# Exercise 5. Automation using Geoprocessing and Python Programming
## GIS in Water Resources, Fall 2013
## Prepared by David G Tarboton and David R. Maidment

## Purpose

The purpose of this exercise is to illustrate the capabilities for automation of repetitive tasks and for the addition of new functionality by using programming in ArcGIS.   In this exercise, you will first examine GRACE satellite data that is published from UT Austin using ArcGIS server functionality.  This illustrates how organizations can publish their own data facilitating its use by others.  GRACE measures total water storage in the earth's surface and subsurface using gravity anomalies and will be used to examine recent droughts.  You will then examine data from the World Water Online services being established at UT Austin that integrate data from USGS, Kisters and Texas sources.  A key idea here is the integration of information from multiple sources.  The services based data infrastructure that is emerging is enabling more comprehensive examination of engineering and water resources problems through the combination of information from different sources.  GIS provides a vehicle for this integration.  You will then use a tool developed at UT Austin for the automated retrieval of North American Land Data Assimilation System (NLDAS) data.  NLDAS includes the NOAH land surface hydrology model that provides gridded estimates of the hydrologic water and energy balance through a combination of observations and modelling.  You will use NLDAS data to examine a watershed water balance.

## Learning objectives

- Examine drought in Texas and Utah using GRACE total water storage estimated from gravity anomaly data.
- Examine streamflow data from World Water Online Services.
- Use Python Programs to automate the retrieval and analysis of NLDAS data to examine the water balance of a watershed.

## Readings

- GRACE http://www.csr.utexas.edu/grace/
- NLDAS http://ldas.gsfc.nasa.gov/index.php

## Computer and Data Requirements

To carry out this exercise, you need to have a computer which runs ArcGIS 10.2 and includes the Spatial Analyst extension.  The necessary data, LDAS Tools are Python Programs are provided in Ex5.zip (http://www.neng.usu.edu/cee/faculty/dtarb/giswr/2013/Ex5.zip).

# Before we start

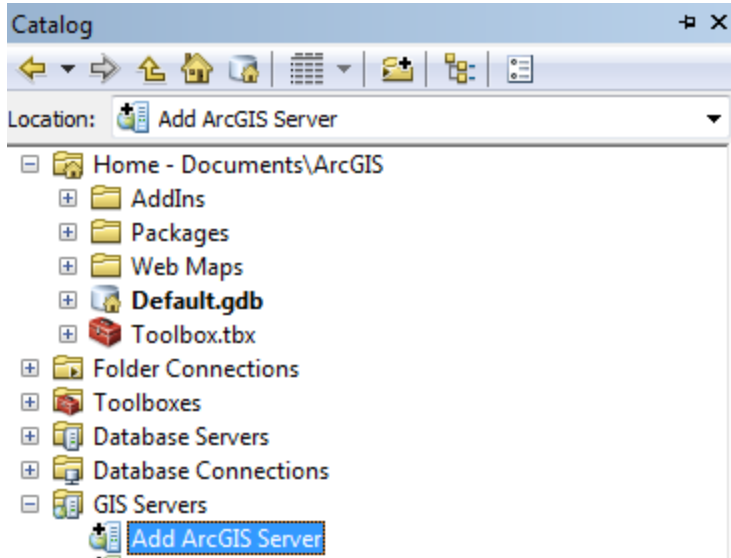Select **Customize** → **Extensions**
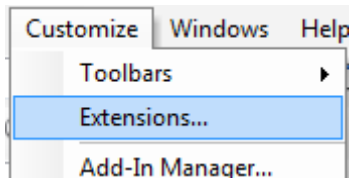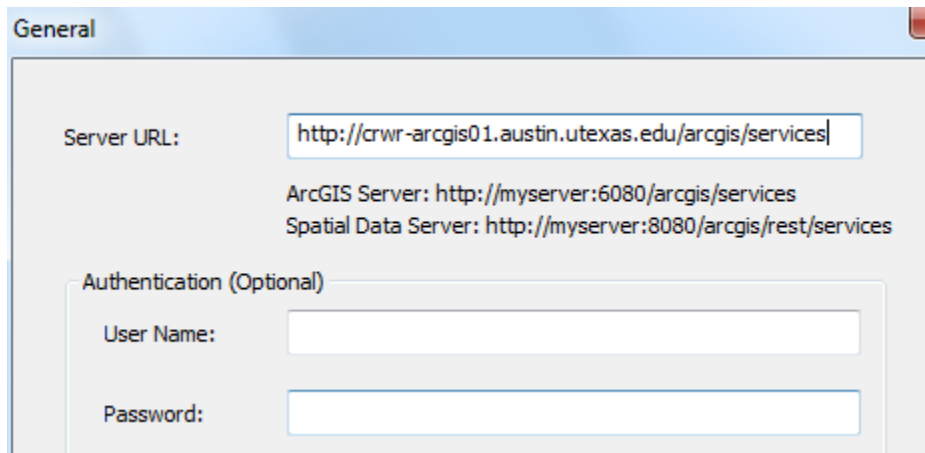


Select the Spatial Analyst extension.  We will be using this during the exercise.
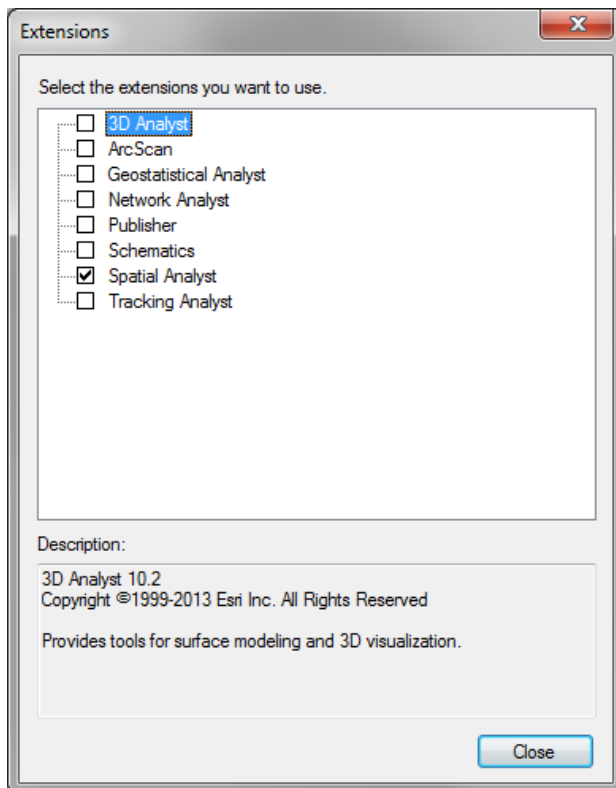
# Total Water Storage from the GRACE Mission

Open ArcGIS Desktop and using the Catalog tab, add an ArcGIS service:



Under Server URL: use http://crwr-arcgis01.austin.utexas.edu/arcgis/services.  There is no need for a login or password.



You should see a new services endpoint available in the services list:



Expand this new service to access the services for World Water Online (WWO):

Drag the GRACE service onto the map display 

Use the **Symbology** properties of the layer to recolor the display
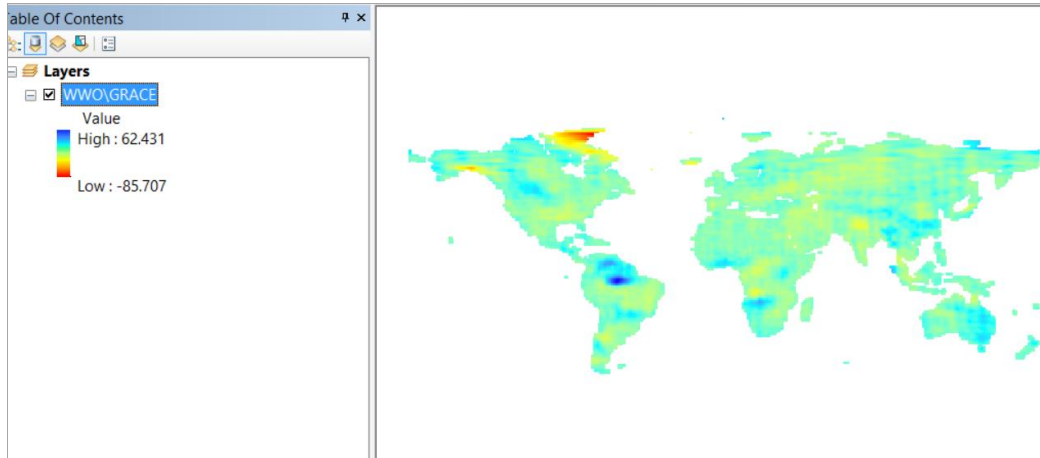
Open the attribute table of the WWO\GRACE layer.

**Table**

WWO\GRACE

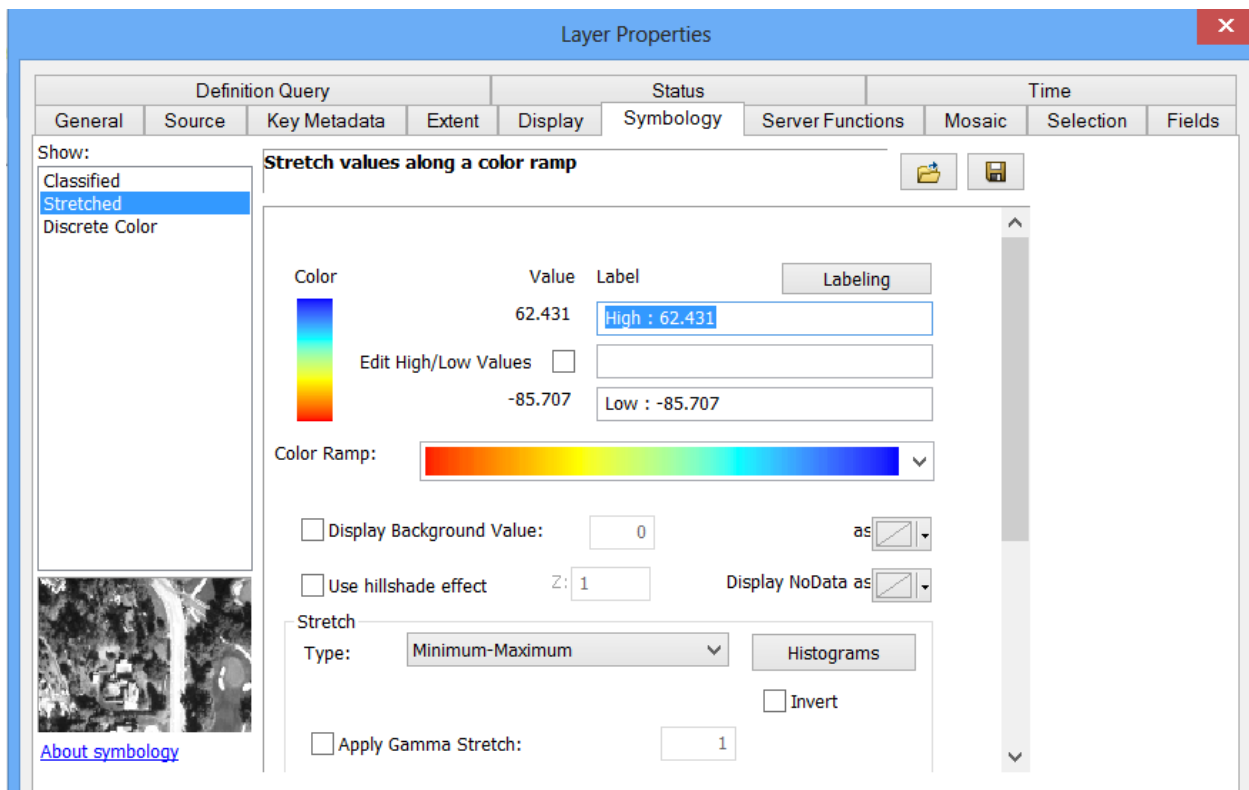| OBJECTID | Shape | RASTER | Name | MinPS | MaxPS | LowPS | HighPS | Category | Tag | Gro | Pro | Cent | Cent | ZOrder | YYYYMM | YYYYMMDD_start | YYYYMMDD_end | Shape |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | Polygon | <Raster> | grace_200302 | 0 | 3 | 1 | 1 | 1 | Dataset | | | 0 | 0 | <Null> | 200302 | 20030201 | 20030301 | |
| 2 | Polygon | <Raster> | grace_200303 | 0 | 3 | 1 | 1 | 1 | Dataset | | | 0 | 0 | <Null> | 200303 | 20030301 | 20030401 | |
| 3 | Polygon | <Raster> | grace_200304 | 0 | 3 | 1 | 1 | 1 | Dataset | | | 0 | 0 | <Null> | 200304 | 20030401 | 20030501 | |
| 4 | Polygon | <Raster> | grace_200305 | 0 | 3 | 1 | 1 | 1 | Dataset | | | 0 | 0 | <Null> | 200305 | 20030501 | 20030522 | |
| 5 | Polygon | <Raster> | grace_200307 | 0 | 3 | 1 | 1 | 1 | Dataset | | | 0 | 0 | <Null> | 200307 | 20030701 | 20030801 | |
| 6 | Polygon | <Raster> | grace_200308 | 0 | 3 | 1 | 1 | 1 | Dataset | | | 0 | 0 | <Null> | 200308 | 20030801 | 20030901 | |
| 7 | Polygon | <Raster> | grace_200309 | 0 | 3 | 1 | 1 | 1 | Dataset | | | 0 | 0 | <Null> | 200309 | 20030901 | 20031001 | |
| 8 | Polygon | <Raster> | grace_200310 | 0 | 3 | 1 | 1 | 1 | Dataset | | | 0 | 0 | <Null> | 200310 | 20031001 | 20031101 | |
| 9 | Polygon | <Raster> | grace_200311 | 0 | 3 | 1 | 1 | 1 | Dataset | | | 0 | 0 | <Null> | 200311 | 20031101 | 20031201 | |
| 10 | Polygon | <Raster> | grace_200312 | 0 | 3 | 1 | 1 | 1 | Dataset | | | 0 | 0 | <Null> | 200312 | 20031201 | 20040101 | |
| 11 | Polygon | <Raster> | grace_200401 | 0 | 3 | 1 | 1 | 1 | Dataset | | | 0 | 0 | <Null> | 200401 | 20040101 | 20040114 | |
| 12 | Polygon | <Raster> | grace_200402 | 0 | 3 | 1 | 1 | 1 | Dataset | | | 0 | 0 | <Null> | 200402 | 20040204 | 20040301 | |
| 13 | Polygon | <Raster> | grace_200403 | 0 | 3 | 1 | 1 | 1 | Dataset | | | 0 | 0 | <Null> | 200403 | 20040301 | 20040401 | |
| 14 | Polygon | <Raster> | grace_200404 | 0 | 3 | 1 | 1 | 1 | Dataset | | | 0 | 0 | <Null> | 200404 | 20040401 | 20040501 | |
| 15 | Polygon | <Raster> | grace_200405 | 0 | 3 | 1 | 1 | 1 | Dataset | | | 0 | 0 | <Null> | 200405 | 20040501 | 20040601 | |

1 ▶ ▶| (0 out of 121 Selected)

WWO\GRACE

Note that this is a combination (in the form of a Mosaic) of 121 GRACE rasters from different dates. It is thus ready for time analysis. There are two time fields: YYYYMMDD_start and YYYYMMDD_end, that represent the year, month and day values of the start date and the end date for each GRACE raster. Generally these are monthly time intervals, but there are some irregularities and data gaps that you'll see when we animate the data. The value that is being reported in the raster is the average gravity anomaly for that interval of time, measured in units of cm of water. Typical values range between -20cm and +20cm or approximately -8 inches to 8 inches of water. When you think about a layer of that depth of water distributed over the whole landscape, you can see that's a lot of water! The gravity "anomaly" refers to the magnitude of the gravity force at that location *relative to its average value since February 2003* which is the beginning time of this dataset. In other words, we are not measuring actual storage of water, but rather the time variation of water storage relative to a constant average value over the last 10 years.

Enable Time on this Layer by clicking in the "**Enable Time**" box in the **Time Properties**, under Layer Time, select "**Each feature has a start and end time field**" Choose **YYYYMMDD_start** as the **Start Time Field**, choose **YYYYMMDD_end** as the **End Time Field**, **YYYYMMDD** as the **field format**, **1 month** as the **time**

**step**, **UTC** Coordinate Universal Time as the **time Zone** and hit **Calculate** so that you see that the Layer time extent gets populated, then reset the **Time Step Interval** to 1 Month, and then hit **Apply** and **Ok**.
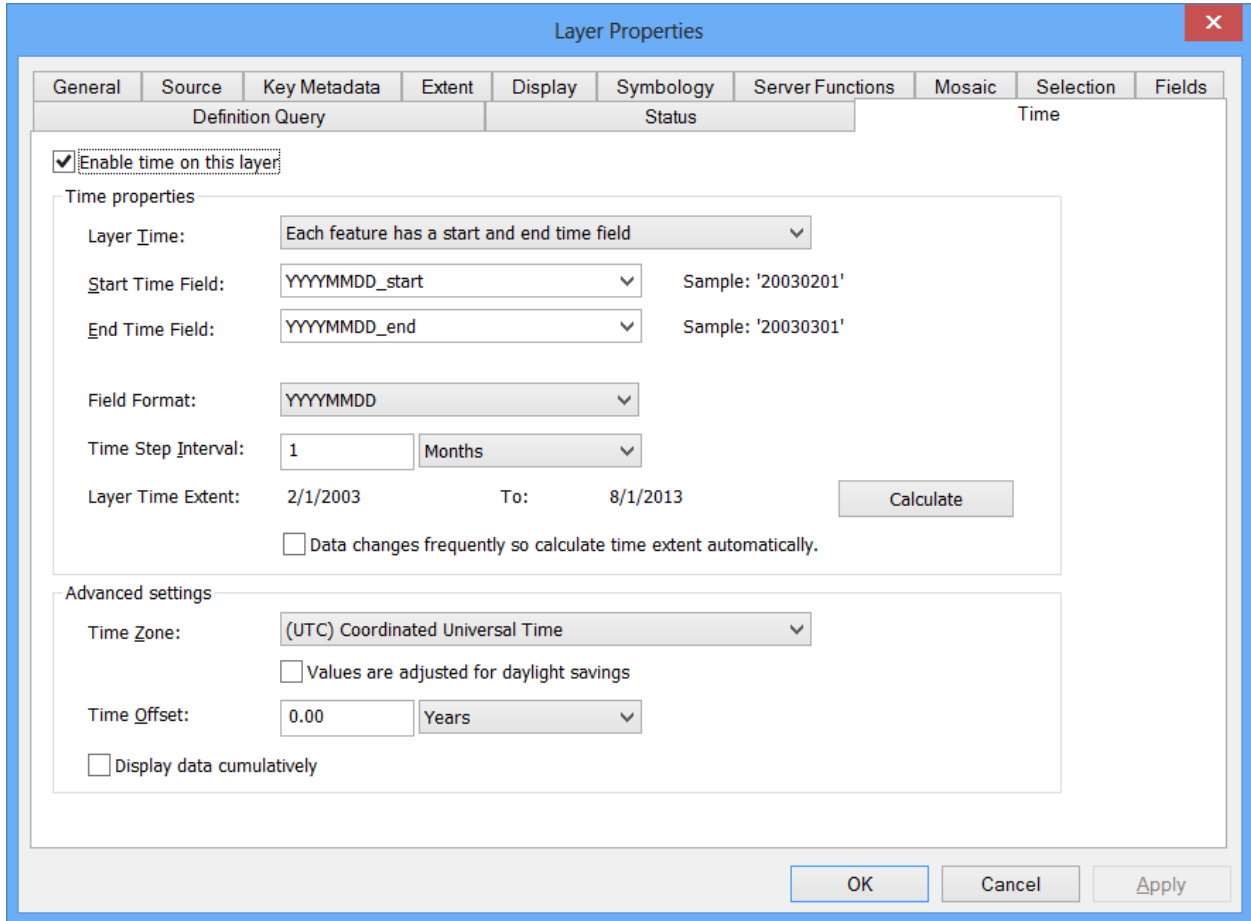


