International Environmental Modelling and Software Society (iEMSs)
7th Intl. Congress on Env. Modelling and Software, San Diego, CA, USA,
Daniel P. Ames, Nigel W.T. Quinn and Andrea E. Rizzoli (Eds.)
http://www.iemss.org/society/index.php/iemss-2014-proceedings

WaM-DaM: A Data Model to Organize and Synthesize Water Management Data

Adel M. Abdallah[1] and David E. Rosenberg[2]
[1] *PhD Student, Dept. of Civil & Env. Engineering, Utah State University (USA)*
amabdallah@aggiemail.usu.edu
[2] *Assistant Professor, Dept. of Civil & Env. Engineering and Utah Water Research Laboratory, Utah State University (USA) david.rosenberg@usu.edu*

Abstract: Water resources data for a particular area of interest are generally scattered across numerous providers, managers and scientists. We present a data model to organize water management data to overcome the semantic (i.e., data terms) and syntactic (i.e., data structure and organization) heterogeneity of this data. The Water Management Data Model (WaM-DaM) presents a set of proposed specifications for a generic, relational, and open-source data model to support water management and hydrologic modeling. The design of WaM-DaM allows the user to construct networks comprised of nodes and links from a set of user defined/customized objects. Model objects can represent components of a water management networks like reservoirs and rivers as well as all their associated data including operations to satisfy demands. The data model organizes data values for formats like parameters, multi-column arrays, and time series. WaM-DaM also supports structured metadata fields that describe for whom, where, and how data for the components were collected. WaM-DaM enforces a set of controlled vocabulary on the names of object and attributes and other features to facilitate data interoperability among models and maintain consistency and homogeneity of metadata. To demonstrate these features to organize and synthesize data and metadata from multiple sources with different formats, we implement WaM-DaM in a relational database for a simple three-node network in the Little Bear River Watershed, Utah. Results show that WaM-DaM organized reservoir data and metadata from four different sources and thus allowed the user to quickly compare values of reservoir attributes with their descriptive metadata like unit and source using one database query. Ongoing work will test WaM-DaM with larger networks and other data sources to further demonstrate and test its capabilities and show the generality and flexibility of its design. Future work also will develop stored procedures of database queries to automate importing data into WaM-DaM and then export it to several water management models in the required semantics and format.

Keywords: water management networks; relational data model; metadata; controlled vocabulary

1. Introduction

There is an increasing challenge to effectively manage and allocate scarce and variable water resources among competing users. State-of-the-art management requires up-to-date, consistent, accessible, well organized, and documented data and its associated metadata (Hey et al., 2009; Rosenberg and Madani, 2014). Currently, data to describe water systems is scattered across numerous sources (e.g., government agencies, states, and cities), data providers (e.g., web services and water managers), and models (e.g., research, policy, and operational). Each source and provider has its own way to organize and store data and uses varied terms and phrases to describe data (e.g., reservoir, dam, lake, etc.). As a consequence, water managers and researchers spend considerable time to compile data from scattered sources to build models of large systems (CUASI, 2005). There are several existing methods to manage one or a few features of water resources and related data. The Arc Hydro Framework links surface and groundwater features like stream networks, monitoring points, watersheds, and wells within the propriety ArcGIS environment (Maidment, 2002). The relational ODM data model is used to manage and publish environmental observations of time series data and provides structured metadata and imposes controlled vocabulary on its metadata (Horsburgh et al., 2008). The ODM is a popular relational data model in the water resources community and several data discovery applications like HydroDesktop (Ames et al., 2012) have been built on top of ODM. The ODM was designed to connect with Arc Hydro to represent time series data for features in ArcHydro (Horsburgh et al., 2008). The Common Hydrologic Feature Model (HY-Features) describes a conceptual model that identifies hydrologic features across scientific sub-disciplines in hydrology like water bodies and watersheds (OGC, 2007). HY-Features focuses on representing general hydrologic features and their connectivity in a watershed context. The WaDE model by the Western States Water Council is a relational data model that provides high-level summary for water use and availability within political and geographical boundaries (Larsen and Young, 2014). HydroPlatform is an-open source software that was developed to manage and visualize network-based models and their data and some metadata (Harou et al., 2010). The Hydrologic Engineering Center (HEC) of the US Army Corps of Engineers designed the data system storage (HEC-DSS) to organize and retrieve large sequential datasets like time series and paired tabular data. The HEC-DSS is the data backbone for all HEC water management models among others (HEC, 2009). There are also simulation and optimization models that manage their data according to their own data models like the Water Evaluation and Planning model (WEAP) (SEI, 2012).

