

# SDA Interpretation demo notes

Sunday 7/16/17

## Contents

SDA Interpretation demo notes.....	1
Introduction .....	1
Resources.....	2
Define a Persistent AOI .....	5
Explore the SDA Soils Layers .....	7
View the SDA Persistent AOI.....	10
SDA Persistent AOIs .....	12
Use Case 1: Import WSS AOI.....	13
Use Case 2: SSA-based AOI .....	17
Use Case 3: Mukey List AOI.....	19
Use Case 4: Ad-Hoc AOI .....	22
Use Case 5: Add Styled Layer Descriptor to AOI.....	25
Other Use Cases.....	31
Interpretations.....	32
Retrieving a Catalog .....	32
Comparing WSS and SDA Interpretation Results.....	40
Specific Interpretation Tests .....	43
Geolocation.....	51
Identify or Draw .....	54
WMS and WFS service links .....	62

## Introduction

Soil Data Access (SDA) now makes it possible to define “Persistent Areas of Interest” (pAOI or AOI) and run “interpretations” (also called “ratings”) and present them through Web maps. This work has been performed under “Soils FY2017 DME task 4” (and earlier) and is viewable in the “NITC Development” environment (“DEV”, access requires the appropriate role and use of the “NAG VPN”).

This document will take you on a quick tour of the new features in SDA. Please note that the “Test” pages presented were created for SDA development and QA testing – they are not “production quality”

and intentionally allow the user to stress SDA in ways that it probably should not be used. They should only be used in Chrome (use in Firefox has not been extensively tested).

## Resources

A number of documents are referenced herein, here are the links:

<b>Soil Data Access</b>	<a href="https://sdmdataaccess-dev.dev.sc.egov.usda.gov/">https://sdmdataaccess-dev.dev.sc.egov.usda.gov/</a>
<b>Web Soil Survey</b>	<a href="https://websoilsurvey-dev.dev.sc.egov.usda.gov/App/WebSoilSurvey.aspx">https://websoilsurvey-dev.dev.sc.egov.usda.gov/App/WebSoilSurvey.aspx</a>
<b>Web Service Help</b>	<a href="https://sdmdataaccess-dev.dev.sc.egov.usda.gov/webservicehelp.aspx">https://sdmdataaccess-dev.dev.sc.egov.usda.gov/webservicehelp.aspx</a>
<b>Layer Help</b>	<a href="https://sdmdataaccess-dev.dev.sc.egov.usda.gov/LayerHelp.htm">https://sdmdataaccess-dev.dev.sc.egov.usda.gov/LayerHelp.htm</a>
<b>"Advanced Queries"</b>	<a href="https://sdmdataaccess-dev.dev.sc.egov.usda.gov/documents/AdvancedQueries.html">https://sdmdataaccess-dev.dev.sc.egov.usda.gov/documents/AdvancedQueries.html</a>
<b>Test Post page</b>	<a href="https://sdmdataaccess-dev.dev.sc.egov.usda.gov/test/testpost.html">https://sdmdataaccess-dev.dev.sc.egov.usda.gov/test/testpost.html</a>
<b>Test WMS page</b>	<a href="https://sdmdataaccess-dev.dev.sc.egov.usda.gov/test/testwms.html">https://sdmdataaccess-dev.dev.sc.egov.usda.gov/test/testwms.html</a>
<b>Query page</b>	<a href="https://sdmdataaccess-dev.dev.sc.egov.usda.gov/query.aspx">https://sdmdataaccess-dev.dev.sc.egov.usda.gov/query.aspx</a>
<b>JSON Formatter</b>	<a href="https://www.freeformatter.com/json-formatter.html">https://www.freeformatter.com/json-formatter.html</a>
<b>XML Formatter</b>	<a href="https://www.freeformatter.com/xml-formatter.html">https://www.freeformatter.com/xml-formatter.html</a>
<b>This document</b>	<a href="https://sdmdataaccess-dev.dev.sc.egov.usda.gov/test/SDA_Interps_demo_notes.pdf">https://sdmdataaccess-dev.dev.sc.egov.usda.gov/test/SDA_Interps_demo_notes.pdf</a>

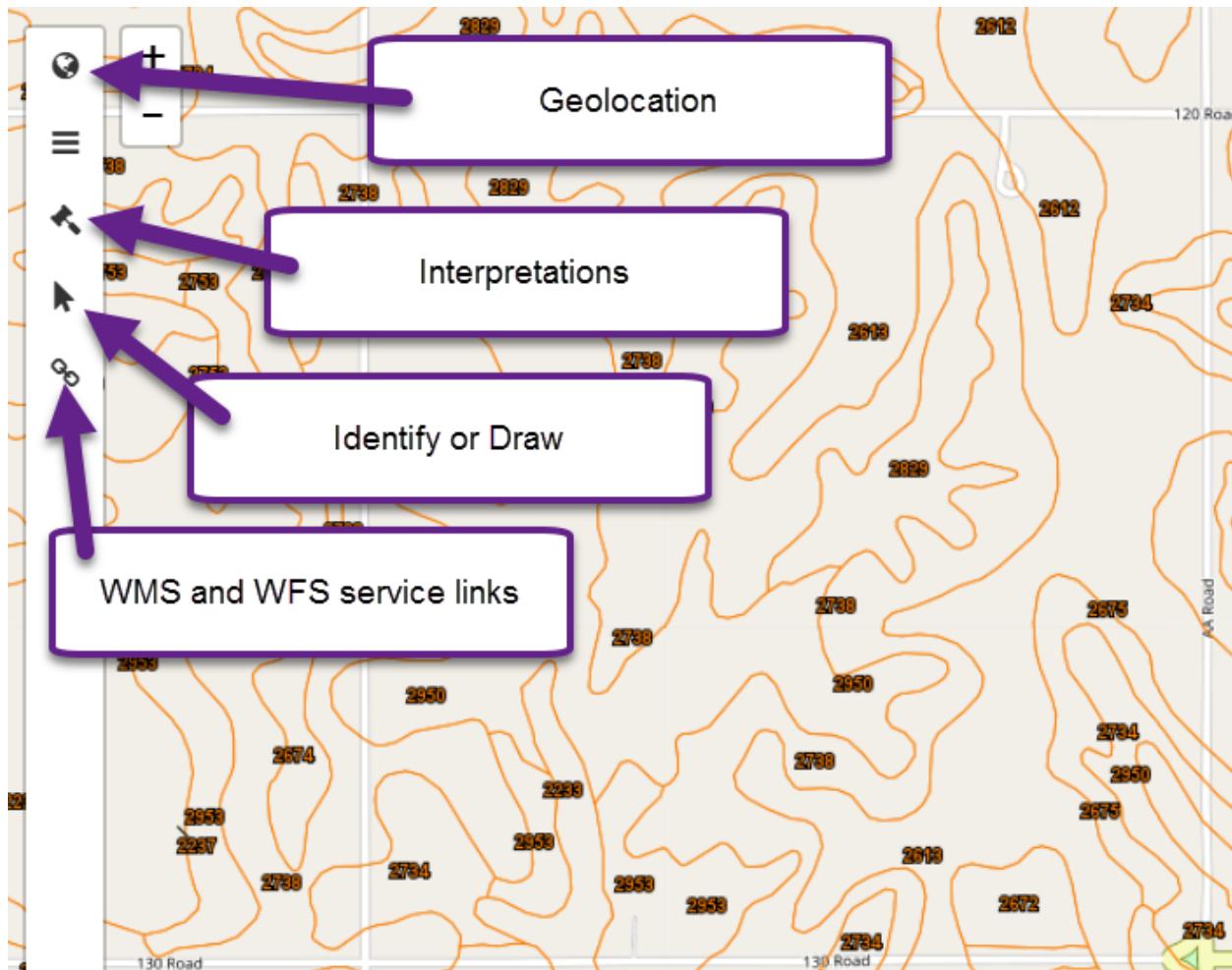
By default the Test WMS (WMS = Web Map Service) page displays a background map provided by Open Street Maps (OSM). You may switch to one of three Microsoft Bing maps by use of a Microsoft-issued access key.

## Late-breaking news!

Some of the screen shots that follow show a “sidebar” that looks like this:



The version just released into the development environment adds a few more pull-out options:



The use of these additional links will be covered in the same-named sections towards the end of this document.

Most of the functionality described here “should” work on a touch device (especially the “drawing” capability documented in the “Identify or Draw” section). This has not yet been tested. In addition the Geolocation capability is known not to work in Internet Explorer version 11 on some CTE-managed notebooks. The TestWMS page was developed using the Chrome browser, it was found to work on Firefox as well. It has not been tested in the Safari browser or on any mobile device.

## Quick Start

The Test Post page will be used to define an SDA “persistent AOI” from a set of coordinates, the Test WMS page will be used to display the fruits of tasks 1, 2 and 3. Begin by establishing a NAG VPN connection and bringing up both of these pages in separate tabs of a Chrome (preferred) or Firefox Web browser.

### Define a Persistent AOI

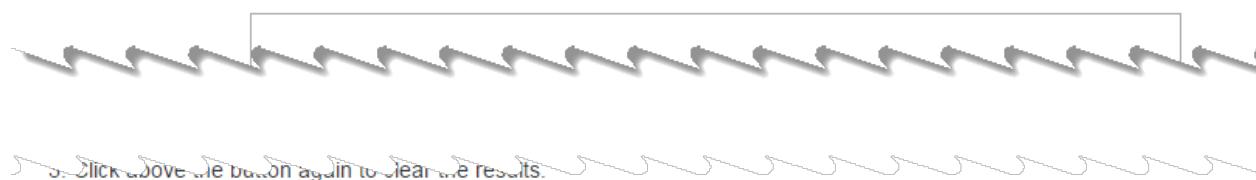
In test Post, scroll down to “Test 2: Perform REST/POST AOI-Create Query”, copy the following “Well Known Text (WKT) description of a rectangle in Smith County KS into the “AoiCoords” box:

```
polygon((-98.617 39.814,-98.617 39.843,-98.551 39.843,-98.551 39.814,-98.617 39.814))
```

## SDM Data Access (Soil Data Access) Test Post

Use this SDA web page to test select POST functions within SDA.

### Test 1: Perform REST/POST Tabular Query



### Test 2: Perform REST/POST AOI-Create Query

For AOI creation, use service=aoi, request=create. AoiCoords, MuKevl, ist\_SSA and WssAoId are mutually exclusive. Filter and/or Partname are only meaningful with AoiCoord. The parameter may be specified in the "(other)" area by using name=value.

Service	aoi
Request	create
AoiCoords	    
...	

Paste WKT here.

After pasting the WKT scroll down a little further and click “run using POST”. After a brief delay note the pAoID (the identifier for the newly-created pAOI) appears below the button. Jot down the number that you see (you’ll need this for the next step):

**Test 2: Perform REST/POST AOI-Create Query**

For AOI creation, use service=aoi, request=create. AoiCoords, MuKeyList, SSA and WssAoid are mutually exclusive. Filter and/or Partname are only meaningful with AoiCoords (when the AoiCoords data is GeoJSON). An additional parameter may be specified in the "(other)" area by using a parameter name/value pair as:

name=value

Service	aoi
Request	create
AoiCoords	<pre>((-98.617 39.814,-98.617 39.843,-98.551 39.843,-98.551 39.814,-98.617 39.814))</pre>
SSA	

(other)

Click here.

1. Click following button to issue a REST/POST request.
2. Confirm that the button click returns result below the button.
3. Click above the button again to clear the results.

### Test 3: GetStyles

WssAoid	
Filter	
PartName	
(other)	<p>In this instance "20601" is the identification number assigned to the newly-created AOI. Yours will be different.</p>

1. Click following button to issue a REST/POST request.
2. Confirm that the button click returns result below the button.  

```
{"id":20601}
```

The SDA AOIs have a limited lifetime – they currently live at least 24 hours after you create them, then they are erased.

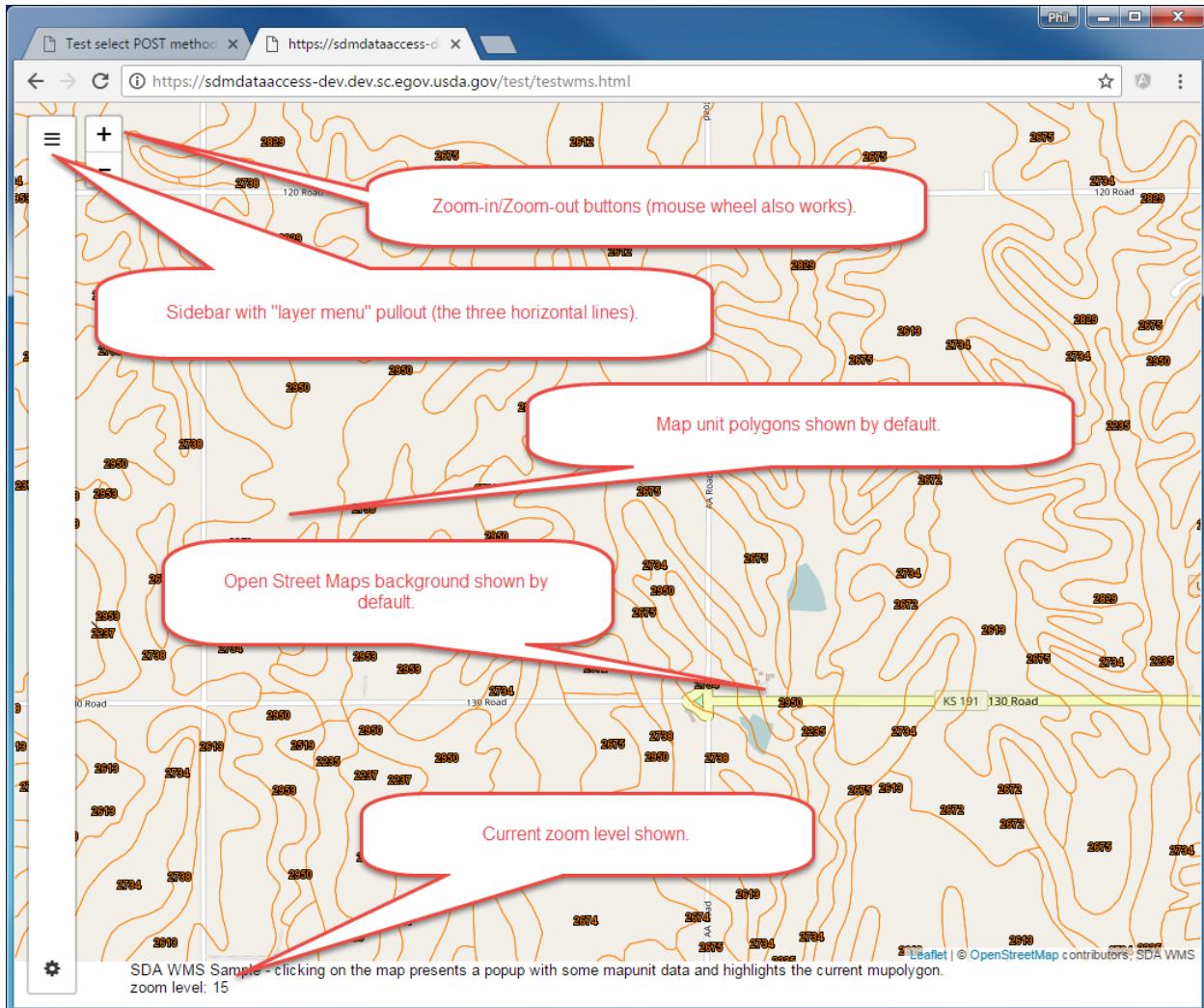
A sanity check: in a third Web browser tab start Web Soil Survey and, once the map is displayed, paste at the end of the URL

```
?aoicoords=(( -98.617 39.814,-98.617 39.843,-98.551 39.843,-98.551 39.814,-98.617 39.814))
```

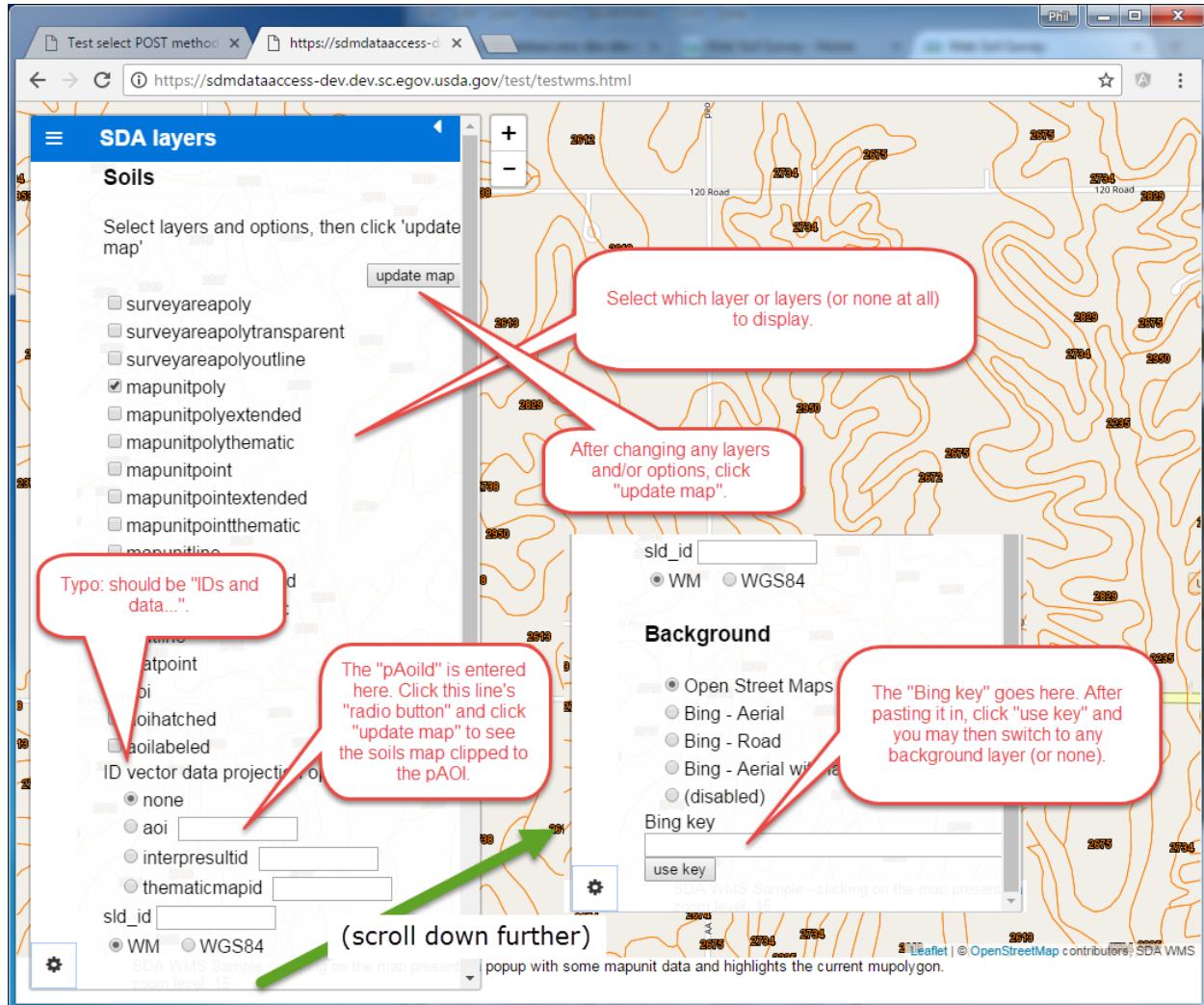
and press enter. Go to the “Soil Map” tab and keep that page open, we’ll return to it a bit later.

