A Data Model to Manage Data for Water Resources Systems Modeling

Adel M. Abdallah* and David E. Rosenberg

Utah Water Research Laboratory and Department of Civil and Environmental Engineering, Utah State University

*Corresponding author

~~September 20~~December 13, 2018

Abstract (150 words limit)
Current practices to identify, organize, analyze, and serve data to water resources systems models are typically model and dataset-specific. Data are stored in different formats, described with different vocabularies, and require manual, model-specific, and time-intensive manipulations to find, organize, compare, and then serve to models. This paper presents the Water Management Data Model (WaMDaM) implemented in a relational database. WaMDaM uses contextual metadata, controlled vocabularies, and supporting software tools to organize and store water management data from multiple sources and models and allow users to more easily interact with its database. Five use cases use thirteen datasets and models focused in the Bear River Watershed, ~~USA~~United States to show how a user can identify, compare, and choose from multiple types of data, networks, and scenario elements then serve data to models. The database design is flexible and scalable to accommodate new datasets, models, and ~~their~~ associated components, attributes, scenarios, and metadata.

Keywords: data management, systems analysis, systems modeling, data fusion, water resources, open-source (up to 6 keywords)

Highlights (up to 5 points with each 85 characters max with spaces)

- We present a data model to organize water resources systems data and models
- Controlled vocabularies link native terms across different datasets and models
- Software tools manage controlled vocabularies and help load datasets
- Modelers can identify and compare available data then serve data to models

Software availability

Name of software: Water Management Data Model (WaMDaM)

Developer: Adel M. Abdallah

Contact: Adel M. Abdallah; 8200 Old Main Hill, Logan, UT 84322, USA; Email amabdallah@aggiemail.usu.edu

Year first available: 2018

Required hardware and software: The WaMDaM data model can be used within any relational database management system or platform. The WaMDaM ~~Data Loader~~ Wizard executable (.exe) is available for use with Microsoft Excel (2007 and ~~after versions~~later) and SQLite3 on Windows 64-bit computers.

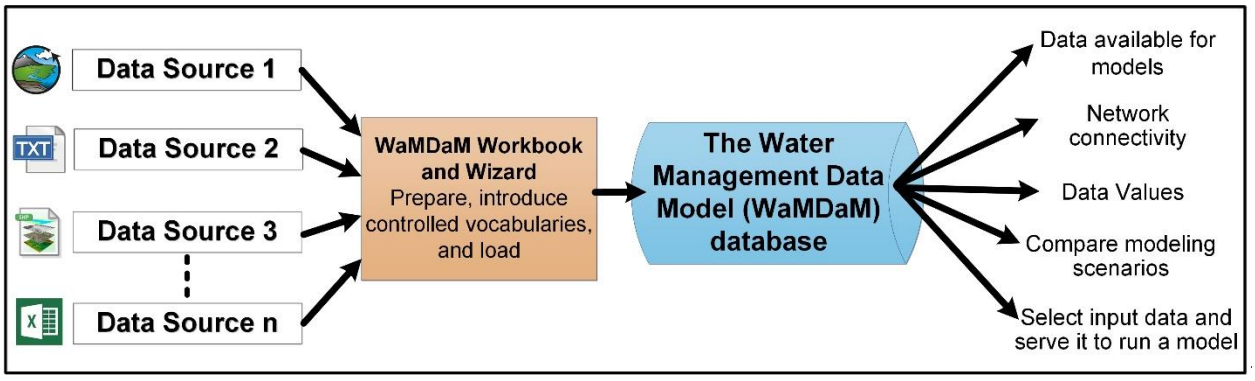
Input data and directions: ~~-~~Documentation of all source code, datasets, use cases, and ~~directions~~instructions to use WaMDaM and replicate results are available on GitHub and facilitated by Jupyter Notebooks at ~~https://github.com/WamdamProject/WaMDaM_JupyterNotebooks~~http://doi.org/10.5281/zenodo.1484581

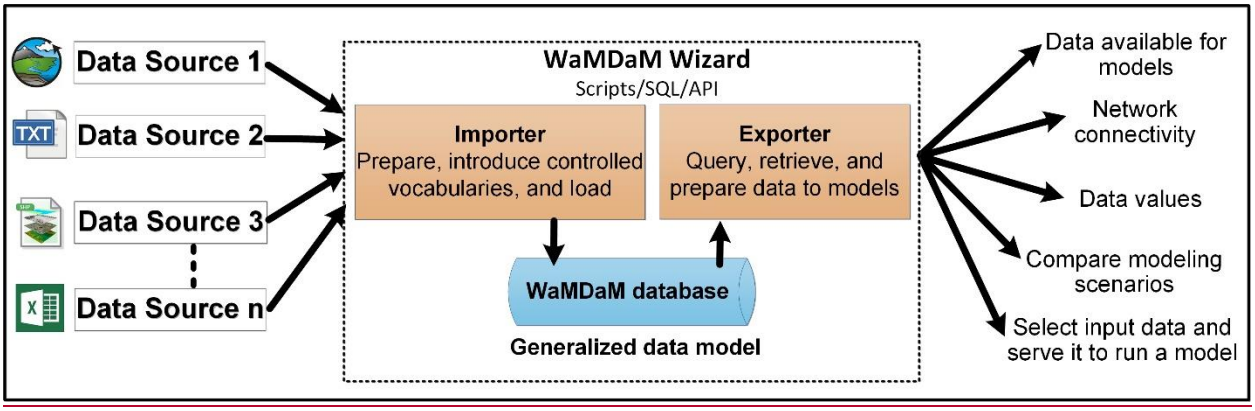
~~[Will create a final DOI before publication]~~

Programming languages: Python 2.7 and Structured Query Language (SQL)

Cost and license: Free. Software and source-code are released under the New Berkeley Software Distribution (BSD) 3-Clause License, which allows for liberal reuse.

Graphical Abstract




Data Source 1
Data Source 2
Data Source 3
Data Source n
WaMDaM Wizard
Scripts/SQL/API
Importer
Prepare, introduce controlled vocabularies, and load
Exporter
Query, retrieve, and prepare data to models
WaMDaM database
Generalized data model
Data available for models
Network connectivity
Data values
Compare modeling scenarios
Select input data and serve it to run a model

1. Introduction

Data analysis and synthesis are fundamental in developing water resources management models (Loucks et al., 2005). Data organization enables or inhibits the analysis that water managers and modelers perform (Brown et al., 2015; Horsburgh et al., 2008). Well organized data can help modelers prepare data for models while poorly organized data can make the process time-consuming and frustrating. Current practices to organize, manipulate, and compare multiple water resources datasets and develop water systems models are typically specific to the data sources, models, and study location (Brown et al., 2015). Source-, model-, and study area-specific practices arise because models have different data requirements for their components, store data in different file formats, have varying spatial and temporal coverage, use inconsistent metadata to describe methods, sources, and units, and use different vocabularies to name similar system components and their attributes (Laituri and Sternlieb, 2014; Maidment, 2016; Miller et al., 2004). These practices limit managers' and modelers' ability to reuse datasets and models in other applications. To reuse, practitioners ~~must~~often spend ~~considerable effort and~~up to 75% of their overall modeling time to modify, subset, transform, convert, and restructure data (Beniston et al., 2012; CUAHSI, 2005; Draper et al., 2003; Hey et al., 2009; Leonard and Duffy, 2013; Maidment, 2008; Michener, 2006; Miller et al., 2004; Ridley and Stoker, 2001; Watkins, 2013). A common database design to organize and manage water resources system data can help modelers and managers spend less time to wrangle with data formats and structures and more effort on ~~modeling and~~ analysis to learn about and model systems.

Water management data describe natural and built water system components like water supply, infrastructure, and demand sites, and these components are typically represented as networks of nodes and links (Brown et al., 2015; Loucks et al., 2005; Rosenberg and Madani, 2014). Each node and link are described with properties that ~~describe~~represent observed values~~,~~ and input data, or variables that store model results. Data can be organized in time series, as seasonal parameters, as multi-variable arrays, or in other ~~formats~~types.

In current practice, a water resources system modeler selects a water management modeling method and then searches for input data that meets the model's requirements (Brown et al., 2015). Modelers often manually search for, download, synthesize, and compare data from disparate datasets to populate input data (Rosenberg and Madani, 2014). In their data search, modelers often use a combination of existing methods to manually gather input data for the different supply and demand system components and their connectivity from local, state, and federal agencies. Searches can also use national data services like the Consortium of

Universities for the Advancement of Hydrologic Science, Inc. (CUAHSI) Water Data Services (Couch et al., 2014; Goodall et al., 2008). Each dataset has a particular file-format, organizational structure, syntax, and descriptive terminology. Some datasets also come with modeling scenarios that represent changes to ~~one or more of the input~~ values ~~that describe base~~of physical, operational, network topology, or socio-economic attributes of the system. Modelers must reconcile ~~heterogeneities in~~ structure and terminology ~~of~~heterogeneities in potential input data.

Many water resources modelers use the U.S. Army Corps of Engineers Hydrologic Engineering Center Data Storage System (HEC-DSS) (HEC, 2009) to store and manage paired variables and time series data. Modelers also use ~~HydraPlatform~~Hydra Platform (Knox et al., 2014) and ArcHydro (Maidment, 2002) for network connectivity. Others may also use the Observations Data Model (ODM) for organizing and storing site-specific time series data (Horsburgh et al., 2008). Other modelers simply organize data into one or many spreadsheets within a Microsoft Excel workbook with consistent column headers (e.g., variables) and units. Still other modelers store data that describe the water system and its operations in proprietary modeling software systems like the Water Evaluation and Planning system (WEAP) (Yates et al., 2005), RiverWare (Zagona et al., 2001), OASIS, ModSim, and others (Loucks et al., 2005; Wurbs, 1993; Wurbs, 2012). Although models like RiverWare (Zagona et al., 2001) and WEAP (Yates et al., 2005) are not strictly used for data management purposes, we consider them data management systems because they contain large amounts of data that describe water systems and house the data used for numerous river basin management studies around the world.

To identify, analyze, or compare water management data stored in one or many of the above systems, modelers often develop source- and model-~~-~~specific workflows to manipulate, join, pivot, sort, aggregate (in time and/or space), and visualize data. Simultaneously, modelers must keep track of metadata, if present, that describe the source of data, methods used for creating the data, and methods used to transform data to a format appropriate for a particular model. These metadata elements are ~~also~~ typically specific to the data source and model. Adding a data source, expanding a study area, or changing the underlying model means the modeler must modify the data preparation workflow or create a new workflow. Modelers then must manually repeat data manipulations and analyses.

Thus, there is a need for a generalized method to more readily and consistently organize, store, join, query, and compare multiple types of water management data and contextual metadata across datasets, models, and study areas (Bajcsy, 2008; Brown et al., 2015; Govindaraju et al., 2009; Vogel et al., 2015). This need arises because of two

fundamental data management challenges related to how data is structured (i.e., syntax) and how key data components are named and described (i.e., semantics). An example of different ~~syntax~~syntaxes is the number and order of headers and rows in a spreadsheet. Examples of different semantics include hydrologic system component names (e.g., “reservoir” versus “storage facility”), attribute names (e.g., “storage” versus “volume”), and system component names (e.g., “Hyrum Reservoir” versus “HYRUM”).

In reviewing more than 40 existing systems to organize water management data (**Appendix A, Table A1**), we found all systems incompletely support structure and syntax issues ~~(Aspen Institute, 2017; Bajcsy, 2008; Blodgett et al., 2016; Govindaraju et al., 2009; Hey et al., 2009; Laniak et al., 2013; Larsen et al., 2016; Loucks et al., 2005; Maidment, 2016; National Research Council, 2012; Order, 2013; Rajaram et al., 2015; Vogel et al., 2015; Wurbs, 2005; Wurbs, 2012).~~. Systems have different and limited capabilities to query and compare multiple datasets and models, no software standards, or no guidelines to organize water management data. Differences include how data is represented in space and time, how data is organized within structures (i.e., data type) (DCMI, 2013), the physical means used to store data (i.e., database, text file, or other formats) (DCMI, 2013), and software technology. The heterogeneity in methods reveals why modelers spend considerable time ~~to prepare~~preparing and ~~move~~transferring data across different models, formats, and technologies.

Several recent efforts to increase data consistency and transparency, such as the Open Water Data Initiative~~,~~ (Blodgett et al., 2016), Observations Data Model 2 (Horsburgh et al., 2016), the Open and Transparent Water Data Act (Cantor et al., 2018; Dodd, 2016), and the Water Data Exchange program (Larsen and Young, 2014) have recommended data standards to integrate fragmented water information data into consistent and interoperable data systems. Such integrations and requests for them aim at improving access to water information to help quantify its availability and use at different scales in the present and future. Here, we contribute a generalizable data model called the Water Management Data Model (WaMDaM) to help organize, join, compare, and analyze multiple water resources datasets and models. ~~The WaMDaM is physically implemented as a relational database.~~We also introduce software tools that demonstrate key functionalities of the design. The WaMDaM design helps answer the overarching research question of: how can data from multiple sources be organized and described in a semantically and syntactically consistent way to facilitate data query, comparison, joining, and analysis that will ultimately help modelers choose input data to build and run water resources systems models? A successful WaMDaM database design ~~requires eight features~~must have: 1) modular and extensible components, 2) networks of nodes and links, 3)

scenarios and version control, 4) reusable contextual metadata, 5) support for multiple data types used by systems models, 6) extensible controlled vocabularies, 7) ~~conditional queries to~~ direct access to subsets of data and metadata, and 8) an open-source environment.

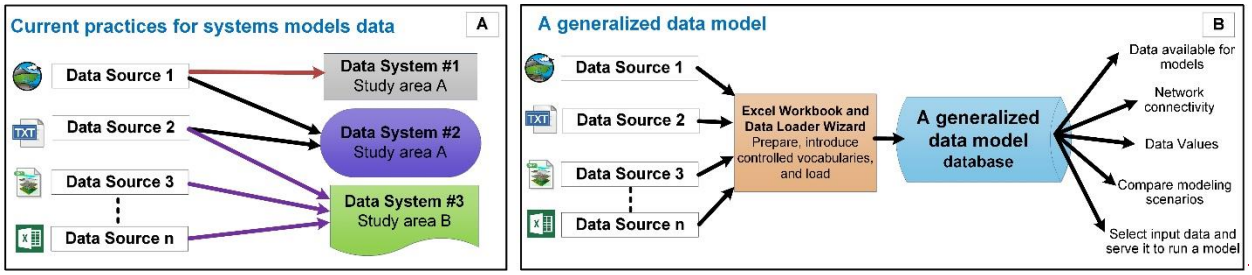
Next, we describe the motivation and design requirements for the WaMDaM system. Section 3 presents the WaMDaM data model design and physical implementations. Section 4 introduces companion software tools. In Section 5, we use WaMDaM to join 13 overlapping local, regional, and national models and datasets. ~~Five use cases~~We demonstrate the utility of the data model in five use cases. The use cases help modelers to identify, compare, and select water supply and demand data, connectivity between engineered infrastructure and natural systems components, ~~and~~ model scenario data, and serve selected data to a WEAP model for the Bear River ~~basin~~Watershed of Utah. Section 6 discusses how modelers can use WaMDaM, limitations ~~and~~, future work, and an invitation to use and improve the design. Section 7 concludes.

