Utah State University

Water Management Data Model

MIS 4330 - Spring 2013

Class Project Report

Due April 25, 2013

JON M. HUNTSMAN SCHOOL OF BUSINESS

The Water Warriors Group
Adel Abdallah, Jason Randall, Jason Smith, and Maxwell Barrus

Introduction

Similar to the massive data generation in daily business transactions, there is a growing mass of data that are collected about water quality and quantity in the United States. In fact, there are tens of thousands of data sensors that measure real-time data in water reservoirs, streams, and rivers. However, data management systems for collected water data have not been developed at the same pace as in the business world. Also, in business, corporations are obligated to share their data with others while water resources data should be shared to manage the same resources across many boundaries.

Currently, water data is being collected separately by dozens of agencies, states, and cities. Each entity uses different standards, semantics, and information systems to collect and describe the same thing. For example, water reservoirs are owned and managed either by the federal government, states, or private entities. Therefore, to get data about reservoirs in Utah you have to search for data from different sources and then spend a considerable amount of time to compile and process the data, even in the same state. There is an incredible need for an integrated national water data management system to help manage our limited water resources among competing users and save time and efforts in the end. Such information systems require ever updated, consistent, accessible, and well organized data and its associated metadata. Metadata is defined as the description of data that can help other people correctly understand and interpret the data.

Successful water management also requires detailed knowledge of the water supply and demand infrastructures and how they are connected. In addition, water management requires observational data about reservoir and canal inflows and releases and why these releases are being made. Thus, there is a need for a data model that can consistently organize, store, and deliver water management data to interested parties.

In this project we focus on reservoir data and integrate physical descriptions about reservoirs. We also incorporate an existing water observation database that measures water inflow and outflow from reservoirs.

1. Motivations and Objectives

In the United State, there are over 10,000 water monitoring stations that collect water data for dozens of variables at a time every 30 minutes. In addition, there are over 8,000 reservoirs that store water for multiple purposes, such as irrigation and flood control. Generally reservoirs differ from each other in their physical description, building materials, owner, operator, as well as many other factors. However, data about reservoirs is not stored in one central database and thus it is hard to answer scientific questions regarding all of the reservoirs in the State of Utah as an example. An example of one important scientific question could be: “What are all of the reservoirs that pose a high hazard to the public in Cache County?” Another challenge in answering scientific questions regarding reservoirs and water flows is that water data is not linked to reservoirs. For example, if a scientist wants to know the temperature of water flowing to Hyrum Reservoir in Utah, he or she has to compile two separate datasets to match the reservoir with the water monitoring station that flows to that reservoir. To solve these issues we designed a database that achieves the following objectives:

- Develop a central data model that stores and organizes reservoir data from across different sources
- Enforce controlled vocabulary/implement constraints on inputting reservoir data to maintain consistency and integrity of data from different sources
- Incorporate and link our reservoir data model with an existing water database that measures water data flowing into or out of reservoirs

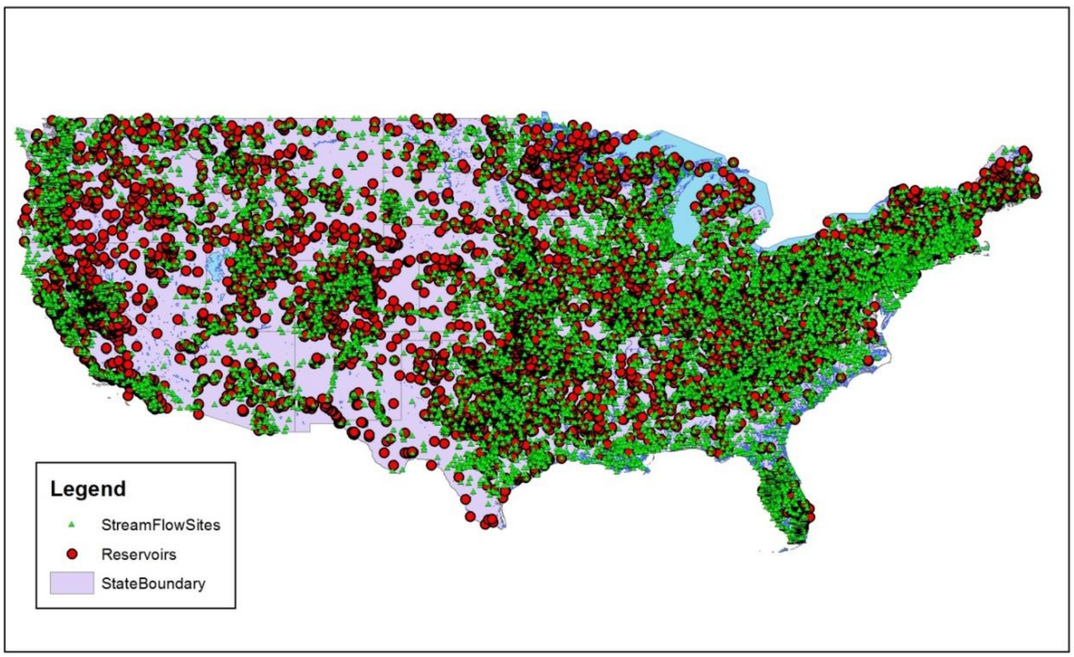

Figure (1): The US network of water monitoring sites and reservoirs

2. Project Scope & Execution

Water managers and researchers are faced by many problems:

- Available data about water supply and demand infrastructure is either held with city water mangers or operators on their computers or even verbally
- Metadata that describes water releases, for example, is either non-existent or not accessible
- Agency's water management data and infrastructure use different vocabularies to describe the same thing

In this project we build on an existing rough water management data model that needs many improvements. The major improvements are mainly: fixing the model relationships, adding more entities, achieving normalization, using controlled vocabularies, and maintaining model integrity.

First we will develop the logical ER water management data model. Here we will review the existing model and consider the right relationships among entities and use controlled vocabularies. Then we will develop the physical model and consider primary and foreign keys and set constraints on values. After that we will implement the data model on SQL Server.

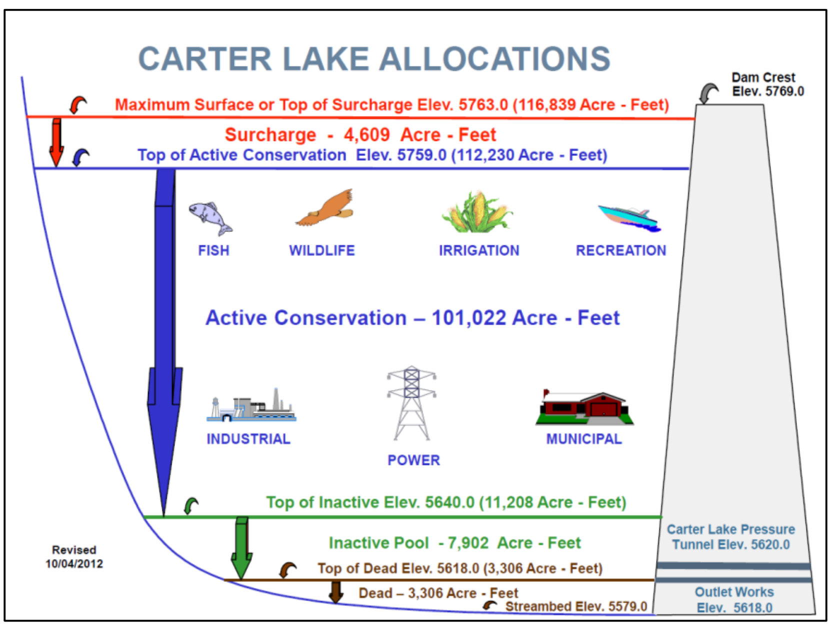

Figure (2): Schematic shows a reservoir physical characteristics and potential water use zones (US BOR 2013)

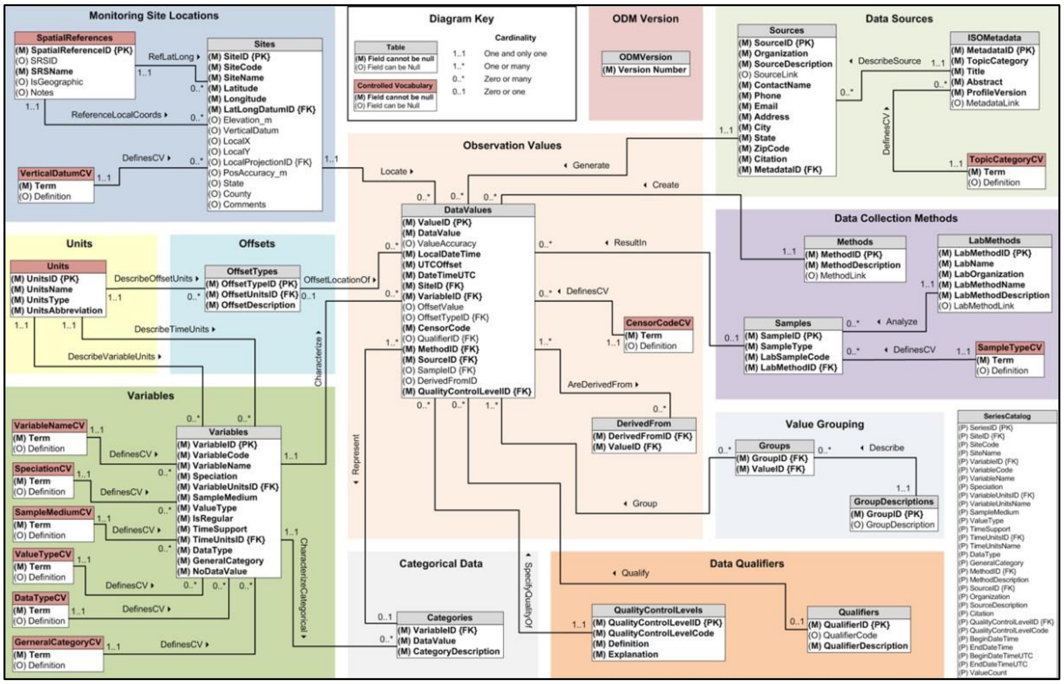

Figure (3): Existing water monitoring sites data model (Horsburgh et al., 2009)