Together, each of these data models supports one or a few of the five key features needed to organize water management data and synthesize it from different data sources (Table 1). Thus, new methods are needed to simultaneously support all features and to organize and synthesize multidisciplinary water resources data consistently in one place to allow integrated understanding of multidisciplinary water systems (Rosenberg and Madani, 2014).

Table 1: Features needed to organize network-based water management data and many of prior work that supports the feature

Feature	Arc Hydro	ODM	HydroPlatform	WEAP	HEC-DSS
Generic, relational, and open source environment		x			
Create dynamic networks	x		x	x	x
Controlled vocabulary	x	x		x	x
Descriptive and explicit metadata		x			
Supports time series, tabular, text, parameters, binary, and file based			x	x	x

Here, we propose the <u>Wa</u>ter <u>Ma</u>nagement <u>Da</u>ta <u>M</u>odel (WaM-DaM) as a method to organize and synthesise network-based water resources data in a systematic and consistent way. WaM-DaM allows users to represent networks comprised of nodes and links that represent water management system components like reservoirs, demand sites, environmental areas, canals, and rivers. A node represents a physical object like a reservoir, demand site, pump station, etc. A link represents an object that connects two nodes and conveys flow between them (e.g., a pipe that connects a pump station and a house or a river reach that connects two reservoirs). Attributes represent physical and operations characteristics of the node and link components. WaM-DaM uses controlled vocabulary to homogenize the use of terms across the database. The data model also supports structured metadata fields that describe who collected data for the network component and how and where the data were collected. Finally, the data model adopts a relational structure to store data values so others can access those values and their meanings through database queries. This paper presents the design principles and features of WaM-DaM and illustrates the implementation and the benefits for a simple three node network in the Little Bear River, UT. WaM-DaM integrates all aspects of water resources data and allows comprehensive understanding of the systems data that is currently scattered across multiple sources. A final section discusses the benefits of using WaM-DaM and future work.

2. WaM-DaM Logical Data Model

There are three types of water management data that we need to capture in WaM-DaM: i) observations of water supply and demand quantity and quality, economic, and ecological data ii) connectivity (topology) between supply and demand elements, and iii) factual or descriptive information about system components like dam owner and release rules. Each of these kinds of data is described by attributes, values, and metadata including the unit of measurement, data source, and collection method. Metadata help researchers and managers understand the context of data and interpret it correctly (Horsburgh et al., 2008). WaM-DaM draws its parts and components from many existing water data systems and implements them physically in relational database. Relational database systems have numerous advantages such as they increase the value of information, minimize data redundancy, improve data integrity, and enforce consistency (Connolly and Begg, 2010)

In WaM-DaM, numeric (e.g., maximum storage) or text (e.g., dam owner) data values are identified by four fundamental characteristics: (1) the location (space) of the data as node or link instances (e.g., Hyrum Reservoir or Logan River) and the object name to which it belongs (e.g., Reservoir, Canal), (2) the date and time data was collected or observed, (3) the attribute that was measured or observed (e.g., elevation, storage volume, dam owner) along with attribute metadata like units, sources, and methods, and (4) for modeling purposes, the user, project, network, and scenario to which the data value belongs.