## Explore the SDA Soils Layers

With the Test WMS page open in either Chrome or Firefox, note the following features:



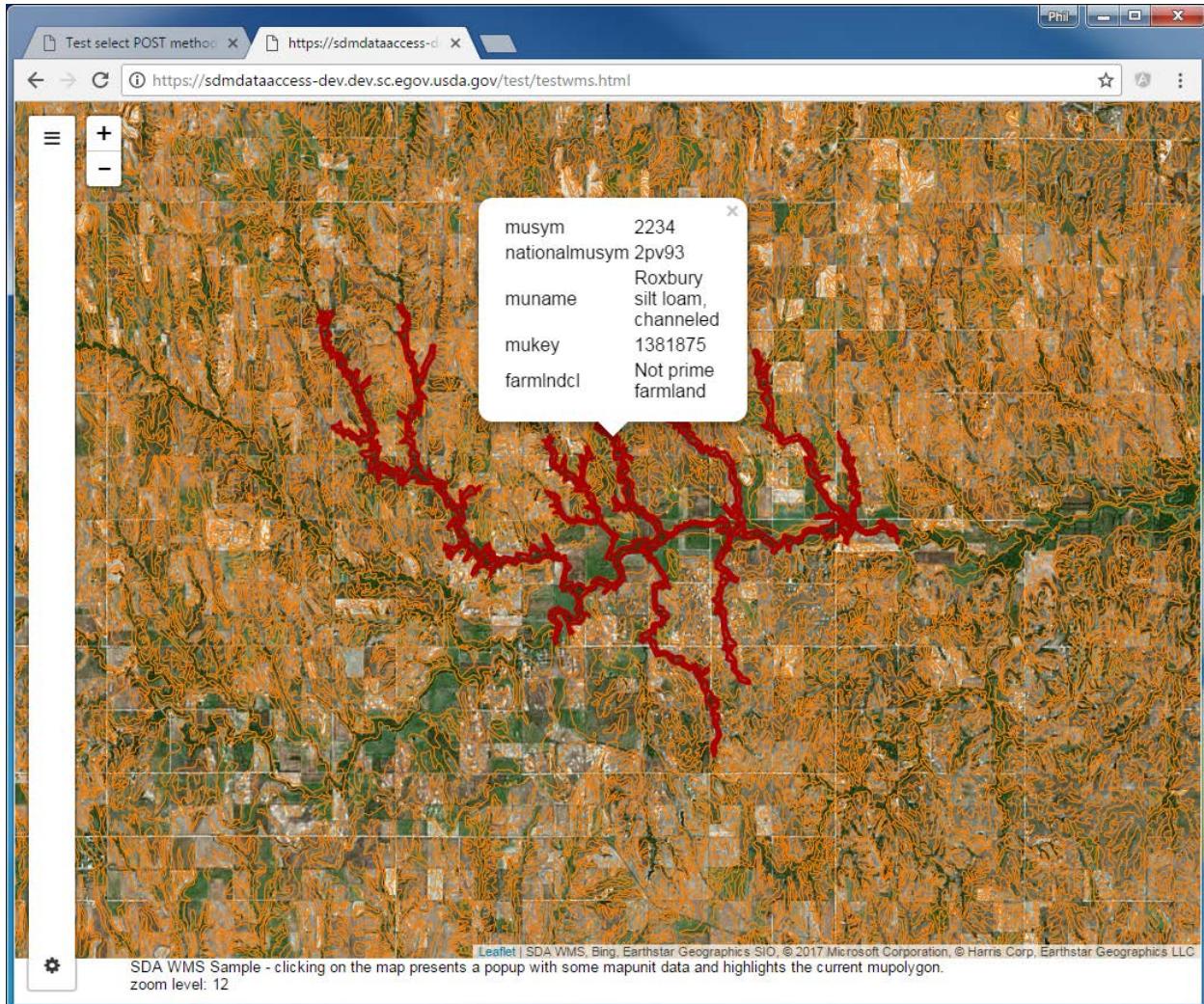
Now click on the “layer menu symbol.



The following discussion assumes you can paste in a “Bing key”.

Click “use key” and then click on the button to the left of “Bing-Aerial”. Close the slide-out menu by clicking on the “layers menu” symbol. Zoom out until you’re at zoom level 12 (either click on the “-“ button towards the top left of the page or roll your mouse’s scroll wheel towards you (map unit polygons are shown over a zoom level from 11 to 18). Zoom back in to 13, pan around by using your mouse.

While you're zoomed all the way in to 12, click inside of one of the larger map unit polygons...



The map is zoomed in or out to show the selected map unit polygon and a few attributes of the associated map unit are displayed (I'll be returning to mukey 1381775 later on).

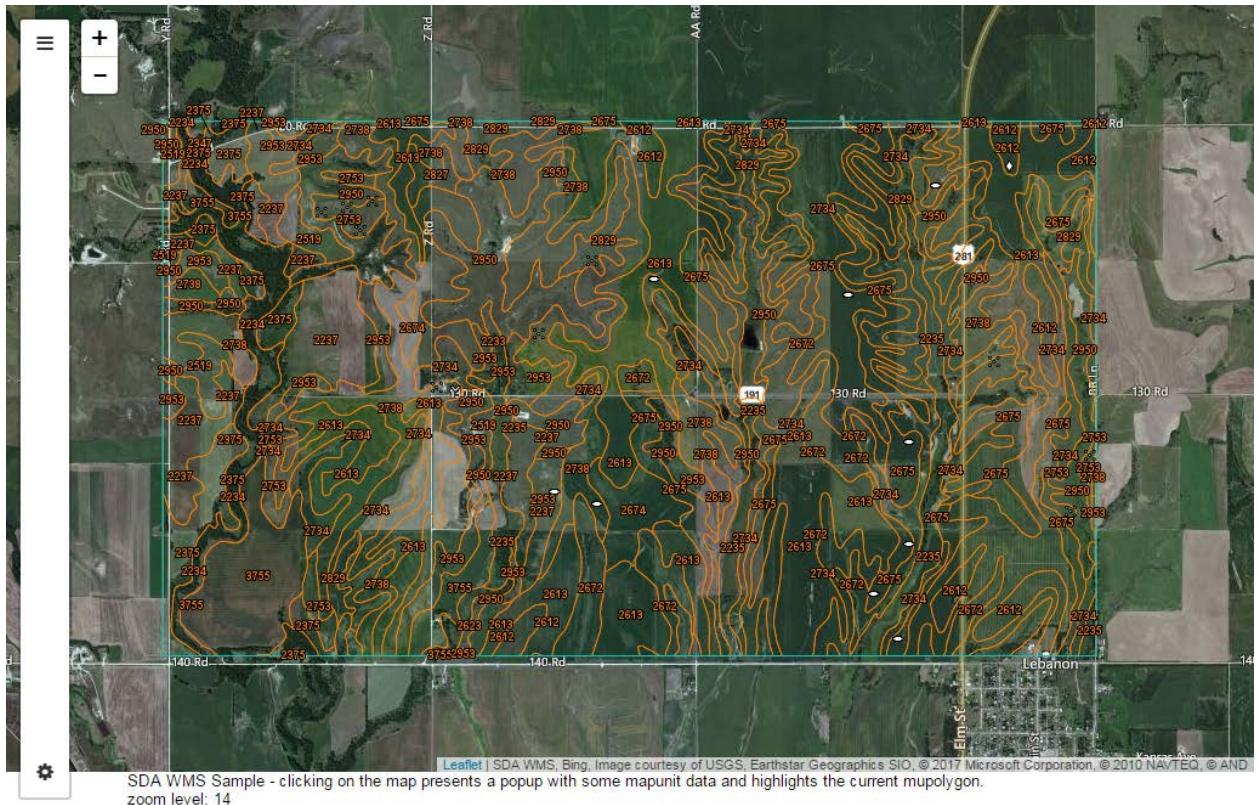
Click again within this polygon (zoom in first if necessary to be a bit more precise) and the popup and red outline will disappear). Using the layers menu select additional layers and “update map”. The symbology used for the layers matches that used in WSS.

### View the SDA Persistent AOI

Pull out the layers menu, switch to the “aoi” option, enter the pAoId (in this example it is “20601”), turn on layers mapunitpoly, mapunitpoint, mapunitline, featline,featpoint, and aoihatched. Click “update map” and hide the layers menu by clicking on the layers menu symbol. Pan around and zoom in and out, the soils layers are not displayed beyond the extent of the AOI. Return to the pAOI’s original location by pulling out the layer menu, change the ID option to “none”, click update map, then turn the “aoi” option back on and click update map again. Finally turn off “aoihatch”, turn on “aoi”, select “Bing – Aerial with Labels”, click update map and close the layers menu. The map should look about the same as that shown by WSS, see next page.

Detour: note the current zoom level (likely to be 14). Zoom out step by step – at 10 the mapunits (orange) disappear and only the aoihatch (cyan) remains, at level 6 and beyond the aoihatch is replaced by a red rectangle that remains all the way out to 1 (or to 0, but the world disappears at that point). (There’s an intermittent bug in TestWMS – the red rectangle sometimes remains when you zoom back in. It can be cleared by selecting “none” in place of the aoiid, updating the map, then reselecting the aoiid and updating yet again.)

The range of zoom levels allowed in Test WMS is beyond what would be appropriate for a real product. The further out you zoom, the longer it takes to draw the soils data (this could be enhanced but



SDA above, WSS below.



## [SDA Persistent AOIs](#)

The authoritative guide to the SDA AOIs is the ““Advanced Queries”” page (this document is still being edited, it does, however, reflect the work done under Task 3). Auxiliary documents are the “Web Service Help” and “Layer Help” pages. The Test Post and Test WMS pages use Web services and advanced query aids documented in these references. The documentation will not be reproduced here, rather I’ll demonstrate some use cases using the Query, Test Post and Test WMS pages.

Use of WSS AOI and thematic map data in SDA remains for Task 4, there’s brief mention in “Advanced Queries” sections “DOCUMENT UPDATES” and “OBSOLETE”. Import of a WSS AOI will be touched upon first, as it duplicates some of what you know from the “Quick Start”.

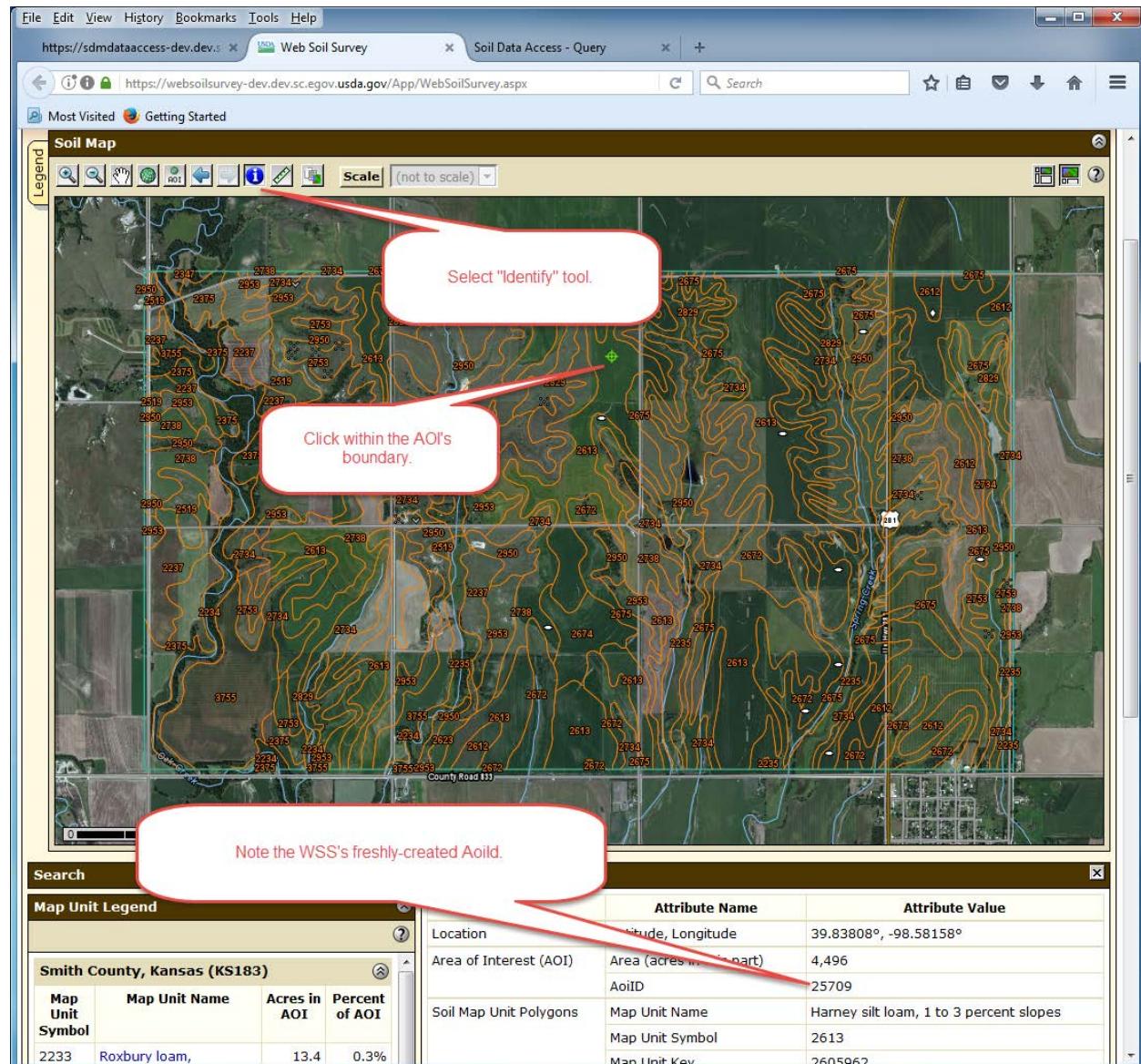
## Use Case 1: Import WSS AOI

In the previously-described “sanity check” it was suggested that you open WSS and after the map was displayed append the following to the URL:

```
?aoicoords=(-98.617 39.814,-98.617 39.843,-98.551 39.843,-98.551 39.814,-98.617 39.814)
```

If you’ve already done this and your WSS session is still active we can continue, otherwise start WSS, append the above to the URL to generate a new wAOI.

On either the “AOI”, “Soil Map” or “Soil Data Explorer” pages use the “Identify” tool and click within the bounds of the AOI. Scroll down and look for the “AoID”:

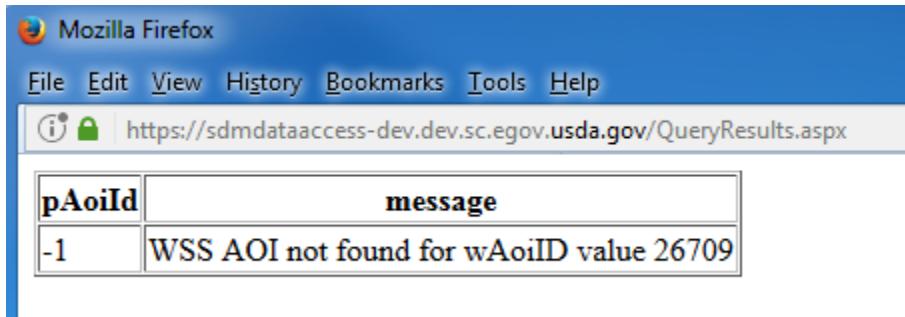


There are two ways to clone the WSS AOI data into an SDA AOI, via the Query page or via a “post.rest” web service such as that offered by the Test Post page. We’ll use the Query page.