3. Methods and Analysis

When developing a working data model and database, planning is critical. In our initial meetings we talked about the many entities and attributes required. These entities and attributes required in depth technical knowledge. Before we could get started building our data model, we had to educate ourselves on some water research concepts. These concepts included:

- Inflow and outflow at monitoring sites
- How sites relate to each reservoir
- Curve measurements and variables
- Different measurements depending on elevation

After discussing these concepts in depth with our stakeholder, we were generally prepared to construct our first draft of the ER diagram. After each subsequent draft, additional questions were uncovered that we needed to research.

A final draft was prepared that we were confident would work. Some assumptions had to be made. We based these assumptions on the normal standards that are used in Utah. If this database is to be used nation-wide, a second look needs to be taken to resolve any issues that might arise. The assumptions we made involved the surveying and research sources. We assumed that only one source was used when gathering information about a particular reservoir. For example: Utah Corps of engineer's gathers all information for an entire reservoir such as Hyrum Dam.

Once we had the design planned out for the reservoir database, we needed to connect it to the existing sites database. A few problems arose with this task. Sites can be considered an inflow to a reservoir, an outflow to another reservoir, or an outflow from one reservoir and an inflow to another. The field of study that operates the sites has not been concerned with documenting which sites are inflows/outflows to each reservoir in the past. In order to connect these two databases together on a large scale, documentation would be needed for each site to associate them with their respective reservoirs. For the scope of our project, we manually associated sites with reservoirs within Cache and Davis County.

Creating a more practical data model and database required many tools and collaboration technologies. Microsoft Visio was used during the brainstorming phase to chart ideas for our ER diagram. Once we cemented down the concepts of our database we used Visio to produce the final ER diagram. We chose to use Microsoft SQL Server to execute our diagram. Using the Server Management Studio we used the GUI to create the entities and attributes. We also used the GUI to connect our tables with primary and foreign keys. For this database to be distributed on a large scale, DDL statements will need to be created so that others can execute and launch the database on their local machines. Much of the test data we used to populate our tables came from Excel spreadsheets. In excel we sorted and normalized the test data. We then imported the data to the tables using SQL Server import wizard. SQL code was written to test the structure of our database. We used questions that water scientists might ask to form our queries.

Collaboration technologies we used included Remote Desktop Connection, Google Docs, and USU Canvas. The database was hosted in an office at the water lab. We used Remote Desktop to gain access to SQL Management Studio during the design and implementation phases, as well as to run queries on the database. Google Docs and USU Canvas were used as a way to create, edit, and share supporting documentation.