Open the Time Slider window in ArcMap and play an animation of the GRACE data from February 2003 to July 2013.  What you are seeing is the time and spatial variation of the gravity force of the earth measured in equivalent cm of water depth on the earth's surface, termed the **Total Water Storage**. This demonstrates the gain and loss of water on the earth's surface and subsurface as the result of endless dynamic forces of global water redistribution across the globe between the land and the oceans.  Red colors represent times when less water is present, blue when more water is present.  The zero value represents the average gravity force or Total Water Storage at that location over the 10-year period from 2003 to 2013.  Gravity force variations are also induced by changes in the ocean and in the polar ice caps, so this **GRACE Research and Education Dataset for Hydrology** has been produced by the Center for Space Research and the Center for Research in Water Resources of the University of Texas at Austin, specifically for study of Total Water Storage variations on the earth's land surface outside of the polar regions.
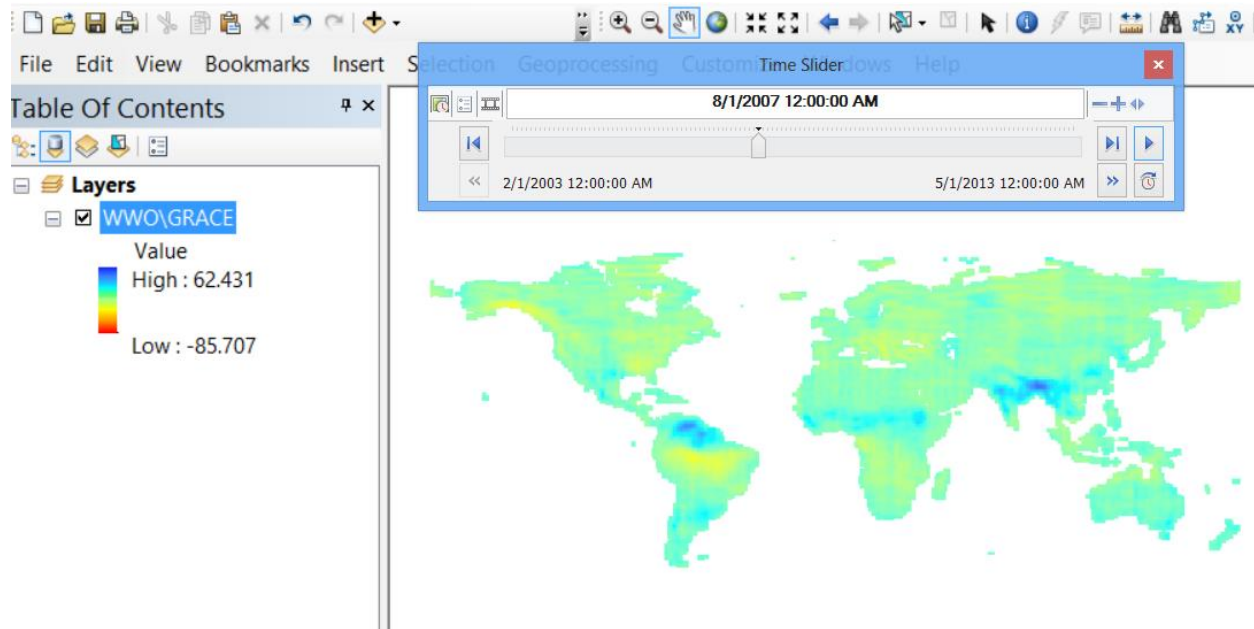
If you find the display moves too quickly, click on the Options button  and under **Playback**, move the speed to a slower pace. You'll see some gaps in the available data because of problems with the measurement and interpretation of the GRACE data – GRACE was initially planned to be a five-year mission but has been so successful that its life has been extended for more than a decade, as long as the batteries and other equipment on the GRACE satellites can remain functioning.
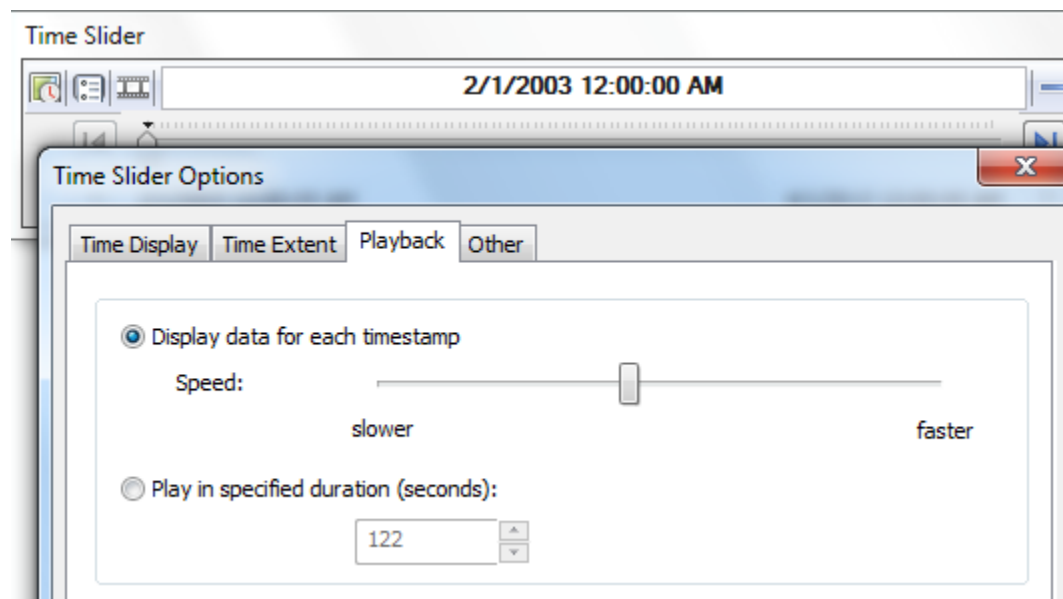


Save the map document as **Ex5GRACE.mxd**

From the zip file http://www.neng.usu.edu/cee/faculty/dtarb/giswr/2013/Ex5.zip extract the geodatabase **BaseLayers.gdb**, add the feature classes for the United States, Texas and Utah to the map display and zoom to the area of the continental United States. Symbolize these as **Hollow** so that just

the outlines show.  Under **Symbology** for the GRACE data, Check the box **Edit High/Low Values** and set the color High/Low values to 20 and -20.
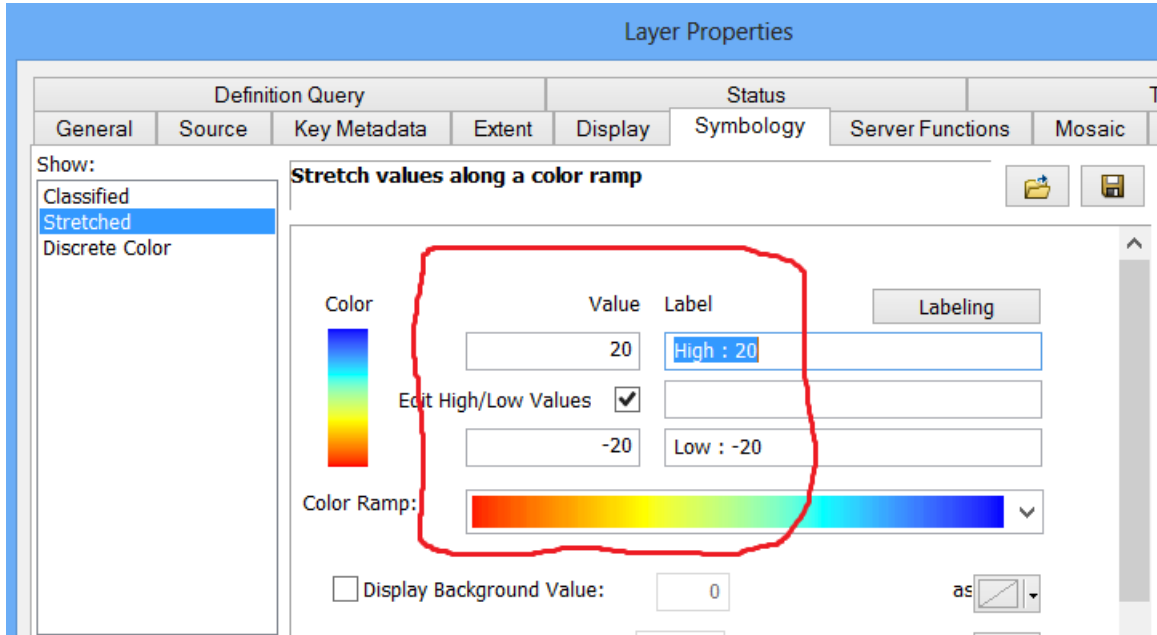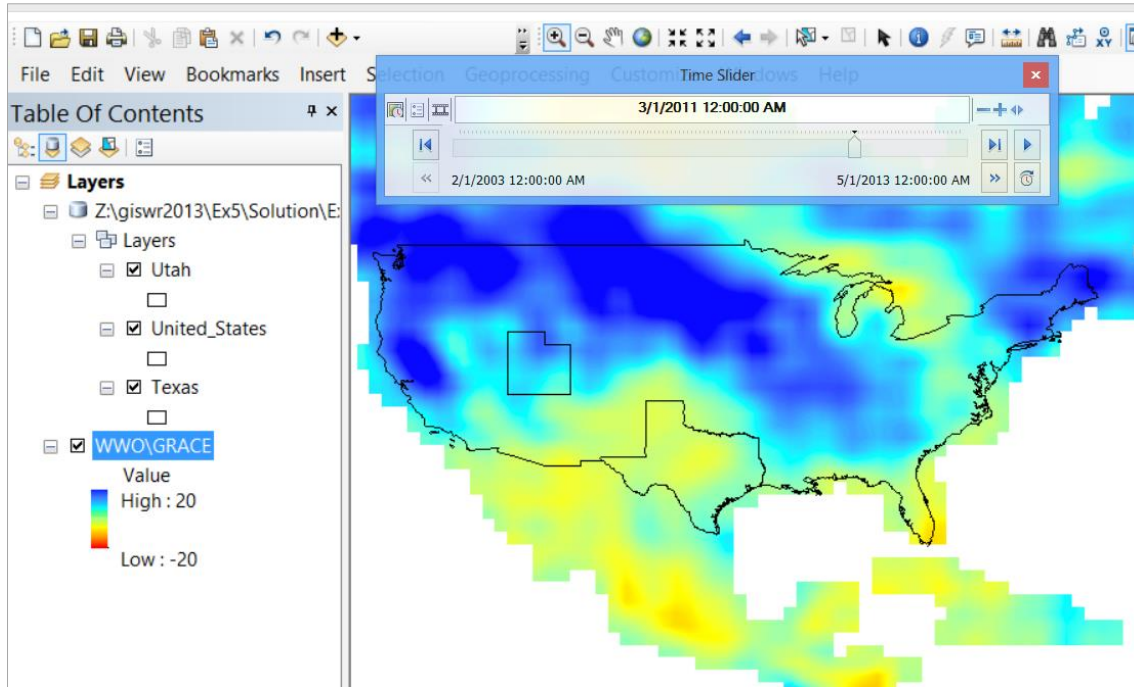


Click on the **Display** tab and set the Resampling to **Bilinear Interpolation**.
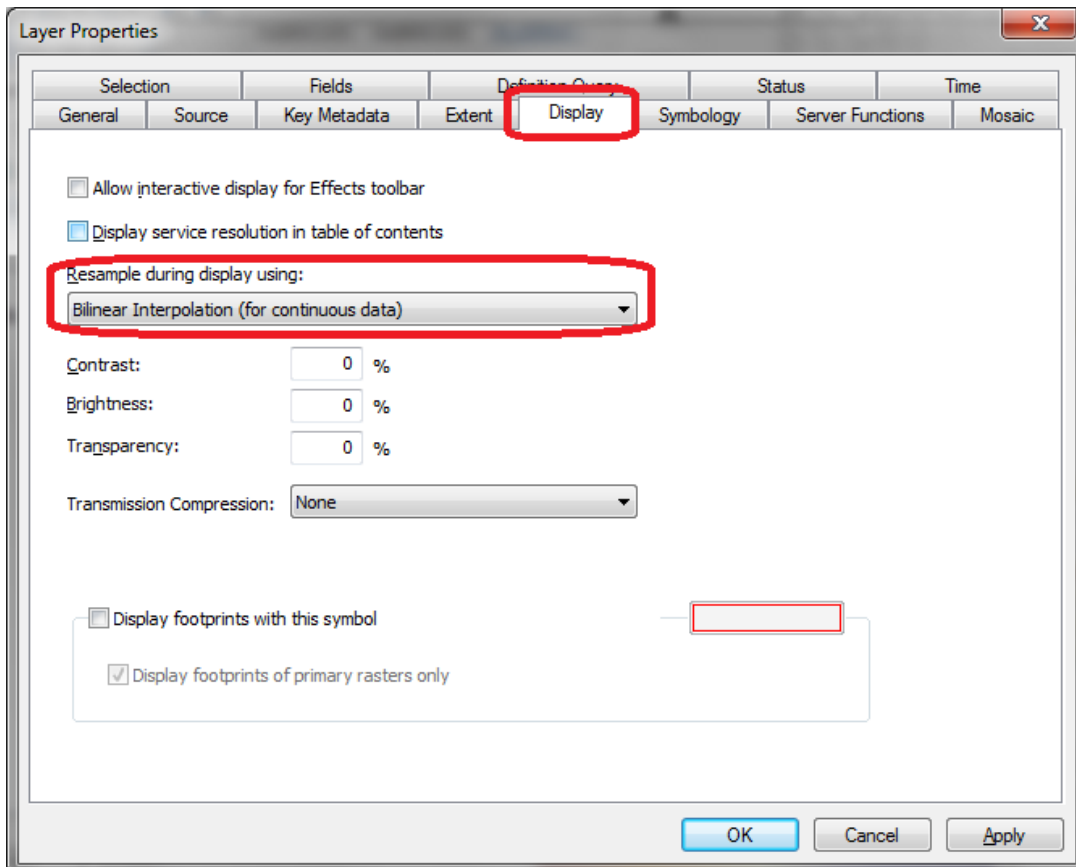
The Time Slider animation now has a smoother look with a consistent image Scale.

Move the time point to March 2011.  Wow.  Drought is starting to occur over Texas and Mexico but much of the rest of the US is relatively wet.



Look later on in September 2011 to see really intense drought in Texas, and over much of the US, except the Midwest, but conditions easing in some parts of Mexico
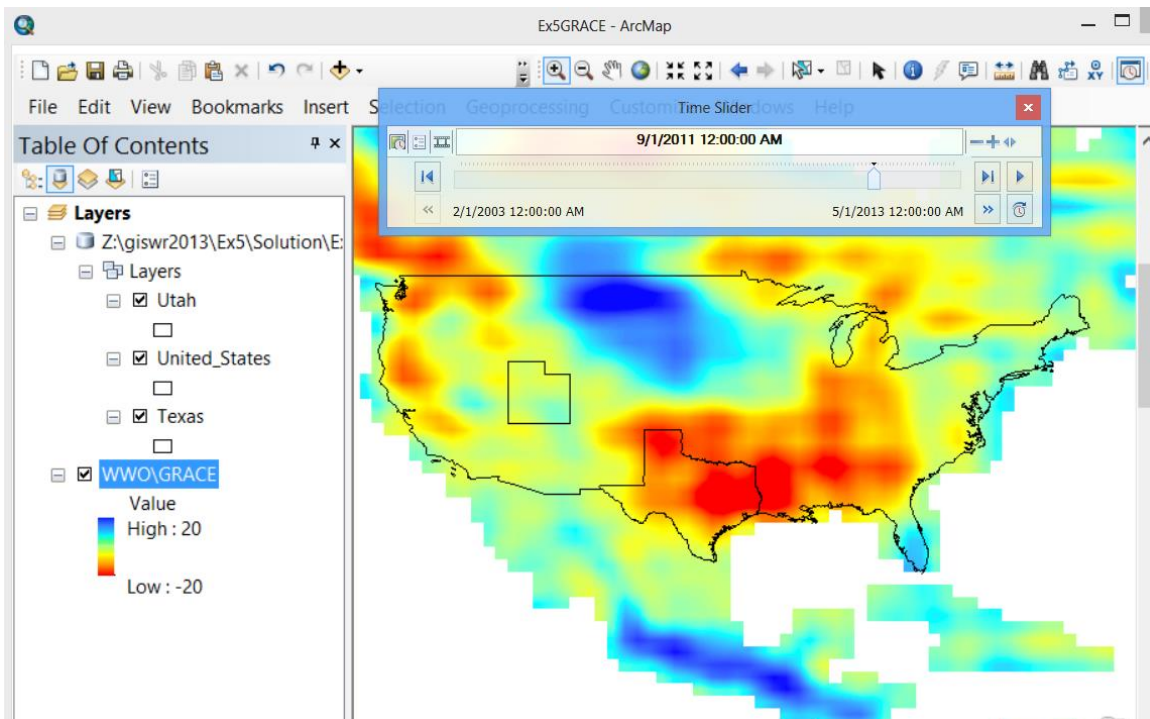
Cycle through the months and you'll see a continuously changing pattern of Total Water Storage anomalies over the United States.

The GRACE anomaly can be averaged over Texas and converted to equivalent units of cubic kilometres of water storage by multiplying the Total Water Storage depth anomaly (cm) by the area of the state. A comparison of the resulting chart for 2003 to 2013 is compared below to the time sequence of drought for Texas, as defined by the US Drought Monitor. You can see that there is a close correlation between Drought Intensity and the GRACE anomaly. At peak drought in 2011, Texas had a Total Water Storage anomaly of about 100 $Km^3$ of water, about 70 times the volume of water storage in Lake Travis, a large surface water reservoir located near Austin, Texas.
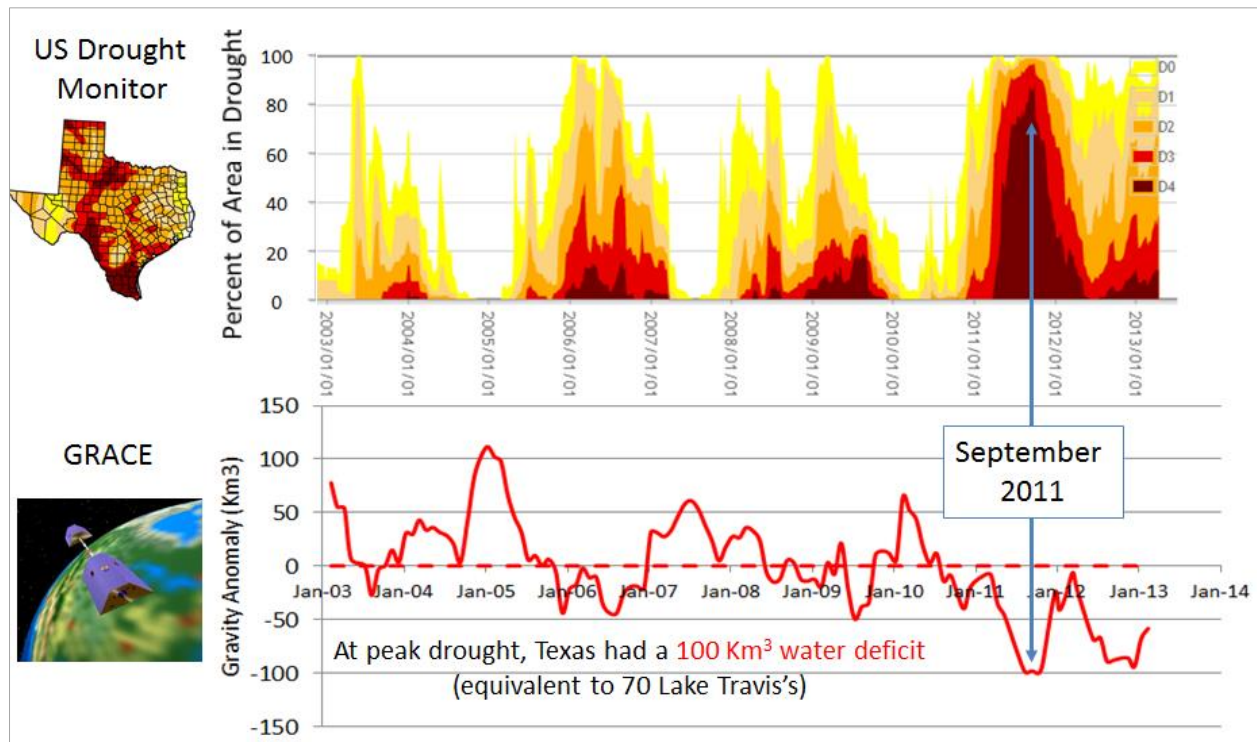