2. Design Motivation

~~Modelers~~WaMDaM focuses on the essential steps to organize, join, compare, analyze, and serve multiple datasets to build a water resources model. Because modelers often use multiple systems to gather, organize, store, join, and query the water management data they need to build models (***Figure 1-A***~~). This~~), they repeat that effort ~~is repeated~~ for each new model, data set, scenario, system component, and element. Modelers would benefit from a general approach that only requires doing the work once but allows others to re-use their effort in their other endeavors (***Figure 1-B***). ~~We identified four~~Five use cases ~~that a modeler would follow to join, query, compare, select, and prepare data to build a~~guide the WaMDaM design by answering key water ~~resources systems model. A fifth~~management data questions. These use case ~~serves the data to a model. The use cases helped guide~~ questions sidestep less important aspects that may overcomplicate the ~~WaMDaM~~ design (Szalay and Blakeley, 2009). The use ~~cases answer the following~~case questions are:

1. What data entered by others can be used to develop a model in a study area?
2. ~~What~~Which network connectivity ~~to use~~should be used in a model?
3. How do data values differ across datasets and which values ~~to choose~~should be chosen for a model?
4. How do scenarios differ and which scenarios ~~to use~~should be chosen in a model?

~~Together, these use cases support a fifth use case to help a water systems modeler select appropriate data, serve data to a model, run the model, and quantify model sensitivity to changes in input data.~~



5. How do the input data developed in earlier use cases affect model outputs?

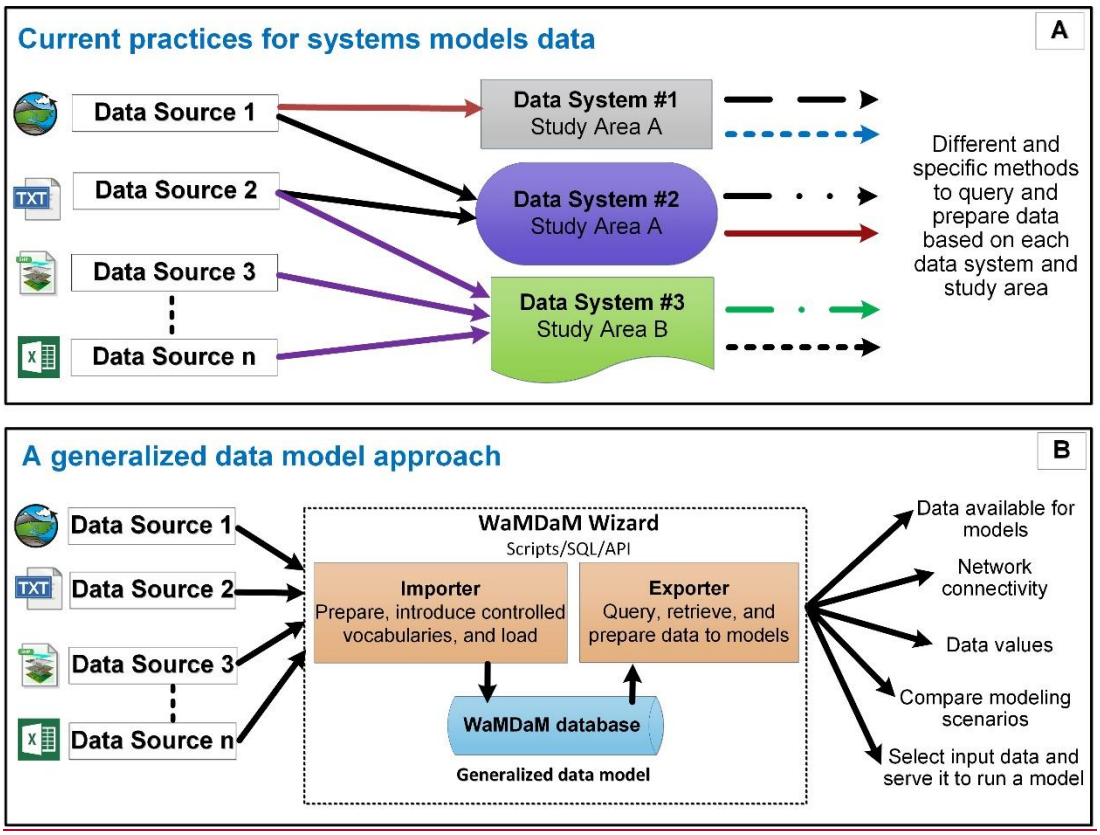


Figure 1: (A) Current data practices use different systems and data manipulation methods for each data source and study area while (B) a generalized data model integrates across the structure and syntax of data sources. The WaMDaM Wizard with scripts, SQL, and ~~allows~~APIs allow modelers to undertake multiple efforts, such as load data, identify data for models, compare networks, data values, and scenarios, and serve data to models.

2.1 Synthesis of design requirements

We synthesized eight design requirements for an integrative data system from 40 prior data management approaches (**Appendix A**, ~~***Table 1***).~~**Table 1**). Below, we define each design requirement and then discuss how the requirement improves over prior approaches.

The first requirement for a modular and extensible design will allow inclusion of multiple model types and their system components (e.g., reservoirs, demand sites, canals) as reusable data objects (i.e., as classes or modules) with properties or attributes (Connolly and Begg, 2010; Knox et al., 2014; Wurbs, 2012; Zagona et al., 2001). Attributes may apply to all network components globally or to individual components. For example, a time series of inflow applies to one reservoir component, while a budget parameter applies to a network. To improve storage efficiency and enable consistent reuse of data, the ~~system~~design must be able to share the same value of an attribute across many ~~systems and~~water resources system components.

~~As part of the modular~~Modular and extensible design~~, the data system must preserve the native terms and descriptions of objects and attributes (terms used in the original datasets and models). For example, a WEAP model uses the term "Reservoir" and RiverWare uses "Storage Reservoir." Supporting native terms allows modelers to continue use of their terms while simultaneously accessing and linking to other data stored with in and features in the WaMDaM system. Controlled vocabulary, another design requirement, will help reconcile heterogeneity in native terms across datasets and modelers. Supporting native vocabulary allows the user to create new object types and attributes for new modeling applications. Although modularity~~ is supported in most existing data systems and water management models~~, it is often limited to a pre-defined set of supported object types. For example both HydraPlatform~~ such as Hydra Platform and the ODM ~~have modular and extensible designs~~ (Harou et al., 2010; Knox et al., 2014)~~ while other~~. Other systems, such as ArcHydro ~~allows~~and WEAP (Maidment, 2002; Yates et al., 2005) allow adding new data objects ~~and attributes. But~~(as in ArcHydro), but users are still forced to ~~work with~~use core components ~~including stream networks, monitoring points, watersheds, and wells~~ and attributes that might not be needed for a case study ~~(Maidment, 2002).~~ .

The second requirement is to represent the spatial configuration of system components as networks of nodes (junctions or points) and links between nodes (arcs, connections, curves, lines, or edges of a directed graph) (HydroLogics, 2009; Rossman, 2000; Zeiler, 1999). Networks help modelers organize and search for system components that are related in purpose (e.g., flow of water through connected pipes), use (e.g., drinking water supply), or in a spatial boundary (e.g., Bear River Watershed) (Loucks et al., 2005). Networks also represent connectivity which is a key principle of water mass-balance fundamental to most systems

models. Although most existing data systems support networks, each system uses ~~a~~ different data organization ~~methods~~method and terms to manage the connectivity of nodes and links. ~~The~~Such different structures require different methods to query network data. While the ODM (Horsburgh et al., 2008) stores time series data for individual nodes or links, ODM cannot describe how the nodes relate to each other (upstream, downstream, etc.). A consistent method to represent networks will allow users to consistently retrieve information about how nodes are connected to each other through links.

Third, the data system must describe and store scenarios that represent changes to the physical, operational, infrastructure, and socio-economic model input data. Scenarios allow modelers to test and run current and proposed water management alternatives. The scenario requirement also includes the ability to track and manage versions of changes from a baseline network. A scenario can be created by one or two potential changes to a water system network: i) change network topology like to add or remove an infrastructure component and ii) change data for one or more attributes of a component such as to expand the capacity of a reservoir or update metadata such as the method or data source. Many existing systems (e.g., WEAP) use scenarios to track changes in input data but cannot track changes in the network components.

Fourth, the data system must allow users to add contextual metadata~~–~~; the additional information to help modelers interpret data. Metadata also helps modelers maintain the data provenance ~~that is~~ needed to track the history and context of sources, methods, people, and organizations that contributed to create the data (Campbell et al., 2013; Carata et al., 2014; DCMI, 2013; Goodman et al., 2014; Gray et al., 2005; Horsburgh et al., 2008; Pokorný, 2006). Some existing systems store metadata in one table that accepts user-specified key-value metadata pairs (e.g. (Knox et al., 2014; Refsgaard et al., 2005). HEC-DSS manages and retrieves large sequential datasets ~~like~~, such as time series and paired tabular data. Support to describe each time series is limited to six metadata parameters that include the variable name, location, and time step. Each parameter must be described in ~~under~~less than 80 characters (HEC, 2009). The ODM uses contextual metadata to describe units, sources, and methods for collecting observational data variables at a site. This requirement mandates ~~an~~ explicit support for the following fundamental metadata elements~~,~~ the unit, source, method, people, and their organization that contributed to creating data. The support to explicit metadata elements guides users to populate, reuse, and later to directly query them.

Fifth, the data system must be able to store and describe multiple data types that modelers use to represent physical, operational, and descriptive attributes of system components: time series, multi-attribute series (e.g., multi-variable for a reservoir bathymetry),

numeric, categorical values (e.g., gate open or closed), and seasonal parameters (e.g., values that are ~~fixes~~the same for months across the years). Many existing systems support multiple data types, but store them as binary data objects which limits users' ability to access stored data outside the software system (Harou et al., 2010; Knox et al., 2014). Supporting multiple data types allows modelers to store, access, and reuse different types of data for properties of water systems components.

Sixth, the data system must support controlled vocabularies (CVs) as sets of terms with definitions for ~~the~~ object ~~type~~types, attributes, and names of nodes and links. CVs allow ~~and encourage~~ modelers to retain the native terms they are familiar with but simultaneously relate ~~their~~ native terms to consistent names that can be reused across datasets and models (Laniak et al., 2013). ~~Thus multiple native terms in~~ For example, the ~~many integrated datasets and models (e.g., storage reservoir, Reservoir Node, reservoir) can be~~ following native terms are related to a single CV term (e.g., ~~Reservoir).~~Reservoir): reservoir (WEAP), storage reservoir (RiverWare), Reservoir Node (Bear River Systems Dynamic Model), reservoir (US Bureau of Reclamation). The CV term then ~~works to link~~links all the fundamentally similar native terms together. ~~For example~~Thus, a query for "Reservoir" ~~will return~~returns all ~~three~~ related native terms. ~~This requirement allows modelers to simultaneously maintain their native terms.~~

Seventh, the data system must support ~~conditional data queries so modelers can load and retrieve subsets of data based on selected water system components, attributes, metadata, networks, scenarios, and data types in space and time without the need for third-party software. Many data systems have pre-defined queries or built-in functionality which prevents users to freely query and compare subsets of systems components, data and metadata. Support for conditional queries allows modelers to synthesize and compare subsets of water management data and use only retrieved data for their working model~~direct access to subsets of data and metadata that enable search and filtering based on a schema. In contrast, unstructured data storage known as the Binary Large OBject (BLOB) formats (Sears et al., 2006) do not allow direct access to subsets of stored values but rather to the entire block of data. Although storing BLOB data such as blocks of time series or arrays as in Hydra Platform and HEC-DSS (HEC, 2009) can be efficient and fast, users must use custom functions to decode and access subsets of the content. In a structured data storage, modelers can load and retrieve subsets of data based on selected water system components, attributes, metadata, networks, scenarios, and data types in space and time without being limited to a custom method.

The eighth requirement is to develop the WaMDaM implementations using free and open-source software tools, to allow access via an open-~~-~~source~~-~~ code repository, promote

reproducibility, and help others further advance the method (Easterbrook, 2014; Gil et al., 2016; Goodman et al., 2014)~~. Many existing data systems like WEAP, RiverWare, and HEC-DSS are proprietary and require specific tools to access their data. The source-code is not available and there is limited documentation of source code. Those proprietary approaches contrast with other customized systems models that use a mix of spreadsheets, text files, and the General Algebraic Modeling System (GAMS) file formats to organize their data and metadata. Examples of customized models include the Watershed Area of Suitable Habitat (WASH) model that allocates water to maximize watershed habitat areas (Alafifi and Rosenberg, In review) and the Bear River Systems Dynamic Model (BRSDM) (Sehlke and Jacobson, 2005) that simulates priority-based water allocation. Subsequent use-cases will feature WEAP, WASH, and BRSDM model instances in the Bear River study area.~~

. At the same time, we recognize that open-source software require documentation to be reusable. Many existing data systems like WEAP, RiverWare, and HEC-DSS are proprietary and require specific tools to access their data. Those proprietary approaches contrast with other customized systems models that use a mix of spreadsheets, text files, and the General Algebraic Modeling System (GAMS) file formats to organize their data and metadata.

2.2 Support for Design Features

To date, existing water resources systems software tools incompletely support the eight requirements (***Table 1***). Thus, we designed WaMDaM to support all eight requirements. The next section describes how WaMDaM is designed and implemented to support the eight requirements, answer four use case questions, and complete a fifth use case that serves data to a model~~-~~.

Table 1: Support for the identified requirements by select data systems and water resources models. An "X" indicates that the system supports the requirement.

	Select Data System / Model					
Data ~~management requirement~~Management Requirement	ODM	~~HydraPlatform~~Hydra Platform	HEC-DSS	ArcHydro	RiverWare	WEAP
Modular and extensible design	X	X				
Supports networks of nodes & links		X		X	X	X
Supports scenarios & version control		X	X		X	X
Reusable contextual metadata	X					
Multiple data types for system models		X	X		X	X
Extensible controlled vocabularies	X					
~~Conditional query to~~ Direct access ~~all~~to subsets of data	X			X		
Open-source environment & license	X	X				

3. WaMDaM Design

We used the eight requirements described in Section 2 to design the WaMDaM data model and its physical implementations to organize, manage, join, query, and compare water resources datasets and models. We aimed for a parsimonious design that minimizes the number of data and metadata entities needed to satisfy the eight requirements and answer the use case questions (Hey et al., 2009)~~. Further descriptions of design decisions, full data dictionary, including definitions of each entity and attribute as well as data type descriptions, can be accessed via HTML-based documentation on the WaMDaM GitHub site (https://github.com/WamdamProject/WaMDaM_Information_Model). We iteratively revised the data model design in five key versions over the course of five years to satisfy the requirements, use cases, and feedback from collaborators. The criteria for a successful design was a design that satisfies the~~. The criteria for a successful design was a design that satisfies the eight requirements and answers the use case questions. Below we present the conceptual design, then show the logical design using an Entity Relationship Modeling (ERM) diagram. Afterwards, we describe physical implementations.

3.1 WaMDaM Conceptual Design

The WaMDaM conceptual design has multiple hierarchal one-to-many relationships; color-coded grouped entities represent key design requirements (**Figure 2**). In general, the color-coded groups define the steps a modeler would follow to populate a physical implementation of the design with data.

