ArcGIS Providing Opportunities for WKU's Kentucky Mesonet

Kevin B. Cary, GISP

Department of Geography and Geology
Western Kentucky University
1906 College Heights Blvd #31066
Bowling Green, KY 42101–1066

June 27, 2014

2014 EdUC Abstract

Western Kentucky University's (WKU) Kentucky Mesonet is an emerging network of weather stations that began as early as 2006 and developed by the Kentucky Climate Center at Western Kentucky University. Data is centralized in an ArcSDE geodatabase for students and faculty. ArcGIS for Server provides map services utilizing the Mesonet ArcSDE geodatabase for web map applications such as ArcGIS API for JavaScript. This introduces a working enterprise GIS environment for WKU's GIScience majors from the front–end to the back–end.

Introduction

In 2001, WKU's Department of Geography and Geology acquired an Esri University Site License as part of a statewide license program with other public universities in Kentucky. The site license includes a suite of Esri products along with other benefits such as Esri online courses and student software. That same year, WKU's Department of Geography and Geology offered Kentucky's first undergraduate Certificate in GIS. Soon after, the department would offer Kentucky's first graduate certificate in GIS, minor in GIS, and in 2009 Kentucky's first undergraduate major in Geographic Information Science. Up until 2009, students pursing GIS as a "major" would major in geography with a concentration in GIS. In addition, the department has been offering a concentration in GIS in its Master of Geoscience program since 2001.

WKU's Geographic Information Science (GIScience) program prepares students to make an easy transition into today's GIS enterprise environment. Courses in program major include *Internet GIS*, *GIS Databases*, and *GIS Programming*. The program major also focuses on spatial analysis by requiring GIScience majors to enroll in courses in an introductory course in mathematical statistics and two geography courses in *Spatial Data Analysis* and *Advance Spatial Analysis*, both geography courses are statistical–based courses requiring a prerequisite of an introductory level course in mathematical statistics and a course in GIS. WKU's major in GIScience program was developed with tomorrow's enterprise GIS in mind and by focusing on the general definition of GIS including storage and retrieval of geospatial data. It's inevitable that tomorrow's jobs as a GIS profesional will be those jobs in an enterprise GIS environment. WKU's GIScience program also prepares students wanting to pursue a graduate degree in GIScience, geoscience, geography, and in the realm of business analytics (Cary, 2014).

The Kentucky Mesonet is housed in the Department of Geography and Geology at WKU along side the Kentucky Climate Center. The Mesonet's first weather station was establised in 2007 and today it has grown to a network of over sixty weather stations across Kentucky (Kentucky Mesonet, 2014). The previous year in 2006, WKU's Department of Geography and Geology began offering Kentucky's first undergraduate major in meteorology . Before 2006, student's interested in meteorology and climatology would major in geography with a concentration in weather and climate (WKU Meteorology, 2013).

Server technology is becoming more main stream and widely available at an affordable cost than ever before and this is good news in higher education for these opportunities provide a hands–on experience. In 2001, WKU's Center for GIS in the Department of Geography and Geology began with one server and now has over five GIS servers dedicated to research, service and teaching. A microcosm of modern day enterprise GIS has been established and WKU's Center for GIS (primarily for the GIScience program) has been pivotal in promoting WKU's Facilities Management and WKU's Department of Planning, Design, and Construction. WKU's GIScience students have served and are still serving as GIS interns with the previous departments mention. Three of which are now WKU staff marking the first staff member in WKU's history with GIS included in the job title, which was in 2012.

To some degree, implementing an enterprise GIS in the past has been arduous due to the lack of literature available for GIS professionals, especially on the "how" with code, connections, and expensive hardware. An extensive amount of knowledge in information technology (IT) is required and with GIS professionals traditionally deriving from a multitude of disciplines then some (if not most) may not have the IT expertise to directly begin establishing an enterprise GIS (URISA Salary Survey, 2011). Today, there is an increasing amount of resources on enterprise GIS architecture; however, getting it connected to provide real–time data to various GIS users or to implement a multiuser editing environment can be limiting for some without consulting directly with GIS or IT professional services or seeking additional technical support from Esri (if using Esri products) with a series of "phone tag" and allocated time.