Open the query page and paste in the following SQL statements, changing the wAoiId to match the one you retrieve from WSS.

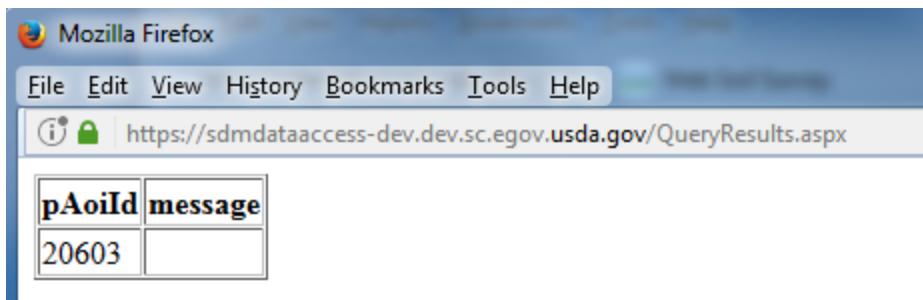
```
~DeclareInt(@wAoiId)~  
Select @wAoiId = 26709  
~CreateAoiFromWssAoi(@wAoiId,@pAoiId,@message)~  
Select @pAoiId as pAoiId,@message as message
```

Submit the query. If your WSS AOI has already expired or you mistyped the wAoiId (as I did above) you'll see



pAoiId	message
-1	WSS AOI not found for wAoiID value 26709

With a valid wAoiid,



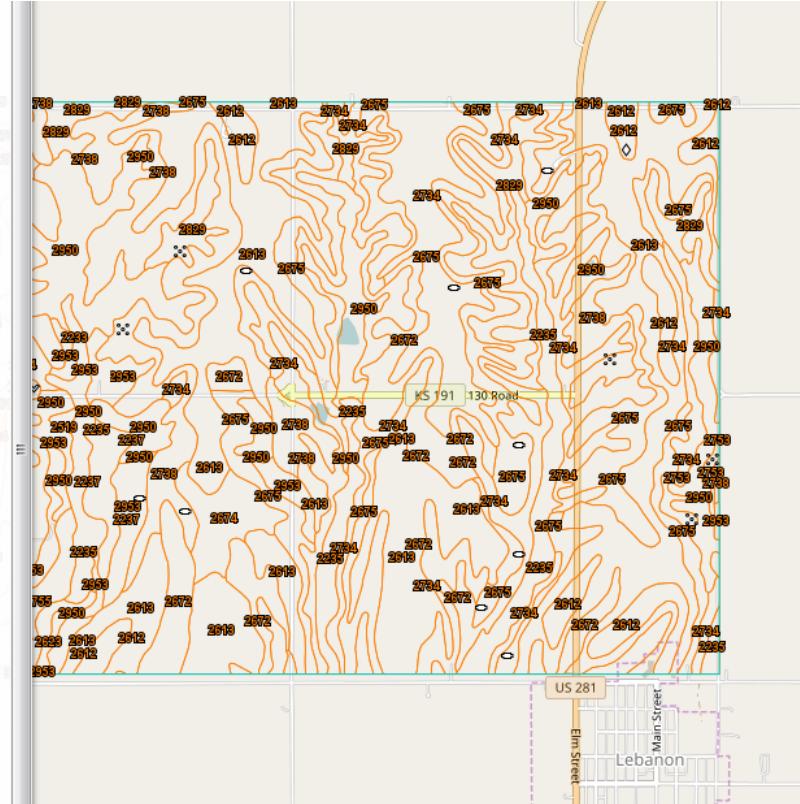
pAoiId	message
20603	

Switch to the Test WMS page, enter the SDA AoID (20603, in this example), update the map and you have demonstrated import of the WSS AOI data into SDA:

mapunitpoly  
 mapunitpolyextended  
 mapunitpolythematic  
 mapunitpoint  
 mapunitpointextended  
 mapunitpointthematic  
 mapunitline  
 mapunitlineextended  
 mapunitlinethematic  
 featline  
 featpoint  
 aoi  
 aoihatched  
 aolabeled  
 ID vector data projection options:  
 none  
 aoi 20603  
 interpresrid [ ]  
 thematicmapid [ ]  
 sld\_id [ ]  
 WM  WGS84

## Background

Open Street Maps



Once the SDA AOI has been defined we can probe the data tables that hold the AOI. For example, what SSA (or SSAs) does this AOI intersect? Using macros and functions documented in “Advanced Queries”, along with your knowledge of the SSURGO data model, a set of SQL statements can be conjured up for use in either the Query or Test Post page. Here I’ll show Test Post with the following statements:

```

~DeclareInt(@pAoild)~
select @pAoild = 20603
select distinct L.areasymbol from
legend L, mapunit MU, SDA_Get_AoiMapunit_By_Aoild(@pAoild) A
where MU.mukey = A.MapUnitKey and L.lkey = MU.lkey
  
```

Enter the statements in the “Query” box of “Test 1: Perform REST/POST Tabular Query”, click “run using POST” and then read the results (KS183) in JSON format (the button name will change after results are returned):

File Edit View History Bookmarks Tools Help

https://sdmdataaccess-dev.dev... x USDA Web Soil Survey x Soil Data Access - Query x https://sdmdataaccess...

← i lock https://sdmdataaccess-dev.dev.sc.egov.usda.gov/test/testpost.html Search

Most Visited Getting Started

## SDM Data Access (SDA) Test Post

Use this SDA web page to test selected functions within SDA.

### Test 1: Perform REST/POST Tabular Query

**Query:**

```
~DeclareInt(@pAoId)~  
select @pAoId = 20603  
select distinct L.areasymbol from legend L, mapunit MU, SDA_Geometry G where MU.mukey = A.MapUnitKey and G.mapunitkey = MU.mukey
```

**Format**

json

1. Click following button to issue a REST/POST request

{"Table": [{"Key": "KS183"}]}

Pasted SQL statements.

Button name changes after results are returned.

The answer: KS183.

## Use Case 2: SSA-based AOI

We'll use the "Test 2: Perform REST/POST AOI-Create Query" section of the Test Post page to define an SDA AOI for KS183. Enter KS183 into the "SSA" field and click "run using POST", the new pAoId will be shown below the button:

File Edit View Bookmarks Tools Help

https://sdmdataaccess-dev.dev. x USDA Web Soil Survey x | Soil Data Access - Query x | https://sdmdataaccess-dev.dev. x

← ⓘ 🔒 https://sdmdataaccess-dev.dev.sc.usda.gov/test/testpost.html ⌛ ⌚ Search

Most Visited Getting Started

3. Click above the button again to clear the results.

### Test 2: Perform REST/POST AOI-Create Query

For AOI creation, use service=aoi, request=create. AoiCoords, MuKeyList, SSA and WssAoId are mutually exclusive. Filter and/or Partname are only meaningful with AoiCoords (when the AoiCoords data is GeoJSON). An additional parameter may be specified in the "(other)" area by using a parameter name/value pair as:

name=value

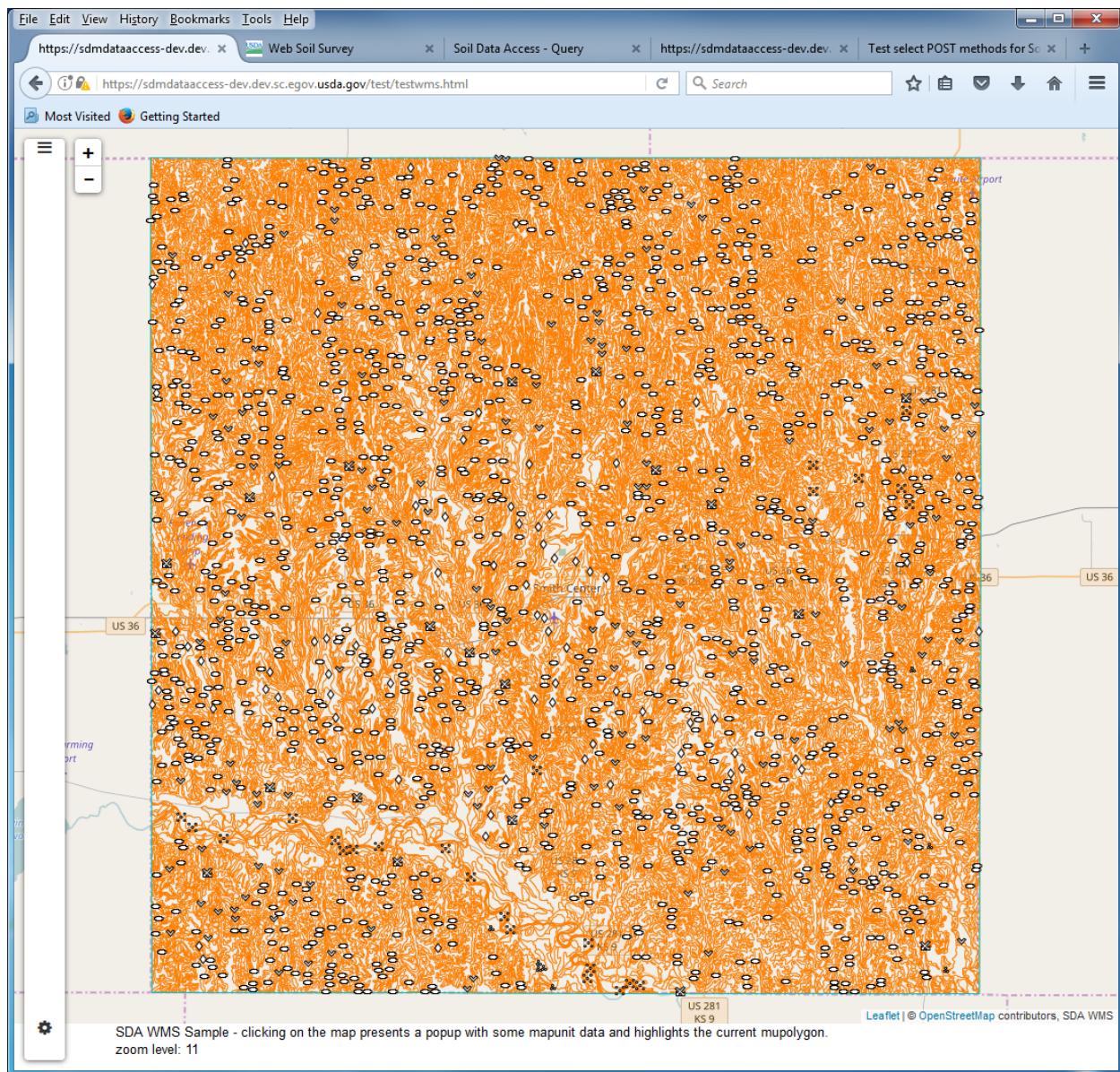
Service	aoi
Request	create
AoiCoords	
SSA	KS183
MukeyList	
WssAoId	
Filter	
PartName	
(other)	"20604" is the newly-created pAoId.

1. Click following button to issue a REST/POST-

clear POST results

{"id":20604}

The new pAoId may be used in Test WMS:



### Use Case 3: Mukey List AOI

A set of mukey values may also be used to define an AOI. Using the following queries in the Query page to create an SDA persistent AOI showing specific “farmlandcl” map units in ND053:

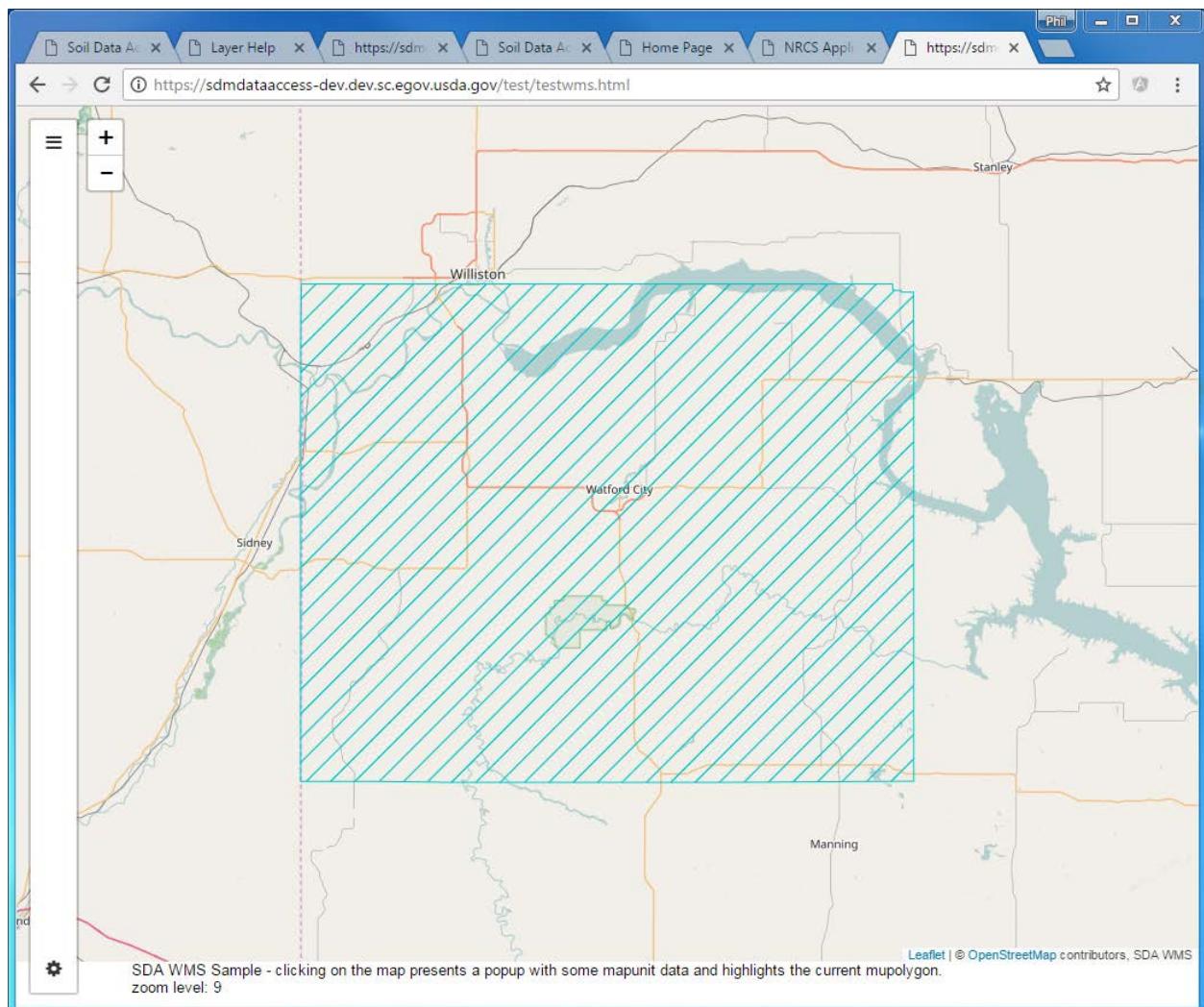
```
~DeclareIntTable(@mukeyList)~
insert into @mukeyList
select mu.mukey
from legend L left join mapunit MU on L.lkey = MU.lkey
where L.areatypename = 'Non-MLRA Soil Survey Area' and L.areasyymbol = 'ND053'
and mu.farmlndcl = 'Farmland of statewide importance';
~CreateAoiFromMukeyList(@mukeyList,@aoiid,@message)~
select count(*) [MukeyCount] from SDA_Get_AoiMapunit_By_AoiId(@aoiid);
select @aoiid [aoiid], @message [message]
```

MukeyCount	
85	

aoiid	message
20605	

In Test WMS, my browser shows zoom level 9 and the map unit polygons are not visible:



Not until zoom level 11 are the map unit polygons visible. For Mukeylist AOIs the AOI boundary is shown on the map as the union of envelopes around the individual map unit's map unit polygons. From the above outline it appears that the select set of map units are found throughout ND053.

The mukey values may also be specified through Test Post, for example:

## Test 2: Perform REST/POST AOI-Create Query

For AOI creation, use service=aoi, request=create. AoiCoords, MuKeyList, SSA and WssAoId are mutually exclusive. Filter and/or Partname are only meaningful with AoiCoords (when the AoiCoords data is GeoJSON). An additional parameter may be specified in the "(other)" area by using a parameter name/value pair as:

name=value

<b>Service</b>	aoi
<b>Request</b>	create
<b>AoiCoords</b>	
<b>SSA</b>	
	58186, 58202,58229,58230
<b>MukeyList</b>	
<b>WssAoId</b>	
<b>Filter</b>	
<b>PartName</b>	
<b>(other)</b>	

1. Click following button to issue a REST/POST request.

{"id":20607}

#### Use Case 4: Ad-Hoc AOI

The “Quick Start” section demonstrated definition of an ad-hoc AOI by WKT specification. In addition to WKT “GeoJSON” may be used (that format allows for part naming and filtering, see “Advanced Queries” for an example).