The first group of blue entities supports a modular and extensible design by allowing the modeler to define the resource type (e.g., a WEAP model), one or many object types (e.g.,

reservoir, river reach, diversion, etc.) for each resource type, and one or many attributes (e.g., storage or diversion capacity, head flow, etc.) for each object type (Requirement #1). ~~A resource type represents the types of input or output data used in a data source such as a "Model Program" as defined in Morsy et al. (2017)~~A resource type represents the types of data (input or output) used in a data provider such as a "Model Program" as defined in Morsy et al. (2017), independent of implementation. For example, a WEAP model resource type has 21 object types (e.g., reservoir, demand site, transmission link, etc.) and each object type has many attributes (e.g., "Storage Capacity", "Net Evaporation"). The resource type entity can also be used for datasets. For example, the U.S. Major Dams Inventory shapefile has a list of 18 attributes that have values for the "Dam" object type. An object type is a system component with typologies such as node or link (e.g., reservoir, canal, water source, or demand site) and can have one or more quantitative or qualitative properties or attributes with units.

The second group of green entities supports networks and scenarios by allowing modelers to define a master network with many scenarios where each scenario can have one or many instances that are either node or links (Requirements #2 and #3). To specify connectivity among instances, links must have start and end nodes.

The third group of orange entities ~~allows~~allow modelers to use reusable, contextual metadata where a modeler affiliates people to an organization and specifies methods and sources that generate data (Requirement #4). The fourth group of red entities ~~allows~~allow modelers to store seven distinct types of data values such as time series or categorical data (Requirement #5). Within a scenario, an attribute for an instance has a source, method, and data type. The fifth group of controlled vocabulary (purple entities) allows modelers to relate native terms for object types, attributes, and instances (Requirement #6).

We satisfied ~~conditional~~direct access to all data ~~queries~~and metadata (Requirement #7) by using relational database theory (also referred to as the Relational Model) to implement the data model entities as interrelated tables (Chen, 1976; Codd, 1970) as further described in Section 3.2. We developed a physical implementation of the data model and software tools in an open-source physical database system (Requirement #8; see Section 3.3). Next, we explain how and why the relationships are implemented to form the WaMDaM Logical Data Model.

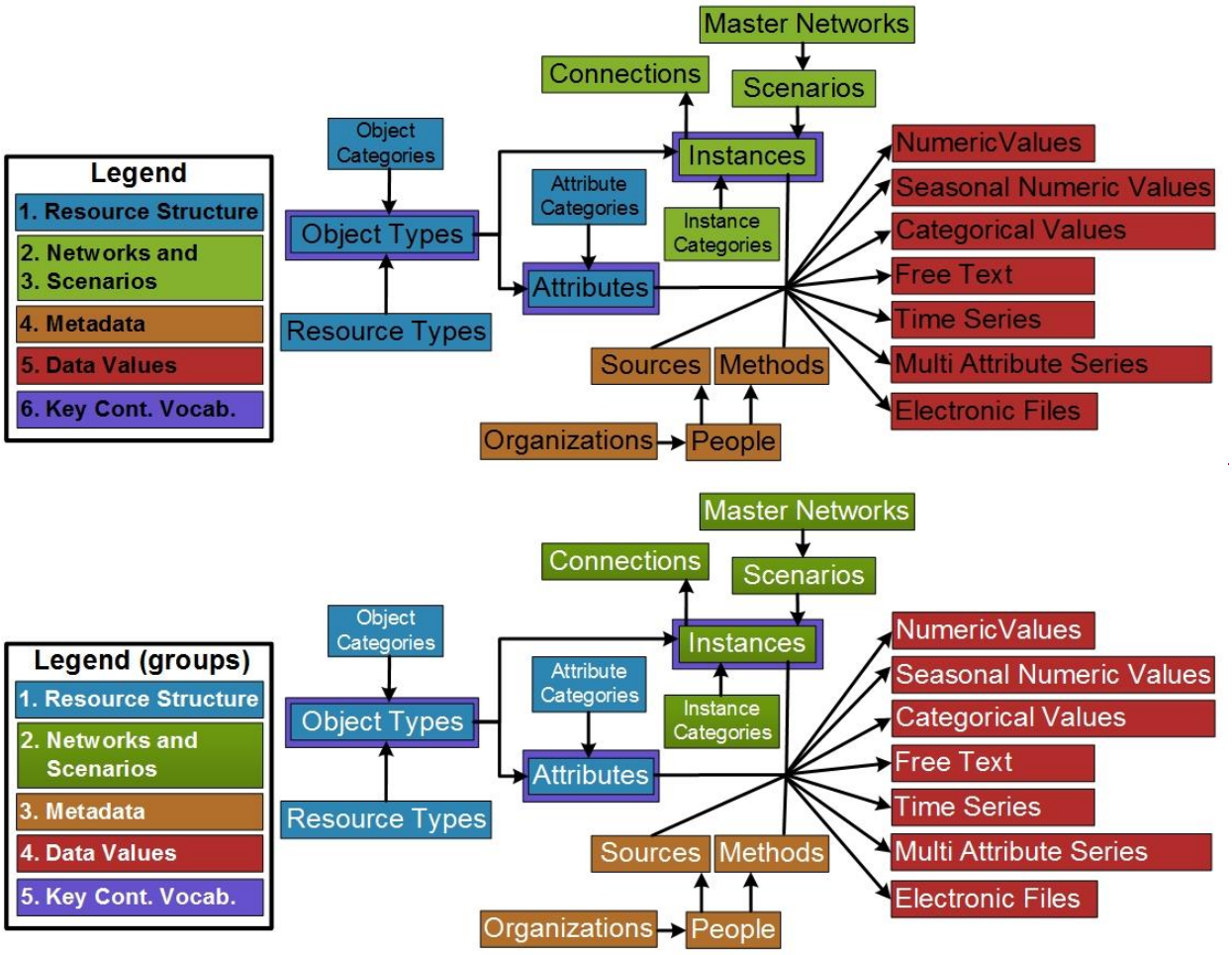


Figure 2: The conceptual diagram relating the first six design requirements for the water management data model. Key controlled vocabularies are introduced to the boxes outlined in purple.

3.2 WaMDaM Logical Data Model

The Logical Data Model schema shows the one-to-one, one-to-many, and many-to-one relationships among database entities ~~(Appendix A, Figure A4).~~(**Figure 3**). Blue, green, orange, red, and purple colors again indicate tables associated with the resource type, networks and scenarios, metadata, data values, and controlled vocabulary design requirements. A WaMDaM data value is described by fourteen required elements (**Appendix A, Table A2**). Here we describe six key ~~features~~requirements that are needed to interconnect schema components and specify the fourteen required elements and design requirements. We pluralize data model entities and list them in italics and capital letters.

First, *ResourceTypes* are datasets (like the U.S. Major Dams Inventory) or models (like WEAP) and have one or more system components called *ObjectTypes* (such as a reservoir, canal, water source, or demand site). *ObjectTypes* have typologies such as node or link and one or more quantitative or qualitative properties called *Attributes* (such as storage capacity, net evaporation, or delivery target). Here we use the broad term attribute, as a contextual property which also may include variables that are measured and might change with time (Sarle, 1995). Attributes could also describe model outputs. Each attribute has a unit, attribute data type, and ~~optionally~~by choice whether it is used as "Input" or "Output" in a water resources model.

Second, an object type such as a "Reservoir" can be specified (i.e., implemented) for zero or more locations as *Instances* (e.g., Hyrum Reservoir, Bear Lake, and Flaming Gorge Reservoir would be three separate reservoir instances). An instance inherits the *Attributes* of its object type and may be geo-referenced as a node in space with longitude and latitude coordinates. Instances can also be a link which has start and end nodes. The *Connections* entity specifies a start and end node for links and avoids a circular reference problem when connecting the *ObjectTypes* table directly to both the *Instances, Attributes*, and *ValuesMapper* tables. A circular reference in a database is problematic to database integrity as it may allow multiple transaction paths to insert or delete data. In the data systems ~~we reviewed,~~ modelers may represent the same water system component, such as reservoir, as a node or a link in a model. Thus, storing nodes and links in the *Instances* table and link connectivity info in the *Connections* table enables modelers to use the same query to access data for nodes or links and improves over prior approaches that require many different queries to access data for node or links (Abdallah and Rosenberg, 2014; Knox et al., 2014; Yates et al., 2005).

Third, one or more node and link *Instances* can be connected into *MasterNetworks* (e.g., water supply/demand, water distribution, or other network for a study area). Each master network contains one or many *Scenarios* in a study area (such as a base case, reduced inflow, or ~~with~~ new infrastructure). *Scenarios* within the same master network may share the same exact network topology or versions of the network and its data. ~~-~~Each scenario also has a start and end date and time step to track the modeling time step and its extent.

Fourth, the *Mappings* bridge entity relates *Instances* to their *ObjectTypes*, *Attributes*, metadata *Sources* and *Methods*, *Scenarios*, and data values. This bridge entity is the central table in the WaMDaM database. This *Mappings* entity is needed because *ObjectTypes* can have i) many *Attributes* (e.g., reservoir object type can have evaporation depth, storage capacity, and volume-area attributes), ii) each *Instance* (e.g., Hyrum Reservoir, Bear Lake,

~~and~~or Flaming Gorge Reservoir) can have shared or instance-specific attribute values, and iii) *Instances* can also have shared or instance-specific *Sources* and *Methods* metadata values.

Fifth, data values are assigned to one of seven supported data types and connected through the *ValuesMapper* entity to the *Mappings* bridge entity. The seven supported data types (numeric, seasonal, categorical, free text, time series, multi-attribute series, electronic file) are commonly used in the models we reviewed (***Appendix A, Table A3***~~,).~~). Similar to prior time-series data models like ODM and ODM2, the *TimeSeries* entity (e.g. flow versus time) captures key global metadata for the entire time series and can have one or many values, time stamps, aggregation statistics (e.g., average, cumulative, etc.), and year types to indicate water year or calendar year. The *MultiAttributeSeries* entity organizes paired data (e.g., area-elevation curve) by referencing multiple *Attributes*. Each paired attribute has one or many values and sequential order to preserve the order and pairing of values across many attributes within the same array. Additional attribute data types can be added and connected to the *ValuesMapper* entity without affecting any of the existing data model relations. The *ValuesMapper* entity helps to reuse and share attribute data across many *Instances* (Requirement #5). This WaMDaM approach of storing values once and sharing them is more efficient and allows the option to register the term one time with a controlled vocabulary.

Sixth, the *ScenarioMappings* bridge entity further allows modelers to share similar *Instance*s, their *Attributes*, metadata, and values across *Scenarios* with no duplication. The WaMDaM ~~Data Loader~~ Wizard, presented later in Section 4, also uses the *ScenarioMappings* bridge entity to query and compare how combinations of *Instances*, their *Attributes*, and data tables change between two *Scenarios* within the same master network.

Seventh, *People, Organizations, Sources*, and *Methods* support four essential key metadata entities needed to interpret *Instances* and values. The *Sources* entity describes the origin or encompassing package of data such as a shapefile, web service, or a model for a study area which may have a citation and a webpage. The *Methods* entity describes how values were created, an instance is defined, ~~or~~data quality, and the resource type works (e.g., simulation or optimization method for a model program). Modelers may document uncertainty in the data and indicate the quality of data within the method that generated it. Each source or method is associated with a person (author) who set up the source or created the method. Each person belongs to an organization. If no person is associated with data, modelers can define a person as “unknown” and relate to the organization that created the source or method. We recognize that there is potential for a more complex and specific representation of metadata. We attempted to balance between the principles and practicality of metadata usage as

recommended by Duval et al. (2002). Complex metadata requirements may discourage modelers to provide metadata while too little metadata might be insufficient to correctly interpret data. Modelers ~~can use~~are required to provide the native unit name for each attribute~~,~~ and ~~they also can~~ are encouraged to relate ~~native terms to~~ the unit with a list of controlled ~~vocabulary terms which can be used~~units. Using controlled unit vocabularies allows the user to convert values into other units.

Eighth, controlled vocabularies have ~~a~~ the following common fields of term, name, ~~term~~category, definition, ~~category,~~ and URL to a source. This approach is similar to the CVs defined for ODM2 (Horsburgh et al., 2016)~~. The CVs attach to *ObjectTypes*, *Attributes*, *Instances*, and *Units* of measurement and allow modelers to relate native terms across *ResourceTypes*.~~

. The key CVs attach to *Object Types*, *Attributes*, and *Instances* to relate native terms and values across *Resource Types*. Each resource type (e.g., model) has its own native terms. Data of different models can be related using three controlled terms, object type (e.g., Reservoir), attribute name (e.g., Volume), and instance name (e.g., Hyrum) (**Figure 4**). Units can be converted using constant or linear multipliers. For example, a value of 1.000 litter has a 0.001 constant fraction in reference to a 1.0 cubic meter volume unit. We adopted the list of controlled units from Hydra Platform (Knox, 2018).

Finally, software business rules (i.e., external code) are used to correctly enforce some of the complex relationships in the data model especially when loading data into the database. For example, software business rules relate an object type and its typology with *Instances* through a dummy attribute and ensure that each link in the *Connections* entity has a start and end node. Another rule relates a resource type with master network through the "NetworkAttributes" object type, the dummy attribute, and a dummy instance to allow modelers to query all the network implementations of a resource type. Correctly representing the many-to-many relationships among the entities within the first six design requirements while attempting to achieve parsimony and relatively simple querying consumed a significant portion of the iterative WaMDaM designs. ~~We summarize the software business rules on GitHub at https://github.com/WamdamProject/WaMDaM_Wizard/blob/master/software_business_rules.md~~ We summarize the software business rules on GitHub (Abdallah, 2018d)

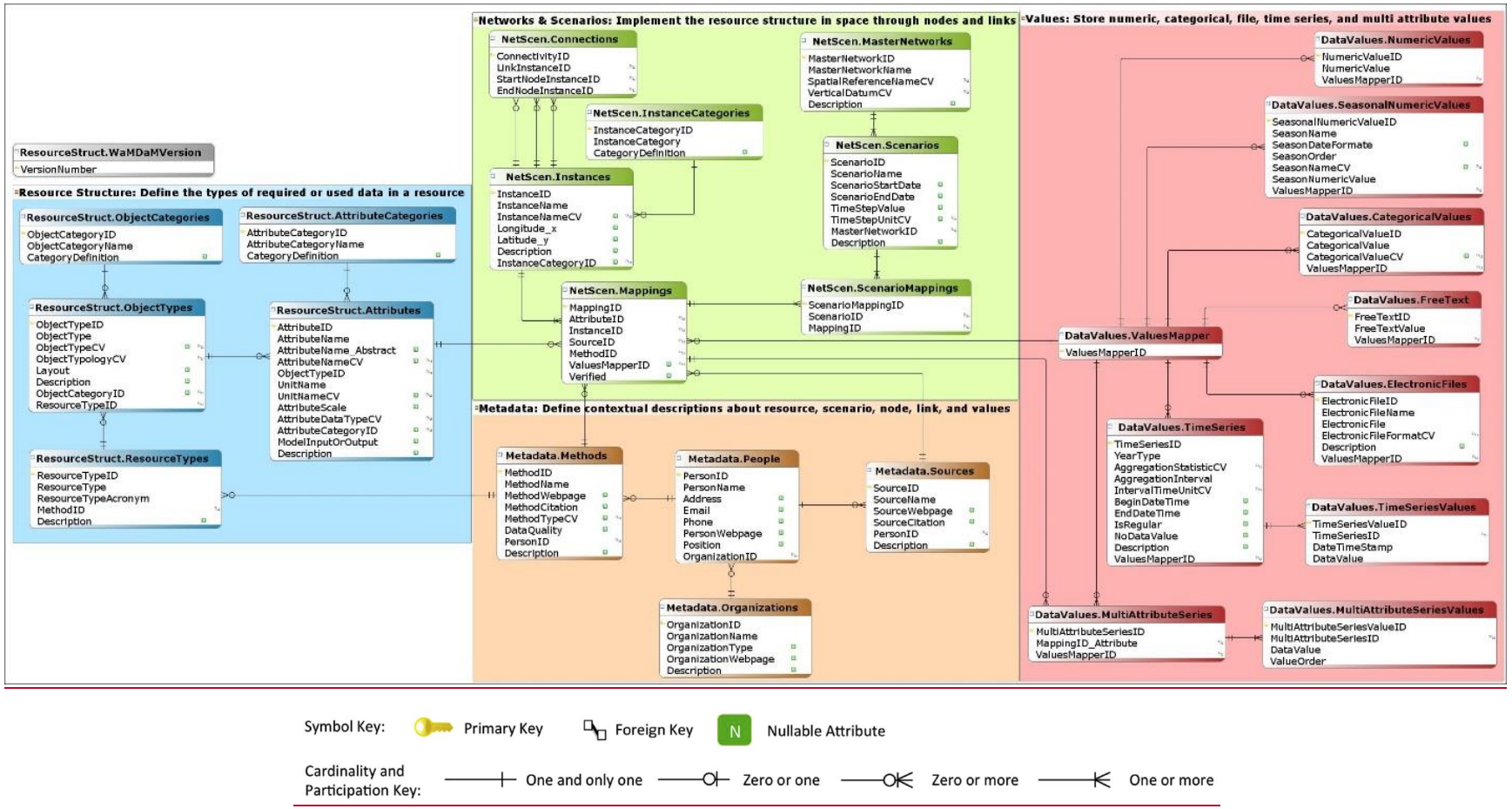


Figure 3: WaMDaM logical model tables grouped into the design requirements. Resource Type (#1), Networks and Scenarios (#2&3), Metadata (#4), and Data Values (#5). The diagram uses the crow's foot notation for relationship cardinality and participation. An interactive html copy is available at http://schema.wamdam.org/diagrams/01_WaMDaM.html (Abdallah, 2018c). Controlled vocabularies tables (#6) are not shown here for simplicity and can be viewed at http://schema.wamdam.org/diagrams/03_CVs.html. Each column name (field) that ends with "CV" indicates that the term is a controlled vocabulary.