"Pushing real–time data into an ArcSDE geodatabase for immediate consumption by end users can be challenging. However, with the advent of Microsoft's SQL Server 2008 and Esri support for built–in SQL Server 2008 spatial data type, real–time updating of ArcSDE feature classes in SQL Server 2008 has gotten easier" (Cooper, 2012). "ESRI has developed a methodology that makes near real–time data and the results of analysis available through all ArcGIS clients. This methodology uses simple batch scripting with a handful of public domain command line utilities to download and preprocessed the feed data. Scripts can run at varying intervals...depending upon the application" (Burke and Dodd, 2009).

This paper will discuss downloading data from Kentucky's Mesonet server in comma delimited format for updating attributes in the mesonet's ArcSDE geodatbase and distributing that data as a feature service via ArcGIS for Server for web map applications and GIS clients. It will focus primarily on software used and code for scripts, i.e., "the meat and potatoes". The below flowchart illustrates the connections among hardware (Figure 1).

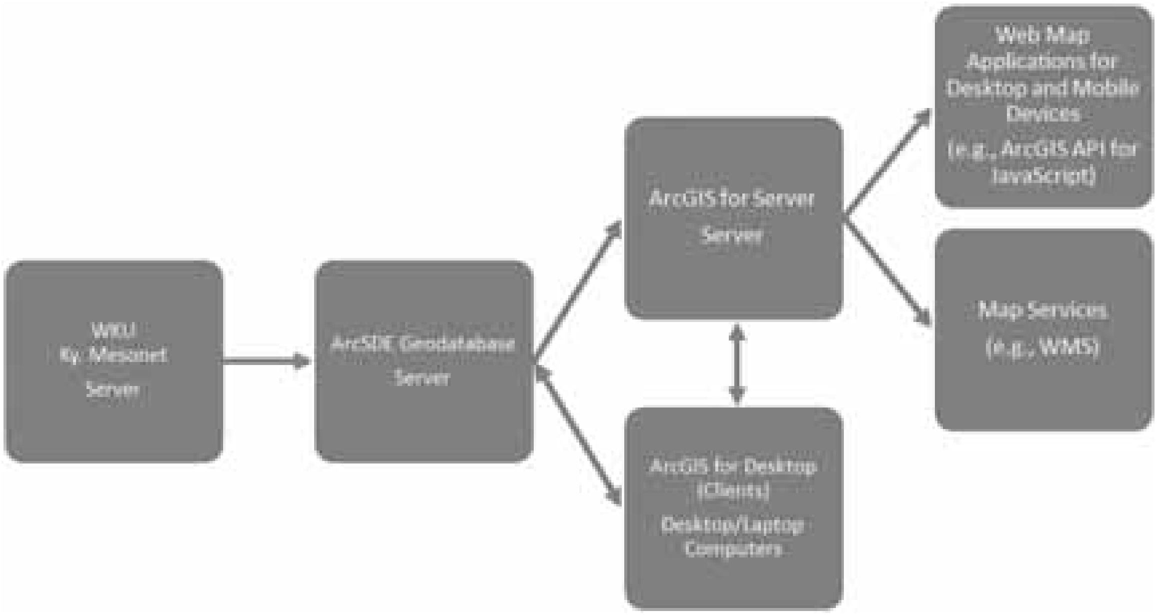

Figure 1: Connections and flow among servers, desktops, and mobile devices.

Software

Below is a list of software used for this project. It is important to note that software below is installed on a server running a Windows 2008 operating system. Also, it is listed in the order of when it is first used or implemented, not in alphabetical order.

Notepad++:

An open source program used as a code editor for this project is Notepad ++. (For Windows users, it's similar to NotePad and supports various languages for scripting.) A nice advantage of Notepad++ is that every line (of code) is numbered and commands within the script are colored coded making it easier to bind like blocks of code or to visually seperate out commands, inputs, text and other parameters. For this paper, NotePad++ is used for writing and editing both Python and MS SQL scripts and to further manipulate ArcGIS API for JavaScript web map viewer. Sometimes a user can execute code from within Notepad ++ like directly into an Internet browser (Ho, 2011).

