBParameters

Input : None

Output :

RemoteSourceUrl : A string, Any ODM WSDL web service URL

RemoteSourceNetworkName : A string Network name of interest on the source URL

ScheduledUpdateFrequencyType: A string of enumeration constraints. Possible values of this one could be Daily, Weekly, Monthly, Yearly, and Adhoc

ScheduledUpdateFrequencyValue: A string for corresponding UpdateFrequencyType Scheme.

Proposed format: yyyy-mm-dd-hh:mm

LastUpdatedOn: A Date of last Updated date of the remote ODM database or NULL.

RemoteSourceParameterCodes: This is an array of parameter codes (in string type) that corresponds to the source URL. This code will be used along with SourceUrl and SourceNetworkName values to fetch the data from remote service (via ODM service). The additional information required for fetching a data would be the date range which will be automatically assumed as one month, week or day based on the frequency of automatic update that we decide.

SourceDescriptionSummary: A string, one liner description for this source.

SourceDescriptionDetail: A string, a paragraph description for this source.

SourceDescriptionLink: A string, the link of this network.

SourceLogoLink: A string, the link for the network’s logo.

The GetServiceDetails function would return and array of ServiceDetail object each one for one unique network.

3. GetTXHISConvertedOutputGEM:

Input : RemoteSourceUrl, A String, indicating WSDL URL of a network

RemoteSourceNetworkName, A String, indicating the name of a network

RemoteSourceParameterCode, A String, indicating the parametercode of the input array

RemoteValue, A float array, an array of input value (to be converted to standard unit/parameter , and currently the size of this array is up to 1000)

Output : TXHISParameterName, A String, indicating the standardized parameter name.

TXHISParameterDescription, A String, describing the parameter name.

TXHISParameterCode, A String, indicating the standardized parameter code

TXHISParameterUnit, A String, indicating the standardized parameter unit

TXHISParameterUnitDescription, A String indicating the standardized parameter unit description, potentially be useful for unit conversion.

TXHISParameterValue, A float array of unit-converted value. Note the element in this array has to correspond to the element of the input value one-on-one.

In a case when there is no unit difference between TWDB parameter and source parameter, GetTXHISConvertedOutput will still be called by the update job in application and the service module can take care of this condition by returning simply returning the same data that was sent.

In this way, whenever a new mapping parameter is added, the application can just do the same process without any difference.

Next page is the backend data caching flow chart:

**Flow Chart:**

Continue Source URL loop until all sources are completed

Continue Site Loop until all sites are completed

Store the translated value, site information, TWDB parameter information into the table

Repeat getting data and translating it and storing it until all parameters are done

Call translation function and get the value translated into TWDB parameter and units

Get the data for the parameter

Loop through each parameter of interest

Get all the sites from the URL in loop selection

Loop through each sites

Loop through each source URL

The information required in the next step is collected by the application.

Email Message Sent

Automatic

User supplies the necessary information required in next step

Email Message Sent

Source URL, Source Network, Parameters, Date

Manual Trigger

Log File Shown

Email to Users