The design of the WaM-DaM logical model follows the relational model design methods described by Connolly and Begg (2010). These methods include among others: identify the conceptual model that abstracts the real world objects into concepts and relations and dependencies between them by using the Unified Modeling Language (UML). Then use the Entity-Relationship (ER) diagram to map out the relations between concepts as tables and fields in a relational data model. We also follow Jim Gray's rules in designing data models by considering the most important twenty questions the users want WaM-DaM to answer (Hey et al., 2009). In WaM-DaM design, these questions belong to three categories: build a network (e.g., what are the existing demand and supply nodes instances in a network?), compare datasets (e.g., what are the scenarios for a network and what are the topological differences among them?), and query factual data about the system (e.g., what is the release rule for a reservoir in a scenario and what is the source of this rule?). Figure 1 shows the full logical model for WaM-DaM. The WaM-DaM tables are color-coded and grouped into five major groups that help organize network-based water management data: i) project metadata, ii) network metadata, iii) controlled vocabulary, iv) attribute metadata, and v) storage of data values. The next subsections elaborate on the five major groups of WaM-DaM data model. The required fields in WaM-DaM are indicated in bold font.

2.1 Project metadata

Green tables in Figure 1 provide general information about the user who creates or gathers data in WaM-DaM, the project which the user is working on, the objects the user creates for their project, and their attributes. These objects and their attributes can be reused for many networks. The **Users** table organizes data about the user name, organization, address, etc. A user can create zero or many projects. The **Projects** table contains all the data and metadata of the user's work. It has metadata like the project name and its description. It can have zero or many networks. The **Objects** table is a container that describes general metadata for nodes and links (e.g., reservoir and canal). Each object can be associated with one or more attributes. For example, a reservoir can have attributes like capacity, dam height, and inflow. The use of node and link object names is controlled by a set of vocabularies like "Reservoir" and "Canal". Finally, the **Attributes** table defines one or more parameters or variables like elevation and storage for an Object.

2.2 Network metadata

Blue tables in Figure 1 contain the instances of objects that exist in the project, their connectivity, and scenarios that in which they appear. An instance is a specific implementation of an object defined in the project metadata table. For example an instance of the object "Reservoir" is Hyrum reservoir in Utah. A node instance can stand by itself but a link instance references a start and end node. The connectivity between node and links instances represents the spatial topology of networks. The instances are connected to their parent objects through the attribute "ObjectName" in the tables of **NodeInstances and LinkInstances.** The **Node and Link Instances** tables contain metadata about instances like name and location. The **Networks** table contains a collection of node and link instances and their data and metadata. The **Scenarios** table contains metadata that describe a scenario like its name, description, and time horizon. A Scenario represents the topological and the data changes in a network from a state to another state like existing case to climate change case. The ScenarioData table serves as a bridge to map out the relation where a scenario can apply to many data attributes and vice versa.

2.3 Controlled vocabulary

Purple tables in Figure 1 enforce consistent metadata terms in WaM-DaM. A controlled vocabulary term describes an object or an attribute in a specific context and this description is enforced throughout the data model. WaM-DaM enforces controlled vocabularies on names of objects (e.g., Reservoir and River) and attributes (e.g., Dam Owner and Storage).