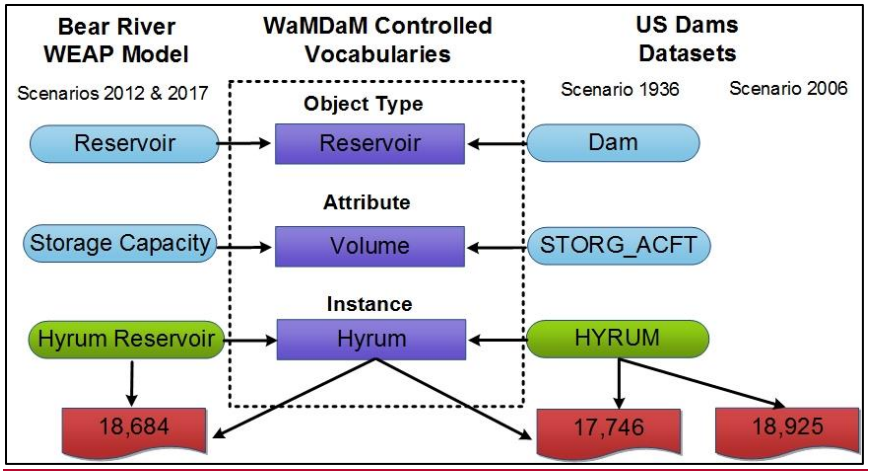


Figure 4: Relating native names with controlled vocabularies for object types, attributes, and instance names allows modelers to query and simultaneously access values across native terms. Identical storage are shared among scenarios of the Bear River WEAP Model while different storage values in the US Dams Datasets are stored separately.

3.3 Physical Model Implementation

We implemented the logical data model schema within four physical Relational Database Management Systems (RDBMS), including PostgreSQL, MySQL, Microsoft SQL Server, and SQLite to demonstrate that WaMDaM is independent of the RDBMS~~.~~ (Abdallah, 2018c).

First, we selected a physical data type for each field in each logical model entity (e.g., integer, varchar) and we imposed physical constraints on each field (e.g., value cannot be null) by following the physical data types convention in the ODM2 (Horsburgh et al., 2016). Second, we adapted an existing Python 2.7 script developed by Horsburgh et al. (2016) to forward engineer a Data Definition Language (DDL) script containing a set of "create" statements for WaMDaM tables for each of the four RDBMS. Finally, we executed the DDL script within each RDBMS to create a physical blank WaMDaM database that modelers can load with data.

We chose to express the logical data model as a relational model to: i) support ~~conditional~~direct access to all data ~~queries~~and metadata (Requirement #7), ii) be platform independent and implement as open-source on different operating systems for different relational database systems (Requirement #8), iii) support a standardized and stable Structured Query Language (SQL), and iv) follow common use and familiarity with the RDBMS within the water resources community (Harou et al., 2010; Horsburgh et al., 2016; Horsburgh et al., 2008; Knox, 2014).

The core contribution of WaMDaM is the description of a generalized design to help organize, join, compare, and analyze multiple water resources datasets and models. Our implementation in a relational database is just one way to solve the problem. Other methods, such as non-relational databases, also known as NoSQL, are increasingly used worldwide (Hoberman, 2014) and could likely satisfy the same use cases. NoSQL implementations may scale and adapt without being limited to a schema. Future work should test WaMDaM's ability to scale and adapt to much bigger and more diverse datasets and models.

3.4 Community Feedback on the Design

We iteratively revised this data model design in five key versions over the course of five years to satisfy the design requirements and use cases. The changes were in response to feedback from collaborators at the University of Manchester, University of California, Davis, and University of Massachusetts, Amherst on WaMDaM design and tools. We acknowledge the need for larger and more diverse community testing and feedback to serve a wider audience of users. We also incorporated feedback on an earlier design and its description (Abdallah and Rosenberg, 2014). The five key designs are available on GitHub (Abdallah, 2018c)

4. WaMDaM Related Software

We created ~~several~~ software tools to demonstrate WaMDaM's functionality and allow users to more easily interact with ~~the WaMDaM~~its database.

4.1 WaMDaM ~~Data Loader~~ Wizard

We developed a WaMDaM ~~SQLite Data Loader~~ Wizard (hereafter the Wizard) in Python 2.7 for SQLite as a simplified demonstration to auto-read input data from an Excel Workbook template into a physical WaMDaM database implementation on the user's local machine~~.~~ (Abdallah, 2018d). The WaMDaM Wizard uses SQL Alchemy to load data into the database and we use direct SQL script to query the database through a Python SQLite3 library. The Wizard provides key functionalities of the design and it is just one of many possible ways to import or export data of the database. We chose Microsoft Excel as a generic input data medium because modelers commonly use it. The Wizard validates entries to comply with the database schema, maps primary and foreign keys, and implements software business rules.

We elected to use SQLite (https://www.sqlite.org/index.html) because it is free, open-source, and server-less to satisfy open-source design (Requirement #8). We also used the DB

Browser for SQLite (https://sqlitebrowser.org/) as an open-source user interface to view tables and execute queries against WaMDaM database tables.

The Wizard has ~~three data utilities~~tools to ~~help modelers~~i) prepare and ~~transform~~pivot a shapefile, time series~~ data,~~, or seasonal data~~, and shapefiles for input to WaMDaM. The Wizard also imports data directly~~ into the data structure of the workbook template, ii) import time series stream flow data from WaterOneFlow CUAHSI web-services~~ and~~, iii) import time-series WaterML files ~~available~~for reservoir inflow, release, storage, elevation from the U.S. Bureau of Reclamation (USBOR) Water Information System web service ~~(https://water.usbr.gov/). The Wizard can also extract any WEAP model's~~(https://water.usbr.gov/), iv) import network and ~~all its~~ data stored in WEAP using ~~WEAP's~~its Application Programming Interface (API) into the workbook template, v) use the provided controlled vocabularies in the workbook to register and relate native terms across sources as discussed in Section 4.2 , vi) adapt and use the example Jupyter Notebooks to execute data query, plots, and ~~then import the~~ analysis across data ~~into a WaMDaM database. The Wizard also provides modelers with automated detailed~~sources, and ~~summary comparisons of network~~serve data into the model, and vii) compare and verify differences in topology~~, metadata, and data between *Scenarios* in the same master network. See instructions to use WaMDaM software at the WaMDaM_JupyterNotebooks GitHub repository.~~ or input data values across modeling scenarios.

4.2 Controlled vocabulary registry

We deployed an online-hosted CVs system to physically implement the CVs design (Requirement # 6), allow multiple modelers to access, reuse, or suggest new consistent vocabularies across WaMDaM database instances and machines. We adapted the existing online CV registry system which is a Python/Django web application API developed by the ODM2 design team (Horsburgh et al., 2016; Horsburgh et al., 2014)~~ to manage WaMDaM CVs (http://vocabulary.wamdam.org).~~ to manage WaMDaM CVs (Abdallah, 2018b) (http://vocabulary.wamdam.org). ~~To get all the native terms registered to a controlled term, modelers can write a simple query against their local WaMDaM database.~~

Because we adopted the CVs moderation system developed by the ODM2 team, modelers have the option to use WaMDaM CVs, submit suggestions to add new terms within the online registry, or use their own native terms without registering them to WaMDaM controlled vocabulary. We populated the CVs system with example WaMDaM CVs for the datasets we worked with and introduce in the next Section. Modelers can use the CVs system seamlessly in an Excel Workbook template and the WaMDaM Wizard. Within the Excel

Workbook template, there is Visual Basic script button that downloads and updates look-up menus for all CVs. Excel sheets in the Workbook template contain a column for the native term and another as a ~~look-up~~ controlled look-up term that register or relates them together. To get all the native terms registered to a controlled term, modelers can write a simple query against their local WaMDaM database.

5. Results

We present five use cases within the Bear River Watershed that help modelers ~~1~~: i) search previously-entered datasets in a WaMDaM database for input data to expand a model to a larger study area, ~~2~~ii) show the spatial configuration and network connectivity of natural and engineered system components, ~~3~~iii) compare retrieved data to help the user decide which data to use, and ~~4~~iv) compare changes in network topology, metadata, and data values among scenarios. These ~~four~~ use cases also support a final common case to ~~5~~v) serve selected data to run a WEAP model ~~and run the model.~~. These five use cases ~~update~~support common operations that water resources systems analysts and modelers perform to develop and ~~expand existing WEAP and WASH~~use models ~~in~~.

The use cases apply one optimization and two priority-based simulation models for the Bear River study area: 1) the Watershed Area of Suitable Habitat (WASH) model that allocates water to maximize watershed habitat areas (Alafifi and Rosenberg, in review), 2) the Bear River Systems Dynamic Model (BRSDM) (Sehlke and Jacobson, 2005), and 3) WEAP model. These use cases expand coverage for the Lower Bear River to more of the ~~entire watershed~~Watershed in Utah, Idaho, and Wyoming (light red to darker red in **Figure 5**).

The use cases assume ~~prior work~~a modeler used WaMDaM CVs, Excel templates, and WaMDaM Data Wizard to load 13 diverse and overlapping U.S. national, regional, and local data sources and models (**Table 2**) into a WaMDaM SQLite database. ~~-~~The database file is 35 Megabytes with 73 *ObjectTypes*, 563 *Attributes*, 15,464 *Instances*, and 214,352 ~~records in the central *Mappings* table. Readers can use instructions and Python 2.7 scripts in Jupyter Notebooks~~rows in the central *Mappings* table. Readers can use the instructions and Python 2.7 scripts in Jupyter Notebooks (Abdallah, 2018a) to load data into the database and replicate queries and figures as well.

Table 2: Data sources used in WaMDaM use cases

#	Data ~~source name~~Source	Instances ~~(number)~~(#)	File ~~format~~Format
1	Water Data Exchange (WaDE) Program of the Western States Water Council ~~http://www.westernstateswater.org/wade~~http://wade.westernstateswater.org/	2	Excel, (Web-service for time series is in progress)
2	WaterOneFlow Web Services (CUAHSI) http://his.cuahsi.org/wofws.html	1	Web-service, WaterML
3	U.S. Bureau of Reclamation Water Information system web service https://water.usbr.gov	2	Web-service, WaterML
4	~~US Hydropower Dataset (N.M. Samu, 2017)~~US Hydropower Dataset (Samu et al., 2017)	2,398	Excel (.xlsx), Shapefile
5	US Major Dams Dataset ~~(U.S. Geological Survey, 2013)~~(U.S. Geological Survey, 2013)	8,121	Shapefile, text files, HTML
6	Bear River Commission Flows (Personal Communications, 2016)	1	Excel (.xlsx, .xls), Quattro Pro (.QPW)
7	Utah Dams Dataset (Craig Miller-Personal Communications, 2016)	910	Shapefile, Excel (.xlsx)
8	Utah Flows Dataset (Craig Miller -Personal Communications, 2016)	893	Shapefile, text file
9	Idaho Flows Dataset (Liz Cresto-Personal Communications, 2016)	164	Shapefile, Excel
10	~~Watershed Area of Suitable Habitat model (WASH) (Alafifi and Rosenberg, In review)~~Watershed Area of Suitable Habitat model (WASH) (Alafifi and Rosenberg, in review)	104	Excel (.xlsx), shapefile
11	Bear River systems Dynamics Model (BRSDM) ~~(Sehlke and Jacobson, 2005)~~(Sehlke and Jacobson, 2005)	237	Excel (.xls)
12	Bear River WEAP Model 2012 for Utah ~~(Rosenberg, 2017)~~(Rosenberg, 2017)	375	CSV, Paradox Database, ~~-~~shapefile
13	Bear River WEAP Model 2017 for Utah and Idaho ~~(Rosenberg, 2017)~~(Rosenberg, 2017)	150	CSV, Paradox Database, ~~-~~shapefile

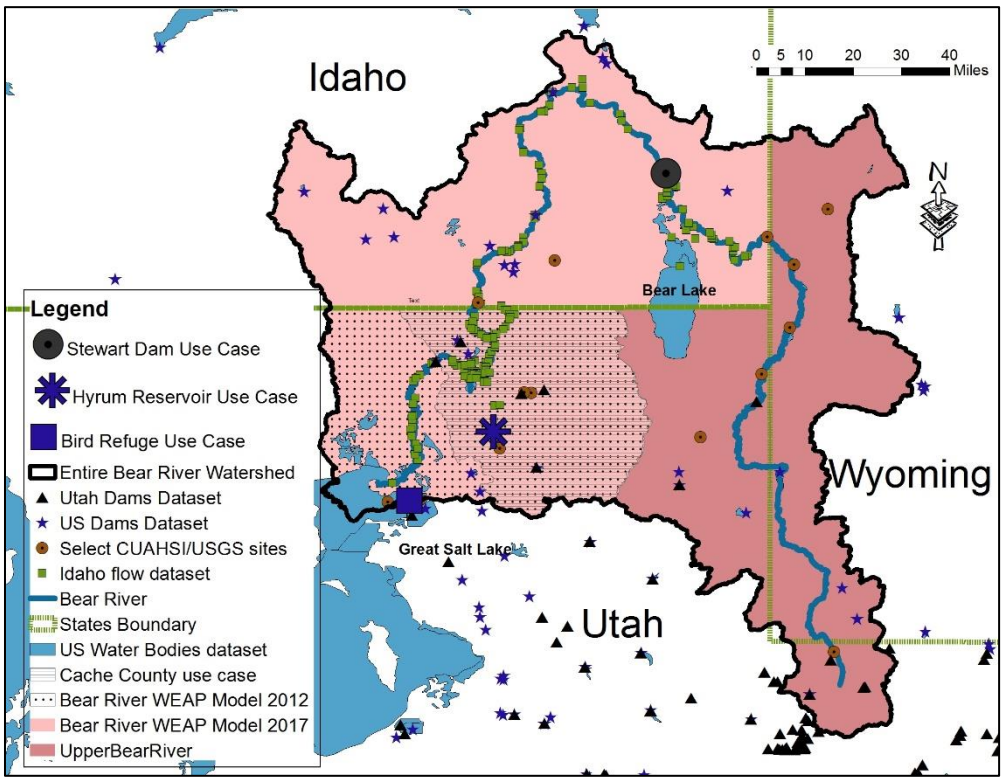


Figure 5: The Bear River Watershed in the western U.S. The dotted area shows the spatial domain of existing WEAP 2012 and WASH models for the ~~lower~~Lower Bear River ~~basin~~Watershed. Lighter red is area for the WEAP 2017 model and dark red is for the ~~upper~~Upper Bear River ~~Basin~~Watershed. Symbols show example available data.