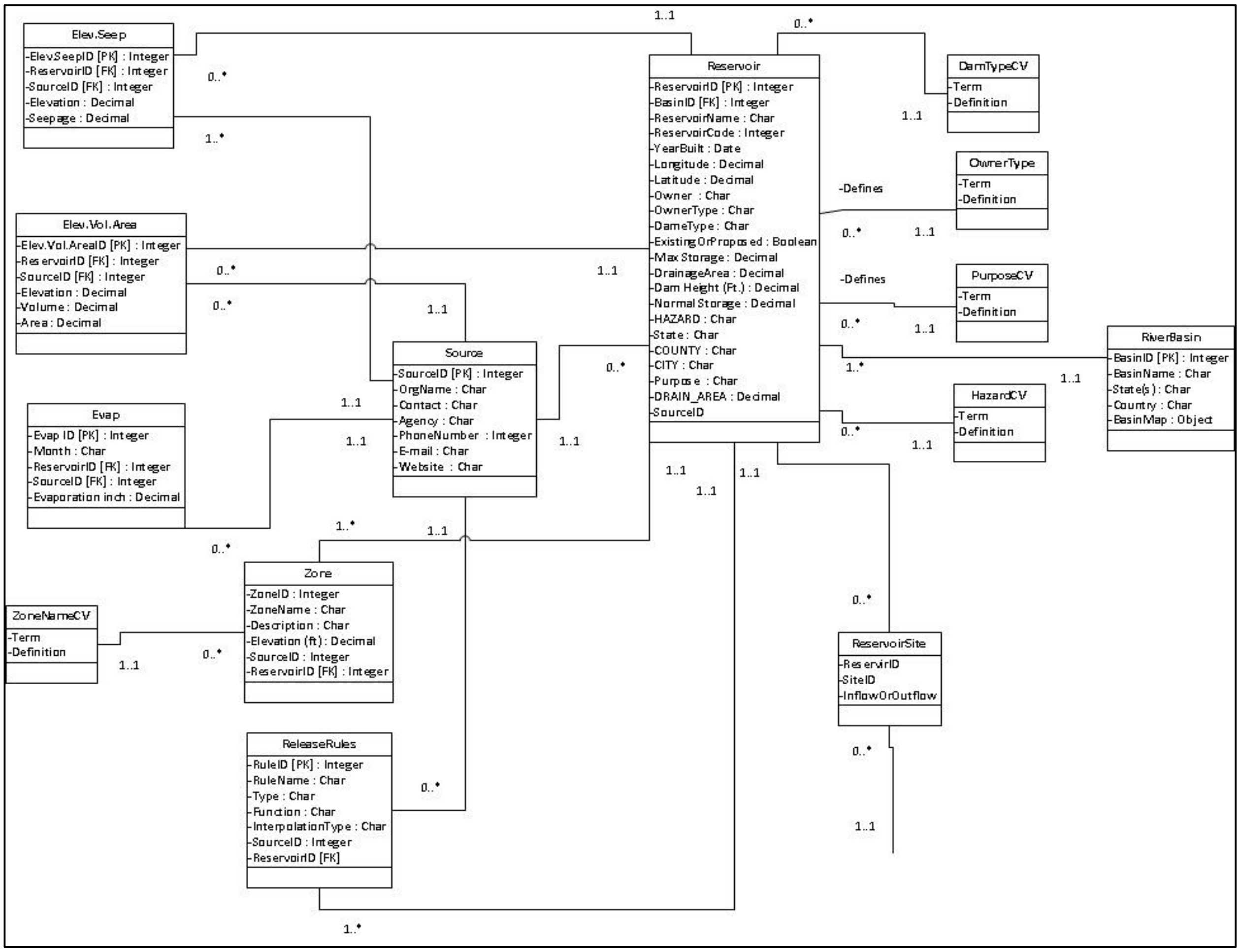

Figure (4): Designed reservoir data model

4. Results

We successfully managed to combine the current water management data model with our new reservoir data model. The new reservoir addition to the current model filled a huge void that separated these two fields of study. Previously, scientists who studied rivers kept their data about rivers, and scientists who studied reservoirs had their information about reservoirs. Rivers and reservoirs are connected, so why wasn't the gathered scientific data connected as well?

First we created a new database to record and store scientifically gathered information about reservoirs across the United States. This information covered everything from reservoir name, volume, and seepage level, to information about the source that gathered the data (see Figure 4). With our created database, reservoir scientists now have a place to store and retrieve data about reservoirs. Also, while creating the reservoir database, we established controlled vocabulary (see figure 5). This made the data clear and consistent. This consistency is crucial for future data gathered by hundreds of scientists across the country.

Owner Type	Definition
un	The owner type is unknown.
F	The dam is owned by a Federal agency.
L	The dam is owned by a County, City, Regional, or other similar local government or government agency.
P	The dam is owned by an individual or individuals, or by a private company.
S	The dam is owned by a State or by a State agency.
U	The dam is owned by a public utility.

Figure (5): Example of reservoir related controlled vocabulary

The next step of our project was tying our new database to the existing water management database (see figure 6). By combining the two we created a large, unified, searchable database, which will help connect the two fields of study. The database connection allows scientists to document and analyze all the information together. For example, we created a view that shows the temperature of the water flowing from the rivers into a reservoir and the name of the reservoir it is flowing into. Our new database will better help scientists analyze and answer questions about water usage, water loss, inflows and outflows, and hazard levels, as well as many other questions.