Python:

A scripting language used is Python, which is an open source programming language. Entry level programmers will not notice that Python uses fewer lines of code than other programs like Visual Basic that came with ArcGIS products prior to version 10. For example, variables are not required to be declared, which is a very nice luxury that will go unnoticed to those new to programming or scripting; however Python, is case sensitive. Python is used as the code executed from within a batch file for downloading a comma delimited text file from WKU's Kentucky Mesonet. Python code will also execute an MS SQL script (Python Software Foundation, 2014).

Microsoft (MS) SQL:

The relational database residing on the "back–end" on the server for this project is Microsoft SQL 2008 standard edition. MS SQL is important for running an ArcSDE geodatabase because it allows geodatabases to be written while in use by other ArcGIS products and its applications (Microsoft SQL Server, 2014). Without it, the geodatabase is locked preventing other users from manipulating it. Furthermore, ArcGIS for Server will not reflect any updates in map layers in the standard map service. ArcGIS for Server must have a feature service with an ArcSDE geodatabase feature class in order for any updates to immediately reflected in a service provided by ArcGIS for Server. An alternative relational database management system choice besides MS SQL is the free open source PostgreSQL.

Windows Task Scheduler:

Windows Server operating systems can automatically execute programs or launch batch files on a scheduler. Windows Task Scheduler is used to run python code every fifteen minutes to download data from Kentucky's Mesonet. The scheduler is part of an administrator privilege that allows programs to be executed on a schedule (Ho, 2011).

Esri Products (ArcSDE, ArcGIS for Desktop, and ArcGIS for Server):

The various Esri products used for editing, publishing and distributing feature map services are ArcSDE, ArcGIS for Desktop 10.1 and ArcGIS for Server 10.1. ArcSDE is the spatial database engine. It "...manages spatial data in a relational database management system (RDBMS) and enables it to be accessed by ArcGIS clients" (ArcSDE, 2014). An advantage to ArcSDE clients is that it enables a multiuser editing environment on the same geodatabase while centralizing the data in one physical (or virtual) location. ArcGIS for Desktop is a professional grade desktop GIS software. It consist of three main applications: ArcCatalog, ArcMap, and ArcToolbox. (The following description for each application is in regards to this project.) ArcCatalag is used for data management of feature classes and geodatbases as well as administration of ArcSDE geodatabases. ArcMap is used primarily for manipulating map documents and publishing map documents to ArcGIS for Server 10.1. ArcToolbox is used for ArcSDE geodatabase administration and management (ArcGIS for Desktop, 2014). ArcGIS for Server 10.1 is the plateform used for hosting feature services and providing services to various clients and web map applications (ArcGIS for Server, 2014).

Internet Information Services (IIS):

IIS is used for hosting websites such as ArcGIS for Server Manager as well as web maps. IIS comes with any Windows operating system, but the edition of the operating system will limit some capabilities. Those limiting capabilities may be found on Esri's website on their system requirements web page for ArcGIS for Server (Internet Information Services, 2014).

Web Map Viewers:

There are two web map viewers readily available to download for free with Esri's product map services as well as sample scripts for JavaScript web maps: ArcGIS Viewer for Flex, ArcGIS Viewer for Silverlight, and ArcGIS API for JavaScript. It's worthy to note that Flex is by Adobe, Silverlight is by Microsoft, and JavaScript is open source. For this project, ArcGIS API for JavaScript is used as the web map application. It was choosen due to the fact that a web map created with JavaScript can run on most smart phone web browsers (ArcGIS API for JavaScript, 2014).

Putting It Together

The first task for this project before anything else is creating an ArcSDE geodatabase of all existing WKU Ky. Mesonet weather stations with the following attributes: station identity, elevation, city, county, UTC time, local time, air temperature in Celsius, relative humidity in percent, dew point temperature, wind speed in miles per hour, wind direction in degrees, max wind speed for that day, incoming solar radiation, and daily total of precipitation in inches. The static attributes are station identity, elevation, city, and county. The remaining attributes will be updated every fifteen minutes and all attribute data are included from the retrieved file downloaded from the WKU Ky. Mesonet server. The UTC time is desirable in Kentucky since the local time attribute will record time in both the eastern and central time because Kentucky is in two time zones; although, theoretically speaking, all of Kentucky should be in one time zone, the central time zone.