A similar comparison can be made to the total volume of water storage in Texas reservoirs, as reported by the Texas Water Development Board http://waterdatafortexas.org/reservoirs/statewide. This shows that the variation in surface water storage is closely correlated with the GRACE total water storage, but most of the variation in Total Water Storage occurs in the soil and groundwater.
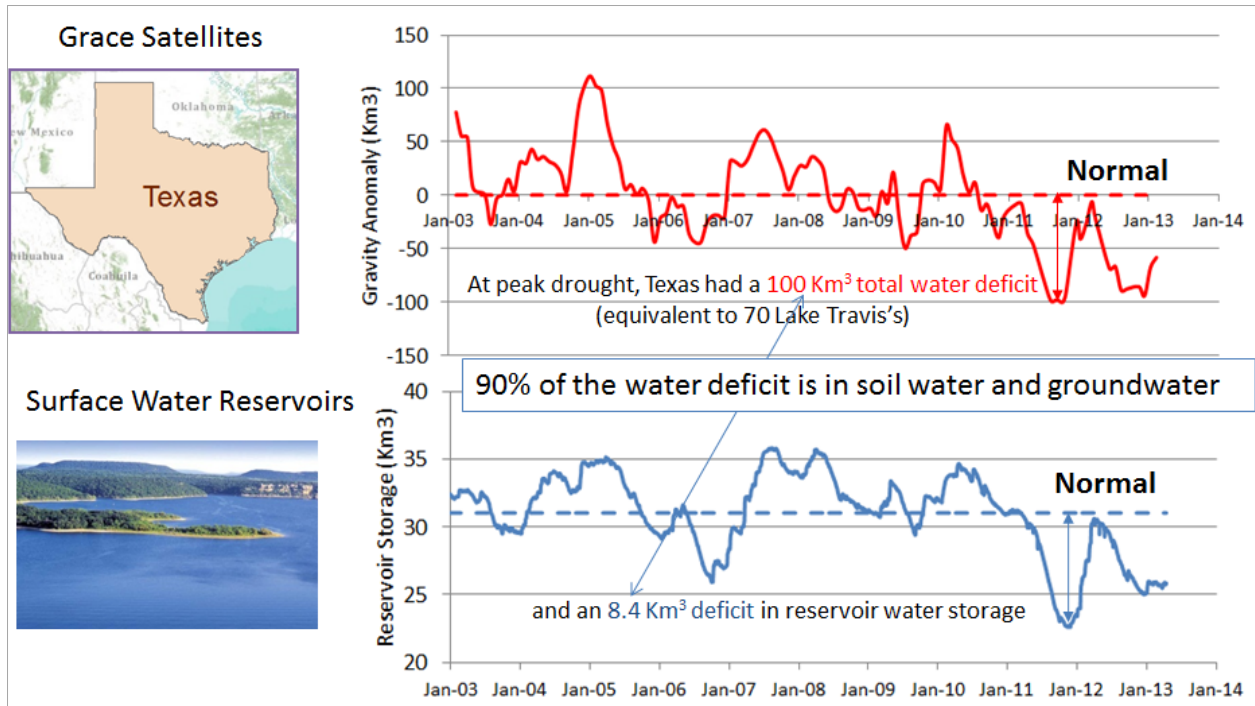
*To be turned in:  Use a sequence of screen shots to discuss the changing patterns of Total Water Storage across the United States and in Texas or Utah in 2011 and 2012.*

Utah students may be interested in comparing the GRACE data in March and April 2011, with March and April 2012.  Think back to the Logan river streamflow in 2011 and 2012.  What was the difference?  GRACE total water storage here includes snowpack which was big in 2011 and small in 2012.  In 2011 we had floods in Utah while Texas was in a drought.

Close ArcMap (saving **Ex5Grace.mxd** again if you wish).
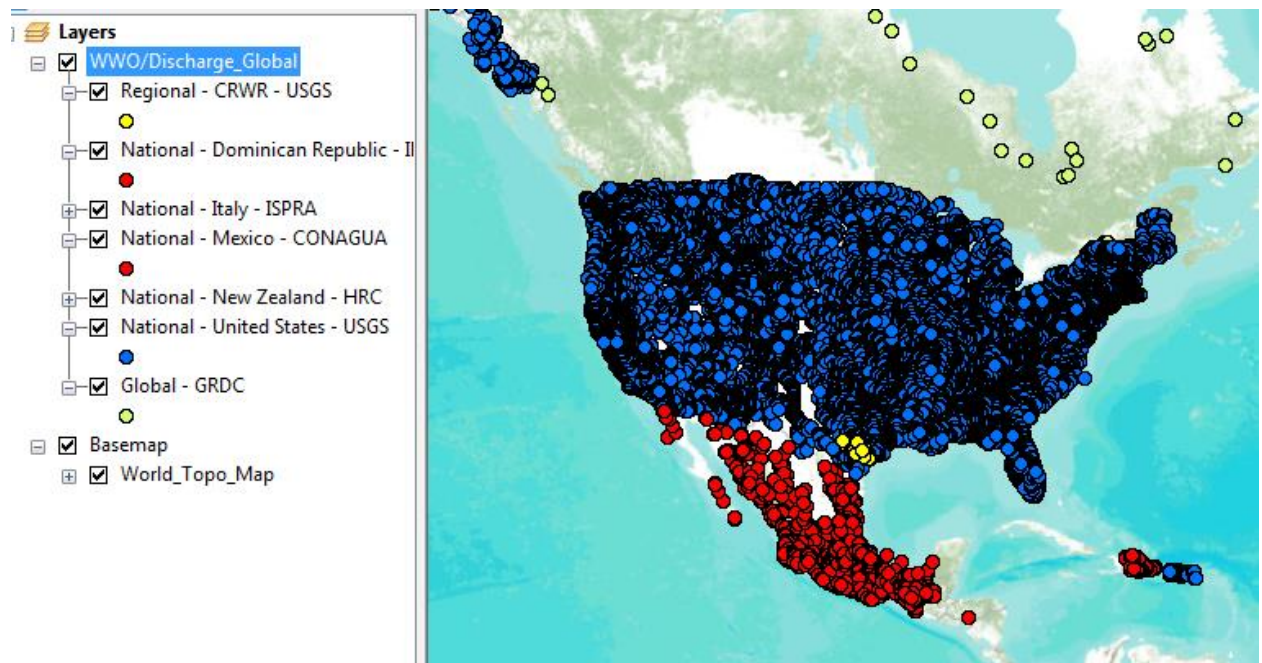
# World Water Online

Open ArcMap, and add the Topographic Base map to your display. Add the World Water Online services as a GIS Server (as in the GRACE part of the exercise). Drag the "Discharge Global" Service onto your display.
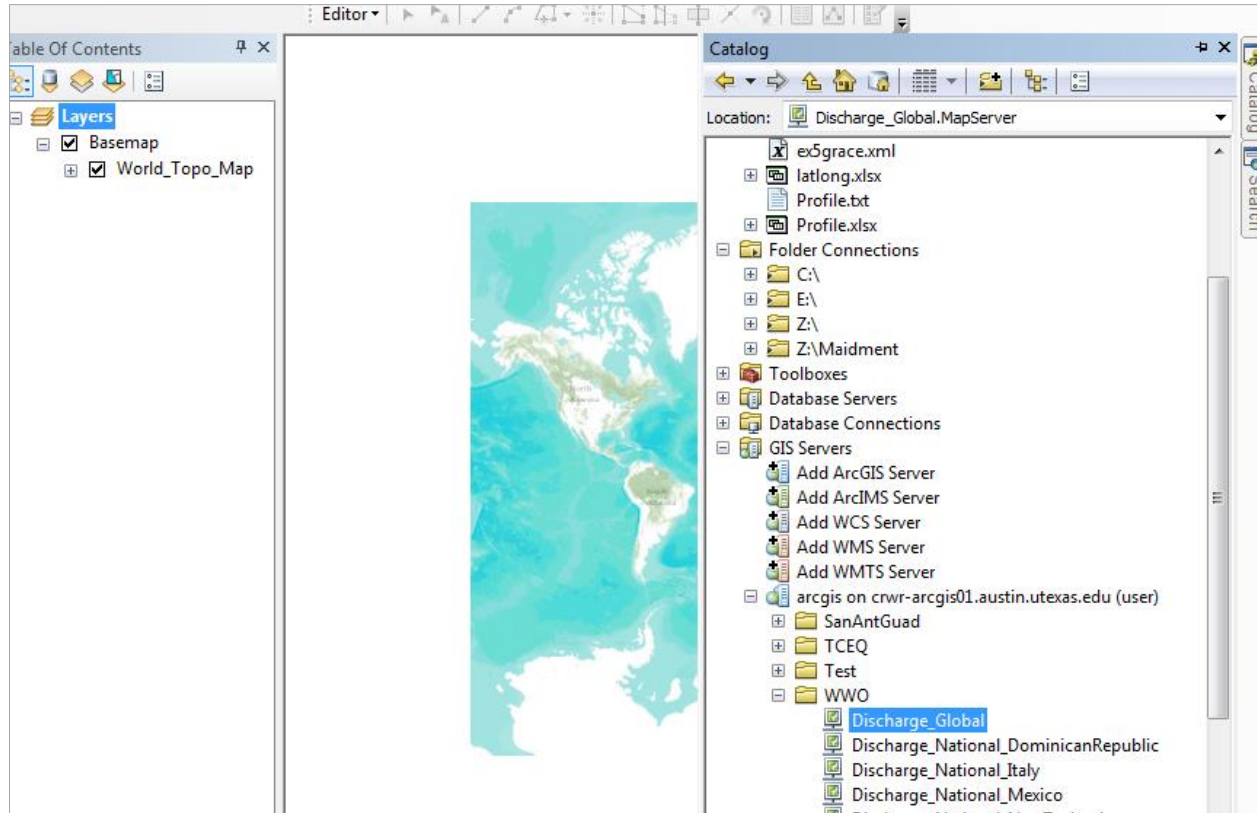


These are locations in the world where streamflow information is presently available as WaterML web services.
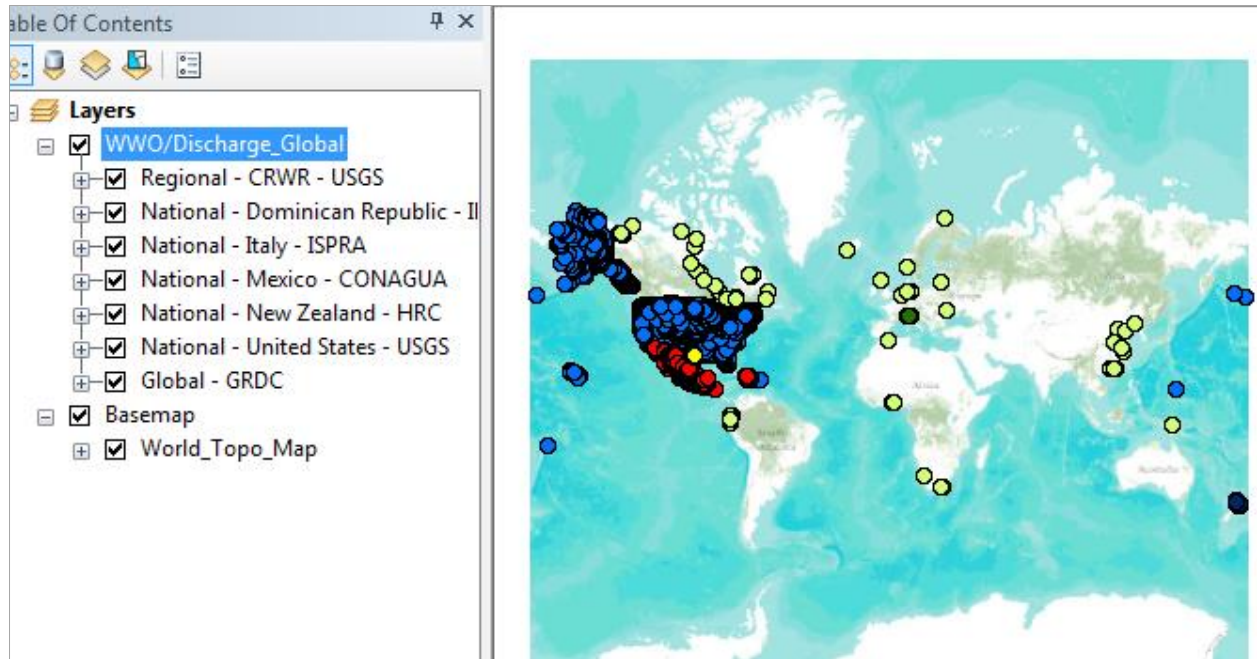
Let's zoom in to North America



The various observation networks are depicted in different colors, blue for US Geological Survey in the United States, Yellow for a special group of USGS flow stations in the San Antonio and Guadalupe basins, red for the Comisión National del Agua (CONAGUA) in Mexico and for Instituto Nacional del Recursos Hidáulicos in the Dominican Republic, green for continental outlet flow stations whose data are stored at the Global Runoff Data Center in Koblenz, Germany.  As more regions and countries adopt the WaterML language for publishing their streamflow information, this map will be further extended.  Right

click on the Regional streamflow coverage CRWR-USGS and zoom to the extent of this layer, and turn off the other streamflow layers.
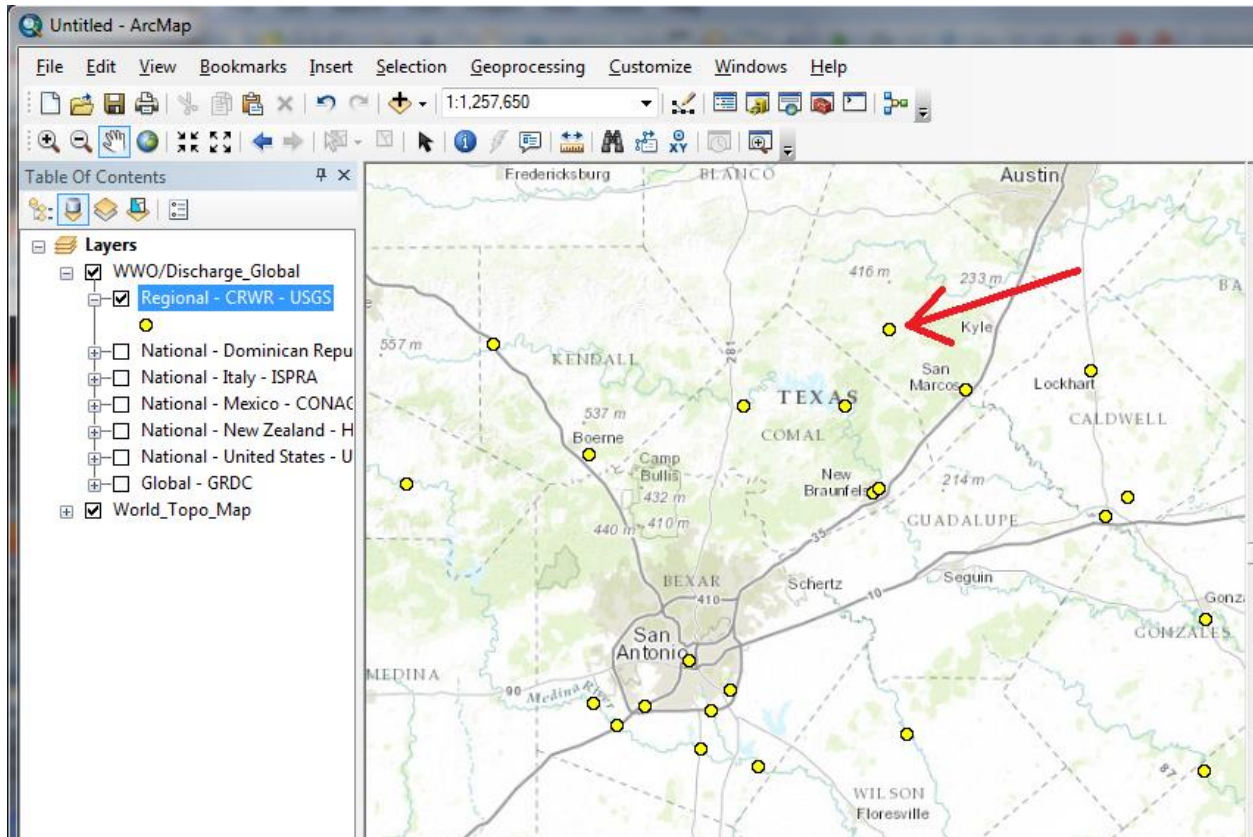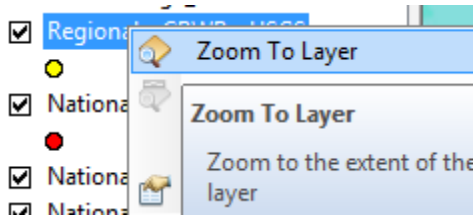




Let's work with data from our old friend, the Blanco River at Wimberley, Texas.

If you hit the Identify button and this site location, you'll get a set of attributes that tell you that this is stream discharge in cubic feet per second that come from a Kisters data server at UT Austin and begin in March 2013 (this is part of the "Central Texas Hub" that provides a cross-network observational system for water conditions in Central Texas http://www.centraltexashub.org)   The time step is 15 minutes.

## Identify                                                                    □ ×

**Identify from:**   | `<Top-most layer>` | ▼

```
⊟ Regional - CRWR - USGS
     └── Blanco Rv at Wimberley, TX
```

                                                                    ⊠□

**Location:** | -98.086957  29.992528 Decimal Degrees

| Field | Value |
|---|---|
| StationCode | 08171000 |
| StationName | Blanco Rv at Wimberley, TX |
| VariableName | Discharge, stream |
| VariableUnit | cubic foot per second |
| DataType | Average |
| StartDate | 2013-03-01 |
| EndDate | Null |
| TimeStep | T15M |
| Latitude | 29.993731 |
| Longitude | -98.088661 |
| GraphURI | http://129.116.104.178:8080/KiWIS/KiWIS?service=kisters&type=queryServices&request=getgraph&datasource=0&for ⚡ |
| WaterML | http://129.116.104.178:8080/KiWIS/KiWIS?service=kisters&type=queryServices&request=gettimeseriesvalues&datasou ⚡ |
| Version | 2 |
| Download | http://129.116.104.178:8080/KiWIS/KiWIS?service=kisters&type=queryServices&request=gettimeseriesvalues&datasou ⚡ |
| Shape | Point |
| Graph | `<p style="color:#21759B;text-decoration: underline;cursor:pointer;" onclick="stDialog=new dijit.Dialog({title:'USGS Streamf` ∨ |

Identified 1 feature

If you click on the **GraphURI**, you'll get a chart of the flow data in which you can see the low flow condition of the Blanco River over the long summer we've just had, and then the high flow pulse that resulted from heavy rain in early October.