Use Case 1: What data entered by others can be used to develop a WEAP water supply/demand model for the entire Bear River ~~basin~~Watershed?

Using the populated ~~version~~instance of the WaMDaM database file, the user first specifies the resource type to search data ~~for~~ (e.g., for WEAP model) and min and max longitudes and latitudes of the ~~upper~~Upper Bear River ~~basin~~Watershed (dark red in **Figure 5**). Next, ~~run~~the user runs the SQL script to identify the available object types and attributes. WaMDaM uses CVs to match native WEAP terms with terms from the other 13 loaded data sources. The workflow is readily repeated for a second resource type like the WASH model. ~~Excluding~~By excluding categories of water quality and cost attributes that are not used in the WEAP 2017 model, the WEAP model has 21 object types with 71 attributes, while the WASH model has six object types with 61 attributes.

WaMDaM found six data sources can provide data ~~in~~for the ~~upper portion of the~~Upper Bear River ~~basin~~Watershed for five WEAP object types and 15 of their attributes (out of 71 needed attributes; **Table 3**). Here, WaMDaM used the Reservoir CV term to mediate between the 13 datasets to return the local native terms “Dam” from the U.S. Dams Dataset and “Reservoir Node” from the BRSDM model. Similarly, the controlled attribute term Volume returns “STORG_ACFT” in the US Major Dam’s Dataset, “Capacity” in the Utah Dams Dataset, and “Max Storage Capacity” in the BRSDM model for the WEAP attribute “Storage Capacity”. To expand the ~~lower~~Lower Bear WASH Model, WaMDaM finds six data sources can provide data for six attributes for demand site and reservoir object types. Data is still needed for 55 attributes. One reason for this mismatch is that the WASH model uses many ecologic parameters that do not have analogues in the other data sources.

This use case demonstrates that the same WaMDaM data search method can be applied to multiple models. Loading more diverse datasets into WaMDaM, such as water right priority to demand sites that are required by WEAP, would allow WaMDaM to identify more data for models.

Table 3: Summary of the identified attributes and node and link instances in WaMDaM database to expand the Bear River WEAP Model 2017 to the entire Bear River Watershed.

Object ~~types~~Types	WEAP Attributes ~~that have~~ ~~data~~with Data	Instances (#)	Resource Type
Reservoir	Inflow, Initial Storage, Max. Turbine Flow, Net Evaporation, Observed, Volume, Storage Capacity, Top of Inactive, Volume Elevation Curve	SULPHUR CREEK, Woodruff Narrows Reservoir, Node 2.02, Node 6.01, Neponset Reservoir, …, Whitney Reservoir (34)	US Dams, Utah Dams, BRSDM
Demand site	Annual Activity Level, Annual Water Use Rate, Consumption, Monthly Demand	Node 1.02, Node 1.02, Bear River ~~Basin~~Watershed ag, Bear River ~~Basin~~Watershed I, Bear River ~~Basin~~Watershed M (4)	WaDE and BRSDM
Flow Requirement	Minimum Flow Requirement	Node 1.02 (1)	BRSDM
Gauge streamflow	Streamflow Data	BEAR RIVER AT BORDER, WY, BEAR RIVER NEAR UTAH-WYOMING STATE LINE (2)	Idaho Flows dataset, CUAHSI
Transmission link	Maximum Flow Volume	NUFFER, RIGBY, SORENSEN, WILLIAMSON (JENSEN) (4)	Idaho Flows dataset

Use Case 2: ~~What~~Which network connectivity ~~to use~~should be used in a model?

After identifying types of data that describe water systems components, modelers must determine how water supply, demand, and other system components are connected to correctly represent modeled system components. Here, CVs, node connectivity, and links help modelers visualize network connectivity~~,~~ and select an appropriate network for a model scenario. We focus the use case on Hyrum Reservoir which is located on the Little Bear River in Utah.

We used SQL to query all links connected to Hyrum Reservoir in the WaMDaM database and then sort them by data source (i.e., model). Next, we used Microsoft Visio to draw query results which show Hyrum Reservoir supplies two demand sites in the Bear River WEAP Model 2012 (**Figure 6-A**) and three different demand sites in each of the Bear River WEAP Model 2017 and WASH models (**Figure 6**). The latter two models also return flow back to Hyrum Reservoir. The WASH ~~model~~Model has the same schematic as the Bear River WEAP Model 2017 model but uses different labels for its nodes and links (**Figure 6-C**). Using its source and methods metadata, the Bear River WEAP Model 2017 model in this area seems to be the most updated and detailed network, so we recommend using the Bear River WEAP Model 2017 model to expand coverage to the ~~upper~~Upper Bear River (**Figure 6-B**).

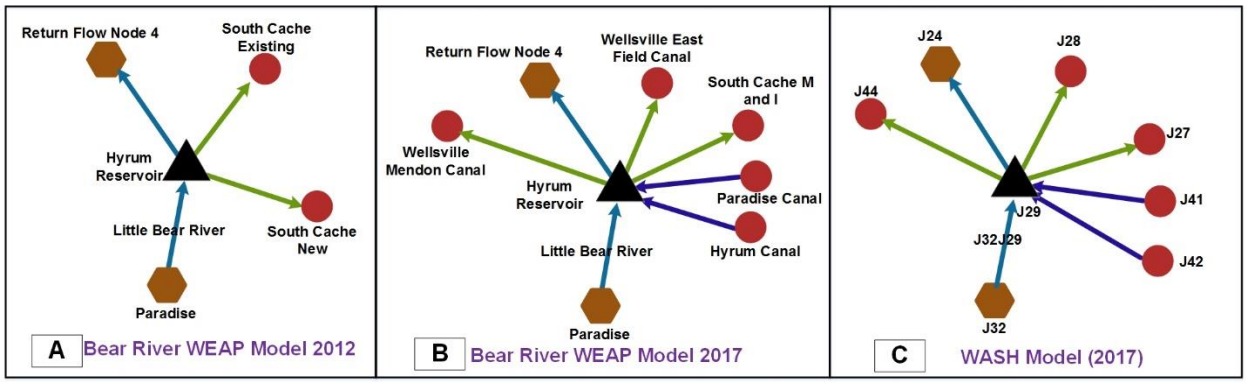


Figure 6: Node-link schematics for flows entering/leaving Hyrum Reservoir for three models in the Lower Bear River Watershed, Utah. Arrows indicate direction of flow. Nodes and links with the same color and shape belong to same controlled object type across models.

Use Case 3: How do data values differ across datasets and which value to choose for a model?

Once modelers have identified the types of data available for a modeling study and the model network, they must choose the data sources and values to use for network components. Here, WaMDaM's multiple attribute data types (e.g., time series, seasonal parameters), CVs, ~~conditional queries~~direct access, and metadata design requirements can help modelers compare datasets, put context to values, and select the appropriate value for a modeling application. We illustrate this process using a subset of the data identified in the first use case for 1) time series and seasonal streamflow below Stewart Dam, Idaho, 2) water use in Cache Valley, Utah, ~~3)~~ and 3) storage elevation curves (i.e., bathymetry) for Hyrum Reservoir in Utah.

Use Case 3.1: What water supply flow values ~~to use~~should a modeler choose at a site (e.g., below Steward Dam)?

Reusing the query for use case #1, controlled vocabulary for the instance and attribute names, and shifting the water year time reference, we identified four data sources with flow data for the site below Stewart Dam in Idaho. The datasets are the USGS, the Utah Division of Water Resources (UDWR), Idaho Department of Water Resources (IDWR), and the Bear River Commission (**Figure 7*:* A**). We used a second SQL query to aggregate and convert all the time series datasets into a comparable cumulative monthly flow in acre-feet per calendar year. The query used the time series metadata of attribute unit, year type, aggregation statistic, and aggregation interval to automate conversions. The four resulting traces span 92 years from 1923-2015 and show data values from the four sources are typically identical except for a few discrepancies in 1996 and 1999 (circles in **Figure 7**: **B**). The source and methods metadata show that the data originates from stream gage data collected by the PacifiCorp power

company. PacifiCorp shares raw data (not available to ~~us~~the authors) with each state~~, which the~~. The states ~~then~~ interpolate ~~if data is~~ missing data points. We recommend using the UDWR dataset which has the longest available record and documented metadata.

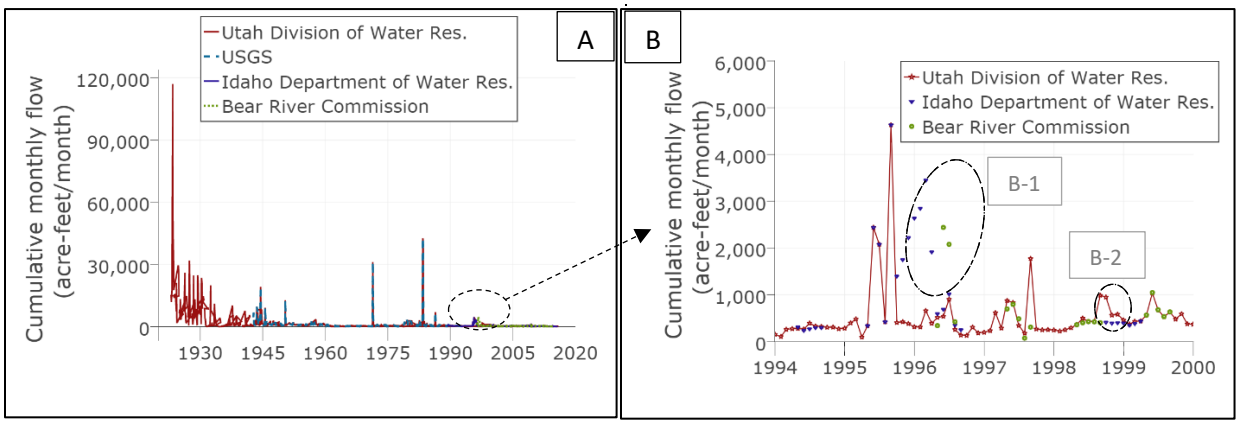


Figure 7: Compiled time series data of flow below Stewart Dam, Idaho reported by different agencies over time. ~~[~~(A~~]~~) 1923 to 2015 and ~~[~~(B~~]~~) a six-year window that highlights similarities and discrepancies (B-1 and B2) among sources after converting the water year into calendar year.

Water management models like WEAP also use seasonal (i.e., average monthly) flow data and modelers need to choose appropriate datasets for them. The same query above also returned seasonal data from a fifth source, the BRSDM model, which has three scenarios for monthly flow (dry, normal, and wet) for the same Stewart Dam site (**Figure 8-A**). The BRSDM materials did not document how seasonal monthly values were derived. However, by comparing ~~query results~~ seasonal values to June high flow values ~~in the longest~~ (UDWR ~~time series record~~ ~~(~~data from 1923 to 2015), we estimated the observed flow is lower 48% of the time than the dry June flow value of 666 acre-ft/month~~, while it~~. We also found the observed flow is higher about 5% of the time than the wet June seasonal flow value of 17,187 acre-ft/month (**Figure 8-B**). These BRSDM model flow values do not capture dry and wet seasons evenly. Thus, we recommend that modelers use newly derived and more representative flow-frequencies from the UDWR dataset like the 5, 50, 95 percentiles which are 184, 702, and -24,900 acre-ft/month for dry, normal, and wet June months.

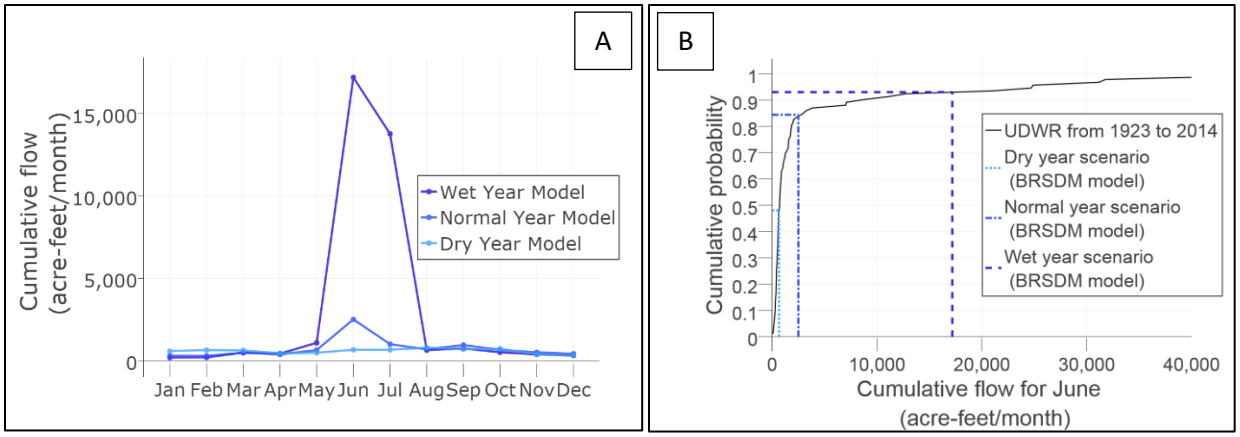


Figure 8: Relating dry, normal, and wet year scenario flows below Stewart Dam, Idaho in BRSDM model (A) to cumulative distribution defined by 91 years of UDWR flow records (B).

Use Case 3.2: What agriculture water use data ~~to use~~should a modeler choose for a demand site?

Systems models often require data for agriculture, and other water uses, which might be derived or estimated. Here, we use CVs, metadata, and multiple attribute data types to query, aggregate, and compare multiple resource types (data sources) for agriculture water use in Cache County in the Lower Bear River, Utah and recommend data to use in a WEAP model. The query used the controlled term diverted flow and returned data from three datasets: WASH ~~and~~model scenarios, WEAP model scenarios, and the WaDE web-service source. The Bear River WEAP Model 2017 uses seasonal demand data for eight sites ~~(**Figure 7A**)~~ and annual demand for two sites. Besides the diverted flow ~~-~~-controlled term, using another controlled term, called "depleted flow", returned a fifth time series form the WaDE source which distinguishes the types of demand (dashed line in **Figure 9~~B~~**~~).~~).