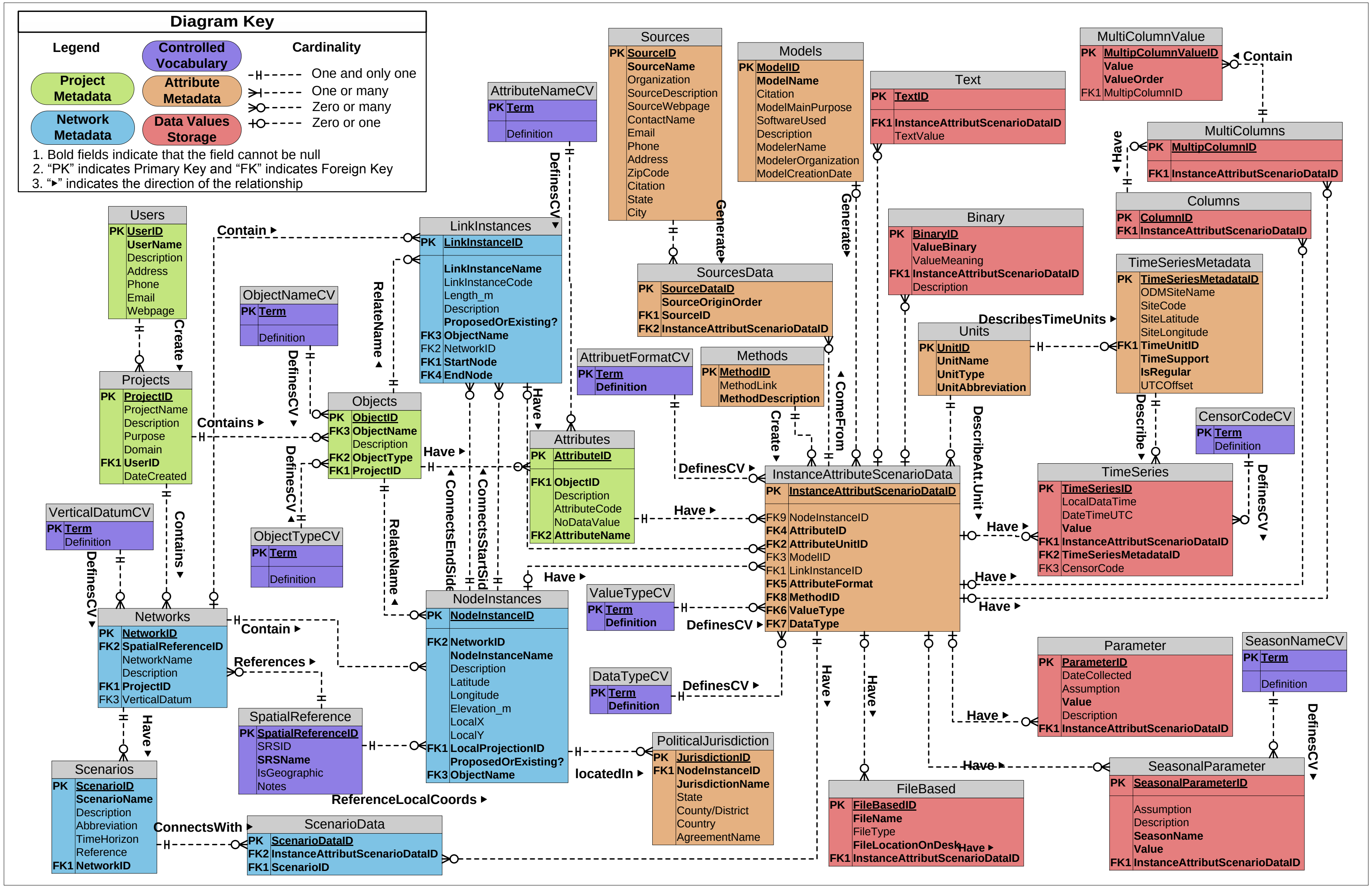

Figure 1: The proposed logical model for the Water Management Data Model (WaM-DaM). Tables are grouped into five groups i) project metadata, ii) network metadata, iii) controlled vocabulary, iv) attribute metadata, and v) data values storage

Other controlled vocabulary tables like Data Type (e.g., average and maximum), value type (e.g., observation and simulation), and Units (e.g., acre-foot and meter) are adopted from ODM (Horsburgh et al., 2008). For example, the term "reservoir" is used in WaM-DaM to describe surface water body that is man-made and is built for water supply purposes. The enforcement of this definition in WaM-DaM avoids the use of other synonyms like water body and dam and therefore reduces confusion on the interpretation of object, object instance, and the associated data. If a user has a dataset of dams and wants to import it to WaM-DaM, first they should define an object to represent their dam. WaM-DaM offers the name "reservoir" that could represent "dams", so the user chooses the name "reservoir" for their object. Then the user can query the two datasets that are organized in WaM-DaM for one object which is reservoir. In this way controlled vocabulary help to consistently integrate multiple datasets in WaM-DaM. We have not fully defined the controlled vocabularies, and should a user not find a suitable term for their modeling needs, they can also define and add their own new controlled vocabulary term. However, it is the user's responsibility to maintain consistent terms.