Timeseries Graph generated 2013-10-25 11:01:57 GMT-06:00
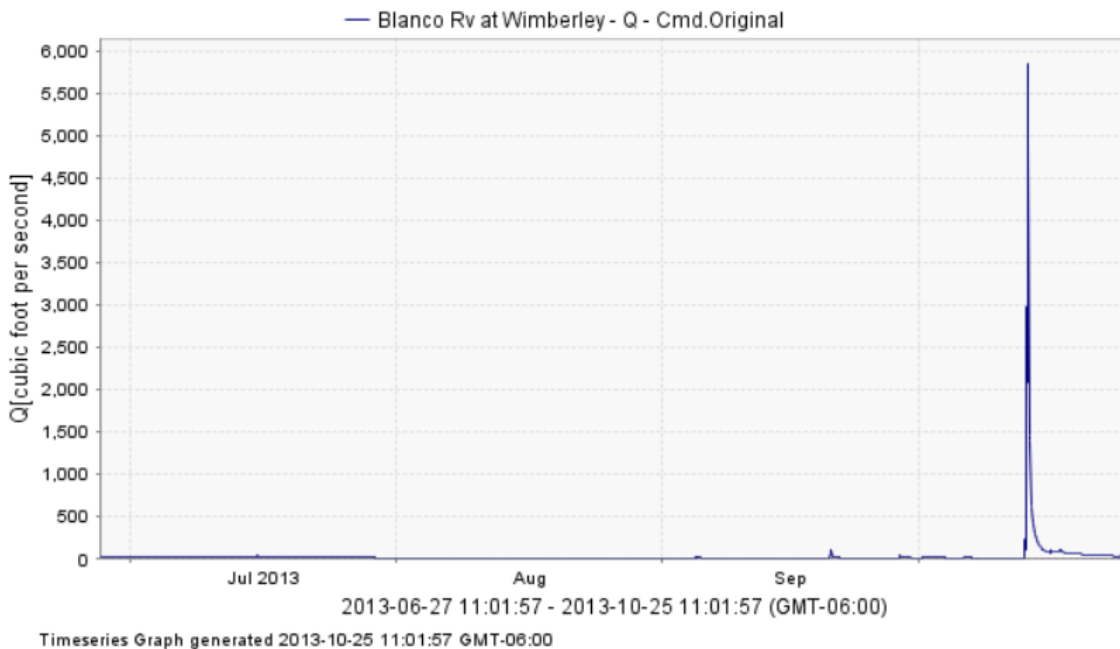
If you click on **WaterML**, you'll get a download of the data in the WaterML2 language, now an international standard for the conveyance of water time series http://www.opengeospatial.org/standards/waterml.  WaterML is a completely self-describing package that tells you where the data came from, what the units are, and then presents the time series as a set of "Time Value Pairs" (TVP).  The Continuous/Instantaneous says that each time value pair refers to the discharge at that instant of time.

```
        <om:procedure xlink:title="Cmd.Original" xlink:href="Cmd.O"/>
        <om:observedProperty xlink:title="Q" xlink:href="Q"/>
        <om:featureOfInterest xlink:title="Blanco Rv at Wimberley" xlink:href="08171000"/>
      - <om:result>
          - <wml2:MeasurementTimeseries gml:id="Ki.Ts.15705042">
              - <wml2:defaultPointMetadata>
                  - <wml2:DefaultTVPMeasurementMetadata>
                        <wml2:qualifier xlink:title="200" xlink:href="200"/>
                        <wml2:uom code="ft³/s"/>
                        <wml2:interpolationType xlink:title="Continuous/Instantaneous" xlink:h
                  </wml2:DefaultTVPMeasurementMetadata>
              </wml2:defaultPointMetadata>
            - <wml2:point>
                - <wml2:MeasurementTVP>
                      <wml2:time>2013-10-11T11:15:00.000-06:00</wml2:time>
                      <wml2:value>11.0</wml2:value>
                  </wml2:MeasurementTVP>
              </wml2:point>
            - <wml2:point>
                - <wml2:MeasurementTVP>
                      <wml2:time>2013-10-11T11:30:00.000-06:00</wml2:time>
                      <wml2:value>11.0</wml2:value>
                  </wml2:MeasurementTVP>
```

And if you click on Download, you'll get a CSV file of the time value pairs that can be implemented in Excel or by programming.  The ISO time stamp 2013-10-11T11:15:00.000-06:00 means that the data time is 11 October 2013 at 11:15AM in the Central time zone which is 6 hours behind Universal Coordinate time.

| | A | B | C | D |
|---|---|---|---|---|
| 1 | #ts_id;15705042 | | | |
| 2 | #rows;1344 | | | |
| 3 | #Timestamp;Value | | | |
| 4 | 2013-10-11T11:15:00.000-06:00;11.0 | | | |
| 5 | 2013-10-11T11:30:00.000-06:00;11.0 | | | |
| 6 | 2013-10-11T11:45:00.000-06:00;11.0 | | | |
| 7 | 2013-10-11T12:00:00.000-06:00;11.0 | | | |
| 8 | 2013-10-11T12:15:00.000-06:00;11.0 | | | |
| 9 | 2013-10-11T12:30:00.000-06:00;11.0 | | | |
| 10 | 2013-10-11T12:45:00.000-06:00;11.0 | | | |
| 11 | 2013-10-11T13:00:00.000-06:00;11.0 | | | |

These three outputs: Graph, WaterML and a .CSV download are typical for time series information – the graph gives you a visual representation, the WaterML gives a complete description, and the .CSV gives a readily usable dataset in Excel.

If you turn on the USGS National Streamflow layer, and click on the Blanco River at Wimberley, you'll find the same set of attributes, but in this instance the data are coming directly from the USGS not through the Central Texas Hub, the time step is 1 day rather than 15 minutes, and the data begin in August 1924 and continue to the present.
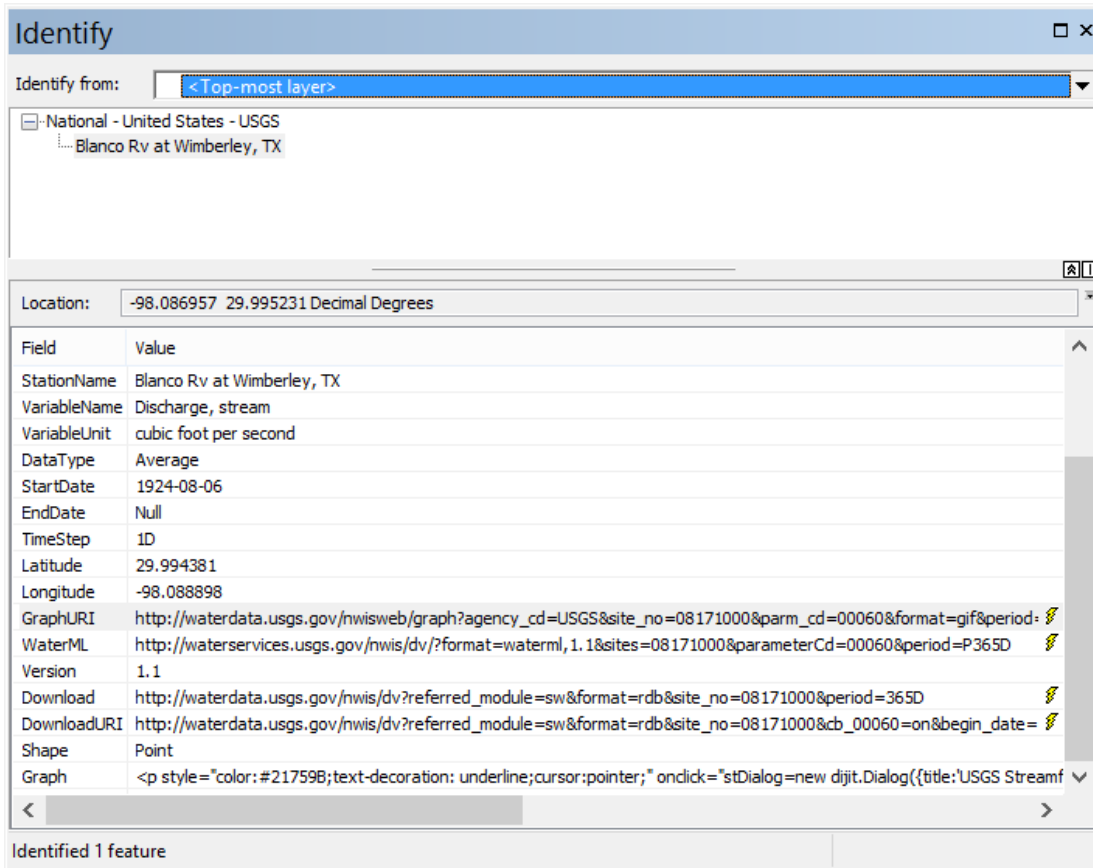


The chart now looks like this (and comes directly from the USGS)
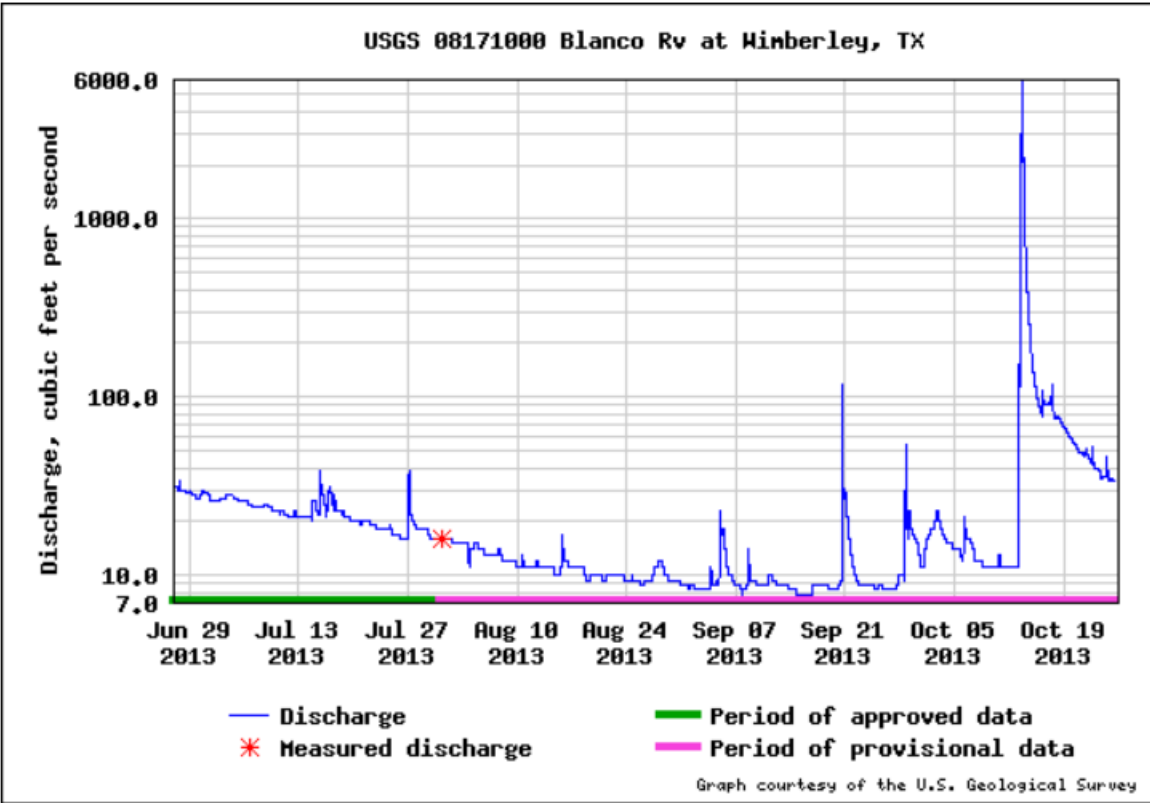
The **WaterML** file looks like this from USGS (this is an earlier version, WaterML version 1.1)

```
<ns1:variable ns1:oid="45807197">
    <ns1:variableCode network="NWIS" variableID="45807197" default="true" vocabulary="NWIS:Un
    <ns1:variableName>Streamflow, ft&#179;/s</ns1:variableName>
    <ns1:variableDescription>Discharge, cubic feet per second</ns1:variableDescription>
    <ns1:valueType>Derived Value</ns1:valueType>
  - <ns1:unit>
        <ns1:unitCode>ft3/s</ns1:unitCode>
    </ns1:unit>
  - <ns1:options>
        <ns1:option name="Statistic" optionCode="00003">Mean</ns1:option>
    </ns1:options>
    <ns1:noDataValue>-999999.0</ns1:noDataValue>
</ns1:variable>
<ns1:values>
    <ns1:value dateTime="2012-10-25T00:00:00.000" qualifiers="A">29</ns1:value>
    <ns1:value dateTime="2012-10-26T00:00:00.000" qualifiers="A">28</ns1:value>
    <ns1:value dateTime="2012-10-27T00:00:00.000" qualifiers="A">27</ns1:value>
    <ns1:value dateTime="2012-10-28T00:00:00.000" qualifiers="A">26</ns1:value>
    <ns1:value dateTime="2012-10-29T00:00:00.000" qualifiers="A">26</ns1:value>
    <ns1:value dateTime="2012-10-30T00:00:00.000" qualifiers="A">25</ns1:value>
    <ns1:value dateTime="2012-10-31T00:00:00.000" qualifiers="A">25</ns1:value>
    <ns1:value dateTime="2012-11-01T00:00:00.000" qualifiers="A">25</ns1:value>
    <ns1:value dateTime="2012-11-02T00:00:00.000" qualifiers="A">25</ns1:value>
```

And in this instance, if you click on **Download URI**, you'll get a text file in RDB format that the USGS uses that is the result of the REST call:

http://waterdata.usgs.gov/nwis/dv?referred_module=sw&format=rdb&site_no=08171000&cb_00060=on&begin_date=2012-01-01&end_date=2013-01-01

What this means is that you are getting data from the USGS National Water Information System, for site no 08172000 (San Marcos River at Luling), for parameter 00060 (Streamflow) beginning at 1 January 2012 and end 1 January 2013.

```
# ------------------------------------------------------------------------
#
# Data provided for site 08171000
#    DD parameter statistic   Description
#    01   00060     00003     Discharge, cubic feet per second (Mean)
#
# Data-value qualification codes included in this output:
#    A  Approved for publication -- Processing and review completed.
#
agency_cd       site_no datetime        01_00060_00003  01_00060_00003_cd
5s      15s     20d     14n     10s
USGS    08171000        2012-01-01      21      A
USGS    08171000        2012-01-02      20      A
USGS    08171000        2012-01-03      20      A
USGS    08171000        2012-01-04      21      A
USGS    08171000        2012-01-05      20      A
USGS    08171000        2012-01-06      20      A
USGS    08171000        2012-01-07      20      A
USGS    08171000        2012-01-08      20      A
USGS    08171000        2012-01-09      20      A
USGS    08171000        2012-01-10      20      A
USGS    08171000        2012-01-11      19      A
USGS    08171000        2012-01-12      18      A
USGS    08171000        2012-01-13      18      A
USGS    08171000        2012-01-14      18      A
USGS    08171000        2012-01-15      18      A
USGS    08171000        2012-01-16      19      A
```

The USGS also provides monthly aggregated streamflow data that is more convenient for a water balance than daily data (or alternatively you could write a program to aggregate this daily data to monthly data).  Go to the USGS NWIS surface water website: http://waterdata.usgs.gov/nwis/sw

Select Statistics / Monthly.

Select Site Number and Submit

Enter site number 08171000 and way at the bottom select submit.

Click on the Site Number in the table that is returned and then select Discharge

Choose the output format as Tab-separated Display in browser and click submit.

The result is a set of monthly aggregated streamflow data for the Blanco River at Wimberley that you can save and work with in Excel.

```
#
#
# US Geological Survey, Water Resources Data
# retrieved: 2013-10-27 15:38:01 EDT      (sdww01)
#
# This file contains USGS Surface-Water Monthly Statistics
#
# Note:The statistics generated from  this site are based on approved daily-mean data a
# The user is responsible for assessment and use of statistics from this site.
# For more details on why the statistics may not match, visit http://waterdata.usgs.gov
#
# ** No Incomplete data have been used for statistical calculation
#
# This file includes the following columns:
#
#
# agency_cd     agency code
# site_no USGS site number
# parameter_cd
# dd_nu
# year_nu Calendar year for value
# month_nu      Month for value
# mean_va monthly-mean value.
#         if there is not complete record
#         for a month this field is blank
#
#
# Sites in this file include:
# USGS 08171000 Blanco Rv at Wimberley, TX
#
# Explanation of Parameter Code and dd_nu used in the Statistics Data
# parameter_cd  Parameter Name                                        dd_nu   Location Name
# 00060         Discharge, cubic feet per second                      1
#
#
agency_cd       site_no parameter_cd    dd_nu   year_nu month_nu        mean_va
5s      15s     5s      3n      4s      2s      12n
USGS    08171000        00060   1       1924    9       62.5
USGS    08171000        00060   1       1924    10      25.4
USGS    08171000        00060   1       1924    11      23.1
USGS    08171000        00060   1       1924    12      23.6
USGS    08171000        00060   1       1925    1       22.4
USGS    08171000        00060   1       1925    2       20.3
USGS    08171000        00060   1       1925    3       18.0
USGS    08171000        00060   1       1925    4       16.0
```

Alternatively the following URL takes you directly to this table

http://waterdata.usgs.gov/nwis/monthly?site_no=08171000&agency_cd=USGS&por_08171000_1=1563,00060,1,1924-08,2013-07&referred_module=sw&format=rdb.

The Logan River is USGS Station Number 10109000.  You can repeat a similar process for it or retrieve the data directly using

http://waterdata.usgs.gov/nwis/monthly?site_no=10109000&agency_cd=USGS&por_10109000_1=448896,00060,1,1953-10,2013-02&referred_module=sw&format=rdb

*To be turned in:  Screen shots of streamflow hydrographs from*

   a) *The Blanco River at Wimberley Texas*
   b) *The Logan River above State Dam Near Logan, Utah.*
   c) *Some other location in the world that has streamflow of interest to you*

*Also include screen shots that show maps of these locations*

   d) *What was the magnitude of the recent (October 2013 high/flood flows in the Blanco River in Texas)?*
   e) *What was the magnitude of the 2011 high/flood flows in the Logan River above State Dam, near Logan Utah?*

Close ArcMap.

## Logan River Water Balance

Open ArcMap and add the Topographic baselayer (World_Topo_Map) and the LoganBasin Feature Class from the Ex5\BaseLayers.gdb\Layers geodatabase and feature dataset in Ex5.zip (http://www.neng.usu.edu/cee/faculty/dtarb/giswr/2013/Ex5.zip).  This is the Feature Class obtained in Exercise 4 projected to a GCS_WGS_1984 datum.  I did this projection so that the feature class is in the same coordinate system as the data retrieved by the LDAS tools because when I tried to programmatically use the LDAS tools I received an error that was to do with a projection datum conversion.  Zoom to the LoganBasin layer.
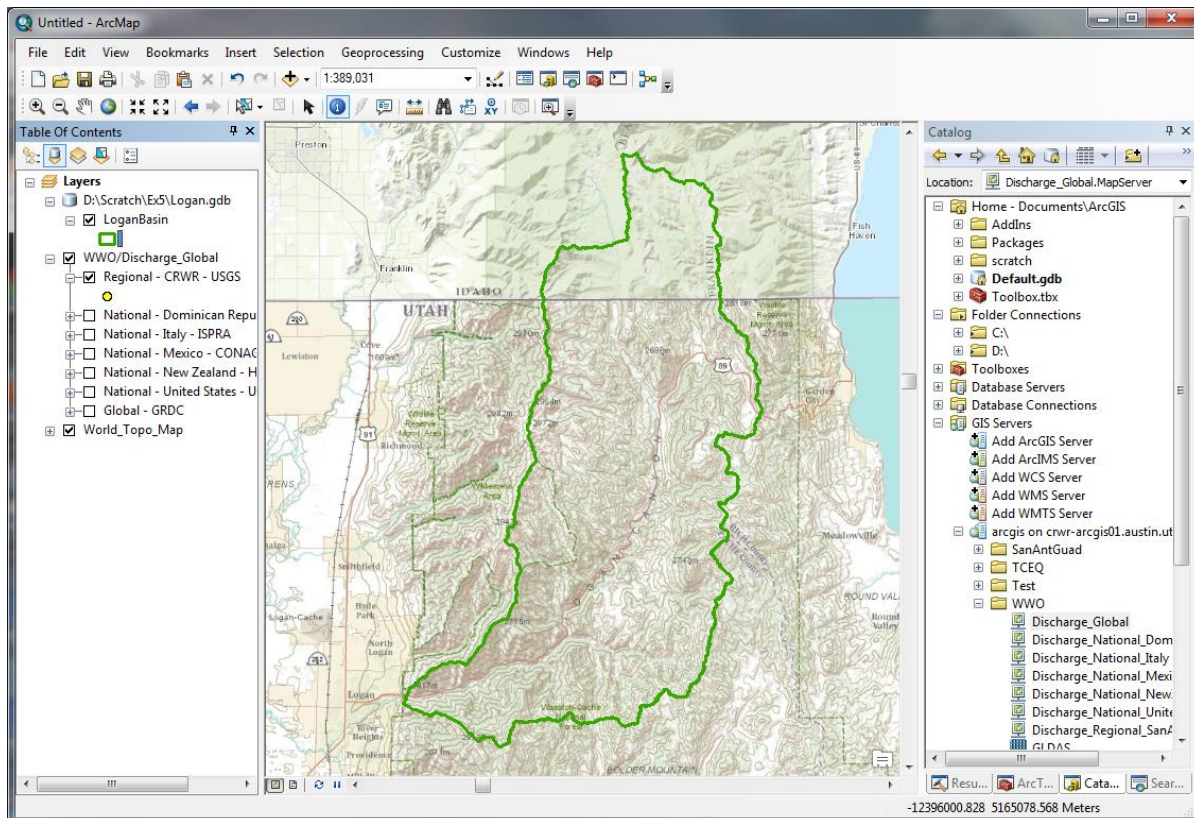
Save your map document as Ex5.mxd in the folder you are working in (where the data was unzipped).