We used the source and method descriptions for attributes, node instances, and scenarios to identify how the data sources represent water use in spatial and time extents. Data either represent i) the entire county area annually in one node as diverted or depleted water like the WaDE dataset (two curves), ii) the entire county seasonally and annually across eight demand sites (WEAP Model 2017), iii) part of the county monthly in one or seven sites as in the Bear River WEAP Model ~~2010~~2012 and WASH models, respectively. The reported annual water use data in WaDE is close to and validates the annual water demand values for the Cache Valley as used in the Bear River WEAP Model 2017. We recommend modelers to use the WaDE "Diversions" data which are annually reported by all water irrigation users in Cache County compared to using demand data that are constant across the years or covers part of

Cache County. Here WEAP accepts input data with daily, monthly, seasonal, and annual spacing and ~~the~~ aggregates or disaggregates them into the ~~models~~model's time step.

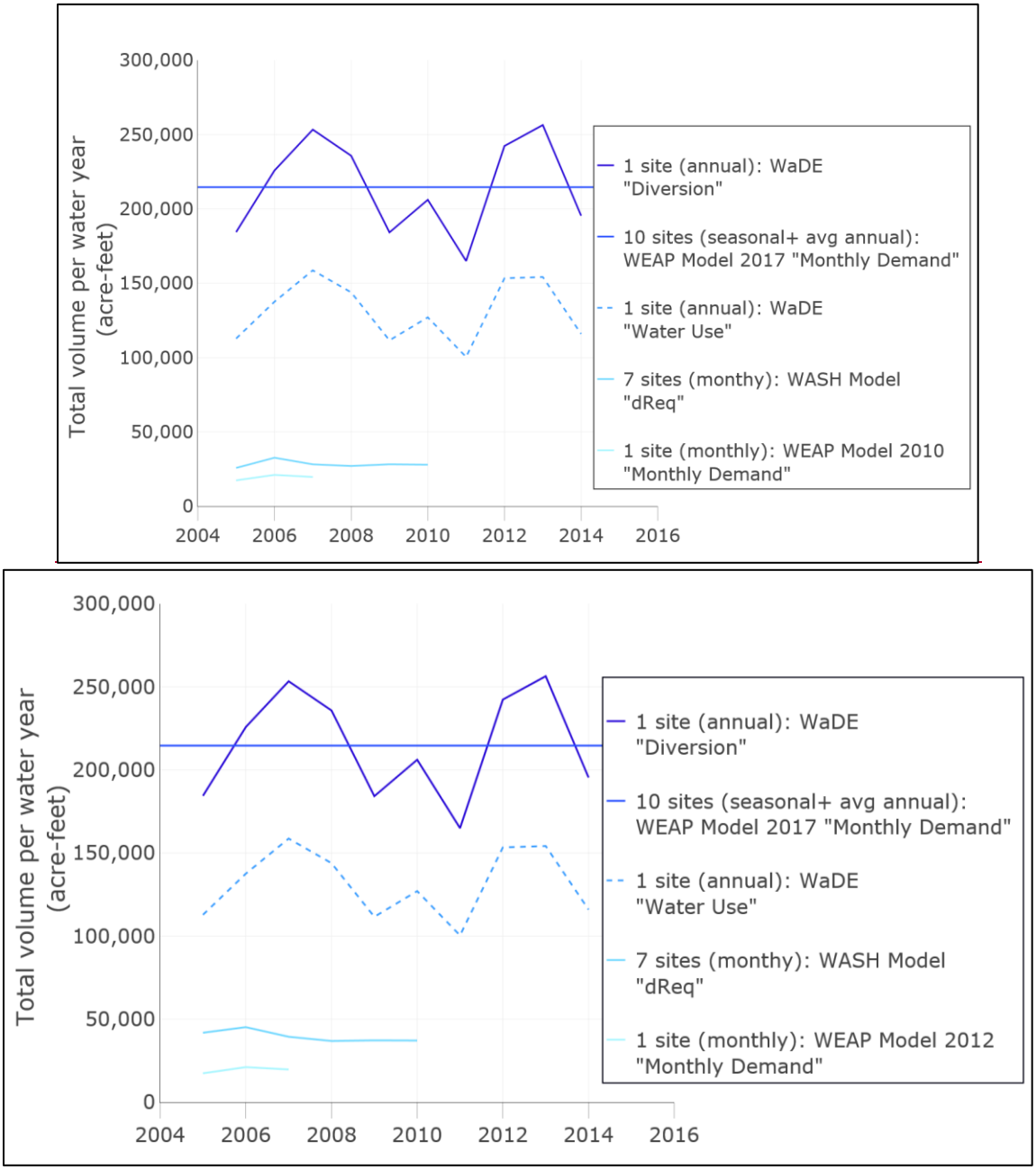


Figure 9: Water demand in Cache County, Utah by source with native attribute term in quotes.

Use Case 3.3: What reservoir volume-elevation curve ~~to use in~~should a modeler choose for a model?

Modelers also search for data describing multi-attribute series such as reservoir bathymetry (elevation versus storage) to represent the physical capacity of reservoirs in their models. Here, we use the controlled instance name of Hyrum Reservoir and controlled attribute names Volume and Elevation to identify ~~five~~four volume-elevation curves for Hyrum Reservoir from the USBOR, Utah Dams, and WEAP model datasets. The USBOR Water Info System dataset has two time series ~~of~~datasets for storage and elevation, which have the same daily time step from January 2010 to May 2017. We plotted both series (**Figure *10***) and used the WaMDaM CVs, metadata, and multiple data types to readily identify and compare multi-attribute bathymetry curves across data sources that had different semantics, measurement periods, and extrapolated versus measured methods. Metadata and semantics are valuable here as misrepresenting the total or live storage or using an old survey could over or under estimate water available to meet demand targets, especially in dry years.

Metadata indicate the ~~five~~four curves originate from two sources: the Utah Dams set and USBOR who owns the dam. The Bear River WEAP ~~models~~model used an older ~~curves~~curve from the UDWR, while Utah Dams and USBOR datasets used USBOR source. Here we report the following three comparison insights, which are related to semantics, the range of data, and date of measurement. First, the top two red curves in ~~**Figure 8**~~**Figure 10** indicate “live storage” which does not account for “dead storage,” while the lower ~~three~~two brown curves reflect “total storage.” The percentage of total storage that is dead storage is relatively high, about 17% in this small reservoir. Second, the slight differences between the two identical lower curves and the top curve are for two bathymetry surveys in 1935 and ~~2016~~2006, respectively. Between the two surveys, total storage decreased by 1,179 acre-feet which is 6% of the original storage due to a decrease in both the dead and live storage potential. Third, the ~~two~~ lower brown ~~curves have~~curve has physical ~~ranges~~range that extend up to 70,000 acre-feet volume and 4,750 feet elevation (not shown) for a future scenario that raised the dam height. From the comparative analysis and metadata, we select the BOR 2006 curve which is for the recent bathymetry survey, used total storage as needed by WEAP, and stayed within the existing operational range of the reservoir.

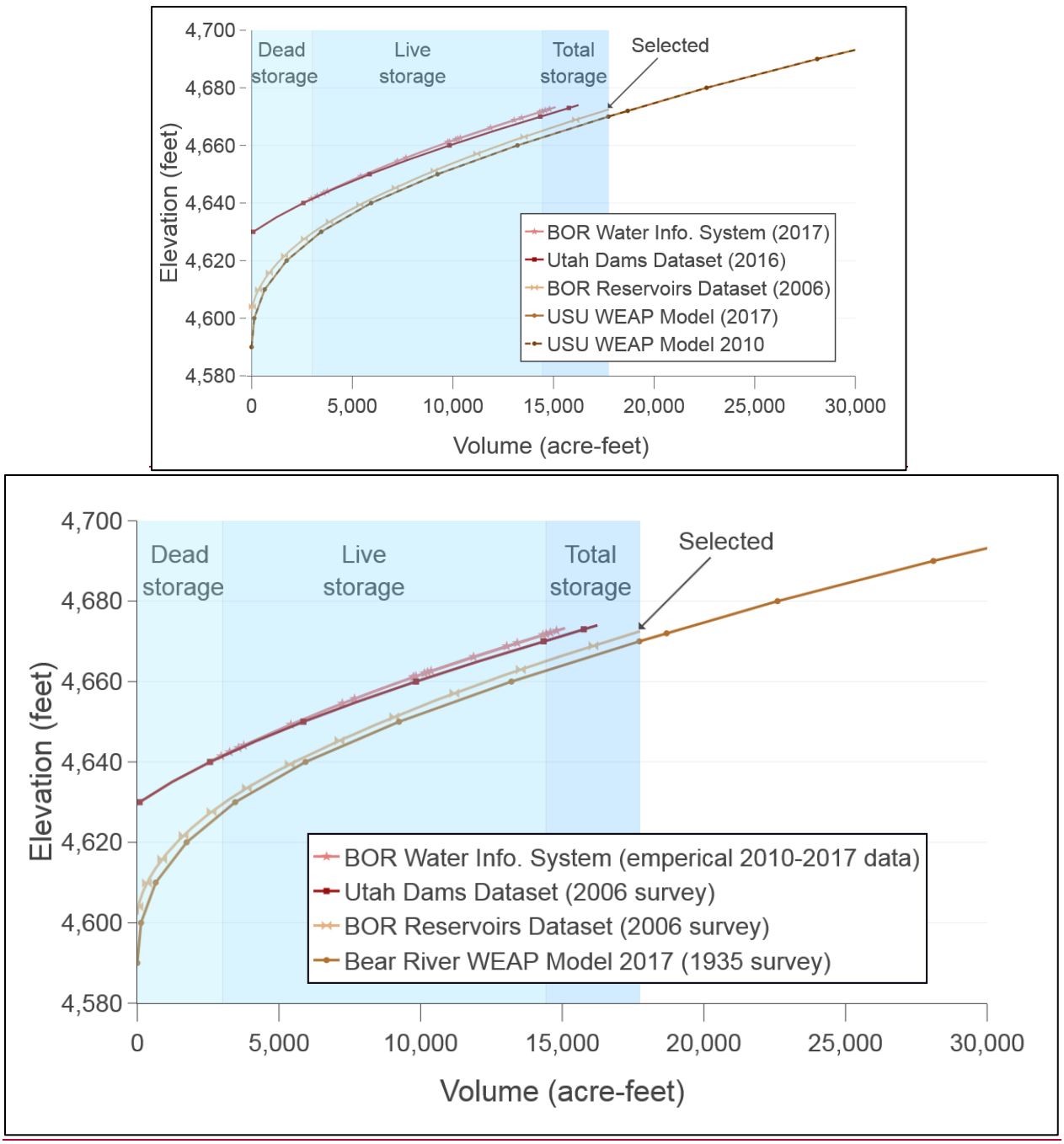


Figure 10: ~~Five~~Four volume-elevation curves for Hyrum Reservoir, Utah. Lighter red and brown curves indicate larger ~~volume~~volumes at the same elevation. Dead, Live, and Total storage zones are from the 2006 ~~BOR~~USBOR survey.

Use Case 4: What ~~is~~are the ~~difference~~differences between two scenarios and which ~~one to~~scenario should a modeler use ~~in a model~~?

Modelers use scenarios to ~~simulate~~evaluate how potential management alternatives can affect system performance. However, ~~models~~scenarios typically have ~~a large number of~~numerous attributes and inputs and it is often difficult to determine the differences in nodes and links, data values, or data sources between multiple scenarios. Here we use the WaMDaM master network, scenario ~~features~~requirement, CVs, and the WaMDaM Wizard Data Loader comparison utility to help a modeler identify differences between existing scenarios in a model. The Wizard executes a script that queries the *ScenarioMappings* table and identifies the data that is shared among and unique to each scenario. Comparison results are exported to an Excel Workbook.

For example, the Bear River WEAP Model 2012 (Utah portion) and Bear River WEAP Model 2017 (Utah and Idaho portions) model scenarios share about ~~17~~12% of the network node and link instances, ~~1%~~22% network metadata, 14% attribute metadata, and ~~5~~14 % data (**Table 4**). ~~Similarity~~Similarly, the BRSDM dry, normal, and wet scenarios have identical master network and metadata for the Wyoming portion of the Bear River ~~basin,~~Watershed and share about ~~80~~93% of data like demand requirements~~,~~ with ~~10~~3.5% unique values to each scenario, such as change in headflows~~.~~ (**Appendix A Table A4**). The larger percentage of shared elements among the BRSDM model scenarios means a correspondingly larger savings in database storage than the WEAP model scenarios.

Because the Bear River WEAP Model 2017 model scenario has more node and link elements, metadata, attributes, and data values, we recommend ~~to use~~using this model scenario as a starting point to expand coverage to the entire ~~basin~~Watershed to include the Wyoming (dark red in **Figure** 5). The BRSDM model network covers the ~~upper~~Upper Bear River in ~~Wyoming (dark red in Figure 3)~~ which can be used as a source to expand the WEAP Bear River WEAP Model 2017 to the entire ~~basin~~Watershed.

Table 4: Unique and shared network nodes and links, metadata (source and method) and data between two WEAP Bear River Watershed model scenarios

***Scenario* comparison element**	**Unique to "Bear River WEAP Model 2012" Scenario** Count of instances (%)	**Shared** Count of instances (%)	**Unique to "Bear River WEAP Model 2017" Scenario** Count of instances (%)
Network nodes and links	~~86 (17.0~~88 (23.5%)	45 (~~8.9~~12%)	~~375 (74.1~~242 (64.5%)
Network metadata	~~218~~88 (20.~~6~~85%)	~~46 (4.4~~92 (21.81%)	~~794 (75.0~~242 (57.35%)

Attributes metadata	~~2,424 (22.2~~1,225 (26.5%)	~~44 (0.4~~654 (14.15%)	~~8,452 (77.4~~2,743 (59.35%)
Data	1,~~217 (19.74~~230 (26.61%)	696 (~~11.29~~13.93 %)	~~4,251 (68.96~~2,748 (59.45%)

Use Case 5: How do annual water shortages at the Bear River Migratory Bird Refuge in the Bear River ~~basin~~Watershed change when serving the Bear River WEAP Model 2017 model with new bathymetry, flow, and demand data selected in use cases 2 and 3?

We selected the Bear River Migratory Bird Refuge (hereafter, the Bird Refuge) at the mouth of Bear River as an environmental demand site to test the sensitivity of water shortages to changes in input of upstream supply, demand, and storage identified in use cases 2 and 3. The site has an annual 425,761 acre-feet water delivery target that is ~~mostly~~primarily required in the winter months. The WaMDaM CVs, consistent data storage, and query method enabled selecting the 1) dry seasonal headflow estimates for the Bear River at Stewart Dam that we derived from the UDWR dataset, 2) total maximum annual demand as reported by the WaDE dataset for the entire Cache County, and 3) bathymetry curve for Hyrum Reservoir from the USBOR dataset. We then used a Python 2.7 script in a local Jupyter Notebook and the WEAP API to export the selected data and populate data automatically in the Bear River WEAP Model 2017. This setup also allowed us to automate the process to create a WEAP scenario for each parameter change, execute the model, and report results for annual unmet demand (shortage) at the Bird Refuge. Each WEAP model run included the simulation period 1966 to ~~2016~~2006.

The modeled annual unmet demand ranged from ~~zero~~0% in wet years to up to 15% of total demand in dry years across the four scenarios ~~(Figure 9).~~(**Figure *11***). Updating Hyrum Reservoir with the new ~~bathometry~~bathymetry (1,179 acre-feet less storage, 6% of capacity) had no observable effect on the annual unmet demand. The average annual unmet demand increased to 1.9% and 2.6% of total demand with higher upstream Cache County irrigation demand and updated headflows for dry years.