The next task is updating the weather attributes every fifteen minutes. Fifteen minute intervals is chosen because the WKU Ky Mesonet weather stations record weather data every fifteen minutes except during severe weather events when data is being recorded in minute intervals. First, a Python script had to be developed in order to have the server with MS SQL call the WKU Ky. Mesonet server to download weather station data in comma delimited format. (Keep in mind that MS SQL and ArcGIS for Server reside on separate servers as seen in figure 1.) The code below is used to do that task.

```
import urllib, os

#Download file from URL.  (This is a comment.)
urllib.retrieve ("http://IPAddress_here/file.csv",
     "drive:\Path_of_CSV_File_Placement\Name_It.csv")

#Access database and run MS SQL script.
os.system ("sqlcmd -U SDE_Admin_Name -P Password -I
     drive:\Path_of_SQL_Script\Name.sql")
```

In the Python code above, it also calls upon an MS SQL script as noted by the .sql extension. The MS SQL script simply joins the downloaded comma delimited data to the attribute table of the ArcSDE geodatabase residing as a database in MS SQL. Then it replaces the attributes values (or old attribute values) in the attribute table of the WKU Ky. weather station ArcSDE geodatabase with the up–to–date values recently downloaded. The key field/attribute (or unique identifier) for joining the tables is the station identity attribute. Below is that code in its entirety, but separated out by “assignments”.

First make a connection to the database and then create a table with attribute names and data types with the appropriate length as a table in an MS SQL database.

```
-- Make a connection.
USE StationsLive  -- StationsLive is name of ArcSDE database in MS SQL.
Go

-- Create Table
CREATE TABLE CSVMesoLive
(NetSiteAbb VARCHAR(5),
UTCTime VARCHAR(20),
LocalTime VARCHAR(20),
ATC DECIMAL(6,2),
RH DECIMAL(6,1),
DP DECIMAL(6,1),
WindSp DECIMAL(6,1),
WindDir DECIMAL(6,1),
WindSpMax DECIMAL(6,1),
SRad DECIMAL(6,1),
DTPrecip DECIMAL(6,1))
Go
```

The script continues with the next block of code. The following imports the data using the existing comma delimited file (.csv).

```
-- Import data using a CSV file.
BULK
INSERT CSVMesoLive
FROM 'Drive:\Path\GISMesonet.csv'
WITH
(
FIELDTERMINATOR = ',',
ROWTERMINATOR = '\n'
)
Go
```

The following block of code isn’t necessary. It was used as a way to validate reading the data by displaying the data in DOS. This code may be omitted but desirable when troubleshooting.

```
-- Check the content of the table.
SELECT *
FROM CSVMesoLive
Go
```