Next we will use the **LDAStools** created by Gonzalo Espinoza to get some data from the NASA Land Data Assimilation System for this watershed.  In Catalog locate the file **LDAStools.gpk** provided in **Ex5.zip**.
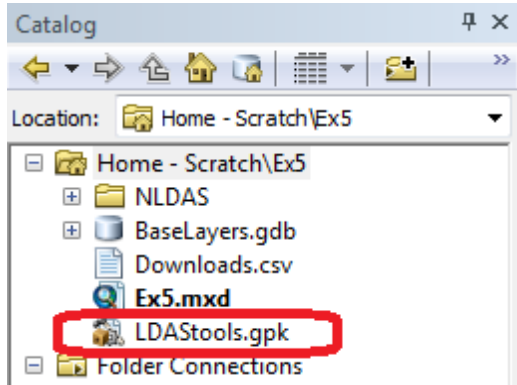


Double click on the **LDAStools.gpk** icon and they will open in the Geoprocessing Results window in ArcMap
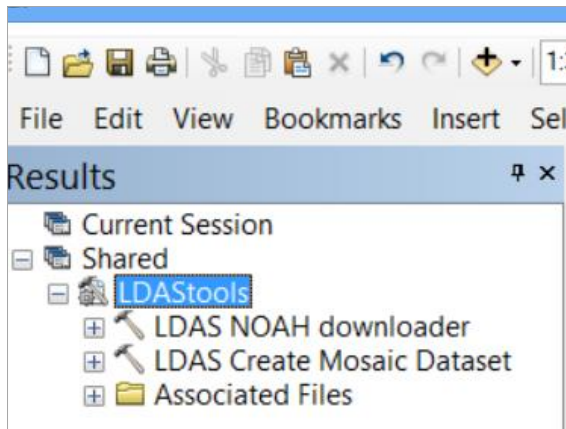


Double Click on the LDAS NOAH downloader and fill in the resulting form as follows.
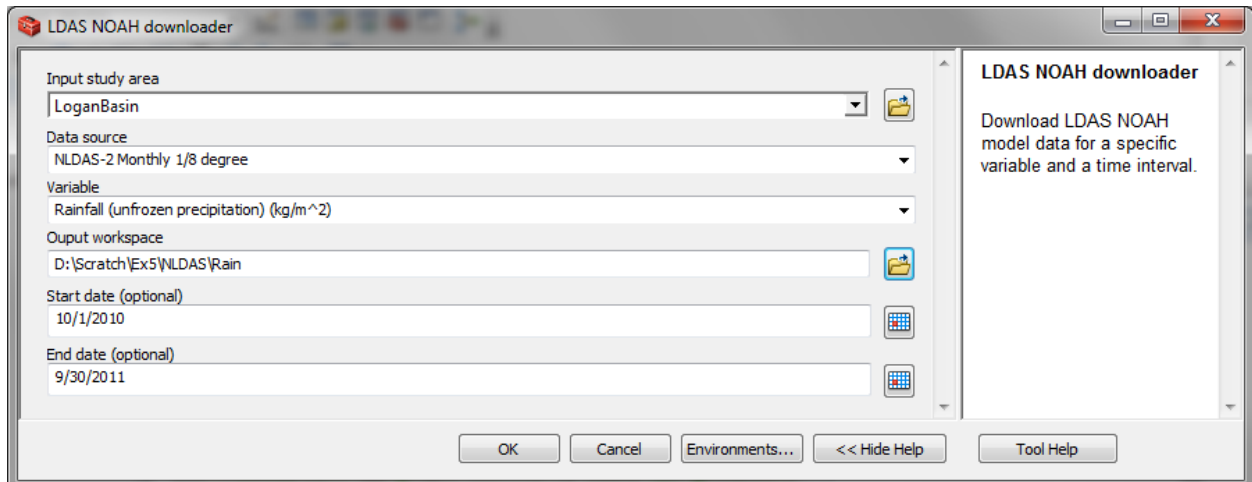
Note that in filling out this form select LoganBasin from Ex5\Logan.gdb. Select Rainfall (unfrozen precipitation). I created the folder NLDAS and Rain to hold the results. I selected the water year of 2011 because I know that was a big flow year and also because the USGS statistics for the current water year are not yet approved and available as monthly statistics.

It takes a little while for this to run, but when it is completed you should see
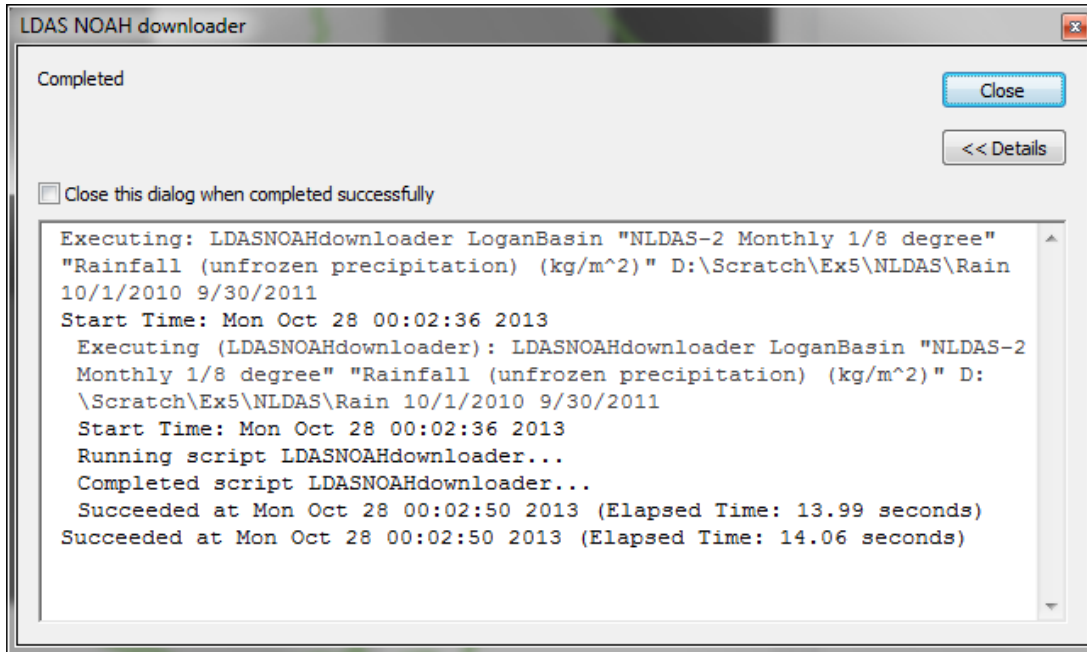


The result is a series of grids of 1/8 x 1/8 degree size whose cells contain the monthly precipitation in mm for this area that are in the folder EX5\NLDAS\Rain
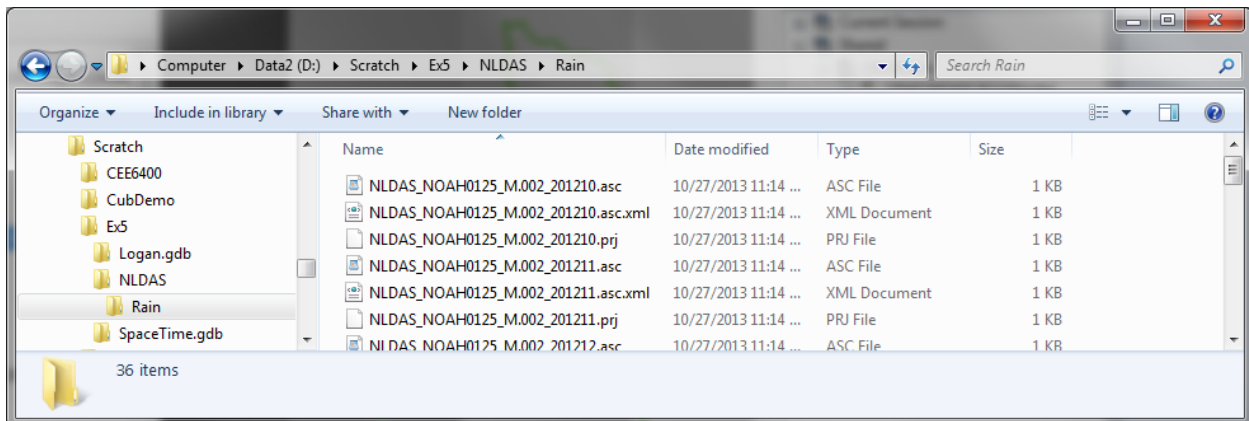


They should also have been loaded into ArcMap.
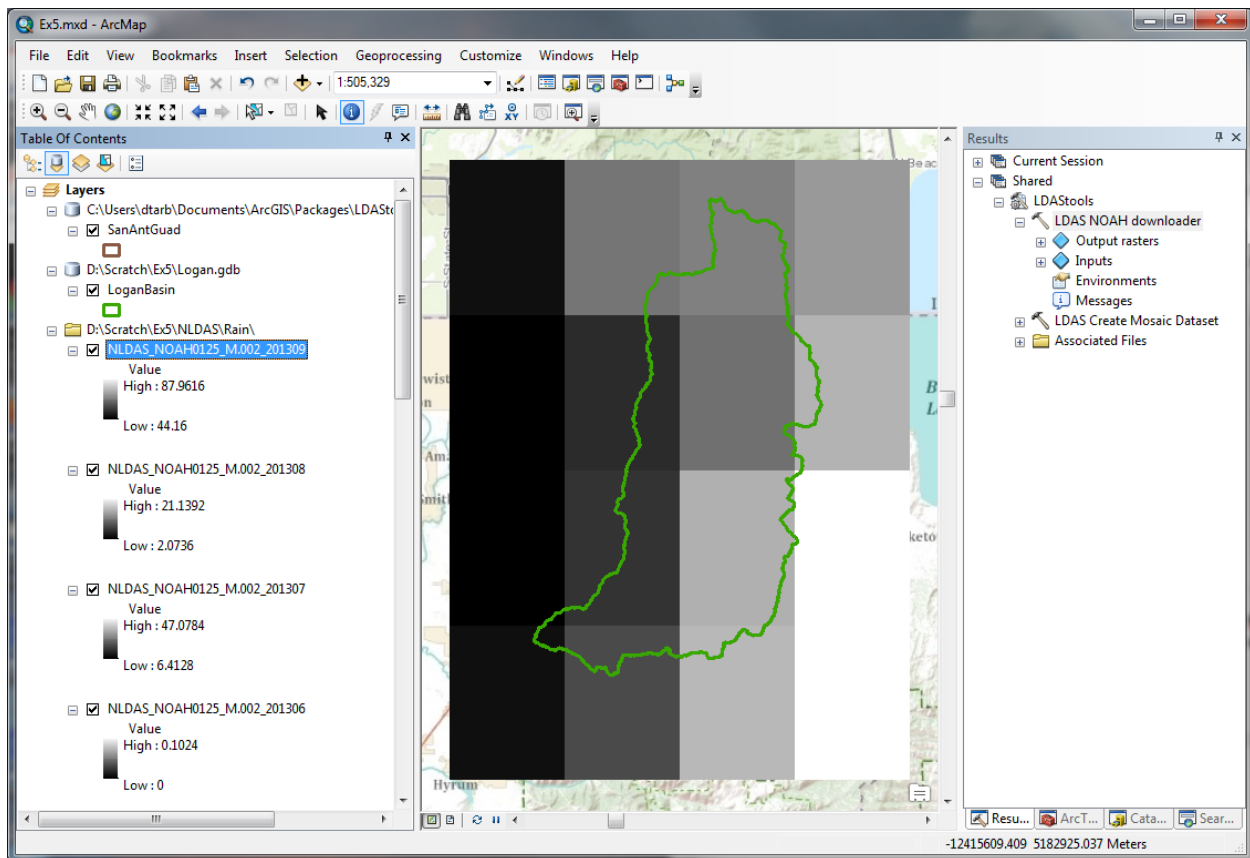
To visualize these better, let's make them into a Mosaic, using an LDAS tool for that purpose.

In **Geoprocessing Results** invoke the **LDAS Create Mosaic Dataset** tool
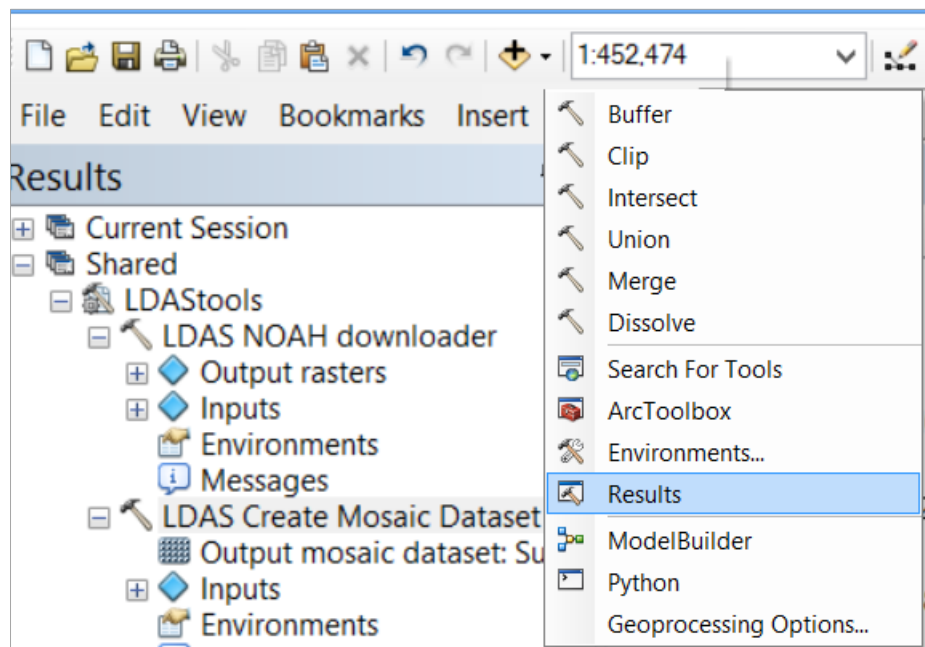
Set the inputs as follows
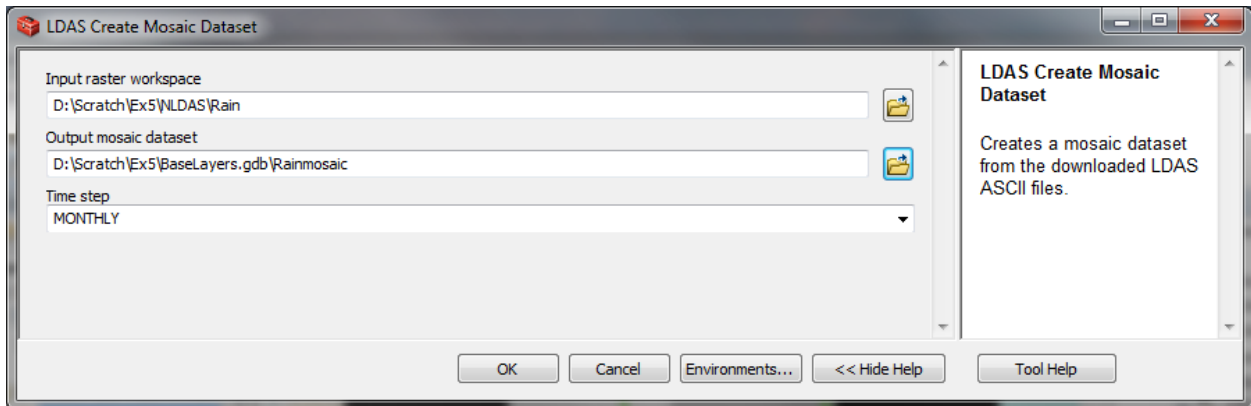


Note that the result was named RainMosaic and placed in the BaseLayers.gdb.  This information will need to be used later, so if you use a different output name, note it.  The result is a mosaic of these rainfall rasters.
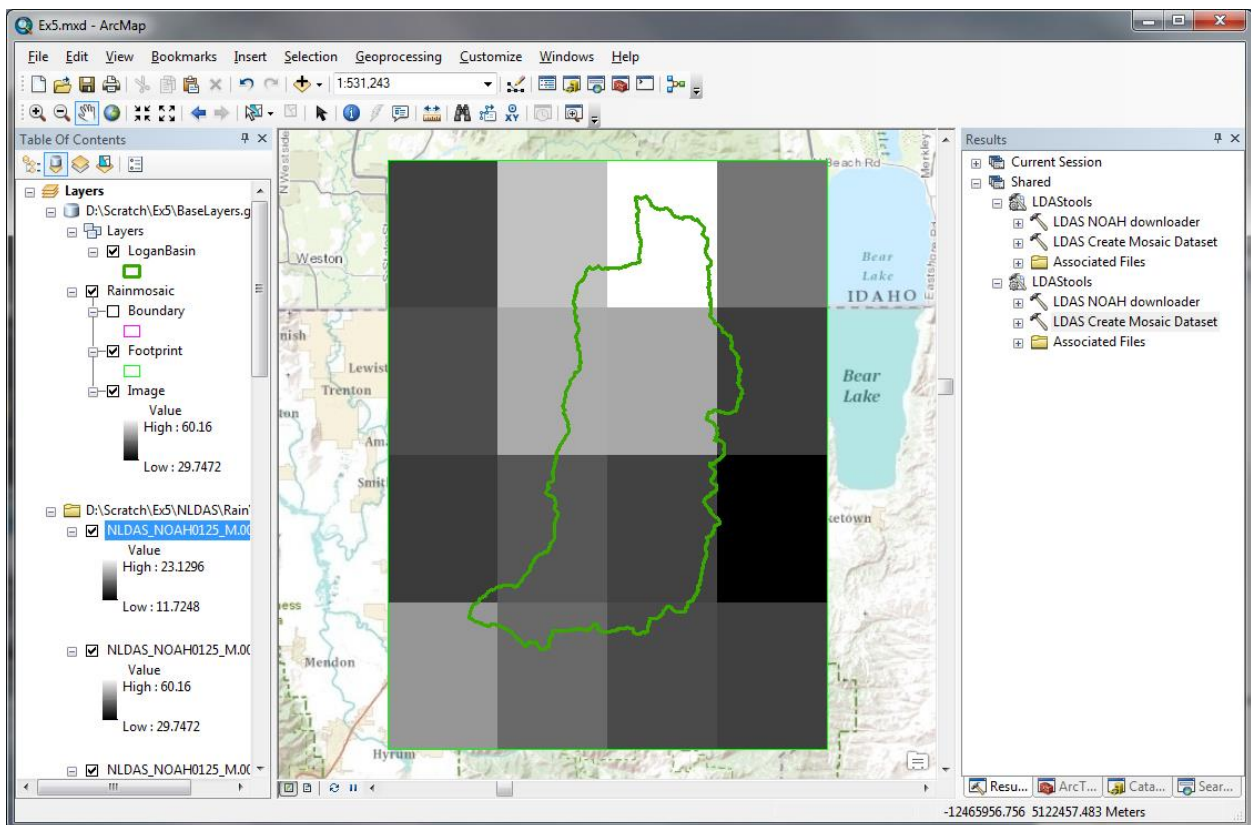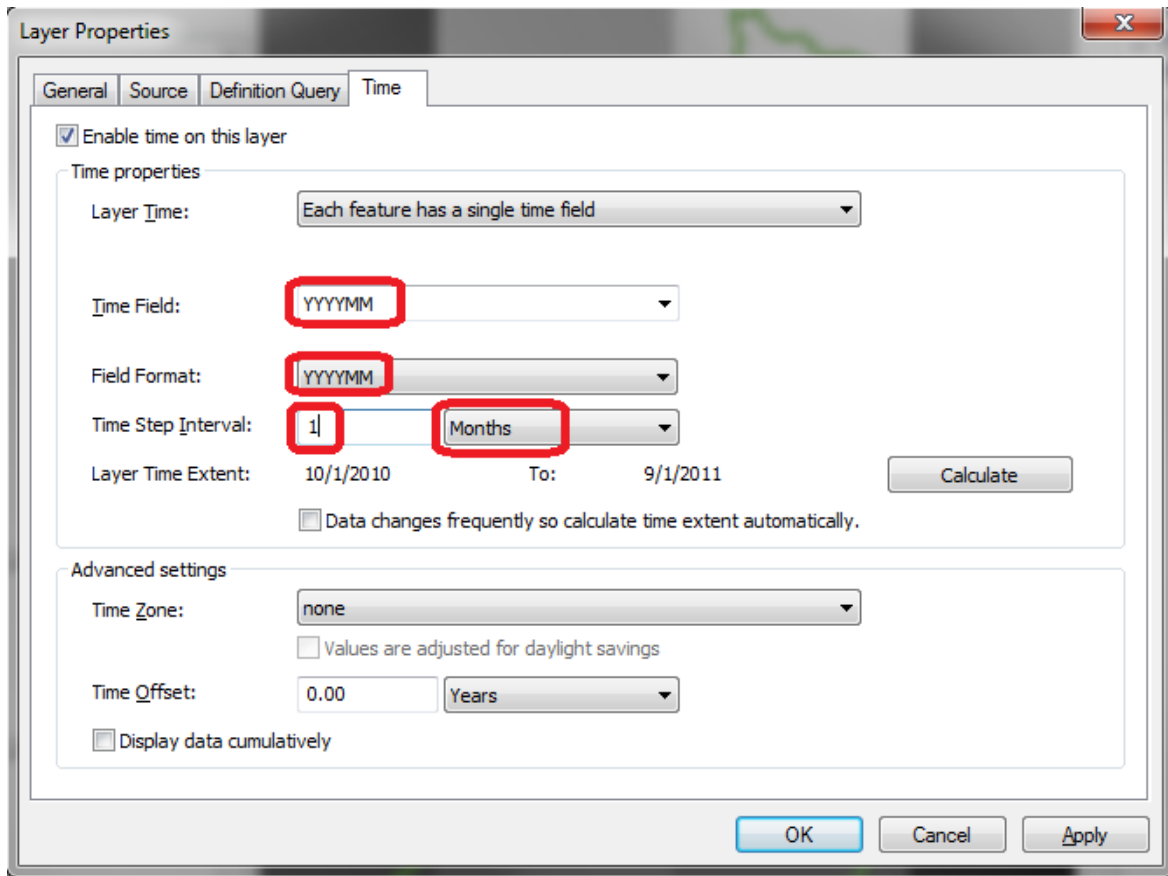