2.1 Attribute metadata

Orange tables in Figure 1 provide metadata that describes attributes of a specific object instance in a particular scenario. The **Methods** table describes the method used to collect an observation or the methods used to release water from a reservoir. The attribute MethodDescription has a text type that allows the user to describe their method in details. The **Sources** Table describes the name of data provider for an attribute of a specific instance. The same attribute for the same instance can have multiple similar or different values that come from one or multiple sources. For example, the Army Corps of Engineers might report the maximum storage capacity for a reservoir as 16,290 Acre-feet while the State of Utah dataset reports it as 15,300 Acre-feet. Thus WaM-DaM stores both values for the two sources. The **Model/Software** table stores metadata like model and modeler name in case the data is a result of model simulations. The **Units** table describes the unit of measurement such as the name and abbreviation for a particular attribute**.** Each unit can be associated with zero or many attributes but an attribute can have only one unit. Text and FileBased data formats take dimensionless units. The **InstanceAttributeScenarioData** table maps the metadata for an attribute of a particular instance to its data values based on the attribute data format selected by the user (e.g., time series). The table has relations (foreign keys) to all metadata (methods, sources, samples models and software). In addition, an attribute for a specific instance in this table connects with a scenario (e.g., base case) through the ScenarioData table.

2.4 Data values storage

Red tables in Figure 1 physically store data values for a specific attribute of a particular object instance. The **TimeSeries** table stores data values and their time stamp. It also captures metadata like CensorCode which indicates whether the observation is censored (i.e., below or above a detection limit). The **TimeSeriesMetadata** table captures metadata for the whole block of time series like the site the data was collected at (see Horsburgh et al. (2008)). The **Text** table stores text data values for an attribute (e.g., names of reservoir zones like dead, conservation, and flood pools). The **Binary** table stores binary data values (i.e., 0, 1) and reports the meaning of these values. For example, status of gates as open=1 or closed=0. The **Parameter** table stores values of single numeric parameter like an elevation of 45.5 m. The **SeasonalParameter** table stores sets of numeric parameters that have seasonal patterns (e.g., water rights that are 20 acre-feet in winter and 5 acre-feet in summer). The seasons here are not necessarily the four seasons but they can be other user-defined periods such as holiday vs non-holiday seasons. The **FileBased** table stores references for data stored in files like maps and images. The **MultiColumns** (tabular) table stores arrays that have multiple paired columns. The next section provides an example of how the MultiColumns data can be stored in a relation way in WaM-DaM.

3. WaM-DaM Implementation and Discussion

WaM-DaM is physically implemented in a relational SQL Server database that comprises thirty-six related tables and 173 fields. Then, WaM-DaM is populated with a simple network of three nodes and two links in the Little Bear River, Utah (Figure 2). Even this small network can show the generality, flexibility, and benefits of using WaM-DaM. For example, we populated this network with data from four different sources that use different semantics (e.g., dam and lake), metadata, and data formats (e.g., time series, multi-column, and parameter). These data sources include: i) shapefile dataset of One Million-Scale Water bodies and Wetlands (Water bodies Dataset, hereafter) (NAUS, 2013a) that includes about 27,000 water bodies and wetlands in the US and has fifteen attributes like lake perimeter, area, code and region for each body, ii) shapefile dataset of the National Inventory of Dams (Dams Dataset, hereafter) (NAUS, 2013b) that includes over 8,000 major dams in the US and has twenty four attributes like dam purpose, storage, and drainage area for each, iii) WEAP model for the Lower Bear River basin created by the authors and derived from a Utah Division of Water Resources GenRes model for the Lower Bear River that includes the Little Bear River as tributary and has data like reservoir bathymetry (elevation-storage-area) and river head flows. The fourth data source is stream discharge time series data that was originally collected by the Little Bear River Test Bed Project (http://littlebearriver.usu.edu/). This time series data is organized according to ODM standards and was downloaded through HydroDesktop (Ames et al., 2012). All data was exported from the sources to Microsoft Excel spreadsheets then imported one at a time to the WaM-DaM database in SQL Server. The use of Excel as an intermediary step was for simplicity; future work will use Python scripts to seamlessly automate loading data to WaMDaM. Figure 2 also shows how the data sources were used to populate each object's instance in the Little Bear River network. The Water bodies, Dams, WEAP, and ODM time series data sources serve as proof of concept for our initial development.