Now to update the attributes in the ArcSDE geodatabase table.

```
-- Update attributes in SDE geodatabase attribute table.
UPDATE SDE.STATIONSLIVE
SET UTCTime = (SELECT CSVMesoLive.UTCTime
                      FROM CSVMesoLive
                      WHERE CSVMesoLive.NetSiteAbb = StationsLive.NetSiteAbb)
WHERE EXISTS (SELECT CSVMesoLive.UTCTime
                      FROM CSVMesoLive
                      WHERE CSVMesoLive.NetSiteAbb = StationsLive.NetSiteAbb)

UPDATE SDE.STATIONSLIVE
SET LocalTime = (SELECT CSVMesoLive.LocalTime
                      FROM CSVMesoLive
                      WHERE CSVMesoLive.NetSiteAbb = StationsLive.NetSiteAbb)
WHERE EXISTS (SELECT CSVMesoLive.LocalTime
                      FROM CSVMesoLive
                      WHERE CSVMesoLive.NetSiteAbb = StationsLive.NetSiteAbb)
```

```
UPDATE SDE.STATIONSLIVE
SET ATC = (SELECT CSVMesoLive.ATC
                            FROM CSVMesoLive
                            WHERE CSVMesoLive.NetSiteAbb = StationsLive.NetSiteAbb)
WHERE EXISTS (SELECT CSVMesoLive.ATC
                            FROM CSVMesoLive
                            WHERE CSVMesoLive.NetSiteAbb = StationsLive.NetSiteAbb)
UPDATE SDE.STATIONSLIVE
SET RH = (SELECT CSVMesoLive.RH
                            FROM CSVMesoLive
                            WHERE CSVMesoLive.NetSiteAbb = StationsLive.NetSiteAbb)
WHERE EXISTS (SELECT CSVMesoLive.RH
                            FROM CSVMesoLive
                            WHERE CSVMesoLive.NetSiteAbb = StationsLive.NetSiteAbb)

UPDATE SDE.STATIONSLIVE
SET DP = (SELECT CSVMesoLive.DP
                            FROM CSVMesoLive
                            WHERE CSVMesoLive.NetSiteAbb = StationsLive.NetSiteAbb)
WHERE EXISTS (SELECT CSVMesoLive.DP
                            FROM CSVMesoLive
                            WHERE CSVMesoLive.NetSiteAbb = StationsLive.NetSiteAbb)

UPDATE SDE.STATIONSLIVE
SET WindSp = (SELECT CSVMesoLive.WindSp
                            FROM CSVMesoLive
                            WHERE CSVMesoLive.NetSiteAbb = StationsLive.NetSiteAbb)
WHERE EXISTS (SELECT CSVMesoLive.WindSp
                            FROM CSVMesoLive
                            WHERE CSVMesoLive.NetSiteAbb = StationsLive.NetSiteAbb)

UPDATE SDE.STATIONSLIVE
SET WindDir = (SELECT CSVMesoLive.WindDir
                            FROM CSVMesoLive
                            WHERE CSVMesoLive.NetSiteAbb = StationsLive.NetSiteAbb)
WHERE EXISTS (SELECT CSVMesoLive.WindDir
                            FROM CSVMesoLive
                            WHERE CSVMesoLive.NetSiteAbb = StationsLive.NetSiteAbb)

UPDATE SDE.STATIONSLIVE
SET WindSpMax = (SELECT CSVMesoLive.WindSpMax
                            FROM CSVMesoLive
                            WHERE CSVMesoLive.NetSiteAbb = StationsLive.NetSiteAbb)
WHERE EXISTS (SELECT CSVMesoLive.WindSpMax
                            FROM CSVMesoLive
                            WHERE CSVMesoLive.NetSiteAbb = StationsLive.NetSiteAbb)

UPDATE SDE.STATIONSLIVE
SET SRad = (SELECT CSVMesoLive.SRad
                            FROM CSVMesoLive
                            WHERE CSVMesoLive.NetSiteAbb = StationsLive.NetSiteAbb)
WHERE EXISTS (SELECT CSVMesoLive.SRad
                            FROM CSVMesoLive
                            WHERE CSVMesoLive.NetSiteAbb = StationsLive.NetSiteAbb)
```

```
UPDATE SDE.STATIONSLIVE
SET DTPrecip = (SELECT CSVMesoLive.DTPrecip
                        FROM CSVMesoLive
                        WHERE CSVMesoLive.NetSiteAbb = StationsLive.NetSiteAbb)
WHERE EXISTS (SELECT CSVMesoLive.DTPrecip
                        FROM CSVMesoLive
                        WHERE CSVMesoLive.NetSiteAbb = StationsLive.NetSiteAbb)
Go
```

The script ends with deleting the comma delimited file (.csv) from the database in MS SQL as seen in the next block of code.

```
-- Deletes the table from database.
DROP TABLE CSVMesoLive
GO
```

At this point, it may be time to configure Windows Task Scheduler, but it can wait until after the next step on creating the ArcSDE geodatabase. The Windows Task Scheduler will run a program on a schedule, but for this project it will launch a batch file (.bat) on a schedule. The Windows Task Scheduler was configured on the MS SQL server and is scheduled to launch the batch file every fifteen minutes. The batch file is what runs the Python code in which downloads the data and then runs the MS SQL script for joining the tables for re–assigning the values from the recently downloaded comma delimited file availalbe at the WKU Ky. Mesonet server to the WKU Ky. weather station ArcSDE geodatabase. Below are the contents of the .bat file. It consists of MS DOS commands in command prompt format.

```
REM REM is for commenting and it stands for REMarks.

H:
REM H: is to change drives where the .py resides.

cd \insert\path\of\python\script\here
REM cd is for changing directories where the .py resides to be launched.

python Name_of_Script.py
REM 'python' is the command to run python code in the .py file.
```