Right click on Rainmosaic > Open > Attribute Table

This opens the attribute table of this Mosaic.



Note that the last column gives date in YYYYMM format.  This will be used to animate this rainfall pattern.  Close the table and open Rainmosaic properties.

Enable time setting the properties as follows

This can now be animated using the Time slider. I found that it was necessary to set the color high/low values as follows so that the image displayed when the range for a time step did not overlap the default which seemed to be the first time step. This is done by right clicking on Image within Rainmosaic.
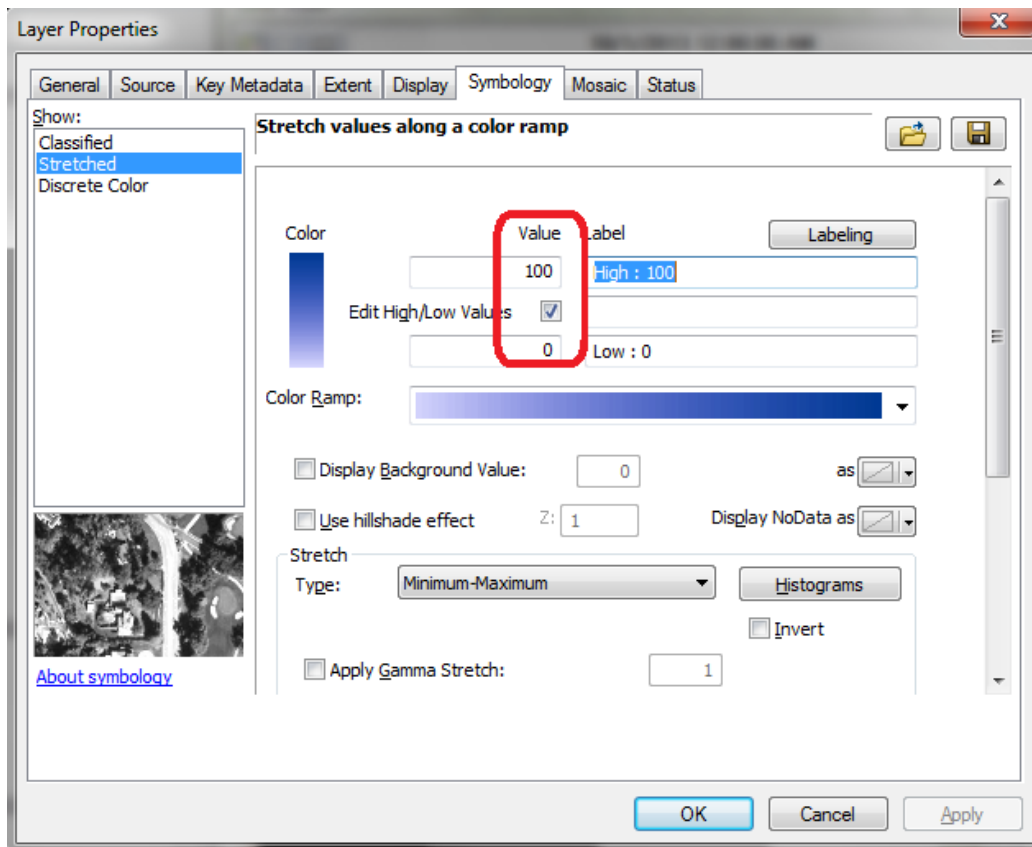
I also adjusted the Image Layer Properties to have the Display use Bilinear Interpolation.
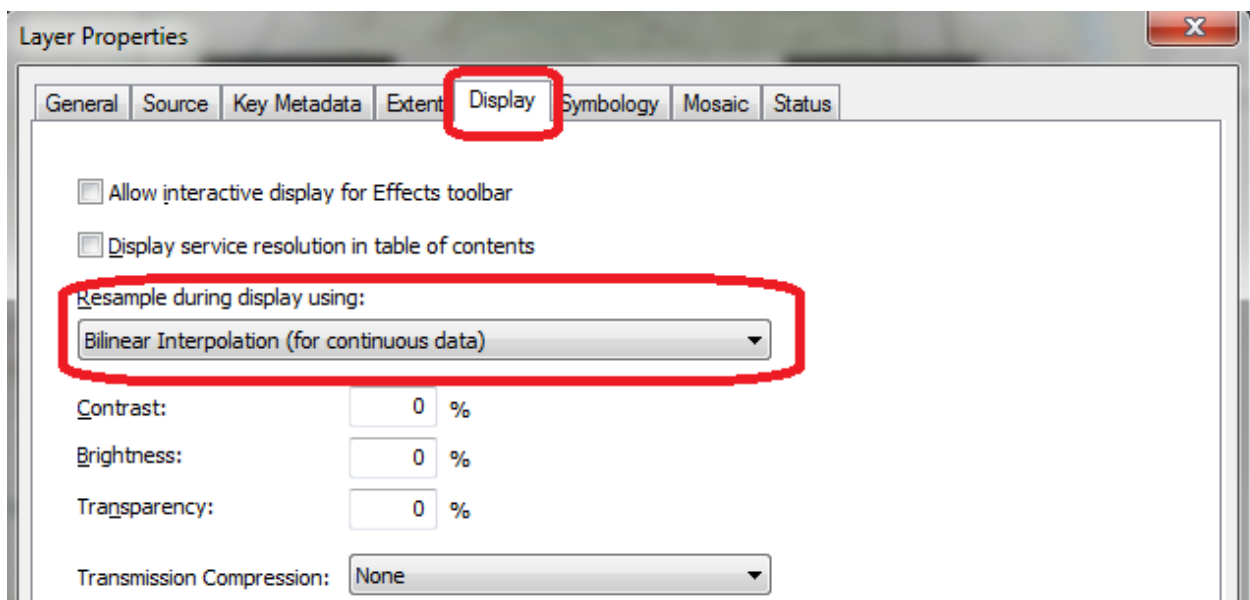


The result is an animation of Rainfall over the Logan River Basin depicted smoothly. However note that the underlying data is still the 1/8 x 1/8 degree grid cells seen earlier.
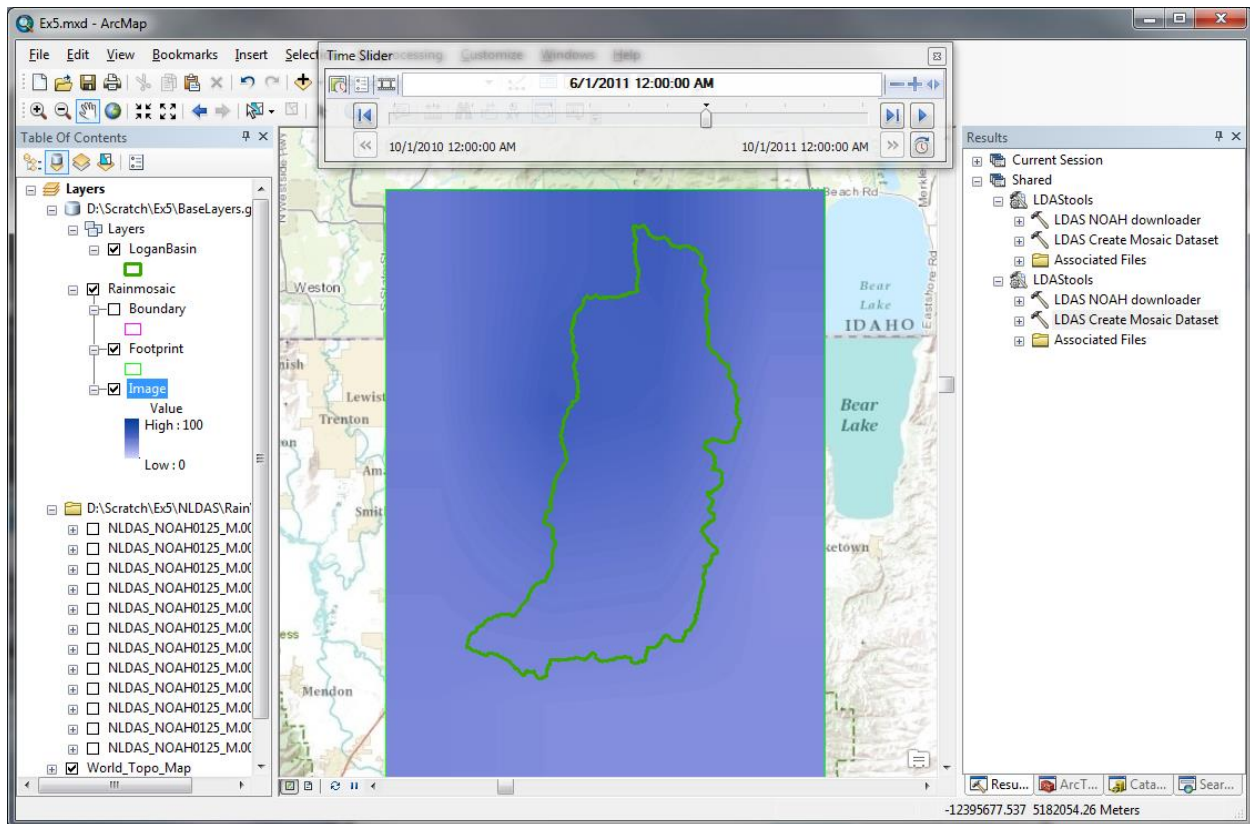
Now we are actually interested in the monthly average rainfall over the watershed. This can be obtained by computing the zonal average for each month. This is a tedious chore, or alternatively a perfect job for programming.

The script **ZonalAverage.py** has been written to achieve this. This has been provided in the Ex5.zip file.

ZonalAverage.py

```
import arcpy, datetime, os
# Python modules used
import arcpy, datetime, os, shutil
from arcpy import env
from arcpy.sa import *

# Inputs -----------------------------------------------------
gdbname="BaseLayers.gdb"   # geodatabase
zones=r"Layers\LoganBasin"  # Name of zone (basins) feature class in geodatabase
mosaic = "Rainmosaic"   # Name of mosaiced layer in geodatabase that is to be zonally averaged
outtab = "Raintab"  # Name of table for zonal averaged data to be written in geodatabase.  The
script will create this file
tempfolder="temp"     # name for temporary folder used for intermediate zone average tables
# End of inputs ----------------------------------------------

arcpy.CheckOutExtension("spatial")   # Check out a spatial analyst license
Folder=os.getcwd() # Use the current working directory as the folder to work in
env.workspace = Folder

# Establish proper paths
zoneshape=gdbname + os.sep + zones  # Basin feature class
theMosaic=gdbname + os.sep + mosaic # The mosaiced layer that is being zonally averaged

# If temporary folder does not exist, create it
```
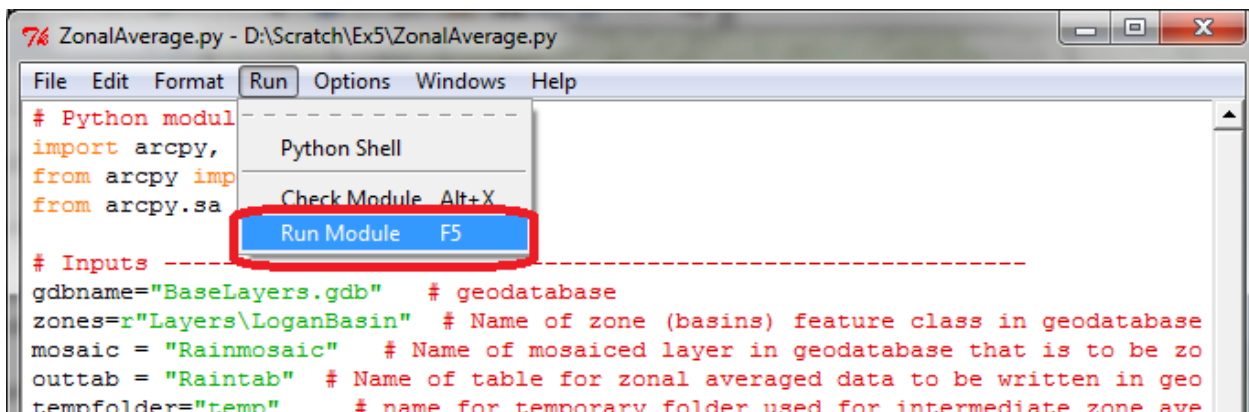
```
if not os.path.isdir(tempfolder):
    os.makedirs(tempfolder)
else:    # Write a message and exit
    print "A folder already exists at the location: " + tempfolder
    print "Delete or move this or use a different temporary folder location"
    quit()

dates=[]  # list to hold dates
vals=[]   # list to hold values
# Loop over all the mosaic values
with arcpy.da.UpdateCursor(theMosaic,["OBJECTID","Raster","YYYYMM"]) as cursor:
    for row in cursor:
        # Deduce date from the YYYYMM column and append it to our list
        dates=dates+[datetime.datetime.strptime(row[2],'%Y%m')]
        #   Create an output table for zonal statistics in the temporary folder
        zonetable=Folder + os.sep + tempfolder + os.sep + "t" + row[2]    # to keep unique name
        outZSaT = ZonalStatisticsAsTable(zoneshape,"OBJECTID",row[1],zonetable,"", "MEAN")
        tableRow = arcpy.UpdateCursor(zonetable)
        # zone table only has one row.  Extract its mean
        for linerow in tableRow:
            meanValue = linerow.MEAN
        vals=vals+[meanValue]  # Append the mean to our list
#  Clean up by removing unused objects to removes locks on the database
del linerow
del tableRow

# Write results to a table
tabnamefull=gdbname+os.sep+outtab
field1="Date"
field2="Value"
arcpy.CreateTable_management(gdbname, outtab,"","")    #  Create Table
arcpy.AddField_management(tabnamefull, field1,"DATE","","","","","")  #  Add date field
arcpy.AddField_management(tabnamefull, field2, "FLOAT","","","","","")  #  Add Value field
rows = arcpy.InsertCursor(tabnamefull,"")
for i in range(len(vals)):
    line = rows.newRow()
    line.setValue(field1,dates[i])
    line.setValue(field2,vals[i])
    rows.insertRow(line)
    print str(dates[i])+","+str(vals[i])

# Clean up
del line
del rows
try:
    if os.path.isdir(tempfolder):  # Remove temporary folder
        shutil.rmtree(tempfolder)
except:
    print "Unable to delete temporary folder: "+tempfolder
print "Done"
```
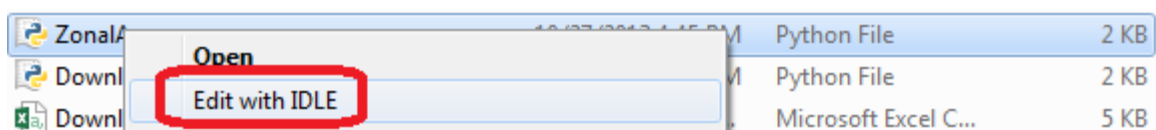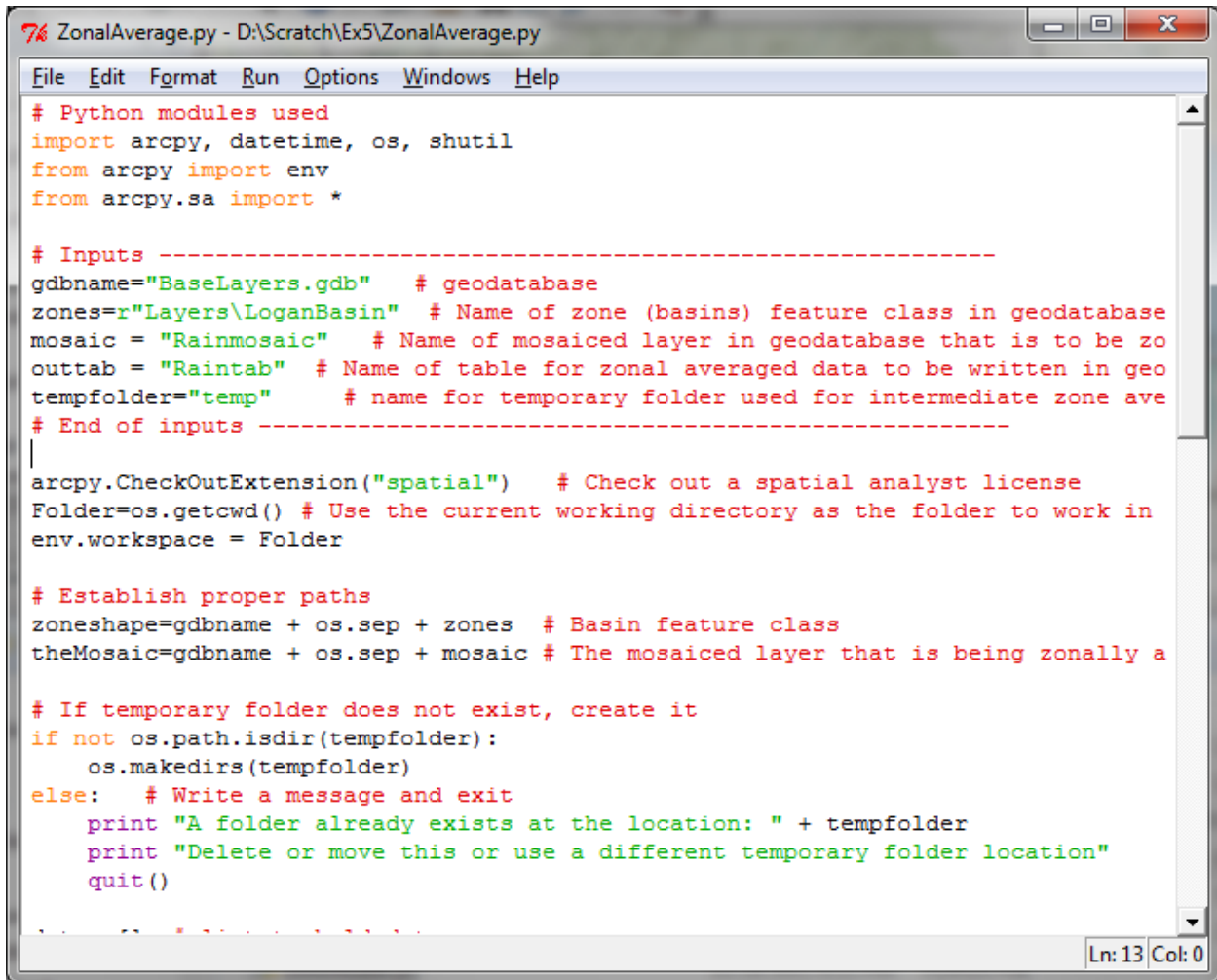
Read this script to get a sense for what it does.  Note that uses as current working directory the directory it is in (location where you unzipped Ex5.zip).  It iterates over each record in the Mosaic and uses a temporary folder to write the tables that result from zonal statistics on the corresponding layer.