Figure 3 illustrates how a MultiColumn reservoir bathymetry can be stored in WaM-DaM with an example on "storage" column format attribute for Hyrum reservoir. In Figure 3, Node and link objects are connected with node instances through the ObjectName (i.e., Reservoir). Attributes (i.e., storage) are connected with objects (i.e., Reservoir) through ObjectID. Units (i.e., Acre-foot), instances (i.e., Hyrum), and attributes (i.e., Storage and ElevationAreaStorage) are connected with InstanceAttributesScenarioData table through NodeInstanceID, AttributeID, and AttributeUnitID. Columns and MultiColumns are connected with the InstanceAttributeScenariosData table and then columns (e.g., storage) constitute the MultiColumns (i.e., ElevationAreaStorage) table. Finally the each of the columns in the MultiColumns table is populated with its data values. The value order in The MultiColumnValue table is stored to preserve the paring between columns data values. Finally a database business rule enforces that there is the same number of entries in each column. Note that not all the tables in WaM-DaM are shown in Figure 3 like Scenarios and Networks. Organizing and storing data in this relational format allows users to perform sophisticated queries for large datasets.

With the data loaded into WaM-DaM, the user can readily answer questions that previously required significant effort and manipulations among multiple data sets. Query results in Table 2 show the values for Hyrum Reservoir attributes that are parameters along with attribute key metadata like attribute name, unit, and source name. The controlled vocabulary term "MaxStorage" was enforced for maintain consistency in describing this attributes compared to the native terms "Total Capacity" as in the sources of the WEAP model and the Utah Division of Water Resources, "MaxStor" in the Dams dataset. These query results allow the user to compare datasets, identify discrepancies and uncertainties, and include uncertainties in preparing model input data. For example, the Water bodies Dataset reports that the maximum storage of Hyrum Reservoir is 14,440 acre-feet whereas the WEAP model reports this capacity as 18,684 acre-feet. Through querying, WaM-DaM can quickly generate this comparison for all reservoir instances in a network or scenario.

On-going work will finalize the data model and test it with extensive data and larger networks. We will develop a complete list of controlled vocabulary to enforce consistency in the data model. We will document and publish detailed design specifications and features of WaM-DaM. Future work will develop programming scripts to export the network configuration (i.e., node and link instances and their connectivity), attributes, and their data values to populate water management models like WEAP in the required semantics and format. We will also develop stored procedures of database queries to automate the searching for, discovering, and importing of data into WaM-DaM. These features along with automated data export and model population will increase the speed, reproducibility, and reliability with which users can search for water management data, format the data, prepare, and run models. As a result, WaM-DaM will allow managers and practitioners to 1) search for data that are published by other agencies, 2) efficiently import searched data into local data systems, 3) better understand integrated data, and 4) use data to run water management models such as WEAP, HEC-ResSim, and others.

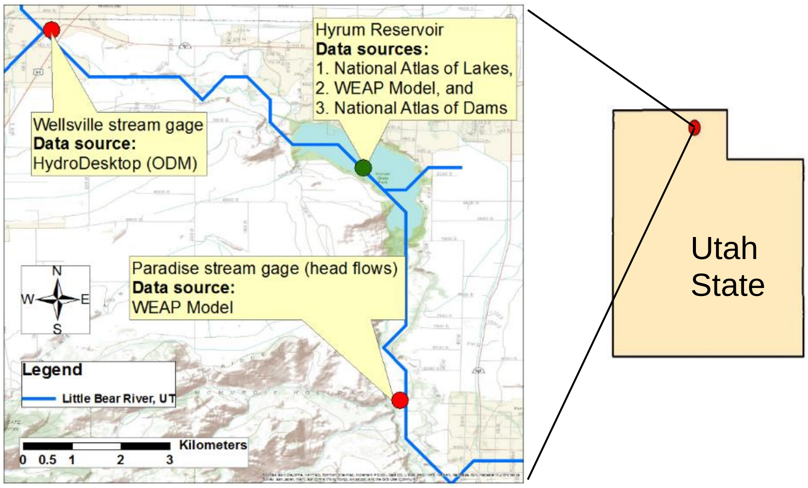

Figure 2: A map of the Little Bear River Network in Utah and the input data sources