Finally, a feature service is to be set up in ArcGIS for Server. The standard "out–of–the–box" map service in ArcGIS for Server is static since it basically takes a snapshot of the map document and its contents. Therefore, it is important to establish a feature service so that WKU Ky. Mesonet service hosted by ArcGIS for Server reflects the updates automatically. The feature service also (like a map service) preserves the symbology in the map document created in ArcMap. ArcMap is used to publish the service to ArcGIS for Server. It is very important that the ArcGIS for Server administrator registers the ArcSDE geodatabase with ArcGIS for Server; otherwise, it will not work. More references, guidlines, and "how–to–do–so" for ArcGIS for Server can be found at the following website: `http://resources.arcgis.com/`.

Now that the feature service is available in ArcGIS for Server, the feature service can be used in any ArcGIS for Desktop application as a web map service and web map applications such as ArcGIS API for JavaScript. The WKU Ky. Mesonet demonstration web map version (Figure 2) is available at the following web address: `http://gisapps.wku.edu/wkymesonetgps.html`. It is also viewable on smart phones such as an iPhone and Android. Those viewing the application outside of Kentucky will noticed that the web map application (if location service is turned on) will zoom into the location of the where a user currently stands. As a result, a user may have to turn off the location service in the their Internet browser in order to pan over to Kentucky. In the web map application, additional layers are used such as Esri's base maps and the National Weather Service radar precipitation map layer. Symbology was designed in ArcMap and take note that the wind barbs reflect both wind speed and wind direction which is possible under the advance button in the symbology tab. The yellow "sunflower" reflects incoming solar radiation and precipitation is represented in colored graduated circles. Most obvious is the temperature labels in Fahrenheit, which is converted from Celsius in the map document created in ArcMap under the Labels tab in the Label Expression of the layer's properties. It's simple and demonstrates the web map application potential on both desktop and mobile applications. To view the web map's source or any web map's source code built with ArcGIS API for JavaScript, simply view the source code in the browser of choice. To view the source code in Internet Explorer, go to *View* in the main menu and select *Source*.