Using the following example containing a “partName” property

```
[  
  {  
    'type': 'FeatureCollection',  
    'features': [  
      {  
        'type': 'Feature',  
        'geometry': {  
          'type': 'Polygon',  
          'coordinates':  
            [[[[-100.0,40.0],[-100.1,40.0],[-100.1,40.1],  
              [-100.0,40.1],[-100.0,40.0]]]  
        },  
        'properties': {'partName': "Sister"}  
      },  
      {  
        'type': 'Feature',  
        'geometry': {  
          'type': 'Polygon',  
          'coordinates':  
            [[[[-100.2,39.8],[-100.3,39.8],[-100.3,39.9],  
              [-100.2,39.9],[-100.2,39.8]]]  
        },  
        'properties': {'partName': 'Brother'}  
      }  
    ]  
  }]
```

Paste it into the Test Post page,

AoiCoords

```
{  
  'type': 'FeatureCollection',  
  'features': [  
    {  
      'type': 'Feature',  
      'geometry': {  
        'type': 'Polygon',  
        'coordinates':  
          [[[[-100.0,40.0],[-100.1,40.0],[-100.1,40.1],  
            [-100.0,40.1],[-100.0,40.0]]]  
      },  
      'properties': {'partName': "Sister" }  
    },  
    {  
      'type': 'Feature',  
      'geometry': {  
        'type': 'Polygon',  
        'coordinates':  
          [[[[-100.2,39.8],[-100.3,39.8],[-100.3,39.9],  
            [-100.2,39.9],[-100.2,39.8]]]  
      },  
      'properties': {'partName': 'Brother' }  
    }  
  ]  
}
```

SSA

MukeyList

WssAoild

Filter

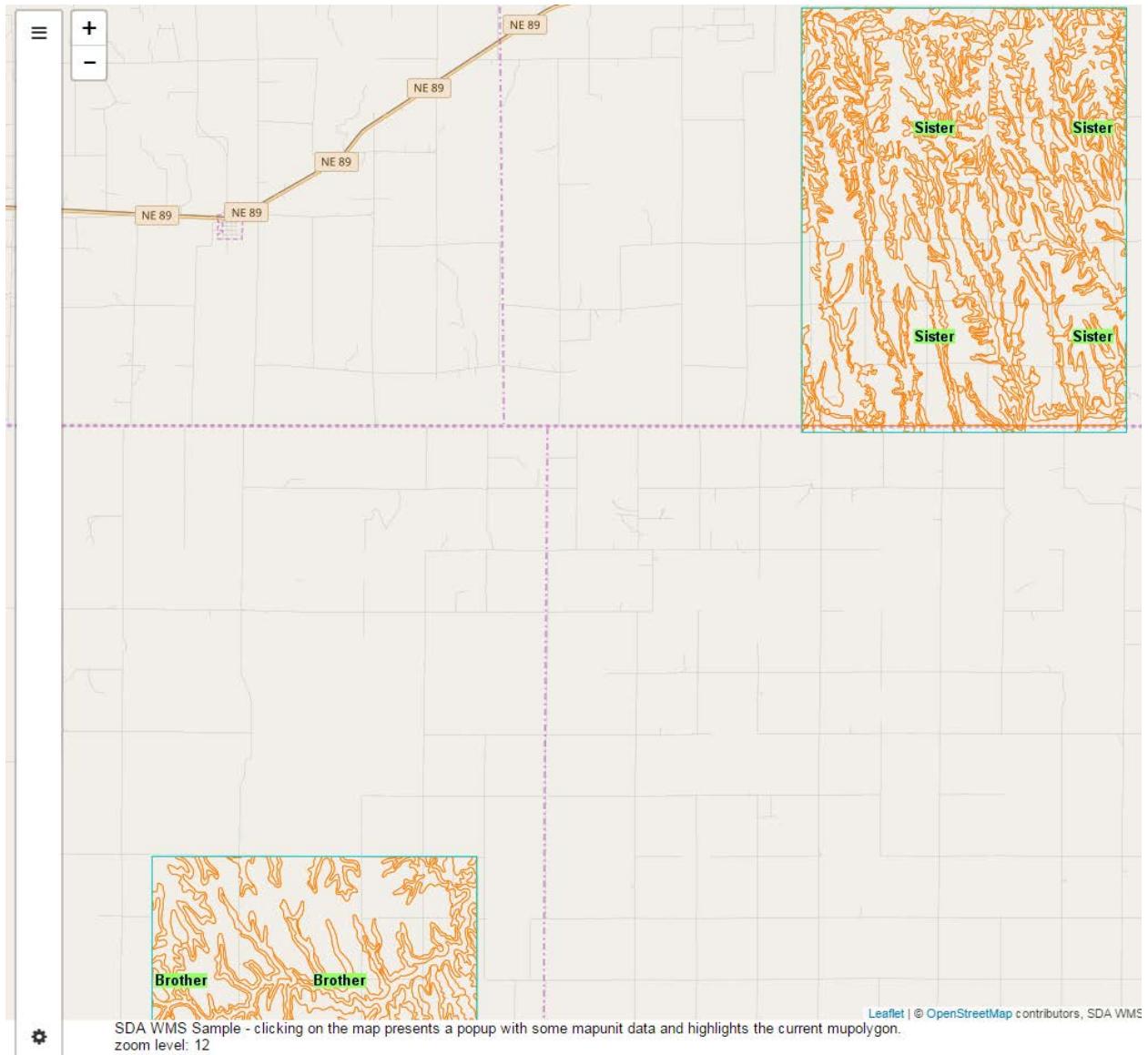
PartName

(other)

1. Click following button to issue a REST/POST request.

{"id":20608}

Within the Test WMS page the “aoilabeled” layer may be selected to show the parts of the multi-part AOI:



## Use Case 5: Add Styled Layer Descriptor to AOI

Earlier in this document an AOI was shown for pAoId = 20603. Let's assume that the polygon boundaries are to be shown in cyan instead of orange. An easy way to get started with an SLD is to grab one from the Test Post page's "Test 3: GetStyles" and "Test 4: GetMap/SLD-Body" sections.

The screenshot shows a web browser window with two sections: "Test 3: GetStyles" and "Test 4: GetMap/SLD-Body".

**Test 3: GetStyles**

Layername: mapunitpoly

1. Click following button to issue a REST/POST request for the layer's styles.  
run using POST  
...cleared...

2. Confirm that the button click returns result.  
3. Click above the button again to clear the results.

This is the layer we want.

**Test 4: GetMap/SLD-Body**

SLD\_BODY:

BBOX W,S,E,N -102.65253,38.18368,-102.59682,38.21139

1. Click following button to issue a REST/POST request for the layer's map.  
run using POST  
...cleared...

2. Confirm that the button click returns result, as a map image, below the button.  
3. Click above the button again to clear the results.

The mapunitpoly's default SLD, formatted for easier reading, will also appear here.

You can edit the text in-situ under Test 4 and see the result below the “run using POST” button. In this example the PolygonSymbolizer / Stroke / CssParameter name="stroke" value will be changed from “#ff8000” (orange) to “00ffff” (cyan). After making the change in the “SLD\_BODY” box and clicking the “run using post” button, a map is produced using the new color specification:

**Test 4: GetMap/SLD-Body**

```

<UserStyle>
  <FeatureTypeStyle>
    <Rule>
      <MinScaleDenominator>0.010000</MinScaleDenominator>
      <MaxScaleDenominator>250000.000000</MaxScaleDenominator>
      <PolygonSymbolizer>
        <Stroke>
          <CssParameter name="stroke">#00ffff</CssParameter>
          <CssParameter name="stroke-width">0.75</CssParameter>
        </Stroke>
      </PolygonSymbolizer>
      <TextSymbolizer>
    </Rule>
  </FeatureTypeStyle>
</UserStyle>

```

**SLD\_BODY:**

BBOX W,S,E,N -102.65253,38.0368,-102.59682,38.21139

1. Click following button to initiate a REST/POST request for the layer's map following the SLD.

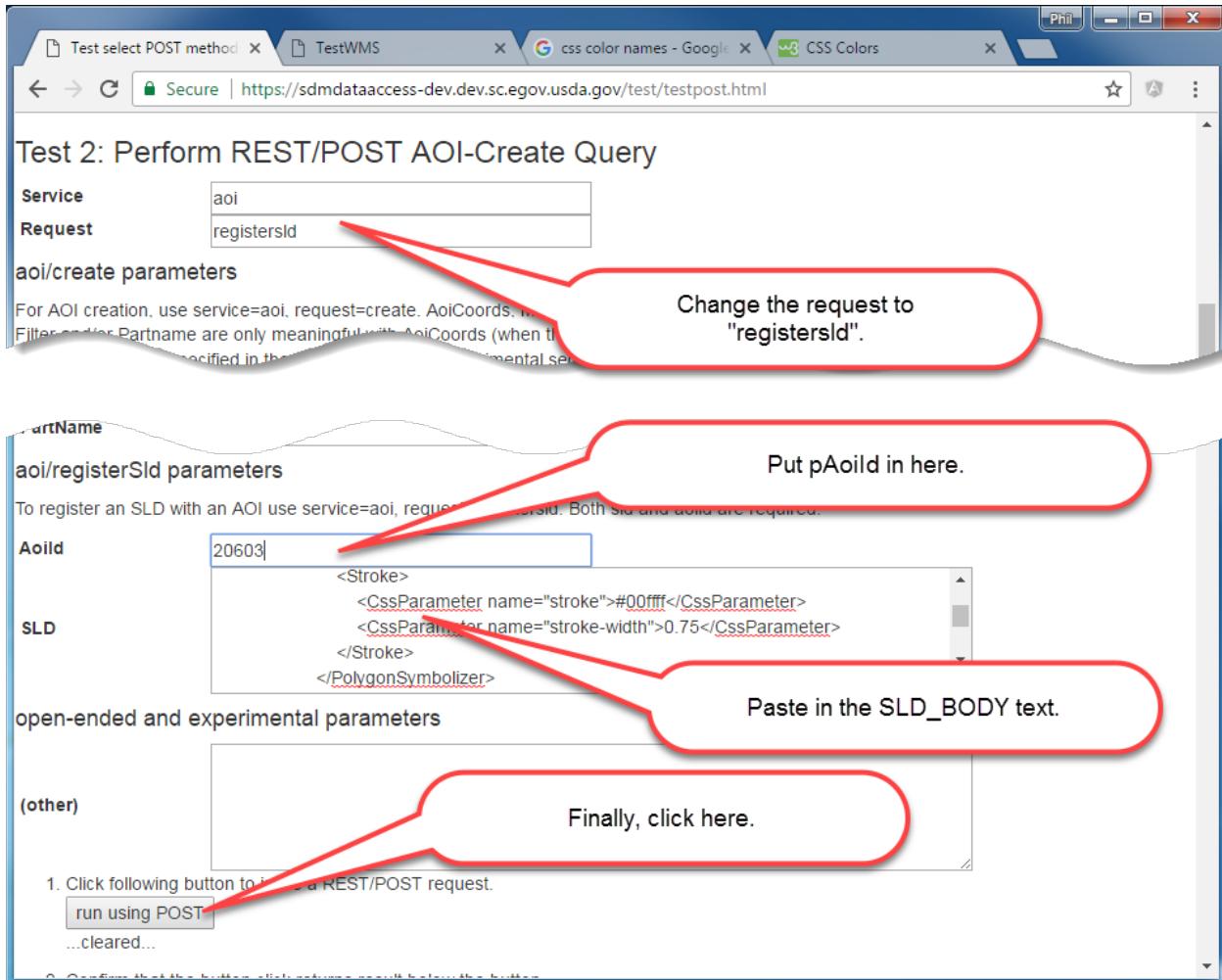
**clear POST results**

Image (image/jpeg) shown below

2. Confirm that the button click returns result, as a map image, below the button.

3. Click above the button again to clear the results.

Your edited SLD\_BODY text can be copied and pasted into the Test Post page to register the SLD with pAoId 20603:



Here's the new sld\_id, created by registering the SLD\_BODY with the pAOI:

The screenshot shows a web browser window with the URL <https://sdmdataaccess-dev.dev.sc.usda.gov/test/testpost.html>. The page contains a form for registering an SLD with an AOI.

**aoi/registerSld parameters**

To register an SLD with an AOI use service=aoi, request=registerSld. Both sld and aoid are required.

**AoId**: 20603

**SLD**: 

```
<Stroke>
<CssParameter name="stroke">#00ffff</CssParameter>
<CssParameter name="stroke-width">0.75</CssParameter>
</Stroke>
</PolygonSymbolizer>
```

**open-ended and experimental parameters**

**(other)**:

1. Click following button to issue a REST/POST request.

**clear POST results**

**{"sld\_id":7}**

The button labeled {"sld\_id":7} is circled in red.

The new SLD\_ID can be used in conjunction with the pAoId in TestWMS:

SDA layers & IDs

Soils

Select layers and options, then click update map

- surveyareapoly
- surveyareapolytransparent
- surveyareapolyoutline
- mapunitpoly
- mapunitpolyextended
- mapunitpolythematic
- mapunitpoint
- mapunitpointextended
- mapunitpointthematic
- mapunitline
- mapunitlineextended
- mapunitlinethematic
- featureline
- feattpoint
- aoi
- aoihatched
- aoilabeled

ID vector data projection options

- none
- aoi 20603
- interpresrid
- thematicmapid
- sld\_id 7
- WM
- WGS84

Background

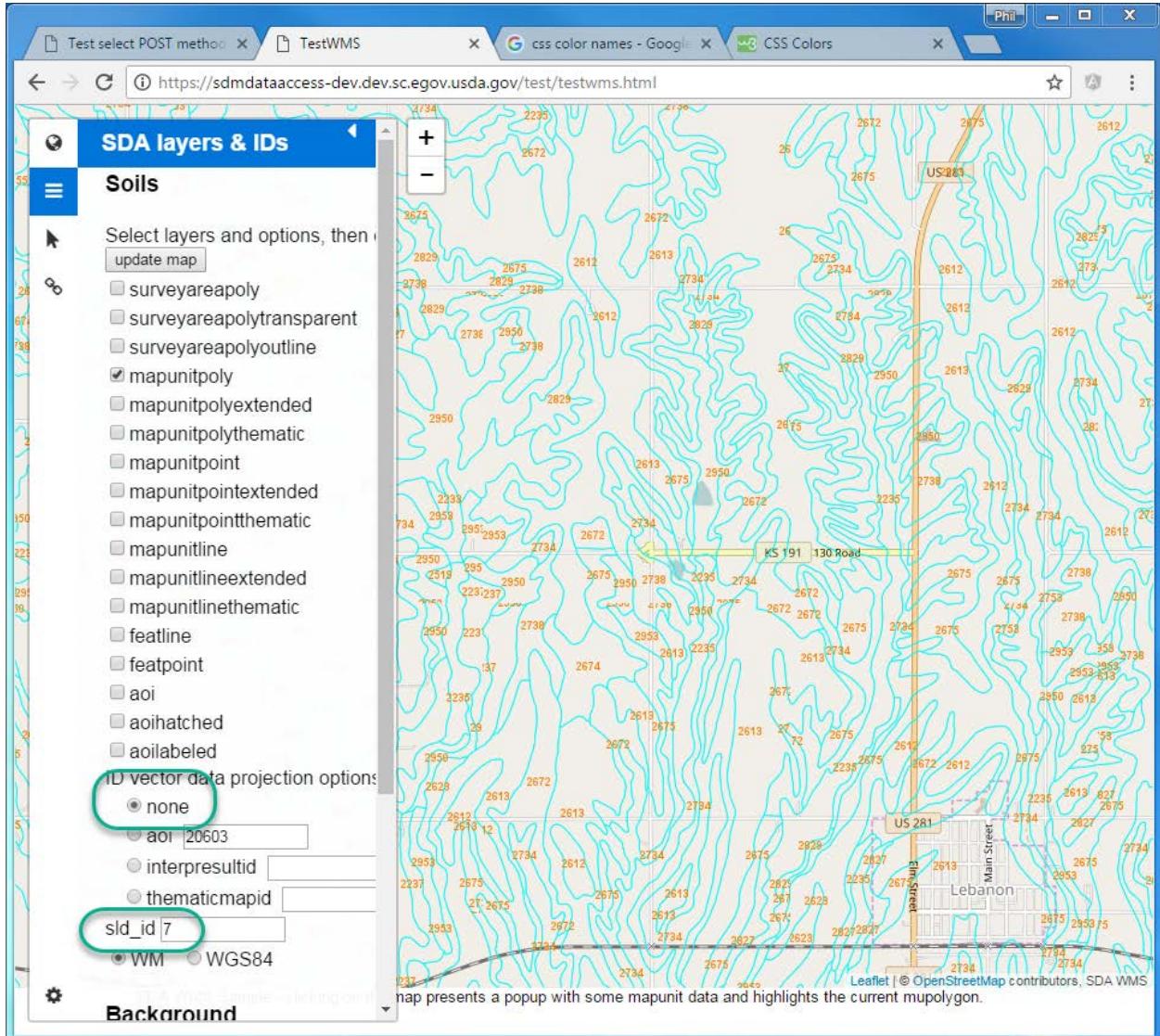
Remember to click here after changing any soil layer selection or ID choice or value.