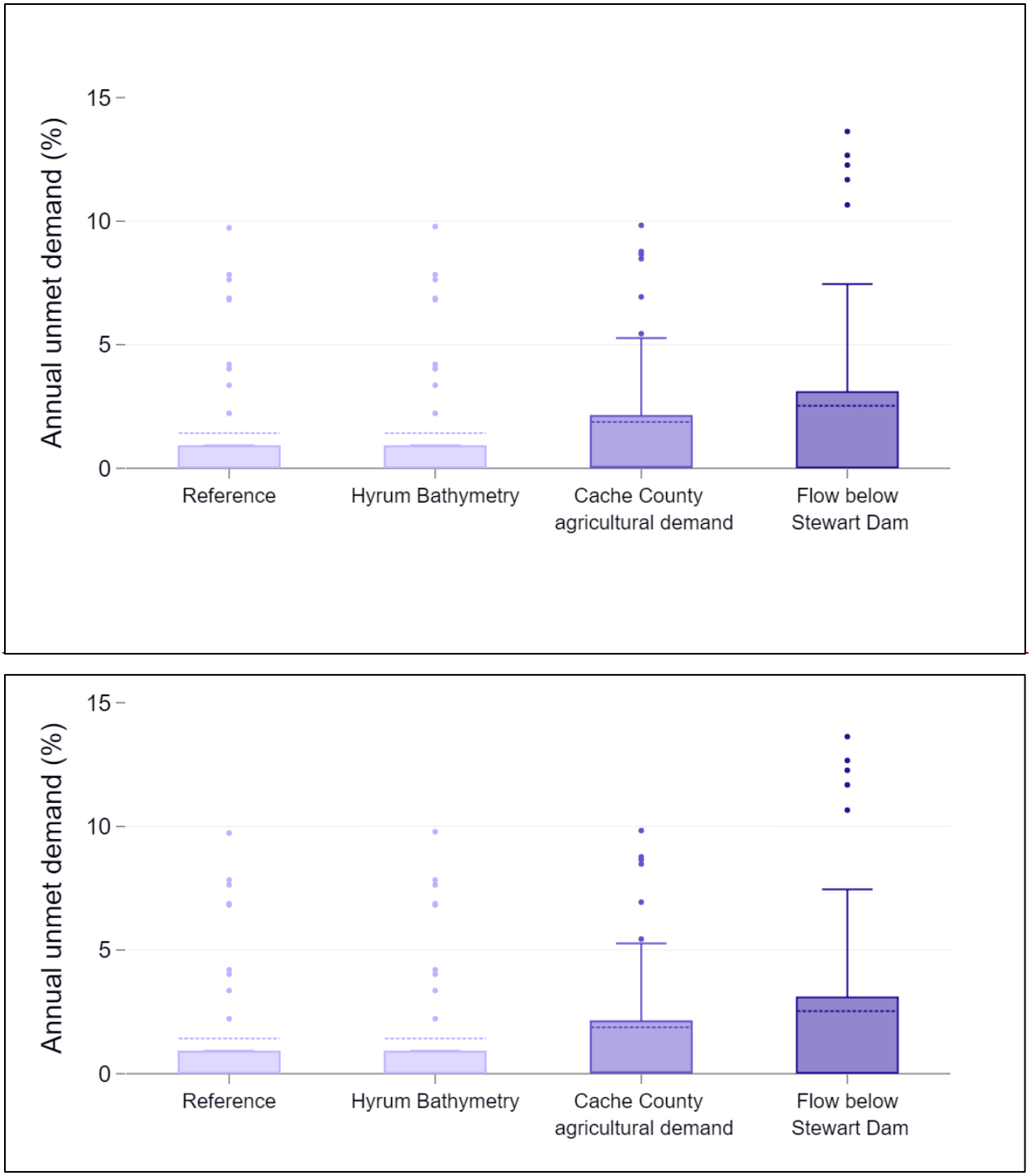


Figure 11: Sensitivity of annual unmet demand at the Bird Refuge, Utah over the simulation period 1966-2006 to changes in upstream storage capacity, demand, and supplies (mean values are in dash lines)

6. Discussion and Further Work

WaMDaM's eight design ~~features~~requirements of modular and extensible components, networks of nodes and links, scenarios, reusable contextual metadata, support for seven data types, extensible controlled vocabularies, ~~conditional queries~~direct access to data, and an open-source environment improve prior work that focused on managing water management data for a single model or ~~data set~~dataset and select systems modeling data types (Horsburgh et al., 2016; Knox et al., 2014)~~. WaMDaM tools help users to prepare datasets and load them into WaMDaM. Tools can also import data from CUAHSI and BOR web services. The use cases show how WaMDaM can help a water resources modeler to identify streamflow, water demand, infrastructure, network, scenario, and other data available to use in a model, then select the appropriate data and serve data to a model like WEAP. These features further allow a water resources modeler to automatically generate and run multiple scenarios that quantify model sensitivity to factors like head flow, water demand, and reservoir storage. The scripting features make it easier for users to set up scenarios, replicate, and extend the work~~. Here we discuss how modelers can use WaMDaM, list limitations of the work, present future work, and invite the community to get involved and provide feedback.

6.1 How can modelers use WaMDaM database and its software?

We show how researchers of five recently published systems modeling studies can use WaMDaM tools to organize, relate, and analyze input data, networks, and scenarios. For example, Ahmadaali et al. (2018) used WEAP to evaluate economic aspects of proposed water management strategies in Urmia Lake, Iran while Angarita et al. (2018) also used WEAP to examine 97 proposed hydropower facilities within a total of 1400 scenarios in the Magdalena River basin, Colombia. Both projects can use the WEAP importer in WaMDaM Wizard to manage the WEAP networks and compare input data for current and future scenarios.

Dogan et al. (2018) developed an open-source version of the California Value Integrated Network (CALVIN) model and separate the model from model data which is stored in a large number of CSV and JSON files in a structured GitHub repository. The researchers could use the WaMDaM Wizard to load input data into the WaMDaM database and compare the input data for different models runs such as for 10 and 40 years' time spans. Wheeler et al. (2018) developed a systems optimization model to identify cooperative management strategies for the large reservoirs on the Eastern Nile Basin. The researchers could use WaMDaM and its scenario comparison tool to track different projected climate change flows for the Nile Basin. Finally, Chini et al. (2018) created a network of virtual water flows for the US electric grid based on six years of empirical data on water use and electricity transfers. The authors could use WaMDaM

to store the created network and its disparate water and energy datasets. WaMDaM can be especially useful to manage the data for the proposed analysis to assess regional interdependencies on a seasonal scale. For each of these studies, storing the modeling data in WaMDaM with its defined schema will allow other researchers to query and reuse data in other studies. This reuse could further increase each study's impact.

6.2 Current limitations

WaMDaM supports numerical, seasonal, categorical, free text, time series, multi-attribute series, and electronic file formats. WaMDaM however does not support gridded data since gridded data are not common to the water resources models we reviewed. The WaMDaM design is implemented in a relational schema which has limitations to adapt and scale compared to NoSQL. The WaMDaM tools help users interact with its SQLite database installed on one machine with no distributed access compared to database servers with API. These software tools are prototypes that are tested using the study datasets on Windows machines. The WaMDaM Wizard is slow to load and validate large datasets.

6.3 Future Work

To improve access and security, future WaMDaM implementations should build web-server APIs with data query functions that distribute and manage the access to many users at the same time and protect the database integrity from unintended changes. Future software tools to load data to the database and export it to models should be time-efficient, more user-friendly, and compatible with Windows, Mac, and Linux. To support more use cases, future work should involve a larger number of diverse datasets, models, and research groups. Future work also should use WaMDaM and web-services to publish, discover, and visualize models and their data and allow multiple users to work with the same datasets. Additionally, future work could leverage scenario and attribute metadata to test use cases that convert data in one time step to other time steps.

In response to earlier feedback, we are collaborating to build a software ecosystem to make WaMDaM interoperable with Hydra Platform web-services~~WaMDaM facilitates these data wrangling tasks by reconciling the disparate datasets into a homogenous structure and by using controlled vocabularies to relate the different native terms across datasets. By loading a dataset or model, a user can access a consistent set of tools to store, organize, query, compare, select, visualize, and share water resources data that the modeler would otherwise have to custom create on their own for their dataset. Loading a dataset also makes it easier to share and reuse~~

data and tools with others. For example, a modeler can export data to run a model or expand an existing model to a larger spatial domain.

Benefits increase as modelers load more datasets, models, and model results sets into WaMDaM, build data exporters, and relate native terms to controlled vocabulary. The CVs can further serve as a basis for a moderated system the water resources systems community can use to relate disparate vocabulary used by different models, analysis methods, and data providers. When needed, users can submit new controlled vocabulary terms at http://vocabulary.wamdam.org/.

WaMDaM supports numerical, seasonal, categorical, free text, timer series, multi-attribute series, and electronic file formats but not gridded data since gridded data is not common to the water resources models we reviewed. The WaMDaM tools help users interact with the data system but can also be made more user-friendly. For example, rather than use a local SQLite database file, future work should build a cross-platform software ecosystem to allow multiple modelers to store, access, visualize, and share common data using a suite of database web-servers and software ecosystem web services that build on the successes of CUAHSI in data publication, discovery (Ames et al., 2012; Couch et al., 2014; Goodall et al., 2008; Horsburgh et al., 2014; Horsburgh et al., 2008), model stores (Knox et al., 2014), and visualization (Rheinheimer, 2018).

In further work, we are also developing workflows to automate the steps to prepare and export all the data needed to run multiple models. These workflows will more readily allow modelers to use the same datasets to run multiple comparison models for the same study domain (e.g., simulation vs optimization) or different domains (e.g. Bear vs. Colorado Rivers). These tasks are now difficult because the modeler must manually build two (or more) models from scratch.

In all these efforts, we seek community involvement to 1) add new datasets and models for new locations, 2) build new exporters to serve data to new models, and 3) further define the system of controlled vocabulary that links the native vocabulary of existing models and datasets. More involvement will further sharing and allow others to reuse tools to store and organize water management data and serve data to models in different locations. We encourage modelers who add data to WaMDaM to share their SQLite files online on GitHub or HydroShare (Tarboton et al., 2014). We also invite the water systems modeling and hydroinformatics communities to provide feedback to improve WaMDaM by submitting issues on GitHub at https://github.com/WamdamProject/WaMDaM-software-ecosystem/issues.

(Knox et al., 2014), OpenAgua (Rheinheimer, 2018), and HydroShare. The ecosystem tools will allow WaMDaM users to import data stored in Hydra Platform as a new source of data. Users will also be able to export WaMDaM data into Hydra Platform and visualize networks and their data in OpenAgua. We are also integrating WaMDaM as a new HydroShare resource type to publish populated WaMDaM SQLite files and extract their metadata to enable search and discovery (Horsburgh et al., 2015). Lastly, we are developing workflows to automate the steps to prepare and export all the data needed to run multiple models. These workflows will more readily allow modelers to use the same datasets to run multiple comparison models for the same study domain (e.g., simulation vs optimization) or different spatial domains (e.g. Bear River vs. Colorado River). These tasks are now difficult because the modeler must manually build two (or more) models from scratch.

6.4 Invitation to community involvement and feedback

Over the past five years, we sought and received feedback from colleagues and collaborators on the WaMDaM design and tools. There is still need for testing and feedback from a larger, more diverse community of users. In all these efforts, we seek community involvement to 1) add new datasets and models for new locations, 2) build new exporters to serve data to new models, and 3) further define the system of controlled vocabulary that can help relate native vocabulary of existing models and datasets. More involvement can benefit a variety of people who work with systems simulation and optimization data and models. WaMDaM can serve as a first step toward a standardized method to store, organize, and share water resources systems modeling data.

7. Conclusions

This paper addressed the problem of needing multiple methods to organize, store, query, and analyze water management data to identify input data to develop or extend a water management model. We contributed a new data model (WaMDaM) implemented in a relational database to organize water management data with contextual metadata and controlled vocabularies to generalize data analysis for multiple data sources, models, and study areas.

The design of WaMDaM integrated eight design requirements that were previously only partially supported by forty prior water resources data systems, models, and standards. The requirements include: 1) modular and extensible components, 2) networks of nodes and links, 3) scenarios and version control, 4) reusable contextual metadata, 5) support for multiple data

types used by systems models, 6) extensible controlled vocabularies, 7) ~~conditional queries to~~ direct access to subsets of data and metadata, and 8) an open-source environment.

We demonstrated the WaMDaM design by using 13 ~~data sets~~datasets and models to answer five use case questions in the Bear River Watershed, ~~USA~~United States. The use cases allowed modelers to: i) search for input data within a model study area, ii) identify flow directions and connections among natural and engineered system components, iii) identify and compare water supply, demand, and reservoir data across multiple ~~data sets~~datasets and models, iv) show data similarities and differences among modeling scenarios, and v) select data, serve the data to a model, and run multiple model scenarios.

Results showed how WaMDaM ~~unifying~~unifies data formats, structures, and controlled vocabulary identified data for 15 attributes (out of 71 needed) from six data sources to expand the spatial extent of a WEAP model. Results also showed discrepancies in river discharge data, demand, and reservoir area-elevation curves. Results helped select input data and develop multiple scenarios. Serving the data to run an existing WEAP model revealed and quantified that shortages at an environmental demand site were sensitive to changes in upstream agricultural water demand and headflows but not reservoir ~~sedimentation~~capacity.

~~In further work, we are~~The WEAP API and SQL make it possible for users to use WaMDaM to set up scenarios, replicate, and extend the work. WaMDaM facilitates these data wrangling tasks by reconciling the disparate datasets into a homogenous structure and by using controlled vocabularies to relate the different native terms across datasets. Modelers can then spend more time on data analysis and synthesis than on time consuming and error-prone steps to manipulate data to set up and run a model.

In further work, we are collaborating on a software ecosystem to make WaMDaM interoperable with Hydra Platform and OpenAgua to visualize networks and their data. We are also developing workflows to automate the steps to serve the same input data already organized in WaMDaM to multiple comparison models for a study area. We also seek community involvement to load larger and more diverse data and model sets which will allow others to reuse data and build models in new areas. These expansions will require more robust methods to define, relate, specify, and expand controlled vocabularies for water management data. ~~Future work will also build a software ecosystem to allow multiple modelers to use a suite of web-service tools to store, access, visualize, and share water management data.~~ We invite the systems modeling and hydroinformatics communities to provide feedback to improve WaMDaM.

Acknowledgments

This work was supported by the National Science Foundation through the CI-Water Project grant 1135482, iUtah grant 1208732, and Utah Mineral Lease Funds. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation. The Intermountain Section of the American Public Works Association, the Utah Chapter of the American Public Works Association, and the Utah Water Users Association provided additional financial support. Jeffery Horsburgh provided extensive feedback on the final design and writings. David Tarboton, Stephen Knox, and David Rheinheimer provided thoughtful feedback and comments on earlier designs and drafts. Hadia Akbar and Jiada Li used the WaMDaM Wizard and ~~Juptyer~~Jupyter Notebooks to replicate use case results. We thank the two anonymous reviewers for their constructive feedback.