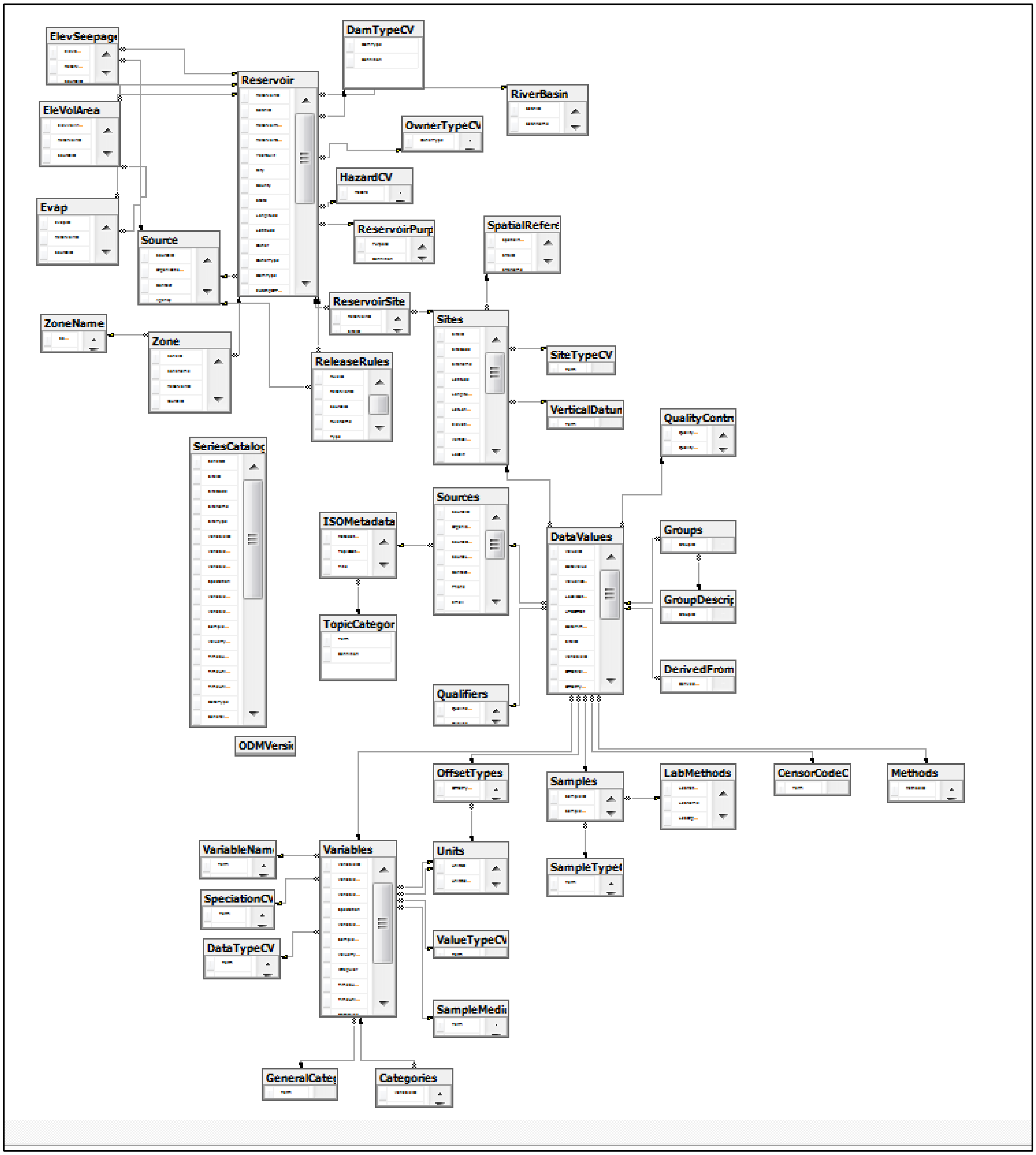

Figure (6): Designed reservoir data model

Once the database was completely finished and populated, we ran queries and created views to demonstrate what data scientist might want to see from our database. The results can be found below.

Example scientific questions that our database answers

- Which reservoirs could cause a major loss of human life if they failed?

```
SELECT r.ReservoirName, r.City, r.[State], r.County, h.Hazard, h.[Definition]
    FROM Reservoir AS R
        JOIN HazardCV AS H
        ON r.Hazard = h.Hazard
    WHERE h.Hazard='H';
```

Answer:

	ReservoirName	City	State	County	Hazard	Definition
1	BIG SKY DAM	WELLSVILLE	UT	CACHE	H	The potential hazard is high. A dam where failure or misoperation will probably cause loss of human life
2	NEW LONDON		UT	CACHE	H	The potential hazard is high. A dam where failure or misoperation will probably cause loss of human life
3	HIGH FALLS	HYRUM	UT	CACHE	H	The potential hazard is high. A dam where failure or misoperation will probably cause loss of human life
4	HUNTINGTON NORTH WEST DIKE	HUNTINGTON	UT	EMERY	H	The potential hazard is high. A dam where failure or misoperation will probably cause loss of human life
5	JOES VALLEY	ORANGEVILLE	UT	EMERY	H	The potential hazard is high. A dam where failure or misoperation will probably cause loss of human life
6	HUNTINGTON NORTH	HUNTINGTON	UT	EMERY	H	The potential hazard is high. A dam where failure or misoperation will probably cause loss of human life
7	MILLER FLAT	HUNTINGTON	UT	EMERY	H	The potential hazard is high. A dam where failure or misoperation will probably cause loss of human life
8	CLEVELAND	HUNTINGTON	UT	EMERY	H	The potential hazard is high. A dam where failure or misoperation will probably cause loss of human life
9	HUNTINGTON NORTH EAST DIKE	HUNTINGTON	UT	EMERY	H	The potential hazard is high. A dam where failure or misoperation will probably cause loss of human life
10	UTAH POWER & LIGHT - ELECTRIC LAKE	HUNTINGTON	UT	EMERY	H	The potential hazard is high. A dam where failure or misoperation will probably cause loss of human life