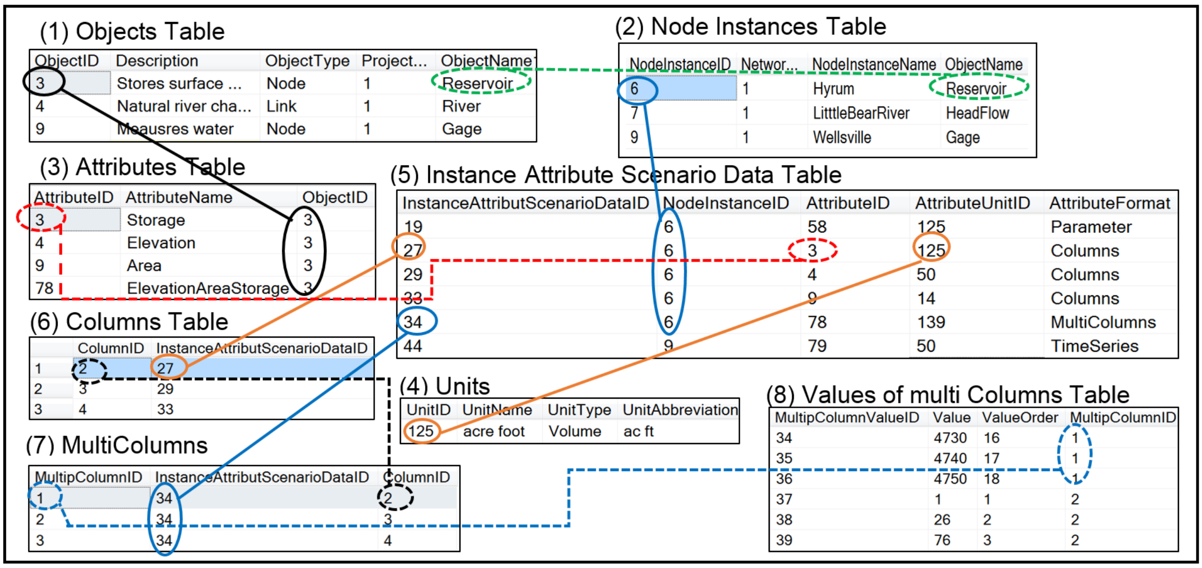

Figure 3: Excerpts from tables illustrating the population of WaM-DaM with MultiColumn reservoir bathymetry data

NodeInstanceName	AttributeName	Value	UnitName	SourceName
Hyrum	Area	451.558	acre	National Atlas Waterbodies
Hyrum	Area	480	acre	National Atlas Major Dams ...
Hyrum	MaxStorage	14440	acre foot	National Atlas Major Dams ...
Hyrum	MaxStorage	18684	acre foot	DavidRosenberg/WEAP
Hyrum	MaxStorage	18684	acre foot	Utah Division of Water Res...
Hyrum	Perimeter	4.343	international mile	National Atlas Waterbodies

Table 2: Results of a query for some of the parameter data that exist in WaM-DaM from multiple sources for Hyrum Reservoir along with key descriptive metadata