Leaflet | OpenStreetMap contributors, SDA WMS

Map presents a popup with some mapunit data and highlights the current mapunit polygon.

The screenshot shows a map interface for soil data. On the left, there's a sidebar titled "SDA layers & IDs" with a section for "Soils". It lists various soil-related layers and features, with "mapunitpoly" checked. Below this is a section for "ID vector data projection options" where "aoi" is selected with the value "20603", and "sld\_id" is set to "7". A red arrow points from the "update map" button in the sidebar to a callout bubble containing the text "Remember to click here after changing any soil layer selection or ID choice or value." The main map area displays contour lines and soil polygons. Some polygons are highlighted in orange, and others have orange numbers (e.g., 2612, 2675, 2734) overlaid. A road labeled "KS 191 - 130 Road" and a town labeled "Lebanon" are visible. A red circle highlights the "sld\_id" input field in the sidebar.

Once an SLD\_BODY is registered and you have an SLD\_ID, that ID can be used with other pAoilds or even with no pAoild at all:



## Other Use Cases

“Advanced Queries” describes other methods for defining persistent SDA AOIs. For example shapefiles may be imported (this requires a Python or Windows Powershell script, samples are available) and as a tentative possibility for CD projects, AOIs based upon PLU may be used as well (the PLU capability will likely not be publically advertised or available).

## Interpretations

SDA now offers the ability to execute “interpretations” (also called “ratings”) against the soils data associated with a pAOI. The “Advanced Queries” describes the capabilities and provides a walkthrough. This text will duplicate the examples presented and then proceed to a more complete test of the interpretations facility. The “Test Post” and “Test WMS” pages will be referenced. Use of SQL Server Management Studio’s querying facility may also be used, that is outside the scope of this document.

In some cases JSON or XML strings will be shown in a multi-line formatted version. This is performed with the “JSON Formatter” or “XML Formatter” (see [Resources](#)), there are numerous alternatives available online.

### Retrieving a Catalog

The set of interpretations that are applicable to an AOI is called the “catalog”. Creation of AOIs has already been addressed, let’s assume that you have created an AOI with:

```
polygon((-102.095 46.686, -102.095 46.717, -102.032 46.717, -102.032 46.686, -102.095 46.686))
```

For this example the aoiid is 515.

The Test Post page now includes a section dedicated to interpretations:

### Test 3: Perform REST/POST Interpretation Requests

#### interpretation request parameters

request	parameters
getusecategories	(none)
getcatalog	aoiid, usecategoryid (optional, defaults to 0)
getruledata	aoiid, format (optional, defaults to "short"), one of {attributekey, interpresultid, thematicmapid}
getrating	shortformdata

Service	interpretation
Request	<input type="text"/>
aoiid	<input type="text"/>
usecategoryid	<input type="text"/>
format	<input type="text"/>
attributekey	<input type="text"/>
interpresultid	<input type="text"/>
thematicmapid	<input type="text"/>
shortformdata	<input type="text"/>

1. Click following button to issue a REST/POST request.

In the “Request” field type “getcatalog”, in “aoiid” insert the aoiid (515 in this example) and click “run using post”:

<b>Service</b>	interpretation
<b>Request</b>	getcatalog
<b>aoiid</b>	515
<b>usecategoryid</b>	
<b>format</b>	
<b>attributekey</b>	
<b>interpresultid</b>	
<b>thematicmapid</b>	
<b>shortformdata</b>	

1. Click following button to issue a REST/POST request.

**run using POST**

The response will appear below the button, here's the start of it:

1. Click following button to issue a REST/POST request.

**clear POST results**

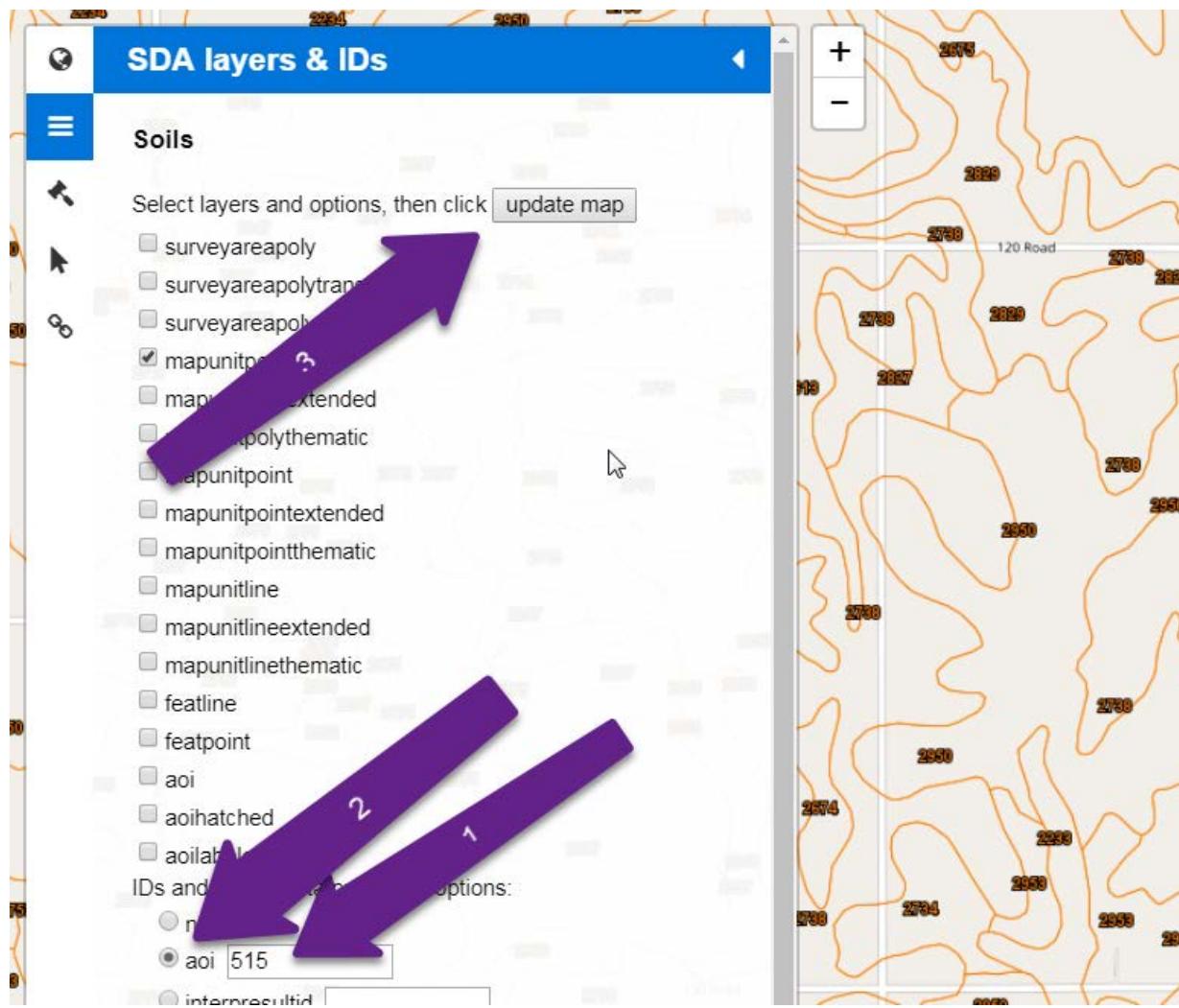
```
{"tables": [{"tablename": "properties", "folders": [{"foldername": "Soil Chemical Properties", "attributes": [{"attributename": "Calcium Carbonate (CaCO3)", "attributekey": "98"}, {"attributename": "Cation-Exchange Capacity (CEC-7)", "attributekey": "28"}, {"attributename": "Effective Cation-Exchange Capacity (ECEC)", "attributekey": "2"}, {"attributename": "Electrical Conductivity (EC)", "attributekey": "59"}, {"attributename": "Gypsum", "attributekey": "47"}, {"attributename": "pH (1 to 1 Water)", "attributekey": "11"}, {"attributename": "Sodium Adsorption Ratio (SAR)", "attributekey": "68"}]}, {"foldername": "Soil Erosion Factors", "attributes": [{"attributename": "K Factor, Rock Free", "attributekey": "19"}, {"attributename": "K Factor, Whole Soil", "attributekey": "51"}, {"attributename": "T Factor", "attributekey": "75"}, {"attributename": "Wind Erodibility Group", "attributekey": "71"}, {"attributename": "Wind Erodibility Index", "attributekey": "4"}]}, {"foldername": "Soil Health", "attributes": [{"attributename": "Soil Health - Organic"}]}]}
```

This JSON response can be formatted with the “JSON Formatter”, the first bit of it looks like:

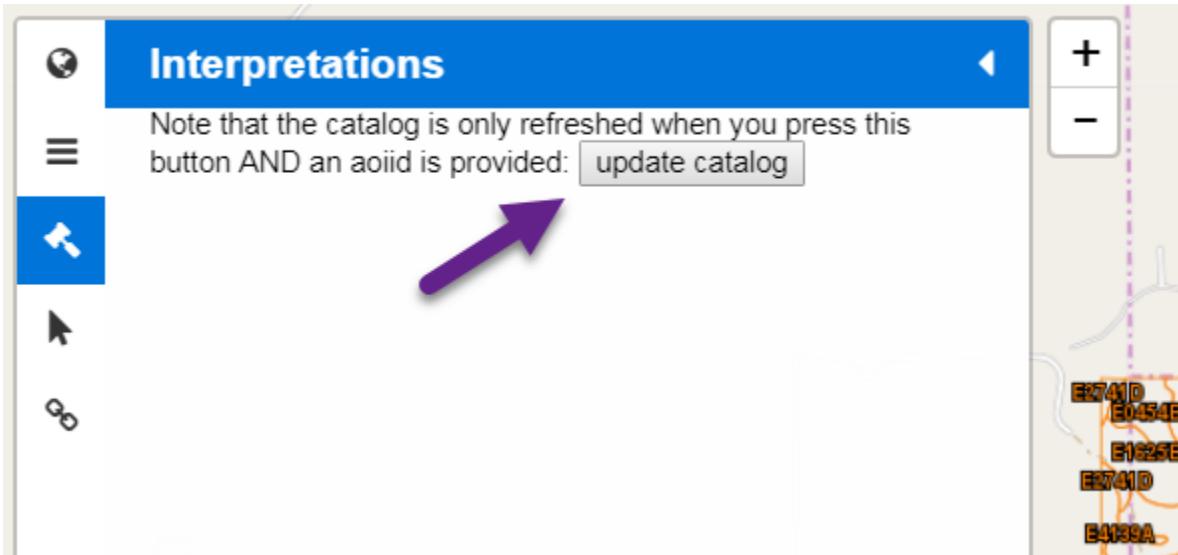
```
{  
  "tables": [  
    {  
      "tablename": "properties",  
      "folders": [  
        {  
          "foldername": "Soil Chemical Properties",  
          "attributes": [  
            {  
              "attributename": "Calcium Carbonate (CaCO3)",  
              "attributekey": "98"  
            },  
            {  
              "attributename": "Cation-Exchange Capacity (CEC-7)",  
              "attributekey": "28"  
            },  
            {  
              "attributename": "Effective Cation-Exchange Capacity (ECEC)",  
              "attributekey": "2"  
            }  
          ]  
        ]  
      ]  
    ]  
  ]  
}
```

Note that each kind of interpretation (or “attribute”) as an associated key number.

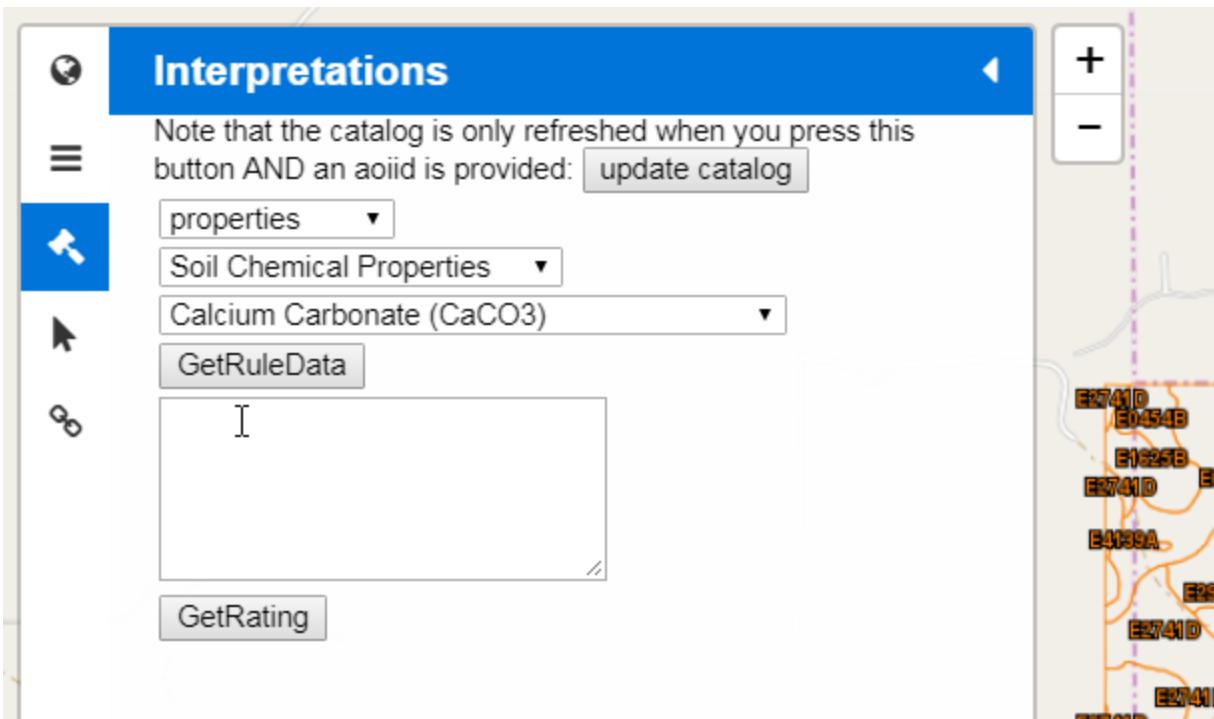
The Test WMS page can perform a similar look-up for you, albeit in a much more concise manner. Start Test WMS, enter the aoiid and turn on the use of aoiid, and then update the map.



Open up the “gavel” pullout and click on “update catalog”:



After a brief delay the catalog will be presented through a set of selection lists:



When you change the first selection to “interpretations”, the second list will change. In that updated second list select “Land Classifications”. The third list will change, in that list select “Soil Taxonomy Classification”. You have just navigated through the hierarchical JSON shown a few pages ago. When you click “GetRuleData” you will get a “short form” version of the input data required for executing “Soil Taxonomy Classification”. Go ahead and click that button:

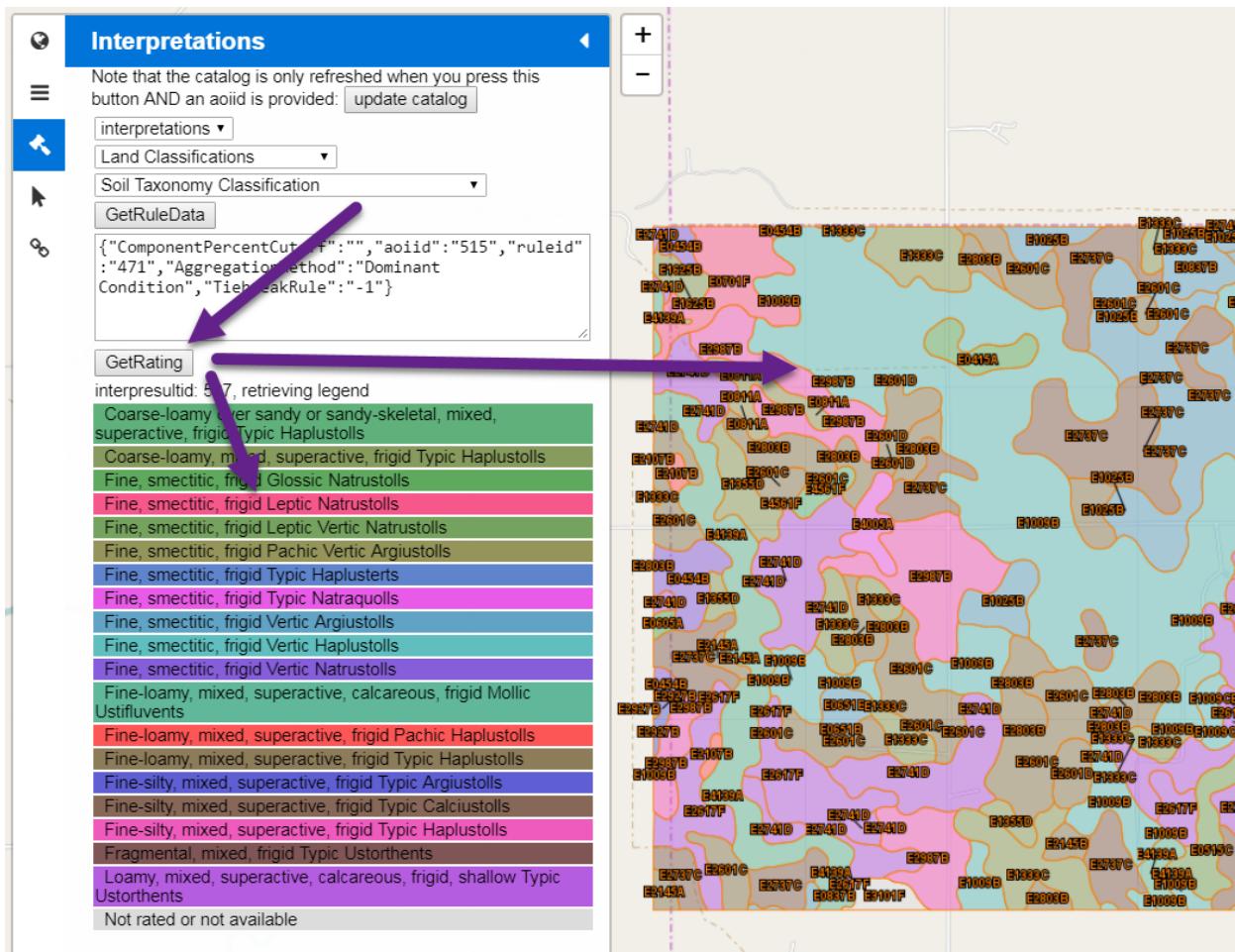
The screenshot shows the “Interpretations” interface. On the left, there are several icons: a globe, a list, a magnifying glass, and a compass. Below these are buttons for “interpretations”, “Land Classifications”, “Soil Taxonomy Classif.”, and “GetRuleData”. A purple callout bubble points to the “GetRuleData” button with the text: “Click this button after making your selections.” Above the “GetRuleData” button, a text box contains: “When a selection is made in any of these three areas, the ones below it are updated.” To the right of the interface is a map showing various soil polygons with labels like E2741D, E0454B, E1625B, E741D, E0701F, E1009B, E4139A, E2987B, E2741D, E0811A, E0811A, E2741D, E2987B, and E0811A.