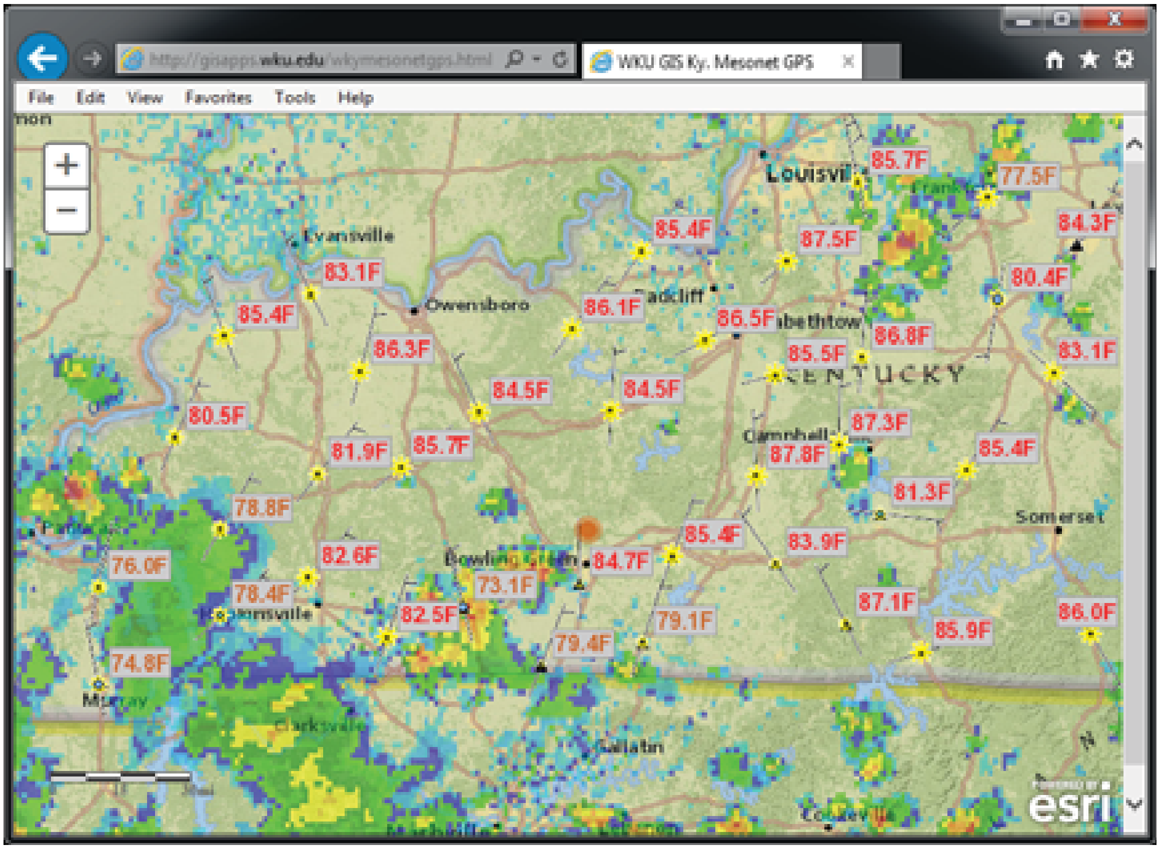

Figure 2: Web map of the demostration version of WKU Ky. Mesonet data.

Conclusions

For this project, it takes a person that is comfortable with today's computer technology and programming or a person that is an advanced GIS user with advanced IT skills and knowledge. Administrative privileges is a must for all applications involved including administrative privileges on any and all desktop computers and servers as well as the programs and applications involved in design and development of this project – that is essential! A basic desktop user of GIS may find their challenges and hang–ups but keep charging forward by troubleshooting to eventually arrive at the desired result.

There is a wealth of resources available on the Internet. Take advantage of the free tutorials available, especially on YouTube.com for MS SQL if not familiar with database management systems, even ArcGIS for Server has a tutorial. In this paper, a lot of the code has been figure out and it works; however, there may be better solutions and a more stable procedure for this scenario. This project is up and running and is still in operation today for over a year as a demonstration of services by WKU's Center for GIS and its possibilities. It is strongly suggested to use Microsoft" products, such as Internet Explorer, before migrating into "non–Microsoft" products, because Esri software seems to "work better" with Microsoft products. In other words, make it work first with Microsoft's Internet Explorer.

References

ArcGIS API for JavaScript (2014). https://developers.arcgis.com/javascript/. Date accessed: June 24, 2014.

ArcGIS for Desktop (2014). http://www.esri.com/software/arcgis/arcgis-for-desktop. Date accessed: June 23, 2014.

ArcGIS for Server (2014). http://www.esri.com/software/arcgis/arcgisserver/features. Date accessed: June 23, 2014.

ArcSDE (2014). http://www.esri.com/software/arcgis/arcsde. Date accessed: June 23, 2014.

Burke, D. and Dodd, P. (2009). Easy Access to Near Real Time Data. *ArcUser*, 9(2):36 – 38. Spring.

Cary, K. (2014). Western Kentucky University Geographic Information Science. http://www.wku.edu/gis/. Date accessed: June 25, 2014.

Cooper, C. (2012). Real-Time Updating of ArcSDE through SQL. *ArcUser*, 15(1):40 – 45. Winter.

Ho, D. (2011). Notepad++. http://notepad-plus-plus.org/. Date accessed: June 23, 2014.

Internet Information Services (2014). http://www.microsoft.com/web/platform/server.aspx. Date accessed: June 23, 2014.

Kentucky Mesonet (2014). http://www.kymesonet.org/. Date accessed: June 25, 2014.

Microsoft SQL Server (2014). http://www.microsoft.com/en-us/server-cloud/products/sql-server-editions/overview.aspx. Date accessed: June 23, 2014.

Python Software Foundation (2014). https://www.python.org/. Date accessed: June 23, 2014.

URISA Salary Survey (2011). *URISA Salary Survey for IT/GIS Professionals.* The Urban and Regional Information Systems Association, 701 Lee Street, Suite 680, Des Plaines, IL 60016. Date accessed: June 26, 2014.

WKU Meteorology (2013). http://www.wku.edu/meteorology/. Date accessed: June 25, 2014.