- What are the temperatures of the water entering High Falls Reservoir?

```
SELECT r.ReservoirName, r.County, r.DamHight, rs.InflowOrOutflow, s.SiteName, v.VariableNam
FROM Reservoir AS r
    JOIN ReservoirSite as rs
    ON r.ReservoirID = rs.ReservoirID
        JOIN Sites as s
        ON rs. SiteID = s.SiteID
            JOIN DataValues AS dv
            ON s.SiteID = dv.SiteID
                JOIN Variables as V
                ON dv.VariableID = v.VariableID
                    JOIN Units as U
                    ON v.VariableUnitsID = u.UnitsID
WHERE r.ReservoirID=28;
```

Answer:

Results | Messages

	ReservoirName	County	DamHight	InflowOrOutflow	SiteName	VariableName	LocalDateTime	DataValue	UnitsName
1	HIGH FALLS	CACHE	18.0000000000000	Inflow	Little Bear River near Wellsville, Utah	Temperature	2010-11-18 14:30:00.000	5.61	degree celsius
2	HIGH FALLS	CACHE	18.0000000000000	Inflow	Little Bear River near Wellsville, Utah	Temperature	2010-11-18 15:00:00.000	5.676667	degree celsius
3	HIGH FALLS	CACHE	18.0000000000000	Inflow	Little Bear River near Wellsville, Utah	Temperature	2010-11-18 15:30:00.000	5.73	degree celsius
4	HIGH FALLS	CACHE	18.0000000000000	Inflow	Little Bear River near Wellsville, Utah	Temperature	2010-11-18 16:00:00.000	5.78	degree celsius
5	HIGH FALLS	CACHE	18.0000000000000	Inflow	Little Bear River near Wellsville, Utah	Temperature	2010-11-18 16:30:00.000	5.825	degree celsius
6	HIGH FALLS	CACHE	18.0000000000000	Inflow	Little Bear River near Wellsville, Utah	Temperature	2010-11-18 17:00:00.000	5.858334	degree celsius
7	HIGH FALLS	CACHE	18.0000000000000	Inflow	Little Bear River near Wellsville, Utah	Temperature	2010-11-18 17:30:00.000	5.869999	degree celsius
8	HIGH FALLS	CACHE	18.0000000000000	Inflow	Little Bear River near Wellsville, Utah	Temperature	2010-11-18 18:00:00.000	5.869999	degree celsius
9	HIGH FALLS	CACHE	18.0000000000000	Inflow	Little Bear River near Wellsville, Utah	Temperature	2010-11-18 18:30:00.000	5.869999	degree celsius
10	HIGH FALLS	CACHE	18.0000000000000	Inflow	Little Bear River near Wellsville, Utah	Temperature	2010-11-18 19:00:00.000	5.868333	degree celsius
11	HIGH FALLS	CACHE	18.0000000000000	Inflow	Little Bear River near Wellsville, Utah	Temperature	2010-11-18 19:30:00.000	5.86	degree celsius
12	HIGH FALLS	CACHE	18.0000000000000	Inflow	Little Bear River near Wellsville, Utah	Temperature	2010-11-18 20:00:00.000	5.848333	degree celsius
13	HIGH FALLS	CACHE	18.0000000000000	Inflow	Little Bear River near Wellsville, Utah	Temperature	2010-11-18 20:30:00.000	5.828333	degree celsius
14	HIGH FALLS	CACHE	18.0000000000000	Inflow	Little Bear River near Wellsville, Utah	Temperature	2010-11-18 21:00:00.000	5.81	degree celsius

Query executed successfully. | (local)\SQLExpress (10.50 RTM) | AdeAbdallah\AdelMAbdal... | ODM | 00:00:00 | 8678 rows

- What is the volume and area of the reservoir at different elevations?

```sql
SELECT r.ReservoirName, r.County, s.OrganizationName, el.Elevation, el.Volume, el.Area
FROM Reservoir as R
    JOIN Source AS S
    ON r.SourceID = s.SourceID
        JOIN EleVolArea as el
        ON s.SourceID = el.SourceID
WHERE r.ReservoirName='High Falls';
```

Answer:

Results | Messages

	ReservoirName	County	OrganizationName	Elevation	Volume	Area
1	HIGH FALLS	CACHE	Utah Division of Water Resources	4590.0000000000000	0.0000000000000	1.0000000000000
2	HIGH FALLS	CACHE	Utah Division of Water Resources	4600.0000000000000	130.0000000000000	26.0000000000000
3	HIGH FALLS	CACHE	Utah Division of Water Resources	4610.0000000000000	649.0000000000000	76.0000000000000
4	HIGH FALLS	CACHE	Utah Division of Water Resources	4620.0000000000000	1739.0000000000000	142.0000000000000
5	HIGH FALLS	CACHE	Utah Division of Water Resources	4630.0000000000000	3456.0000000000000	205.0000000000000
6	HIGH FALLS	CACHE	Utah Division of Water Resources	4640.0000000000000	5937.0000000000000	291.0000000000000
7	HIGH FALLS	CACHE	Utah Division of Water Resources	4650.0000000000000	9236.0000000000000	367.0000000000000
8	HIGH FALLS	CACHE	Utah Division of Water Resources	4660.0000000000000	13206.0000000000000	427.0000000000000
9	HIGH FALLS	CACHE	Utah Division of Water Resources	4670.0000000000000	17721.0000000000000	473.0000000000000
10	HIGH FALLS	CACHE	Utah Division of Water Resources	4672.0000000000000	18684.0000000000000	479.0000000000000
11	HIGH FALLS	CACHE	Utah Division of Water Resources	4680.0000000000000	22600.0000000000000	516.0000000000000
12	HIGH FALLS	CACHE	Utah Division of Water Resources	4690.0000000000000	28100.0000000000000	572.0000000000000

Query executed successfully. | (local)\SQLExpress (10.50 RTM) | AdeAbdallah\AdelMAbdal... | ODM | 00:00:00 | 18 rows

- What is the evaporation at High Falls Reservoir throughout the year?

```
SELECT r.ReservoirName, r.County, s.OrganizationName, ev.EvapINCH, ev.[Month]
FROM Reservoir as R
    JOIN Source AS S
    ON r.SourceID = s.SourceID
        JOIN Evap as ev
        ON s.SourceID = ev.SourceID
WHERE r.ReservoirName='High Falls';
```

Answer:

Results | Messages

	ReservoirName	County	OrganizationName	EvapINCH	Month
1	HIGH FALLS	CACHE	Utah Division of Water Resources	0.2700000000000	2012-10-01
2	HIGH FALLS	CACHE	Utah Division of Water Resources	0.0100000000000	2012-11-01
3	HIGH FALLS	CACHE	Utah Division of Water Resources	0.0000000000000	2012-12-01
4	HIGH FALLS	CACHE	Utah Division of Water Resources	0.0000000000000	2012-01-01
5	HIGH FALLS	CACHE	Utah Division of Water Resources	0.0000000000000	2012-02-01
6	HIGH FALLS	CACHE	Utah Division of Water Resources	0.0000000000000	2012-03-01
7	HIGH FALLS	CACHE	Utah Division of Water Resources	0.1000000000000	2012-04-01
8	HIGH FALLS	CACHE	Utah Division of Water Resources	0.4400000000000	2012-05-01
9	HIGH FALLS	CACHE	Utah Division of Water Resources	0.5100000000000	2012-06-01
10	HIGH FALLS	CACHE	Utah Division of Water Resources	0.5600000000000	2012-07-01
11	HIGH FALLS	CACHE	Utah Division of Water Resources	0.5000000000000	2012-08-01
12	HIGH FALLS	CACHE	Utah Division of Water Resources	0.3200000000000	2012-09-01

Query executed... | (local)\SQLExpress (10.50 RTM) | AdeAbdallah\AdelMAbdal... | ODM | 00:00:00 | 12 rows

- Create view of all temperature values in High Falls reservoir

```sql
GO
CREATE VIEW [ReservoirTemp] AS
SELECT Reservoir.ReservoirName, Reservoir.County, Reservoir.DamHight, ReservoirSite.InflowO
FROM Reservoir
    JOIN ReservoirSite
    ON Reservoir.ReservoirID = ReservoirSite.ReservoirID
        JOIN Sites
        ON ReservoirSite.SiteID = Sites.SiteID
            JOIN DataValues
            ON Sites.SiteID = DataValues.SiteID
                JOIN Variables
                ON DataValues.VariableID = Variables.VariableID
                    JOIN Units
                    ON Variables.VariableUnitsID = Units.UnitsID
WHERE Reservoir.ReservoirID=28;

SELECT *
FROM ReservoirTemp;
```

Answer:

Results | Messages

	ReservoirName	County	DamHight	InflowOrOutflow	SiteName	VariableName	LocalDateTime	DataValue	UnitsName
1	HIGH FALLS	CACHE	18.0000000000000	Inflow	Little Bear River near Wellsville, Utah	Temperature	2010-11-18 14:30:00.000	5.61	degree celsius
2	HIGH FALLS	CACHE	18.0000000000000	Inflow	Little Bear River near Wellsville, Utah	Temperature	2010-11-18 15:00:00.000	5.676667	degree celsius
3	HIGH FALLS	CACHE	18.0000000000000	Inflow	Little Bear River near Wellsville, Utah	Temperature	2010-11-18 15:30:00.000	5.73	degree celsius
4	HIGH FALLS	CACHE	18.0000000000000	Inflow	Little Bear River near Wellsville, Utah	Temperature	2010-11-18 16:00:00.000	5.78	degree celsius
5	HIGH FALLS	CACHE	18.0000000000000	Inflow	Little Bear River near Wellsville, Utah	Temperature	2010-11-18 16:30:00.000	5.825	degree celsius
6	HIGH FALLS	CACHE	18.0000000000000	Inflow	Little Bear River near Wellsville, Utah	Temperature	2010-11-18 17:00:00.000	5.858334	degree celsius
7	HIGH FALLS	CACHE	18.0000000000000	Inflow	Little Bear River near Wellsville, Utah	Temperature	2010-11-18 17:30:00.000	5.869999	degree celsius
8	HIGH FALLS	CACHE	18.0000000000000	Inflow	Little Bear River near Wellsville, Utah	Temperature	2010-11-18 18:00:00.000	5.869999	degree celsius
9	HIGH FALLS	CACHE	18.0000000000000	Inflow	Little Bear River near Wellsville, Utah	Temperature	2010-11-18 18:30:00.000	5.869999	degree celsius
10	HIGH FALLS	CACHE	18.0000000000000	Inflow	Little Bear River near Wellsville, Utah	Temperature	2010-11-18 19:00:00.000	5.868333	degree celsius
11	HIGH FALLS	CACHE	18.0000000000000	Inflow	Little Bear River near Wellsville, Utah	Temperature	2010-11-18 19:30:00.000	5.86	degree celsius
12	HIGH FALLS	CACHE	18.0000000000000	Inflow	Little Bear River near Wellsville, Utah	Temperature	2010-11-18 20:00:00.000	5.848333	degree celsius

Query executed successfully. | (local)\SQLExpress (10.50 RTM) | AdeAbdallah\AdelMAbdal... | master | 00:00:00 | 8678 rows

5. Future developments

The scope of the current database is limited to working only within a relatively small region, such as the state of Utah. Without some improvements and reengineering, it most likely would not work on a larger scale because of some of the assumptions that we mentioned previously. In addition, in order to have this model be implemented across the country or across the world, much more time and work would be needed than the few months that our small group had.

Adel and the water lab will take this database and continue to refine and improve it for their needs. One feature that would be very useful to end users is creating a front-end website for easier access. We also wanted to create drop downs on this front-end site for the controlled vocabulary and make the user input process very streamlined. Our hope is that, with some modifications and improvements, this reservoir data model can become a model for the nation.

We attempted to use Matlab in order to import large datasets automatically, but we couldn't get it to work properly. Below is the code we used:

```
% Load River Basin Data using ODBC connection
conn = database('ODM','sa','amabdallah')

colname={'BasinID','BasinName','State','County'};

exData={93,'Bear River','UT','Cache'};

insert(conn,'RiverBasin',colname,exData);

close(conn);
```

6. What we would do better next time

One of the biggest challenges we ran in to was data importation. If we did this project over, we would better plan out data importation techniques. We imported a lot of data from Excel and ran into some common errors a few times that could have been remedied if we had better planned out our data types.

Another problem that we had was understanding the field of study to the extent needed before starting. The specific vocabulary used inside the field of water science is somewhat difficult to understand for the common person. Some of the relationships between entities are not as clear as they are in a typical transactional type database model, so a lot of thinking and explanation from our expert was necessary before we understood how and why the data is used a particular way. This even caused us to design our ER diagram incorrectly a few times before we really understood the interactions properly. After we had a clear understanding of the entities and their relationships, we were able to create a successful data model and eventually a successful database.

Lastly, we would try to use better tools for collaboration. It was difficult for us to collaborate outside of Google Docs or Canvas because we didn't have the ability to log in to Adel's office computer (where the database was housed) because of some USU security issues. If we could all remotely access the database, it would have been easier to work on it on our own time.

7. Conclusion

In this project we developed a relational data model that stores and organizes consistent data about water reservoirs in the United States. In addition, the relational data model incorporates an existing water database that measures data about inflow and outflow water from or into reservoirs. We applied concepts we learned in class to create a working data model and database. We first developed the conceptual data model using ER diagramming techniques. We then implemented the conceptual model to produce the physical database. We populated the database with Utah test data and then tested the model using some example queries. The developed reservoir data model enables scientists to answer critical and important scientific questions about both reservoirs and water measuring sites at the same time across many boundaries. Future work can test the model for large national datasets.

8. References

- Adel Abdallah, 2012. "Design Reservoir Conceptual Data Model" class term project: Hydroinformatics Fall 2012 Utah State University
- US Bureau of Reclamation (US BOR) http://www.usbr.gov/gp/lakes_reservoirs/index.htm accessed Jan 2013
- Horsburgh, J. S., D. G. Tarboton, D. R. Maidment and I. Zaslavsky, (2008), A Relational Model for Environmental and Water Resources Data, Water Resources Research, 44: W05406, doi:10.1029/2007WR006392.