After clicking you will see (the text box has been stretched for clarity):

The screenshot shows the “Interpretations” interface after the “GetRuleData” button was clicked. The “GetRuleData” button is now highlighted in blue. The text box on the right contains the following JSON data: {“ComponentPercentCutoff”: “”, “aoiid”: “515”, “ruleid”: “471”, “AggregationMethod”: “Dominant Condition”, “TiebreakRule”: “-1”}. The map to the right remains the same, showing soil polygons with labels like E2741D, E0454B, E1625B, E741D, E0701F, E1009B, E4139A, E2987B, E2741D, E0811A, E0811A, E2741D, E2987B, and E0811A.

The contents of the text box is “short form” data that drives execution of the rating.

Click on “GetRating”:



Several things happened:

1. The “short form” data was used to control interpretation of the soils in the AOI.
2. The “mapunitpolygon thematic” layer was turned if it was not already on.
3. The “interpresultid” number was pushed into the “Layers” ID section and it was turned on.
4. The legend for the interpretation was painted on the Interpretations pull-out.

We can peek at the Layers pull-out to confirm this:

The screenshot shows the 'SDA layers & IDs' interface for 'Soils'. On the left, a sidebar lists several options with checkboxes. Two checkboxes are checked: 'mapunitpoly' and 'mapunitpolythematic'. A purple arrow points from the 'mapunitpolythematic' checkbox to the 'aoilabeled' section below. In the 'aoilabeled' section, there are four radio button options: 'none', 'aoi [E139]', 'interpresrid [527]', and 'thematicmapid [ ]'. The 'interpresrid' option is selected, and a purple arrow points from its value to a text input field. To the right of the sidebar is a map preview showing various soil polygons labeled with codes like E2741D, E0454B, E1625B, etc. There are also zoom controls (+, -, arrows) and a legend icon.

Almost all “properties” require modification of the short form data as not all valid defaults are in place.

Determining options that may be specified in this “short form” is described in “Advanced Queries” where there are extensive notes on the “short form” data and its partner “long” and how to work around incomplete sets of options.

For example, if you had selected “properties”, “Soil Chemical Properties” and “Calcium Carbonate (CaCO<sub>3</sub>)” and clicked “GetRuleData” and “GetRating”, you would receive:

GetRating

```
<?xml version='1.0' encoding="UTF-8" standalone="no" ?>
<ServiceExceptionReport xmlns="http://www.opengis.net/ogc"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/ogc
    http://schemas.opengis.net/wms/1.1.1/OGC-exception.xsd">
  <ServiceException> An invalid Top Depth was specified.
  </ServiceException> </ServiceExceptionReport>
```

“Advanced Queries” tells all.

## Comparing WSS and SDA Interpretation Results

Both SDA and WSS place their interpretation results in database tables. We can confirm that SDA is working properly by comparing results between the two, here's an example.

Define an AOI in WSS for SSA CA795 (Mojave National Preserve Area, California), run the Soil Taxonomy classification on it and then use the "Identify" tool to get the WSS thematicmapid (in this example it is 1504). Do not clear the WSS AOI or the WSS AOI records will be immediately deleted.

In the Test Post page define an SDA AOI for CA795. In this example the aoiid is 516. In Test WMS run the Soil Taxonomy Classification on this aoiid and get the interpresultid, in this exampe it is 528. (You may need to turn on the "aoihatched" layer to see the extent of the AOI and then zoom in around "Castle Mountains National Monument", in the northeast part of the AOI, until your down to zoom level 12 in order to see mapunit polygons).

In either the Query page or the Test Post query area, compare the legend results with this set of query statements:

```
~DeclareInt(@thematicmapid)~  
~DeclareInt(@interpresultid)~  
  
select @thematicmapid = 1504;  
select @interpresultid = 528;  
  
select from  
WSS_Get_AoiSoilThematicMapLegend_By_AoiSoilThematicMapId (@thematicmapid);  
  
select RgbString, LegendText from  
SDA_Get_AoiSoilThematicMapLegend_By_AoiSoilThematicMapId (@interpresultid);
```

Running this in the Query page yielded:

RgbString	LegendText
#5f5fd3	Fine-loamy, mixed, superactive, thermic Ustic Paleargids
#5fa95f	Loamy-skeletal, mixed, superactive, thermic Ustic Haplargids
#805555	Loamy-skeletal, mixed, superactive, thermic, shallow Cambidic Haplodurids
#fc5656	Loamy-skeletal, mixed, superactive, thermic, shallow Typic Haplargids
#dcdcdc	Not rated or not available

RgbString	LegendText
#5f5fd3	Fine-loamy, mixed, superactive, thermic Ustic Paleargids
#5fa95f	Loamy-skeletal, mixed, superactive, thermic Ustic Haplargids
#805555	Loamy-skeletal, mixed, superactive, thermic, shallow Cambidic Haplodurids
#fc5656	Loamy-skeletal, mixed, superactive, thermic, shallow Typic Haplargids
#dcdcdc	Not rated or not available

The above shows that the interpretation's legends match.

Similarly for the rating results,

```
~DeclareInt(@thematicmapid)~
~DeclareInt(@interpresultid)~

select @thematicmapid = 1504;
select @interpresultid = 528;

select MapUnitKey, MapUnitRatingString, MapUnitRatingNumeric, RgbString from
WSS_Get_AoiSoilThematicMapRating_By_AoiSoilThematicMapId(@thematicmapid);

select MapUnitKey, MapUnitRatingString, MapUnitRatingNumeric, RgbString from
SDA_Get_AoiSoilThematicMapRating_By_AoiSoilThematicMapId(@interpresultid);
```

Which yields:

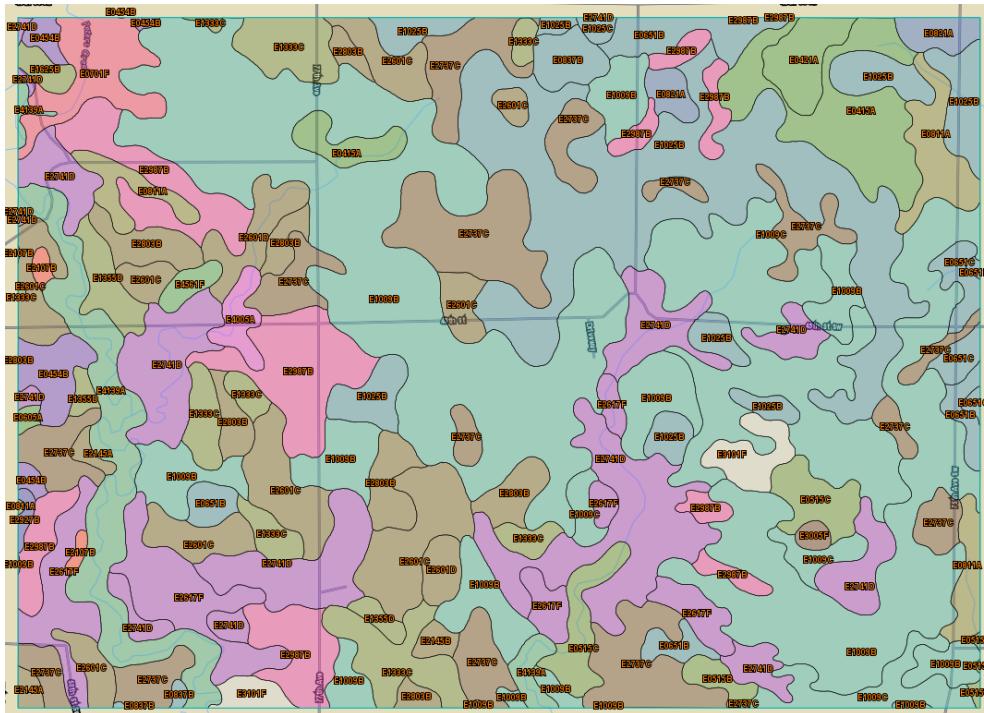
MapUnitKey	MapUnitRatingString	MapUnitRatingNumeric	RgbString
1860695	Loamy-skeletal, mixed, superactive, thermic, shallow Typic Haplargids		#fc5656
1860696	Loamy-skeletal, mixed, superactive, thermic, shallow Typic Haplargids		#fc5656
1860697	Loamy-skeletal, mixed, superactive, thermic Ustic Haplargids		#5fa95f
1860698	Loamy-skeletal, mixed, superactive, thermic, shallow Cambidic Haplodurids		#805555
1860699	Fine-loamy, mixed, superactive, thermic Ustic Paleargids		#5f5fd3
1860700	Loamy-skeletal, mixed, superactive, thermic Ustic Haplargids		#5fa95f
1900535			#dcdcdc

MapUnitKey	MapUnitRatingString	MapUnitRatingNumeric	RgbString
1860695	Loamy-skeletal, mixed, superactive, thermic, shallow Typic Haplargids		#fc5656
1860696	Loamy-skeletal, mixed, superactive, thermic, shallow Typic Haplargids		#fc5656
1860697	Loamy-skeletal, mixed, superactive, thermic Ustic Haplargids		#5fa95f
1860698	Loamy-skeletal, mixed, superactive, thermic, shallow Cambidic Haplodurids		#805555
1860699	Fine-loamy, mixed, superactive, thermic Ustic Paleargids		#5f5fd3
1860700	Loamy-skeletal, mixed, superactive, thermic Ustic Haplargids		#5fa95f
1900535			#dcdcdc

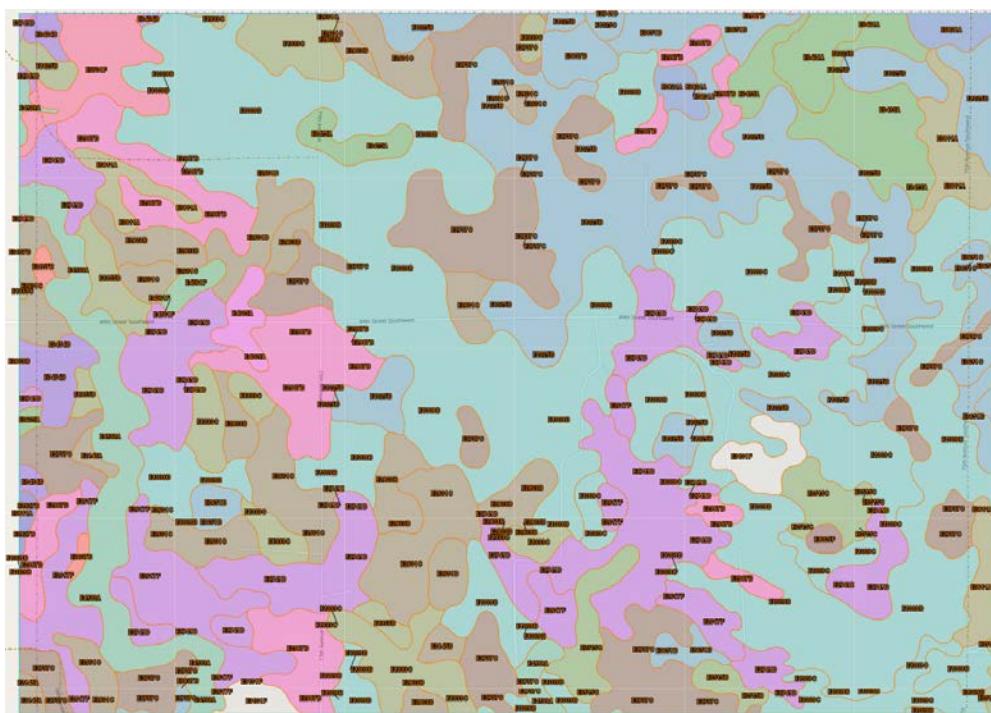
Once you've compared a few of these data sets you may want to move on to a much faster comparison technique, comparing the map images themselves between WSS and SDA. Turn the background layers off in WSS in order to get a better comparison.

Here for example are the Soil Taxonomy Classification maps from WSS and SDA for  
((-102.095 46.686,-102.095 46.717,-102.032 46.717,-102.032 46.686,-102.095 46.686))

WSS:



SDA:



## Specific Interpretation Tests

Following are a suggested series of tests. Where the "short form" data must be edited I've shown the edited version in formatted form, you may either use that formatted form in Test Post (I'll show that when it comes up) or in Test WMS. In addition the short form data should contain an "aoiid" that reflects the current SDA aoid for the test.

In all cases the maps, legend and rating data for WSS should match those from SDA. After a few you may choose to forgo the actual data comparison and stick to the maps.

Test #1: attributekey 471: interpretations/Land Classifications/Soil Taxonomy  
Classification  
SSA AOI District of Columbia DC001

Test #2 attributekey 471: interpretations/Land Classifications/Soil Taxonomy  
Classification  
Ad-hoc AOI District of Columbia DC001  
((-76.963 38.922,-76.920 38.892,-76.962 38.892,-76.963 38.922))

Test #3 attributekey 31: interpretations/Land Classifications/Ecological Site Name  
Ad-hoc AOI ND037 Grant County, North Dakota  
((-102.095 46.686,-102.095 46.717,-102.032 46.717,-102.032 46.686,-102.095 46.686))  
(basic option defaulted to "Forest Suitability Groups")

This test requires that the edited “short form” be entered. I’ll show how this would be done on the Test Post page.

```
Test #4 attributekey 31: interpretations/Land Classifications/Ecological Site Name  
Ad-hoc AOI ND037 Grant County, North Dakota  
((-102.095 46.686,-102.095 46.717,-102.032 46.717,-102.032 46.686,-102.095 46.686))  
(basic option set to "NRCS Rangeland Site")  
{  
    "ComponentPercentCutoff": "",  
    "aoiid": "517",  
    "ruleid": "31",  
    "ecoclassstypename": "NRCS Rangeland Site",  
    "AggregationMethod": "Dominant Condition",  
    "TiebreakRule": "-1"  
}
```

### Test 3: Perform REST/POST Interpretation Requests

interpretation request parameters

request	parameters
getusecategories(none)	
getcatalog	aoiid, usecategoryid (optional, defaults to {apid})
getruledata	aoiid, format (optional, defaults to "short")
getrating	shortformdata

Service Request

aoiid	interpretation
usecategoryid	getrating
format	
attributekey	
interpresultid	
themematicapid	

shortformdata

```
{  
    "ComponentPercentCutoff": "",  
    "aoiid": "524",  
    "ruleid": "95",  
    "AggregationMethod": "D",  
    "TiebreakRule": "1"  
}
```

Specify "getrating".

The edited (if required) and formatted (if desired) short form data goes here.

Click here.

1. Click following button to issue a REST/POST request.

run using POST

...cleared...