4. Conclusions

We propose the Water Management Data Model (WaM-DaM) as a method to organize and synthesise network-based water management data. WaM-DaM integrates five features of water resources data that are used separately in prior work. It represents networks with customizable node and link objects, supports structured metadata, accommodates several data formats, stores data in a relational structure, and imposes controlled vocabulary to maintain the use of homogeneous terms from different users. We demonstrated how WaM-DaM can organize and synthesize water management data that originates from multiple sources that use different semantics and syntactic structure. The use of explicit and consistent metadata and consistent metadata allow users to unambiguously interpret and merge data that originates from different sources. Having all these datasets organized and fully described in WaM-DaM and in one place advances out understanding of water systems data. In the Little Bear River example, WaM-DaM organized reservoir data and metadata from four different data sources and allowed the user to quickly compare values of reservoir attributes like maximum storage and surface area with their descriptive metadata like unit and origin using one database query. Users can also compare datasets and identify discrepancies such as the different maximum storage capacities of Hyrum reservoir reported in the Water bodies and Dams datasets. Ongoing work will refine the design of WaM-DaM and populate it with large datasets. Future work also will develop stored procedures of database queries to automate importing data into WaM-DaM and then export it to several water management models in the required semantics and format. Finally, the use of WaM-DaM can foster integrated understandings of water systems.

ACKNOWLEDGMENTS

WaM-DaM is developed as part of the project "Cyber-Infrastructure to Advance High Performance Water Resource Modelling" (CI-Water) http://ci-water.org. The CI-Water Project is funded by the National Science Foundation through grants EPS-1135482 and EPS-1135483. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

REFERENCES

Ames, D.P., Horsburgh, J.S., Cao, Y., Kadlec, J., Whiteaker, T., Valentine, D., (2012). HydroDesktop: Web services-based software for hydrologic data discovery, download, visualization, and analysis. Environmental Modelling & Software 37(0) 146-156.

Connolly, T.M., Begg, C.E., (2010). Database systems : a practical approach to design, implementation, and management. Addison-Wesley, Boston, Mass.

Consortium of Universities for the Advancement of Hydrologic Science Inc (CUAHSI), (2005). Hydrologic Information System Status Report, In: Maidment, D.R. (Ed.).

Harou, J.J., Pinte, D., Tilmant, A., Rosenberg, D.E., Rheinheimer, D.E., Hansen, K., Reed, P.M., Reynaud, A., Medellin-Azuara, J., Pulido-Velazquez, M., Matrosov, E., Padula, S., Zhu, T., (2010). An open-source model platform for water management that links models to a generic user-interface and data-manager In: David A. Swayne, W.Y., A. A. Voinov, A. Rizzoli, T. Filatova (Ed.), International Congress on Environmental Modelling and Software, Modelling for Environment's Sake ed. International Environmental Modelling and Software Society (iEMSs) Ottawa, Ontario, Canada.

Hey, A., Tansley, S., Tolle, K., (2009). The Fourth Paradigm: Data-Intensive Scientific Discovery. Microsoft Research, Redmond, Wash.

Horsburgh, J.S., Tarboton, D.G., Maidment, D.R., Zaslavsky, I., (2008). A relational model for environmental and water resources data. Water Resour. Res. 44(5) W05406.

Larsen, S.G., Young, D., (2014). WaDE: An Interoperable Data Exchange Network for Sharing Water Planning and Use Data. Journal of Contemporary Water Research & Education(153) 33-41.

Maidment, D.R., 2002. Arc hydro : GIS for water resources. ESRI Press, Redlands, Calif.

National Atlas of the United States, (2013)a. 1:1,000,000-Scale Waterbodies and Wetlands of the United States: National Atlas of the U.S., 14-Jan-2013 ed. http://nationalatlas.gov: Rolla, MO.

National Atlas of the United States, (2013)b. National Inventory of Dams (Shapefile of dams), 30-May-2013 ed. http://nationalatlas.gov.

Open Geospatial Consortium (OGC), (2007). HY_Features: a Common Hydrologic Feature Model Discussion Paper, In: Rob Atkinson, Dornblut, I. (Eds.), OGC 07-041r1. Open Geospatial Consortium Inc.

Rosenberg, D., Madani, K., (2014). Water Resources Systems Analysis: A Bright Past and a Challenging but Promising Future. Journal of Water Resources Planning and Management 140(4) 407-409.

Stockholm Environment Institute, (2012). Water Evaluation And Planning (WEAP) 3.3001 ed.

US Army Corps of Engineers Hydrologic Information Center (HEC), (2009). HEC Data Storage System, 2.0 ed. US Army Corps of Engineers Institute for Water Resources Hydrologic Engineering Center (HEC) Davis, CA.