References

Abdallah, A., 2018a. WaMDaM JupyterNotebooks. doi: https://zenodo.org/record/1484581, url: https://github.com/WamdamProject/WaMDaM_JupyterNotebooks

Abdallah, A.M., 2018b. WaMDaM Controlled Vocabularies Source Code. doi: https://zenodo.org/record/1484579, url: https://github.com/WamdamProject/WaMDaM_ControlledVocabularies

Abdallah, A.M., 2018c. WaMDaM Information Model Source Code. doi: https://doi.org/10.5281/zenodo.1484575, url: https://github.com/WamdamProject/WaMDaM_Information_Model

Abdallah, A.M., 2018d. The WaMDaM Wizard Source Code. doi: https://zenodo.org/badge/latestdoi/92693785, url: https://github.com/WamdamProject/WaMDaM_Wizard

Abdallah, A.M., Rosenberg, D.E., 2014. WaM-DaM: A Data Model to Organize and Synthesize Water Management Data, In: Ames, D.P., Quinn, N., Rizzoli, A.E. (Eds.), 7th International Congress on Environmental Modelling and Software. International Environmental Modelling and Software Society (iEMSs).

Ahmadaali, J., Barani, G.-A., Qaderi, K., Hessari, B., 2018. Analysis of the Effects of Water Management Strategies and Climate Change on the Environmental and Agricultural Sustainability of Urmia Lake Basin, Iran. Water 10(2) 160.

Alafifi, A., Rosenberg, D., ~~In~~in review. Systems Modeling to Improve River, Riparian, and Wetland Habitat Quality and Area. Environmental Modelling & Software.

~~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.~~

~~Aspen Institute, 2017. INTERNET OF WATER: Sharing and Integrating Water Data for Sustainability.~~

Angarita, H., Wickel, A.J., Sieber, J., Chavarro, J., Maldonado-Ocampo, J.A., Herrera-R, G.A., Delgado, J., Purkey, D., 2018. Basin-scale impacts of hydropower development on the Mompós Depression wetlands, Colombia. Hydrol. Earth Syst. Sci. 22(5) 2839-2865.

Bajcsy, P., 2008. A Perspective on Cyberinfrastructure for Water Research Driven by Informatics Methodologies. Geography Compass 2(6) 2040-2061.

Beniston, M., Stoffel, M., Harding, R., Kernan, M., Ludwig, R., Moors, E., Samuels, P., Tockner, K., 2012. Obstacles to data access for research related to climate and water: Implications for science and EU policy-making. Environmental Science & Policy 17(0) 41-48.

~~Blodgett, D., Read, E., Lucido, J., Slawecki, T., Young, D., 2016. An Analysis of Water Data Systems to Inform the Open Water Data Initiative. JAWRA Journal of the American Water Resources Association 52(4) 845-858.~~

Brown, C.M., Lund, J.R., Cai, X., Reed, P.M., Zagona, E.A., Ostfeld, A., Hall, J., Characklis, G.W., Yu, W., Brekke, L., 2015. The future of water resources systems analysis: Toward a scientific framework for sustainable water management. Water Resources Research.

Campbell, J.L., Rustad, L.E., Porter, J.H., Taylor, J.R., Dereszynski, E.W., Shanley, J.B., Gries, C., Henshaw, D.L., Martin, M.E., Sheldon, W.M., Boose, E.R., 2013. Quantity is Nothing without Quality: Automated QA/QC for Streaming Environmental Sensor Data. BioScience 63(7) 574-585.

Cantor, A., Michael Kiparsky, Rónán Kennedy, Susan Hubbard,, Roger Bales, L.C.P., Kamyar Guivetchi,, Christina McCready, a.G.D., 2018. Data for Water Decision Making: Informing the Implementation of California's Open and Transparent Water Data Act through Research and Engagement.

Carata, L., Akoush, S., Balakrishnan, N., Bytheway, T., Sohan, R., Seltzer, M., Hopper, A., 2014. A primer on provenance. Commun. ACM 57(5) 52-60.
Chen, P.P.-S., 1976. The entity-relationship model - Toward a unified view of data. ACM Trans. Database Syst. 1(1) 9-36.
Chini, C.M., Djehdian, L.A., Lubega, W.N., Stillwell, A.S., 2018. Virtual water transfers of the US electric grid. Nature Energy.
Codd, E.F., 1970. A relational model of data for large shared data banks. Commun. ACM 13(6) 377-387.
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.).
Couch, A., Hooperb, R., Pollakb, J., Martinb, M., Seulb, M., 2014. Enabling Water Science at the CUAHSI Water Data Center, iEMSs 2014 Conference, p. 1.
Dodd, B., 2016. The Open and Transparent Water Data Act, In: California, S.o. (Ed.), AB-1755.
Dogan, M.S., Fefer, M.A., Herman, J.D., Hart, Q.J., Merz, J.R., Medellín-Azuara, J., Lund, J.R., 2018. An open-source Python implementation of California's hydroeconomic optimization model. Environmental Modelling & Software 108 8-13.
Draper, A., Jenkins, M., Kirby, K., Lund, J., Howitt, R., 2003. Economic-Engineering Optimization for California Water Management. Journal of Water Resources Planning and Management 129(3) 155-164.
Dublin Core Metadata Initiative (DCMI), 2013. Dublin Core Metadata Element Set, Version 1.1 ANSI/NISO Z39.85-2012
National Information Standards Organization (NISO): Baltimore, MD.
Duval, E., Hodgins, W., Sutton, S., Weibel, S.L., 2002. Metadata principles and practicalities. D-lib Magazine 8(4) 16.
Easterbrook, S.M., 2014. Open code for open science? Nature Geosci 7(11) 779-781.
Gil, Y., David, C.H., Demir, I., Essawy, B.T., Fulweiler, R.W., Goodall, J.L., Karlstrom, L., Lee, H., Mills, H.J., Oh, J.-H., Pierce, S.A., Pope, A., Tzeng, M.W., Villamizar, S.R., Yu, X., 2016. Toward the Geoscience Paper of the Future: Best practices for documenting and sharing research from data to software to provenance. Earth and Space Science 3(10) 388-415.
Goodall, J.L., Horsburgh, J.S., Whiteaker, T.L., Maidment, D.R., Zaslavsky, I., 2008. A first approach to web services for the National Water Information System. Environmental Modelling & Software 23(4) 404-411.
Goodman, A., Pepe, A., Blocker, A.W., Borgman, C.L., Cranmer, K., Crosas, M., Di Stefano, R., Gil, Y., Groth, P., Hedstrom, M., 2014. Ten Simple Rules for the Care and Feeding of Scientific Data. PLoS computational biology 10(4) e1003542.
Govindaraju, R., Engel, B., Ebert, D., Fossum, B., Huber, M., Jafvert, C., Kumar, S., Merwade, V., Niyogi, D., Oliver, L., Prabhakar, S., Rochon, G., Song, C., Zhao, L., 2009. Vision of Cyberinfrastructure for End-to-End Environmental Explorations (C4E4). Journal of Hydrologic Engineering 14(1) 53-64.
Gray, J., Liu, D.T., Nieto-Santisteban, M., Szalay, A., DeWitt, D.J., Heber, G., 2005. Scientific data management in the coming decade. SIGMOD Rec. 34(4) 34-41.
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.

Hoberman, S., 2014. Data Modeling for MongoDB: Building Well-Designed and Supportable MongoDB Databases. Technics Publications.

Horsburgh, J.S., Aufdenkampe, A.K., Mayorga, E., Lehnert, K.A., Hsu, L., Song, L., Jones, A.S., Damiano, S.G., Tarboton, D.G., Valentine, D., Zaslavsky, I., Whitenack, T., 2016. Observations Data Model 2: A community information model for spatially discrete Earth observations. Environmental Modelling & Software 79 55-74.

Horsburgh, J.S., Morsy, M.M., Castronova, A.M., Goodall, J.L., Gan, T., Yi, H., Stealey, M.J., Tarboton, D.G., 2015. Hydroshare: Sharing diverse environmental data types and models as social objects with application to the hydrology domain. JAWRA Journal of the American Water Resources Association.

Horsburgh, J.S., Tarboton, D.G., Hooper, R.P., Zaslavsky, I., 2014. Managing a community shared vocabulary for hydrologic observations. Environmental Modelling & Software 52(0) 62-73.

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.

HydroLogics, 2009. User Manual for OASIS WITH OCL.

Knox, S., 2018. Hydra Platform handling of units and dimensions.

Knox, S., Meier, P., Harou, J., 2014. Web service and plug-in architecture for flexibility and openness of environmental data sharing platforms, In: Ames, D.P., Quinn, N., Rizzoli, A.E. (Eds.), 7th International Congress on Environmental Modelling and Software. San Diego, California, USA.

Laituri, M., Sternlieb, F., 2014. Water Data Systems: Science, Practice, and Policy. Journal of Contemporary Water Research & Education 153(1) 1-3.

Laniak, G.F., Olchin, G., Goodall, J., Voinov, A., Hill, M., Glynn, P., Whelan, G., Geller, G., Quinn, N., Blind, M., Peckham, S., Reaney, S., Gaber, N., Kennedy, R., Hughes, A., 2013. Integrated environmental modeling: A vision and roadmap for the future. Environmental Modelling & Software 39(0) 3-23.

Larsen, S~~., Hamilton, S., Lucido, J., Garner, B.,~~.G., Young, D., ~~2016. Supporting Diverse~~2014. WaDE: An Interoperable Data ~~Providers in the Open~~Exchange Network for Sharing Water ~~Data Initiative: Communicating Water Data Quality~~Planning and ~~Fitness of~~ Use~~. JAWRA~~ Data. Journal of ~~the American Water Resources Association 52(4) 859-872~~Contemporary Water Research & Education(153) 33-41.

Leonard, L., Duffy, C.J., 2013. Essential Terrestrial Variable data workflows for distributed water resources modeling. Environmental Modelling & Software 50(0) 85-96.

Loucks, D.P., Van Beek, E., Stedinger, J.R., Dijkman, J.P., Villars, M.T., 2005. Water resources systems planning and management: an introduction to methods, models and applications. Paris: UNESCO.

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

Maidment, D.R., 2008. Bringing Water Data Together. Journal of Water Resources Planning and Management 134(2) 95-96.

Maidment, D.R., 2016. Open Water Data in Space and Time. JAWRA Journal of the American Water Resources Association n/a-n/a.

Michener, W.K., 2006. Meta-information concepts for ecological data management. Ecological Informatics 1(1) 3-7.

Miller, R.C., Guertin, D.P., Heilman, P., 2004. Information Technology in Watershed Management Decision Making. JAWRA Journal of the American Water Resources Association 40(2) 347-357.

Morsy, M.M., Goodall, J.L., Castronova, A.M., Dash, P., Merwade, V., Sadler, J.M., Rajib, M.A., Horsburgh, J.S., Tarboton, D.G., 2017. Design of a metadata framework for

environmental models with an example hydrologic application in HydroShare. Environmental Modelling & Software 93(Supplement C) 13-28.
N.M. Samu, S.-C.K., and P.W. O'Connor, 2017. Existing Hydropower Assets [series] FY17Q4, In: National Hydropower Asset Assessment Program (Ed.): Oak Ridge National Laboratory, Oak Ridge, TN.
National Research Council, 2012. Challenges and opportunities in the hydrologic sciences.
Order, E., 2013. Executive order--making open and machine readable the new default for government information.
Pokorný, J., 2006. Database architectures: Current trends and their relationships to environmental data management. Environmental Modelling & Software 21(11) 1579-1586.
Rajaram, H., Bahr, J.M., Blöschl, G., Cai, X., Scott Mackay, D., Michalak, A.M., Montanari, A., Sanchez-Villa, X., Sander, G., 2015. A reflection on the first 50 years of Water Resources Research. Water Resources Research n/a-n/a.
Refsgaard, J.C., Nilsson, B., Brown, J., Klauer, B., Moore, R., Bech, T., Vurro, M., Blind, M., Castilla, G., Tsanis, I., Biza, P., 2005. Harmonised techniques and representative river basin data for assessment and use of uncertainty information in integrated water management (HarmoniRiB). Environmental Science & Policy 8(3) 267-277.
Rheinheimer, D., 2018. OpenAgua: Collaborative water system modeling for a new generation.
Ridley, M., Stoker, C., 2001. Data Management Tools, In: Shelke, D.P.G. (Ed.), Bridging the Gap: Meeting the World's Water and Environmental Resources Challenges. American Society of Civil Engineers: Orlando, Florida, United States, pp. 1-10.
Rosenberg, D., 2017. Bear River WEAP Models (2012 and 2017).
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.
Rossman, L.A., 2000. EPANET 2: Users Manual. US Environmental Protection Agency: Cincinnati, Ohio.
Samu, N.M., Kao, S.-C., O'Connor, P.W., 2017. Existing Hydropower Assets [series] FY17Q4, In: National Hydropower Asset Assessment Program (Ed.): Oak Ridge National Laboratory, Oak Ridge, TN.
Sarle, W.S., 1995. Measurement theory: Frequently asked questions. Disseminations of the International Statistical Applications Institute 1(4) 61-66.
Sears, R., Ingen, C.v., Gray, J., , 2006. To BLOB or Not To BLOB: Large Object Storage in a Database or a Filesystem? , Microsoft Research Microsoft Microsoft Corporation One Microsoft Way Redmond, WA 98052
Sehlke, G., Jacobson, J., 2005. System Dynamics Modeling of Transboundary Systems: The Bear River Basin Model. Ground Water 43(5) 722-730.
Szalay, A.S., Blakeley, J.A., 2009. Gray's laws: database-centric computing in science.
Tarboton, D.G., Idaszak, R., Horsburgh, J.S., Heard, J., Ames, D., Goodall, J.L., Band, L., Merwade, V., Couch, A., Arrigo, J., 2014. Hydro share: Advancing collaboration through hydrologic data and model sharing, 7th International Congress on Environmental Modelling and Software, iEMSs 2014. International Environmental Modelling and Software Society.
U.S. Geological Survey, 2013. USGS Small-scale Dataset - Major Dams of the United States 200603 Shapefile: U.S. Geological Survey, 30-May-2013 ed. http://nationalatlas.gov: Reston, VA.
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.

Vogel, R.M., Lall, U., Cai, X., Rajagopalan, B., Weiskel, P.K., Hooper, R.P., Matalas, N.C., 2015. Hydrology: The interdisciplinary science of water. Water Resources Research 51(6) 4409-4430.

Watkins, D.W., 2013. Water resources systems analysis through case studies data and models for decision making. Environmental Water Resources Institute. American Society of Civil Engineers., Reston, Virginia.

Wheeler, K.G., Hall, J.W., Abdo, G.M., Dadson, S.J., Kasprzyk, J.R., Smith, R., Zagona, E.A., 2018. Exploring Cooperative Transboundary River Management Strategies for the Eastern Nile Basin. Water Resources Research 0(ja).

Wurbs, R.A., 1993. Reservoir-System Simulation and Optimization Models. Journal of Water Resources Planning and Management-Asce 119(4) 455-472.

Wurbs, R.A., ~~2005. Comparative evaluation of generalized river/reservoir system models. Texas Water Resources Institute: College Station, TX.~~

~~Wurbs, R.A.,~~ 2012. Reservoir/River System Management Models. The Texas Water Journal 3(1) 16.

Yates, D., Sieber, J., Purkey, D., Huber-Lee, A., 2005. WEAP21—A Demand-, Priority-, and Preference-Driven Water Planning Model. Water International 30(4) 487-500.

Zagona, E.A., Fulp, T.J., Shane, R., Magee, T., Goranflo, H.M., 2001. RiverWare: A Generalized Tool for Complex Reservoir System Modeling. JAWRA Journal of the American Water Resources Association 37(4) 913-929.

Zeiler, M., 1999. Modeling our world : the ESRI guide to geodatabase design. Environmental Systems Research Institute (ESRI) Press, Redlands, CA.