There are many ways to execute a Python Script.  Python (which was installed with ArcGIS) includes an editor IDLE.  Right click on ZonalAverage.py (in a Windows Explorer window) and select Edit with IDLE.

You should get the following Window.



This provides an editor in which you can change the code.  For this first run we do not need to change the code.

Select Run > Run Module.

After a short while this script will complete.  You should see the following in the Python Shell



The numbers printed are the dates and zonal average statistics that the script printed to the screen as well as writing to a table.  Close the Python Shell and Idle editor window.

In Catalog right click on BaseLayers and select Refresh.
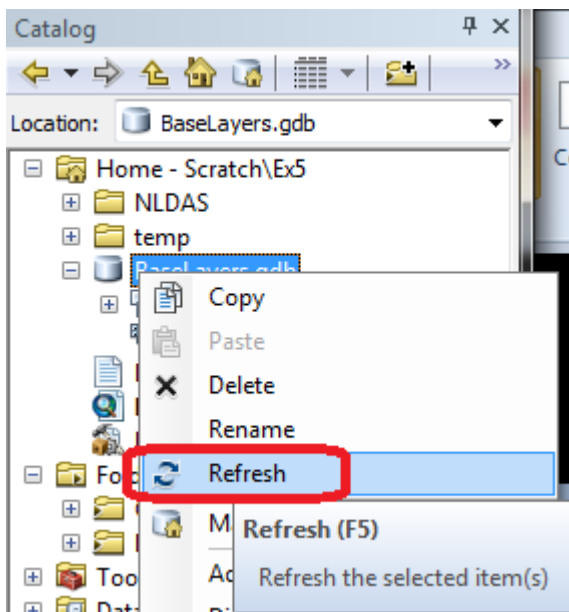


You should now see Raintab in the BaseLayers geodatabase.
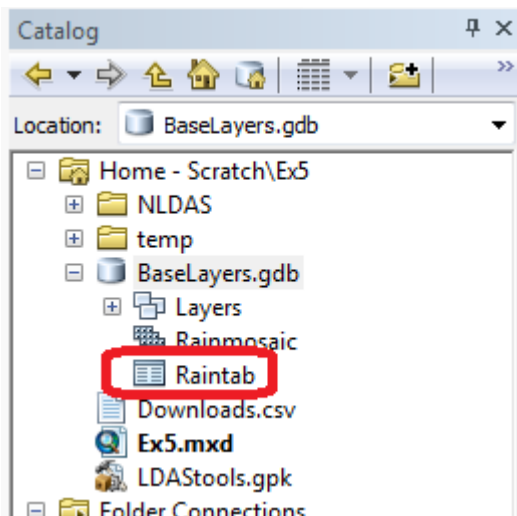
Drag this onto the table of contents and open it.



This is a table of zonally averaged monthly rainfall data for this watershed from zonal statistics on each raster in the Rainfall mosaic.

Use Create Graph to create a simple chart of this data.
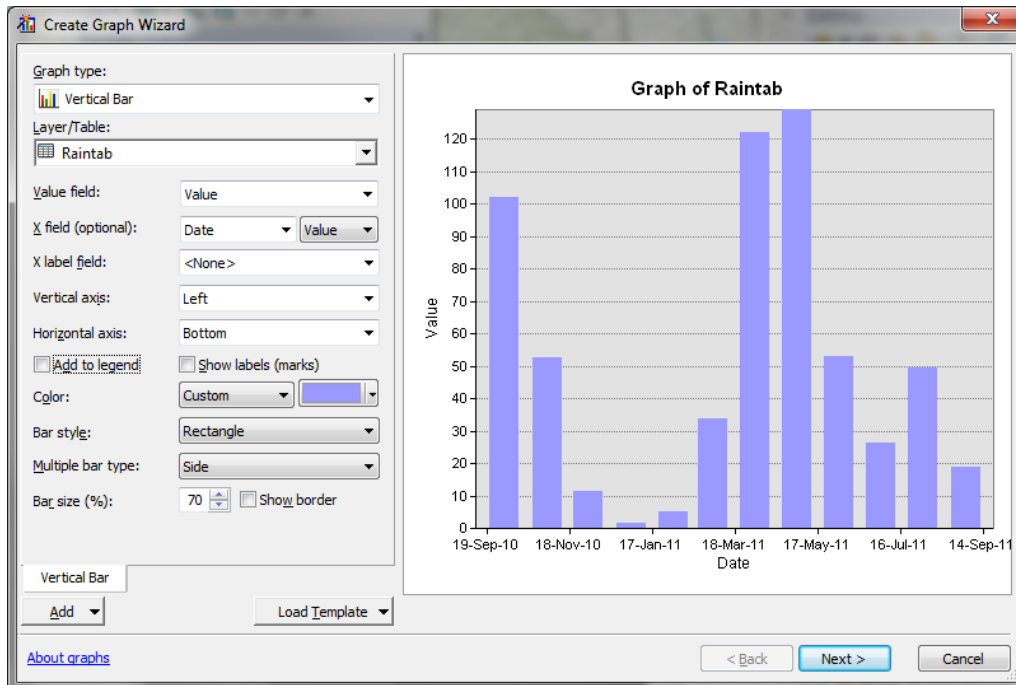
The rainfall values are in kg/m$^2$ which since the density of water is 1000 kg/m$^3$ and since there are 1000 mm/m, correspond to mm of precipitation totalled for each month.

To complete a water balance export this table and work with it in Excel.  Also import into Excel the monthly streamflow from the USGS NWIS gage (Logan is 10109000 identified above) for the corresponding period.  The area of the Logan River basin is 214 mi$^2$.  To convert discharge from cfs to Runoff in mm/month averaged over the area the conversion factor is:

$$Days\ in\ month \times 24 \times 3600/(214 \times 5280^2) \times 304.8 = Days\ in\ month \times 0.004414$$

Note that I am using here the term "Discharge" to represent a volumetric flow rate, and "Runoff" to represent a per unit area flow rate.  The resulting spreadsheet that I worked out is:

| | Area | | 214 mi^2 | Conversion | | 0.00441415 |
|---|---|---|---|---|---|---|
| Date | Rainfall | Year | Month | Discharge | Days | Discharge (mm) |
| 10/1/2010 0:00 | 101.9264 | 2010 | 10 | 98.4 | 31 | 13.46 |
| 11/1/2010 0:00 | 52.4032 | 2010 | 11 | 114.6 | 30 | 15.18 |
| 12/1/2010 0:00 | 11.42613 | 2010 | 12 | 112.3 | 31 | 15.37 |
| 1/1/2011 0:00 | 1.5872 | 2011 | 1 | 109.9 | 31 | 15.04 |
| 2/1/2011 0:00 | 4.945067 | 2011 | 2 | 101.8 | 28 | 12.58 |
| 3/1/2011 0:00 | 33.8432 | 2011 | 3 | 133.7 | 31 | 18.30 |
| 4/1/2011 0:00 | 121.8816 | 2011 | 4 | 302.3 | 30 | 40.03 |
| 5/1/2011 0:00 | 129.1157 | 2011 | 5 | 769 | 31 | 105.23 |
| 6/1/2011 0:00 | 52.81067 | 2011 | 6 | 1309 | 30 | 173.34 |
| 7/1/2011 0:00 | 26.22933 | 2011 | 7 | 1047 | 31 | 143.27 |
| 8/1/2011 0:00 | 49.32267 | 2011 | 8 | 405.5 | 31 | 55.49 |
| 9/1/2011 0:00 | 18.97387 | 2011 | 9 | 248.2 | 30 | 32.87 |
| | | | | | | |
| Total | **604.47** | | | | | **640.15** |

This indicates 604 mm of rainfall over the Logan Basin and a total of nearly 800 mm of Runoff!  Does this balance?  What is missing?  Snow!!

The above calculations used only "Rainfall (unfrozen precipitation)".  There is also a variable in NLDAS for snow, "Snowfall (frozen precipitation)".

Use the LDAS NOAH downloader to download "Snowfall (frozen precipitation)" for the Logan River. Repeat the procedure above to evaluate zonal averages.  Compute total precipitation as the sum of rainfall and snowfall.  Use Excel to prepare a graph of Monthly Discharge and Monthly Total Precipitation for the Logan River Basin for the 2011 water year.  Determine the runoff ratio as the ratio of annual discharge (expressed in mm) to annual precipitation.

*To be turned in:  A graph showing the total monthly precipitation, snowfall and monthly discharge (expressed as mm/month) for the Logan River basin for the 2011 water year.  Report the total precipitation (in mm), total runoff (in mm)*

**Other LDAS variables.**

You may have noticed other variables as options to download using the LDAS NOAH downloader.  NOAH is a complete land surface hydrology model used as the land surface parameterization in some climate and weather models.  As such it performs quite elaborate water and energy balance calculations.  The following variables are involved in the LDAS total land surface water balance.

Flux variables

| | |
|---|---|
| RF | Rainfall (unfrozen precipitation) (kg/m^2) |
| SF | Snowfall (frozen precipitation) (kg/m^2) |
| Qsurf | Surface runoff (non-infiltrating) (kg/m^2) |

Qbase       Subsurface runoff (baseflow) (kg/m^2)

ET          Total evapotranspiration (kg/m^2)

Storage variables

SM          0-200 cm Total column soil moisture content (kg/m^2)

SWE        Accumulated snow water-equivalent (kg/m^2)

Cstor      Plant canopy surface water (kg/m^2)
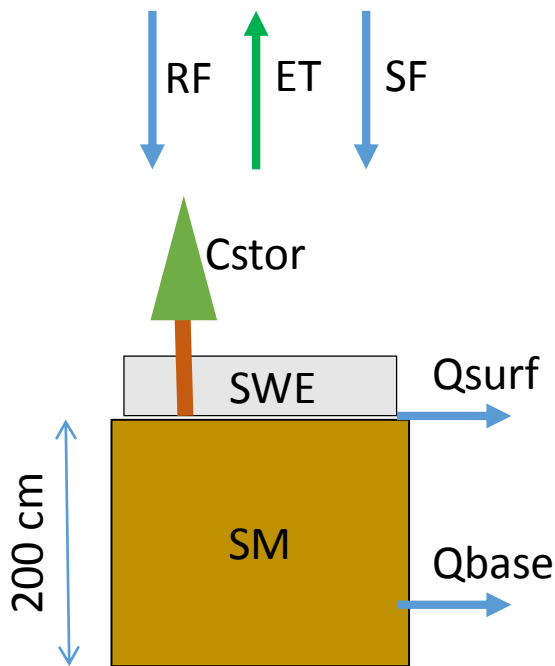


A complete model water balance can be evaluated as

$$(SM + SWE + Cstor)_{t2} - (SM + SWE + Cstor)_{t1} = RF + SF - Qsurf - Qbase - ET$$

In evaluating this energy balance it is more precise if the storages are recorded instantaneously at the beginning and end of the period involved. Thus it is better for the storage variables to retrieve NLDAS hourly variables and for the flux variables to retrieve NLDAS monthly variables (if a monthly water balance is sought).

This is a whole lot of data to download and manage, really a job for programming.

The Script Downloads.py was developed to automate the retrieval of these variables.

Downloads.py

```
# -------------------------------------------------------------------------
# Downloads.py
# Created on: 2013-10-27
```

```
#    David Tarboton
# Description:
#   This script automates the repetitive downloads of NLDAS data over a watershed
#   The toolbox location needs to be edited for your system
#   The file Downloads.csv needs to be edited to control the variables to download
# -------------------------------------------------------------------------

# Import arcpy module
import arcpy, datetime, os
from arcpy import env

# Inputs -----------------------------------------------------------
gdbname="BaseLayers.gdb"    # geodatabase
zones=r"Layers\LoganBasin"  # Name of zone (basins) feature class in geodatabase
infile="Downloads.csv"
# End of inputs ----------------------------------------------------

# Load the LDAS toolbox.  Adjust this line to where it occurs in your system
arcpy.ImportToolbox("C:/Users/dtarb/Documents/ArcGIS/Packages/LDAStools/v101/LDAStools
.tbx")

# Local variables:
# Use the current working directory as the folder
Folder=os.getcwd()  # or r"D:\Scratch\Ex5"
env.workspace = Folder

Basin = Folder+os.sep+gdbname+os.sep+zones

Outfold=Folder+os.sep+"NLDAS"
# If this folder does not exist make it
if not os.path.isdir(Outfold):
    os.makedirs(Outfold)

f=open(infile)
line=f.readline()  # reads the header
for line in f:
    print(line)
    cols=line.split(",")
    Outfolder=Outfold+os.sep+cols[0]
    if not os.path.isdir(Outfolder):
        os.makedirs(Outfolder)
    Dataset=cols[1]
    Var=cols[2]
    Begdate=cols[3]
    Enddate=cols[4]
    arcpy.LDASNOAHdownloader_LDASNOAHdownloader(Basin, Dataset,Var, Outfolder,
Begdate, Enddate)
```

Downloads.py reads from the file **Downloads.csv**, and downloads the variables indicated for the
specified date range.

<u>Downloads.csv</u>
```
Folder,Dataset,Variable,Begdate,Enddate
RF,NLDAS-2 Monthly 1/8 degree,Rainfall (unfrozen precipitation)
(kg/m^2),10/1/2012,9/1/2013
SF,NLDAS-2 Monthly 1/8 degree,Snowfall (frozen precipitation)
(kg/m^2),10/1/2012,9/1/2013
Qsurf,NLDAS-2 Monthly 1/8 degree,Surface runoff (non-infiltrating)
(kg/m^2),10/1/2012,9/1/2013
```

```
Qbase,NLDAS-2 Monthly 1/8 degree,Subsurface runoff (baseflow)
(kg/m^2),10/1/2012,9/1/2013
ET,NLDAS-2 Monthly 1/8 degree,Total evapotranspiration (kg/m^2),10/1/2012,9/1/2013
...
```

Columns in Downloads.csv are

- Folder:  The name of the Folder into which rasters should be retrieved
- Dataset:  The name of the NLDAS source dataset to use, either Hourly or Monthly
- Variable:  The name of the NLDAS variable to retrieve
- Begin Date:  The start date of the interval to use
- End date:  The end date of the interval to use

These correspond to the following inputs in the LDAS NOAH downloader tool
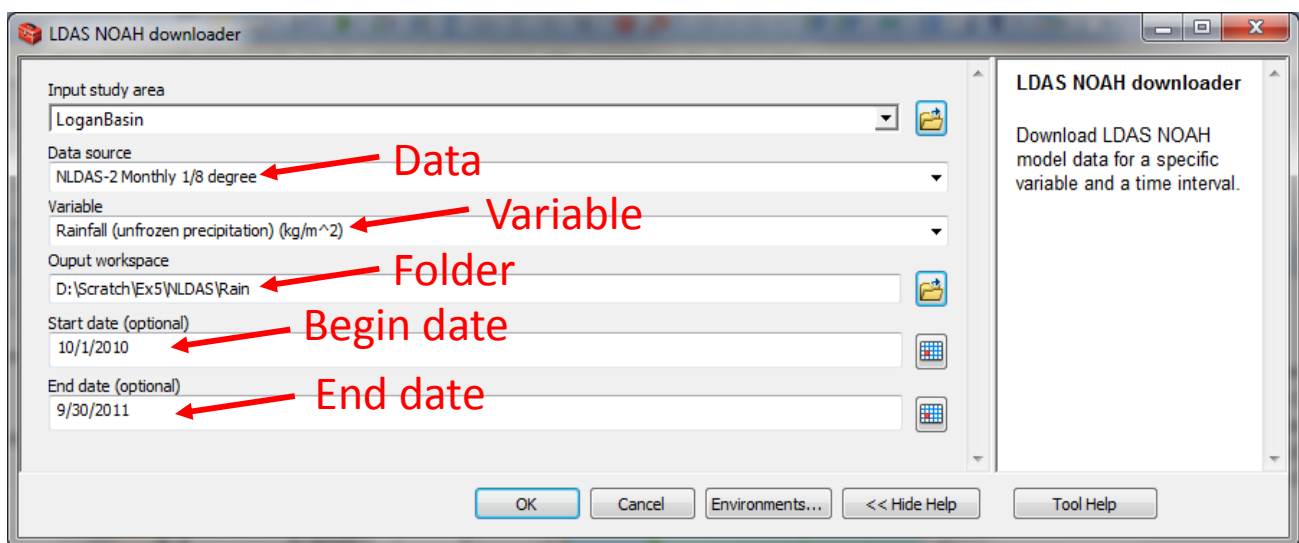


Downloads.csv as provided has been configured to download all the data needed for the 2011 water year water balance.  Outputs are written to folders in a folder named NLDAS.

Delete the folders Rain and Snow in the NLDAS folder so as to not mix this data with the data you area about to download.

Open Idle and examine the **Downloads.py** script.  Note in line 22 that it uses the LDAStools toolbox that was provided.  When this tool is opened from Catalog at the step on page 22
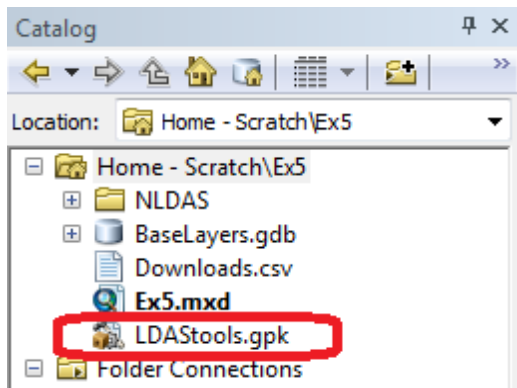
before downloading NLDAS data it results in a toolbox being placed in a user's default ArcGIS documents folder.  In my case this is "C:/Users/dtarb/Documents/ArcGIS/Packages/LDAStools/v101/LDAStools.tbx".  Locate this folder on your computer and once you have determined the correct path to it, edit line 22 in Downloads.py to where it occurs on your system.



Then run **Downloads.py**.   Click Yes if prompted to save the script.

The result should be a series of folders in the NLDAS folder each of which contains the requested rasters in ASCII format.

The folder name from the first column in Downloads.csv identifies the variable.

The program **ZoneAverageAll.py** has been written to perform zonal averages of all these files.