Test #5 attributekey 25: interpretations/Vegetative Productivity/Yields of Non-Irrigated Crops (Component)

Ad-hoc AOI Pickaway County, Ohio OH129

((-83.037 39.679,-83.037 39.697,-82.990 39.697,-82.990 39.679,-83.037 39.679))

(basic option defaulted to "Alfalfa hay"/Tons)

Test #6 attributekey 25: interpretations/Vegetative Productivity/Yields of Non-Irrigated Crops (Component)

Ad-hoc AOI Pickaway County, Ohio OH129

((-83.037 39.679,-83.037 39.697,-82.990 39.697,-82.990 39.679,-83.037 39.679))

(basic option set to Oats/Bu)

```
{  
    "ComponentPercentCutoff": "",  
    "aoiid": "20837",  
    "ruleid": "25",  
    "cropname": "Oats",  
    "yldunits": "Bu",  
    "AggregationMethod": "Weighted Average",  
    "TiebreakRule": "1",  
    "InterpretNullsAsZero": "True"  
}
```

Test #6 attributekey 25: interpretations/Vegetative Productivity/Yields of Non-Irrigated Crops (Component)

Ad-hoc AOI Pickaway County, Ohio OH129

((-83.037 39.679,-83.037 39.697,-82.990 39.697,-82.990 39.679,-83.037 39.679))

(basic option set to Pasture/AUM)

```
{  
    "ComponentPercentCutoff": "",  
    "aoiid": "20837",  
    "ruleid": "25",  
    "cropname": "Pasture",  
    "yldunits": "AUM",  
    "AggregationMethod": "Weighted Average",  
    "TiebreakRule": "1",  
    "InterpretNullsAsZero": "True"  
}
```

Test #7 attributekey 77: interpretations/Vegetative Productivity/Forest Productivity (Cubic Feet per Acre per Year)

Ad-hoc AOI Piscataquis County, Maine, Southern Part ME615

((-69.089 45.342,-69.089 45.403,-68.931 45.403,-68.931 45.342,-69.089 45.342))

(basic option defaulted to "American beech"/"Hampf 1965 (165)")

```

Test #8 attributekey 77: interpretations/Vegetative Productivity/Forest Productivity
(Cubic Feet per Acre per Year)
Ad-hoc AOI Piscataquis County, Maine, Southern Part ME615
((-69.089 45.342,-69.089 45.403,-68.931 45.403,-68.931 45.342,-69.089 45.342))
(basic option set to "balsam fir"/"Gevorkiantz 1956a (010)")
{
    "ComponentPercentCutoff":"",
    "aoiid":"20838",
    "ruleid":"77",
    "plantcomname":"balsam fir",
    "siteindexbase":"Gevorkiantz 1956a (010)",
    "AggregationMethod":"Dominant Component",
    "TiebreakRule":"1",
    "InterpretNullsAsZero":"False"
}

Test #9 attributekey 77: interpretations/Vegetative Productivity/Forest Productivity
(Cubic Feet per Acre per Year)
Ad-hoc AOI Piscataquis County, Maine, Southern Part ME615
((-69.089 45.342,-69.089 45.403,-68.931 45.403,-68.931 45.342,-69.089 45.342))
(basic option set to "balsam fir"/"Lloyd 1970a (020)"")
{
    "ComponentPercentCutoff":"",
    "aoiid":"20838",
    "ruleid":"77",
    "plantcomname":"balsam fir",
    "siteindexbase":"Lloyd 1970a (020)",
    "AggregationMethod":"Dominant Component",
    "TiebreakRule":"1",
    "InterpretNullsAsZero":"False"
}

Test #10 attributekey 98: properties/Soil Chemical Properties/Calcium Carbonate
(CaCO3)
Ad-hoc AOI Bottineau County, North Dakota (ND009)
((-101.221 48.920,-101.221 49.000,-101.023 49.000,-101.023 48.920,-101.221 48.920))
WSS advanced option settings:
    "Aggregation Method"/"Dominant Component"
    "Component Percent Cutoff"/
    "Tie-break Rule"/"Higher"
    "Interpret Nulls as Zero"/Yes
    "Layer Options"/"Depth Range"
    "Top Depth"/10
    "Bottom Depth"/500
    "Units of Measure"/"Centimeters"
Shortformdata:
{
    "AggregationMethod":"Dominant Component",
    "ComponentPercentCutoff":"",
    "TiebreakRule":"1",
    "InterpretNullsAsZero":"True",
    "HorizonAggregationMethod":"Depth Range",
    "TopDepth":"10",
    "BottomDepth":"500",
    "DepthUnits":"Centimeters",
    "aoiid":"20839",
    "ruleid":"98"
}

```

```

Test #11 attributekey 98: properties/Soil Chemical Properties/Calcium Carbonate
(CaCO3)
Ad-hoc AOI Bottineau County, North Dakota (ND009)
((-101.221 48.920,-101.221 49.000,-101.023 49.000,-101.023 48.920,-101.221 48.920))
WSS advanced option settings:
    "Aggregation Method"/"Minimum or Maximum"
    "Component Percent Cutoff"/
    "Tie-break Rule"/"Higher"
    "Interpret Nulls as Zero"/Yes
    "Layer Options"/"Depth Range"
    "Top Depth"/10
    "Bottom Depth"/500
    "Units of Measure"/"Centimeters"
Shortformdata:
{
    "AggregationMethod": "Minimum or Maximum",
    "ComponentPercentCutoff": "",
    "TiebreakRule": "1",
    "InterpretNullsAsZero": "True",
    "HorizonAggregationMethod": "Depth Range",
    "TopDepth": "10",
    "BottomDepth": "500",
    "DepthUnits": "Centimeters",
    "aoiid": "20839",
    "ruleid": "98"
}

Test #12 attributekey 98: properties/Soil Chemical Properties/Calcium Carbonate
(CaCO3)
Ad-hoc AOI Bottineau County, North Dakota (ND009)
((-101.221 48.920,-101.221 49.000,-101.023 49.000,-101.023 48.920,-101.221 48.920))
WSS advanced option settings:
    "Aggregation Method"/"Dominant Component"
    "Component Percent Cutoff"/"75"
    "Tie-break Rule"/"Higher"
    "Interpret Nulls as Zero"/Yes
    "Layer Options"/"Depth Range"
    "Top Depth"/10
    "Bottom Depth"/500
    "Units of Measure"/"Centimeters"
Shortformdata:
{
    "AggregationMethod": "Dominant Component",
    "ComponentPercentCutoff": "75",
    "TiebreakRule": "1",
    "InterpretNullsAsZero": "True",
    "HorizonAggregationMethod": "Depth Range",
    "TopDepth": "10",
    "BottomDepth": "500",
    "DepthUnits": "Centimeters",
    "aoiid": "20839",
    "ruleid": "98"
}

```

```

Test #13 attributekey 98: properties/Soil Chemical Properties/Calcium Carbonate
(CaCO3)
Ad-hoc AOI Bottineau County, North Dakota (ND009)
((-101.221 48.920,-101.221 49.000,-101.023 49.000,-101.023 48.920,-101.221 48.920))
WSS advanced option settings:
    "Aggregation Method"/"Dominant Component"
    "Component Percent Cutoff"/""
    "Tie-break Rule"/"Lower"
    "Interpret Nulls as Zero"/Yes
    "Layer Options"/"Depth Range"
    "Top Depth"/10
    "Bottom Depth"/500
    "Units of Measure"/"Centimeters"
Shortformdata:
{
    "AggregationMethod": "Dominant Component",
    "ComponentPercentCutoff": "",
    "TiebreakRule": "-1",
    "InterpretNullsAsZero": "True",
    "HorizonAggregationMethod": "Depth Range",
    "TopDepth": "10", "BottomDepth": "500",
    "DepthUnits": "Centimeters",
    "aoiid": "20839", "ruleid": "98"
}

Test #14 attributekey 98: properties/Soil Chemical Properties/Calcium Carbonate
(CaCO3)
Ad-hoc AOI Bottineau County, North Dakota (ND009)
((-101.221 48.920,-101.221 49.000,-101.023 49.000,-101.023 48.920,-101.221 48.920))
WSS advanced option settings:
    "Aggregation Method"/"Dominant Component"
    "Component Percent Cutoff"/""
    "Tie-break Rule"/"Lower"
    "Interpret Nulls as Zero"/No
    "Layer Options"/"Depth Range"
    "Top Depth"/10
    "Bottom Depth"/500
    "Units of Measure"/"Centimeters"
Shortformdata:
{
    "AggregationMethod": "Dominant Component",
    "ComponentPercentCutoff": "",
    "TiebreakRule": "-1",
    "InterpretNullsAsZero": "False",
    "HorizonAggregationMethod": "Depth Range",
    "TopDepth": "10", "BottomDepth": "500",
    "DepthUnits": "Centimeters",
    "aoiid": "20839", "ruleid": "98"
}

```

```

Test #15 attributekey 98: properties/Soil Chemical Properties/Calcium Carbonate
(CaCO3)
Ad-hoc AOI Bottineau County, North Dakota (ND009)
((-101.221 48.920,-101.221 49.000,-101.023 49.000,-101.023 48.920,-101.221 48.920))
WSS advanced option settings:
    "Aggregation Method"/"Dominant Component"
    "Component Percent Cutoff"/""
    "Tie-break Rule"/"Lower"
    "Interpret Nulls as Zero"/No
    "Layer Options"/"Surface Layer"
    "Top Depth"/10
    "Bottom Depth"/500
    "Units of Measure"/"Centimeters"
Shortformdata:
{
    "AggregationMethod": "Dominant Component",
    "ComponentPercentCutoff": "",
    "TiebreakRule": "-1",
    "InterpretNullsAsZero": "False",
    "HorizonAggregationMethod": "Surface Layer",
    "TopDepth": "", "BottomDepth": "",
    "DepthUnits": "Centimeters",
    "aoiid": "20839", "ruleid": "98"
}

Test #16 attributekey 98: properties/Soil Chemical Properties/Calcium Carbonate
(CaCO3)
Ad-hoc AOI Bottineau County, North Dakota (ND009)
((-101.221 48.920,-101.221 49.000,-101.023 49.000,-101.023 48.920,-101.221 48.920))
WSS advanced option settings:
    "Aggregation Method"/"Dominant Component"
    "Component Percent Cutoff"/""
    "Tie-break Rule"/"Lower"
    "Interpret Nulls as Zero"/No
    "Layer Options"/"All Layers"
    "Top Depth"/10
    "Bottom Depth"/500
    "Units of Measure"/"Centimeters"
Shortformdata:
{
    "AggregationMethod": "Dominant Component",
    "ComponentPercentCutoff": "",
    "TiebreakRule": "-1",
    "InterpretNullsAsZero": "False",
    "HorizonAggregationMethod": "All Layers",
    "TopDepth": "", "BottomDepth": "",
    "DepthUnits": "Centimeters",
    "aoiid": "20839", "ruleid": "98"
}

```

Test #17 attributekey 82: properties/Water Features/Depth to Water Table  
Ad-hoc AOI Bottineau County, North Dakota (ND009)  
((-101.221 48.920,-101.221 49.000,-101.023 49.000,-101.023 48.920,-101.221 48.920))

Test #18 attributekey 82: properties/Water Features/Depth to Water Table  
Ad-hoc AOI Bottineau County, North Dakota (ND009)  
((-101.221 48.920,-101.221 49.000,-101.023 49.000,-101.023 48.920,-101.221 48.920))

WSS advanced option settings:

- "Aggregation Method"/"Dominant Component"
- "Component Percent Cutoff"/""
- "Tie-break Rule"/"Lower"
- "Interpret Nulls as Zero"/"No"
- "Layer Options"/"All Layers"
- "Beginning Month"/"May"
- "Ending Month"/"October"

Shortformdata:

```
{  
    "ComponentPercentCutoff": "",  
    "aoiid": "20839", "ruleid": "82",  
    "AggregationMethod": "Dominant Component",  
    "TiebreakRule": "-1",  
    "InterpretNullsAsZero": "False",  
    "BeginningMonth": "May",  
    "EndingMonth": "October"  
}
```

## Geolocation

Web applications, using the “Geolocation API”, can determine your device’s current location (for privacy reasons, you are asked for permission to report location information). This location determination is approximate and depends upon a number of factors (see <http://www.andygup.net/html5-geolocation-api-%E2%80%93-how-accurate-is-it-really/> and [https://en.wikipedia.org/wiki/W3C\\_Geolocation\\_API](https://en.wikipedia.org/wiki/W3C_Geolocation_API)).

Per the Wikipedia article,

Usually geolocation will try to determine a device's position using one of these several methods.

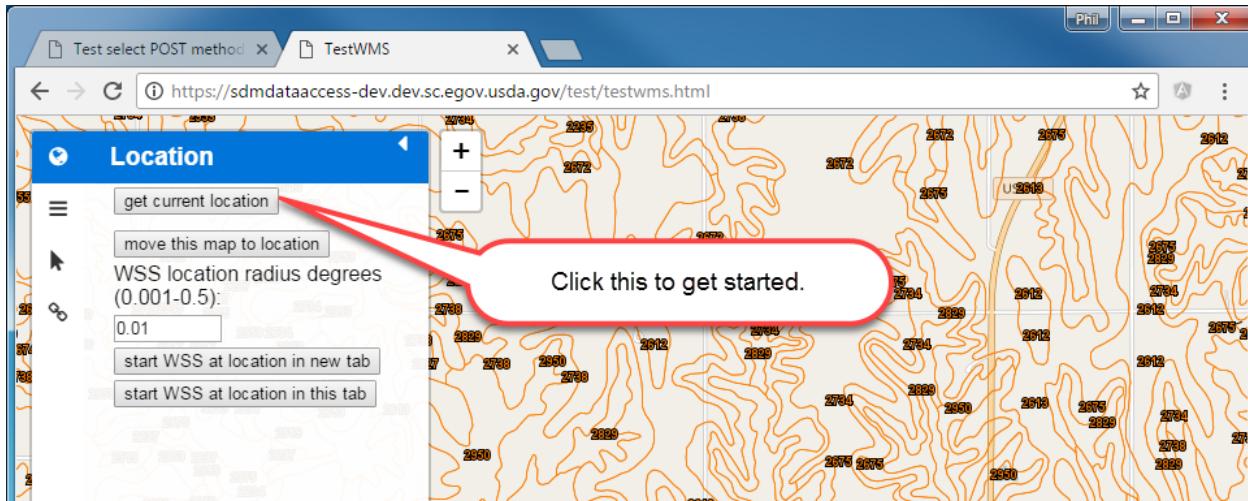
### GPS (Global Positioning System)

This happens for any device which has GPS capabilities. A smartphone with GPS capabilities and set to high accuracy mode will be likely to obtain the location data from this. GPS calculate location information from the satellite signal. It has the highest accuracy; in most Android smartphones, the accuracy can be up to 10 metres.

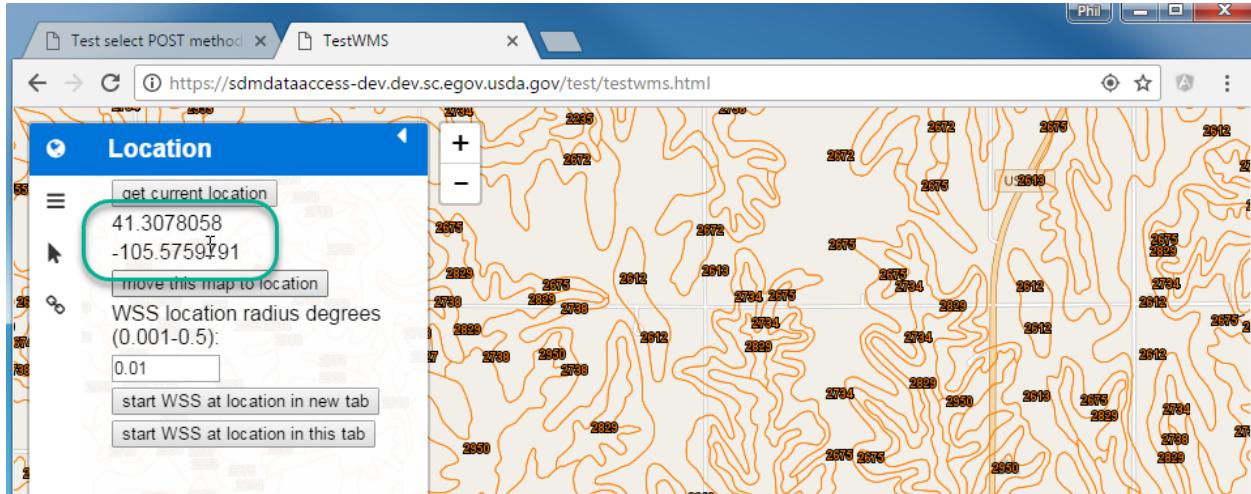
Your mileage will vary.

By default the TestWMS page starts roughly at the center of the 48 contiguous US states, we can move to our current location.