ZoneAverageAll.py

```python
# Python modules used
import sys
import arcpy, datetime, os, shutil
from arcpy import env
from arcpy.sa import *

# Inputs -------------------------------------------------------
gdbname="BaseLayers.gdb"   # geodatabase
zones=r"Layers\LoganBasin"  # Name of zone (basins) feature class in geodatabase
outtable="zstats"
# End of inputs -------------------------------------------------

# Check out the ArcGIS Spatial Analyst extension for using the zonal statistics
function
arcpy.CheckOutExtension("spatial")

# Use the current working directory as the folder to work in
Folder=os.getcwd()  # or r"D:\Scratch\Ex5"
env.workspace = Folder

# Full file paths
zoneshape=gdbname + os.sep + zones  # Basin feature class
WorkFolder = Folder+os.sep+"NLDAS"  # This is the folder used for input data
TempFolder = Folder+os.sep+"temp"  # This is the folder where temporary tables are
written
if not os.path.isdir(TempFolder):
    os.makedirs(TempFolder)
VarFolders=os.listdir(WorkFolder)
tabnamefull=gdbname+os.sep+outtable
field1="Varcode"
field2="Date"
field3="Value"
field4="TimeSupport"
arcpy.CreateTable_management(gdbname, outtable,"","")   #  Create Table
arcpy.AddField_management(tabnamefull, field1,"TEXT","","","","","")  #  Add code
field
arcpy.AddField_management(tabnamefull, field2, "DATE","","","","","")  #  Add date
field
arcpy.AddField_management(tabnamefull, field3,"FLOAT","","","","","")  #  Add value
field
```
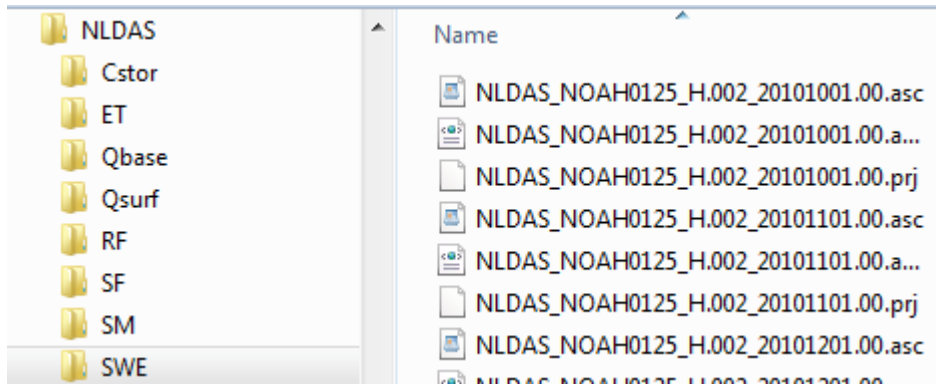
```
arcpy.AddField_management(tabnamefull, field4, "TEXT","","","","","")  #  Add support
field
rows = arcpy.InsertCursor(tabnamefull,"")

for VarFolder in VarFolders:
    VarFiles=os.listdir(WorkFolder+os.sep+VarFolder)
    for theFile in VarFiles:
        if theFile.endswith(".asc"):  # Only ASC files
            if(theFile[-6:-4]=="00"):    # This occurs for an hourly file
                dateString = theFile[-15:-7]
                timeSupport="Hour"
            else:
                dateString = theFile[-10:-4]
                timeSupport="Month"
            #print VarFolder + " " + theFile + " " + dateString

            zonetable=TempFolder + os.sep +VarFolder+ dateString   # to keep unique
name
            fullFile=WorkFolder+os.sep+VarFolder+os.sep+theFile
            outZSaT = ZonalStatisticsAsTable(zoneshape, "OBJECTID", fullFile,
                                    zonetable, "", "MEAN")
            tableRow = arcpy.UpdateCursor(zonetable)
        # zone table only has one row.  Extract its mean
            for linerow in tableRow:
                meanValue = linerow.MEAN
            print VarFolder + " " + dateString + " " + str(meanValue)
            line = rows.newRow()
            if timeSupport=="Hour":
                date=datetime.datetime.strptime(dateString,'%Y%m%d')
            else:
                date=datetime.datetime.strptime(dateString,'%Y%m')
            line.setValue(field1,VarFolder)
            line.setValue(field2,date)
            line.setValue(field3,meanValue)
            line.setValue(field4,timeSupport)
            rows.insertRow(line)
# Clean up
del line
del rows
del linerow
del tableRow
# redo last zonal statistics.  Append "t" to name to make it different.  This seems
necessary to delete the locks
outZSaT = ZonalStatisticsAsTable(zoneshape, "OBJECTID", fullFile,
                                zonetable+"t", "", "MEAN")
try:
    if os.path.isdir(TempFolder):  # Remove temporary folder
        shutil.rmtree(TempFolder)
except:
    print "Unable to delete temporary folder: "+TempFolder
print "Done"
```

Open Idle and run **ZoneAverageAll.py**.

The result is a table **zstats** in the BaseLayers geodatabase.  Open this table to examine the zonally averaged water balance variables.

| | OBJECTID * | Varcode | Date | Value | Time Support |
|---|---|---|---|---|---|
| ▶ | 1 | Cstor | 10/1/2010 | 0 | Hour |
| | 2 | Cstor | 11/1/2010 | 0.499867 | Hour |
| | 3 | Cstor | 12/1/2010 | 0.499933 | Hour |
| | 4 | Cstor | 1/1/2011 | 0.5 | Hour |
| | 5 | Cstor | 2/1/2011 | 0.5 | Hour |
| | 6 | Cstor | 3/1/2011 | 0.5 | Hour |
| | 7 | Cstor | 4/1/2011 | 0.5 | Hour |
| | 8 | Cstor | 5/1/2011 | 0.357933 | Hour |
| | 9 | Cstor | 6/1/2011 | 0.166667 | Hour |
| | 10 | Cstor | 7/1/2011 | 0 | Hour |
| | 11 | Cstor | 8/1/2011 | 0 | Hour |
| | 12 | Cstor | 9/1/2011 | 0 | Hour |
| | 13 | Cstor | 10/1/2011 | 0 | Hour |
| | 14 | ET | 10/1/2010 | 24.2128 | Month |
| | 15 | ET | 11/1/2010 | 10.83043 | Month |
| | 16 | ET | 12/1/2010 | 3.780733 | Month |
| | 17 | ET | 1/1/2011 | 4.3698 | Month |

1 ▶ ▶I | (12 out of 99 Selected)

zstats

Use select by attributes to identify a specific variable of interest.  These variables can be plotted using the ArcGIS chart tool, but you can probably more easily get the flexibility you need exporting the data to Excel and plotting it.

Note that the downloads.csv file was configured to download beginning of the month hourly values for the storage variables and monthly averages for the flux variables.  This is so that the flux during a period can be related to the change in storage from the beginning to the end of the period.
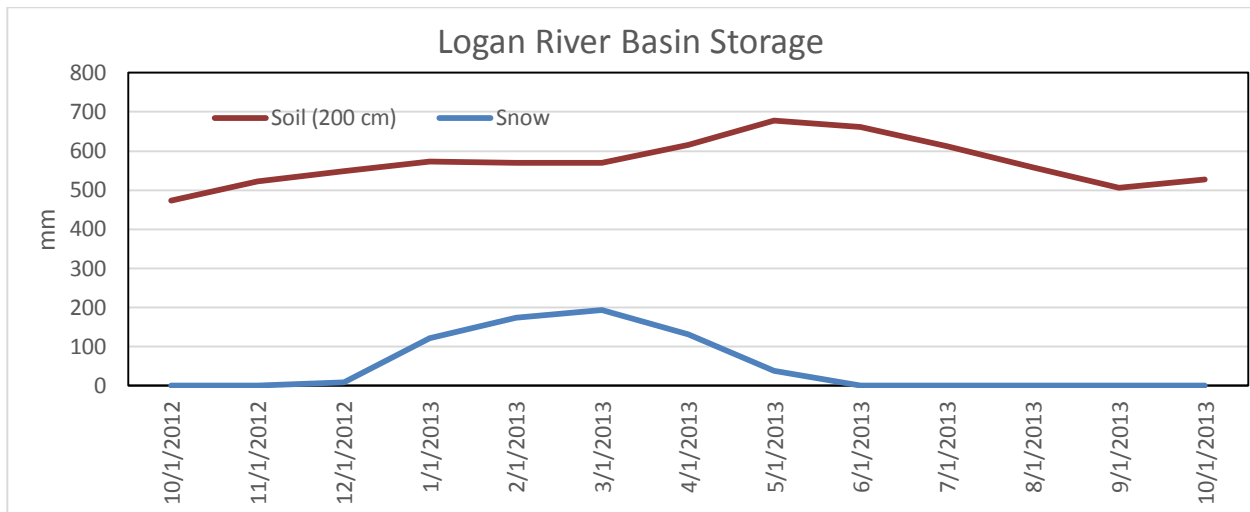
Following are the sort of results that can be produced from this data.
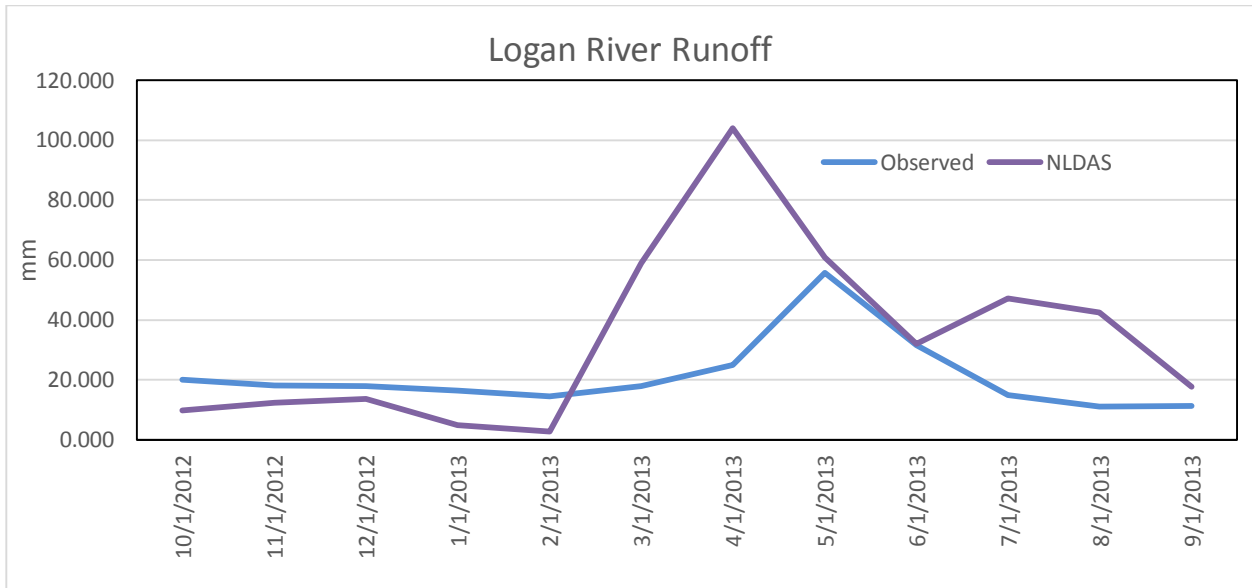
Note the difference between NLDAS modeled and observed runoff.
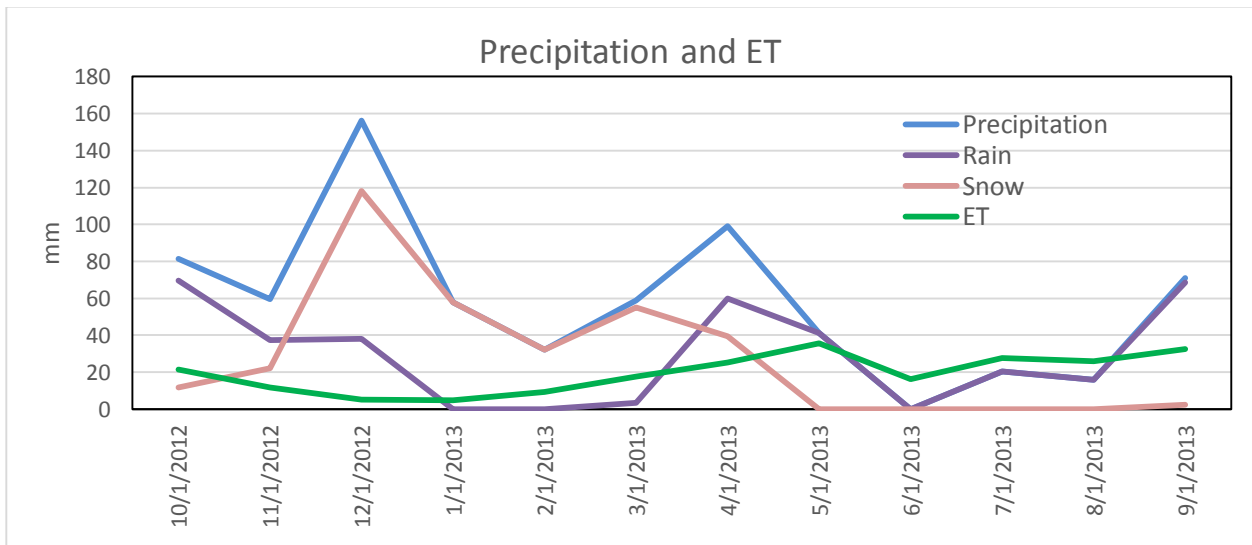


Note the seasonality of the Rain, Snow and ET fluxes.

Logan River Basin Storage

Note the soil column and snow storage present in the Logan River Basin.

The following spreadsheet gives the 2013 water year Logan River water balance.

| LDAS/NOAH Water Balance | | | | |
|---|---|---|---|---|
| Logan River 2013 Water Year.  Storages in kg/m^2 = mm | | | | |
| | | | | |
| | 10/1/2012 | 10/1/2013 | Change/flux | |
| SM | 472.9 | 526.6 | 53.8 | |
| SWE | 0.0 | 0.0 | 0.0 | |
| Cstor | 0.0 | 0.0 | 0.0 | |
| RF | | | 354.6 | |
| SF | | | 338.7 | |
| Qbase | | | 194.6 | |
| Qsurf | | | 211.5 | |
| ET | | | 233.4 | |
| | | | | |
| | In | RF+SF | 693.3 | |
| | Out | Qbase+Qsurf+ET | 639.4 | |
| | Storage Increase | SM+SWE+Cstor | 53.8 | |
| | | In -Out | 53.8 | |
| | | Closure Error | 0.1 | |

*To be turned in:   Figures such as the above for the Logan River basin for the 2011 water year.  Report the following quantities for the 2011 water year (10/1/2010-9/30/2011).*

- *Year precipitation (in mm)*
- *Year runoff from NLDAS (in mm)*
- *Year runoff observed (in mm)*
- *Year Evaporation (in mm)*
- *Year precipitation in the form of rain (in mm)*

44

- *Year precipitation in the form of snow (in mm)*
- *Net change is soil moisture storage during the year (in mm)*

OK, you are done!

## Summary of Items to turn in.

1. *Use a sequence of screen shots to discuss the changing patterns of Total Water Storage across the United States and in Texas or Utah in 2011 and 2012.*

2. *Screen shots of streamflow hydrographs from*
    a) *The Blanco River at Wimberley Texas*
    b) *The Logan River above State Dam Near Logan, Utah.*
    c) *Some other location in the world that has streamflow of interest to you*

   *Also include screen shots that show maps of these locations*

    d) *What was the magnitude of the recent (October 2013 high/flood flows in the Blanco River in Texas)?*
    e) *What was the magnitude of the 2011 high/flood flows in the Logan River above State Dam, near Logan Utah?*

3. *A graph showing the total monthly precipitation, snowfall and monthly discharge (expressed as mm/month) for the Logan River basin for the 2011 water year. Report the total precipitation (in mm), total runoff (in mm)*

4. *Figures such as the above for the Logan River basin for the 2011 water year. Report the following quantities for the 2011 water year (10/1/2010-9/30/2011).*
    - *Year precipitation (in mm)*
    - *Year runoff from NLDAS (in mm)*
    - *Year runoff observed (in mm)*
    - *Year Evaporation (in mm)*
    - *Year precipitation in the form of rain (in mm)*
    - *Year precipitation in the form of snow (in mm)*
    - *Net change is soil moisture storage during the year (in mm)*

## Appendix:  Alternative Python Environment.

Python scripts can be run from any number of Python environments.  I used Pyscripter.  This may be obtained from http://code.google.com/p/pyscripter/downloads/list.  Download **PyScripter-v2.5.3-Setup.exe** from this website.  Even if you have a 64 bit computer, use the 32 bit version as the 64 bit version does not seem to work with ArcGIS.  Double Click on **PyScripter-v2.5.3-Setup.exe** to install it.

To use Pyscripter right click on a .py file and Edit with PyScripter.
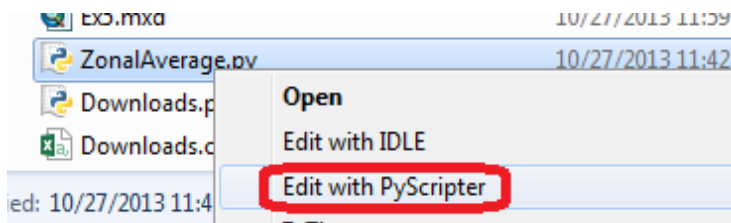


The following PyScripter interface opens
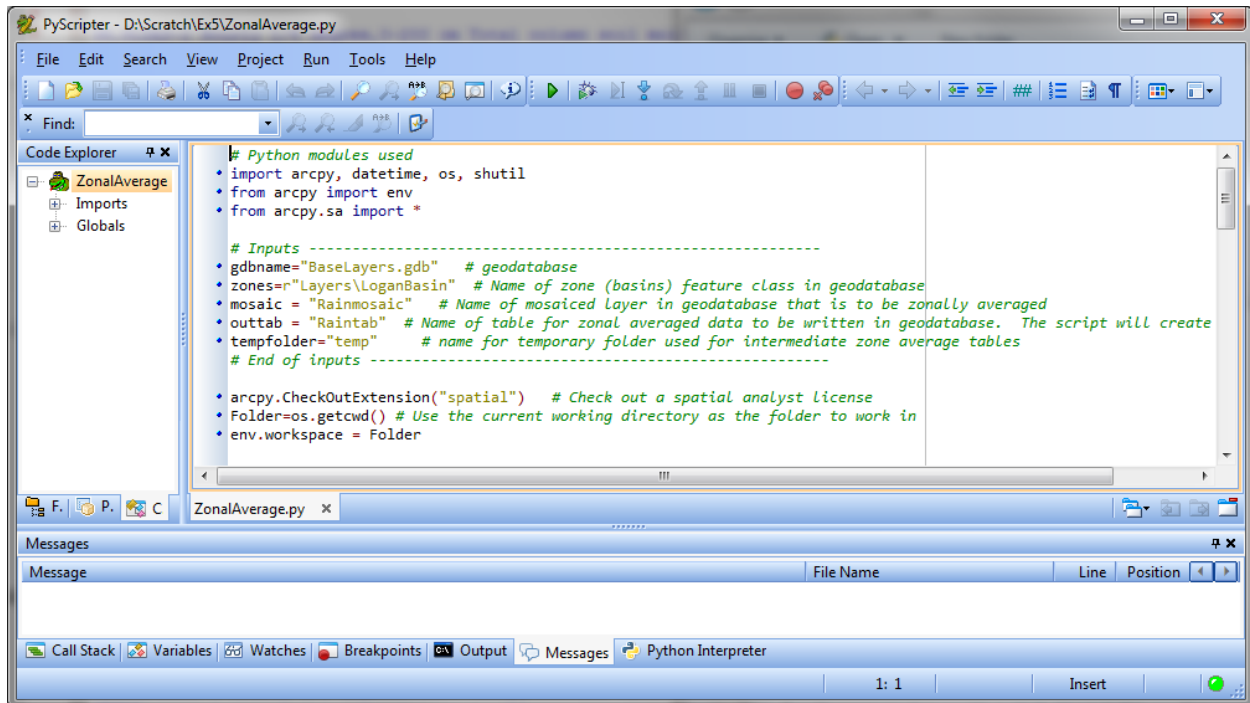


PyScripter provides a more powerful interactive debugging environment that can be used to set break points and step through a Python script. This process is useful in developing these programs where trial and error iteration is necessary.