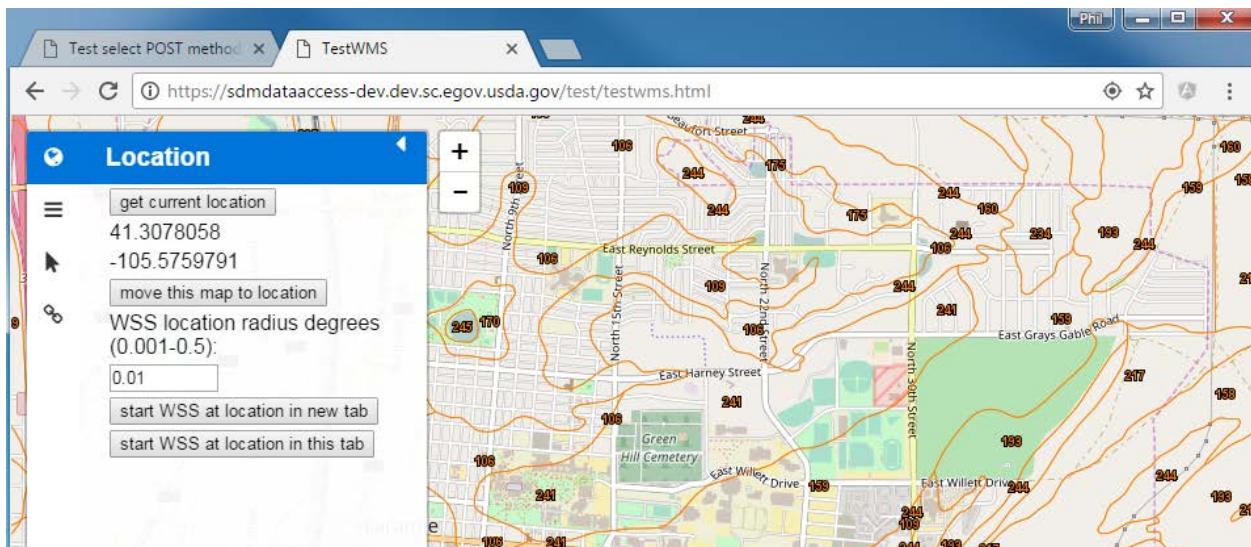
Pull out the Geolocation tab on the sidebar:



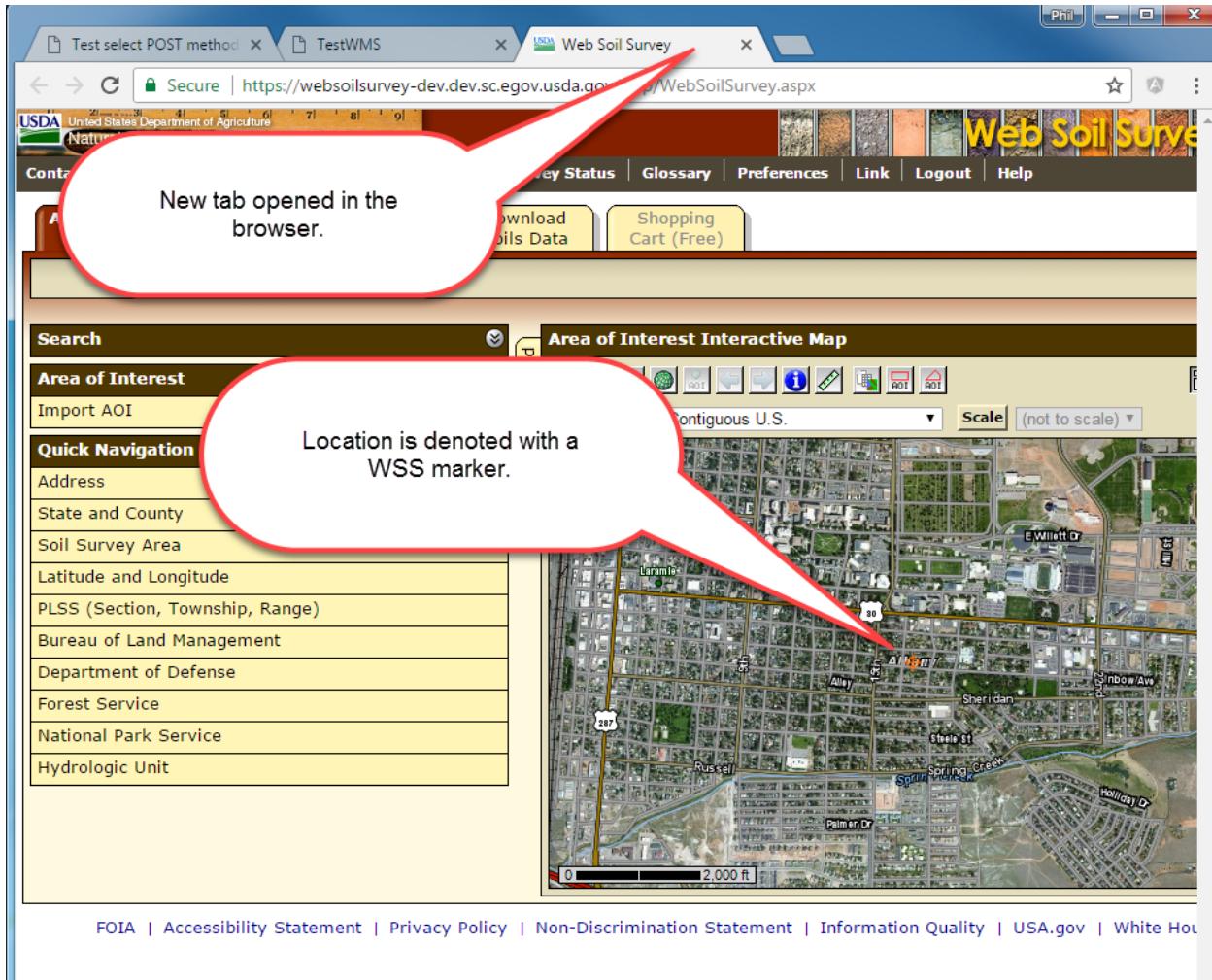
Once your location has been determined (you'll get a popup asking for your permission, Internet Explorer will likely deny you permission),



you can use any of the remaining buttons. For example, clicking on the “move this map to location” pans the map to the identified location:



There are two alternatives to starting Web Soil Survey at the identified location. By default WSS is started with a displayed area about 0.01 degrees around the identified point, this can be changed as desired before clicking on the either of the “start WSS...” buttons. Here I’ve clicked on the “start WSS at location in new tab” button:



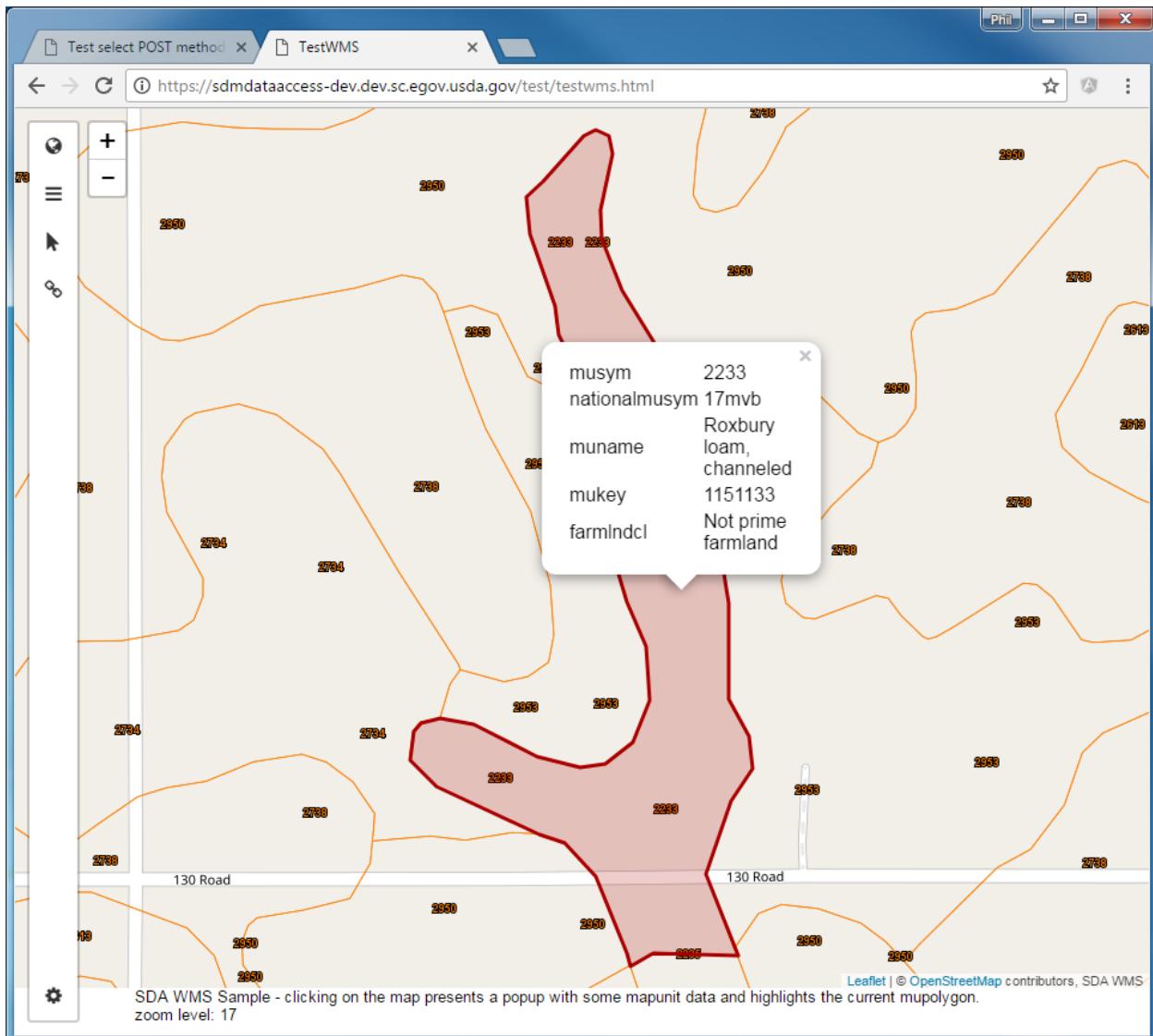
### What can go wrong here?

1. WSS was previously opened in the browser and you did not log out but you did close the browser’s WSS tab. In that case you may see a WSS “Session Ended” page. Close the WSS tab, return to Test WMS and click on the button again.
2. WSS has gone “to sleep” in the development environment (this is unlikely in the production environment where WSS is in constant use). In this case, after a delay, WSS will show up with the default map extent displayed. Close the WSS tab, return to Test WMS and click on the button again.

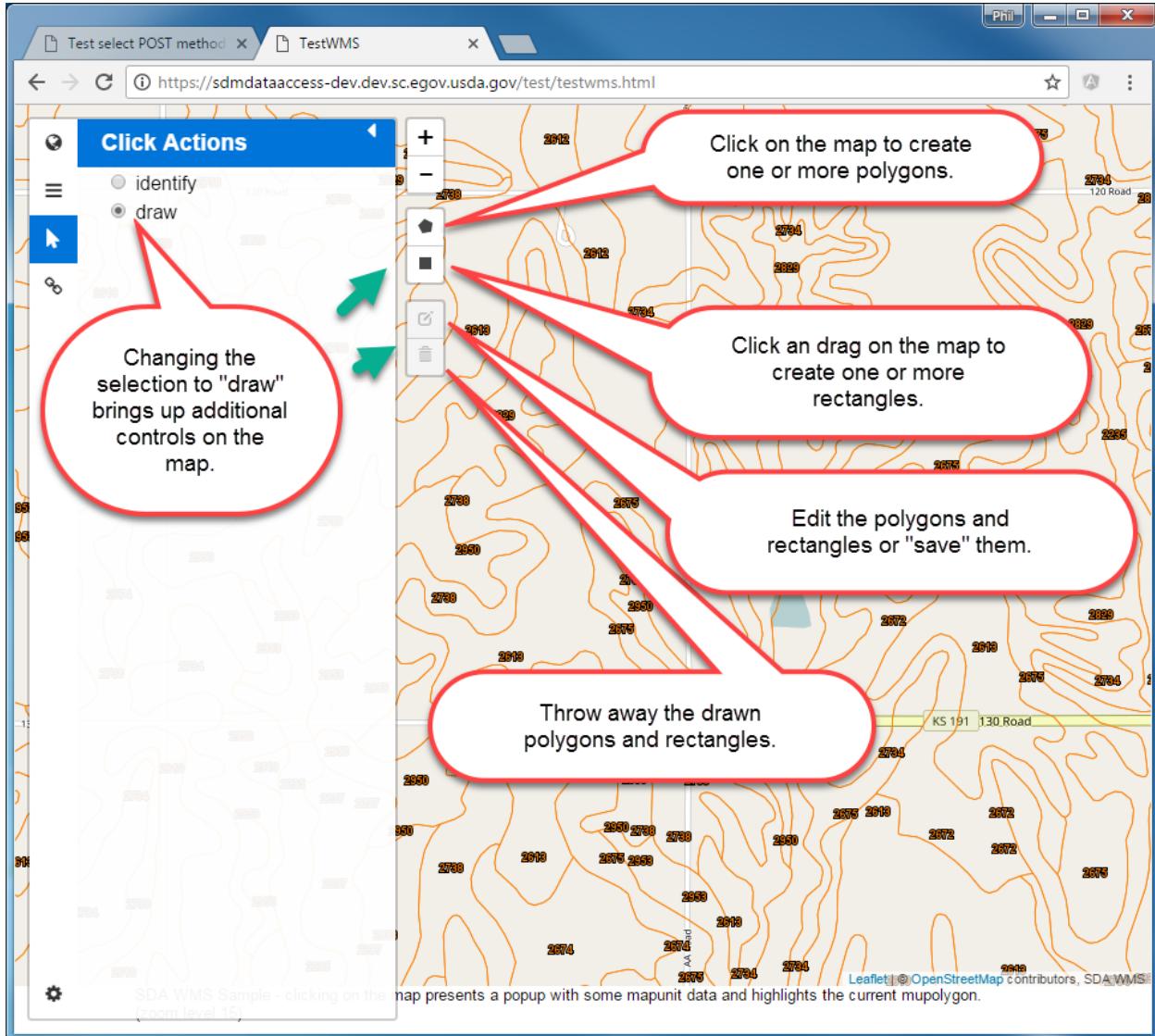
The method used to start WSS is still experimental. Should this capability be useful it can likely be improved upon. The other button, “start WSS at location in this tab” replaces the TestWMS page with WSS. If WSS is fully awake this approach is reasonable.

## Identify or Draw

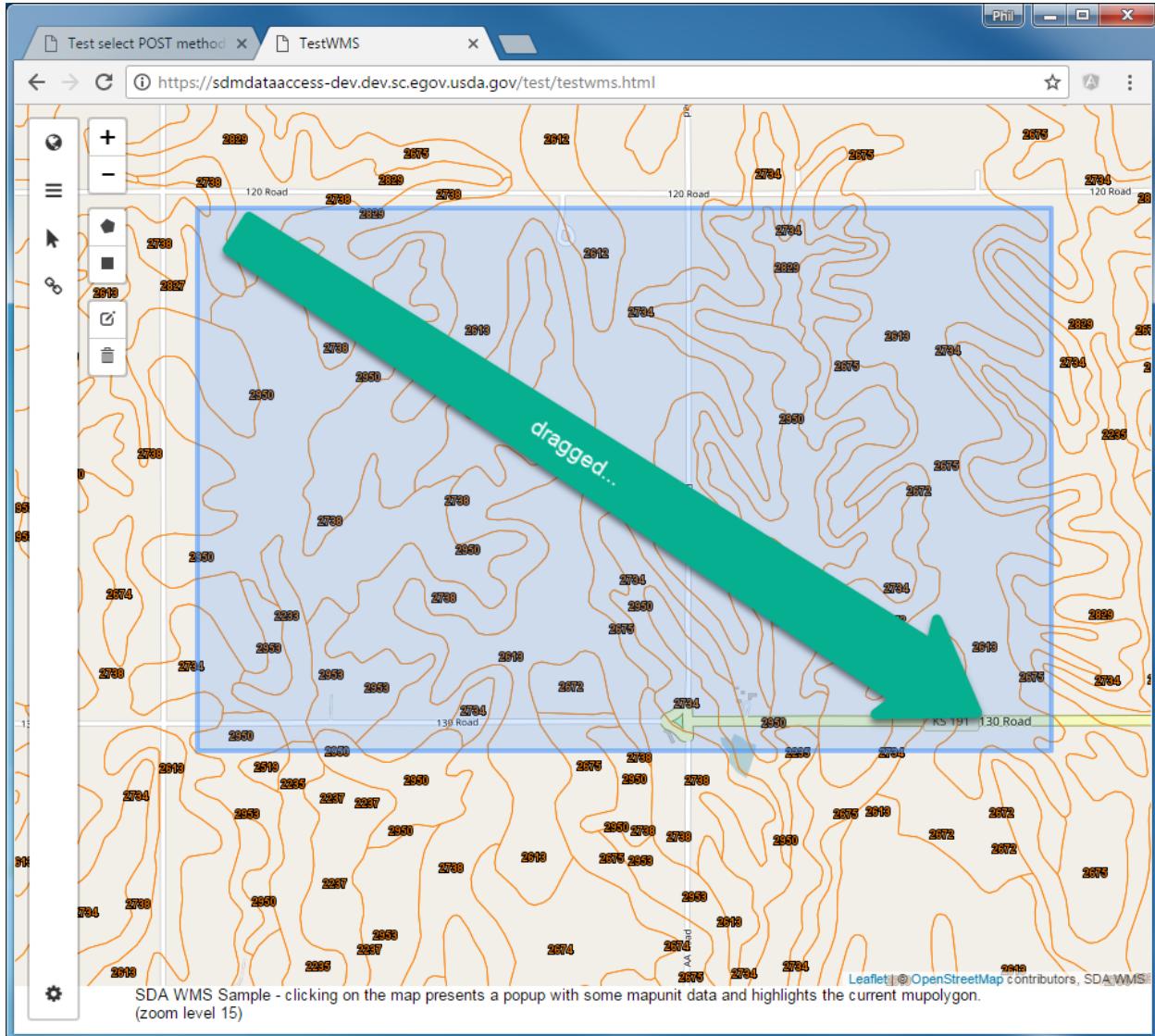
By default TestWMS lets you click on the map and get thumbnail information about the mapunit polygon where you clicked:



In place of this default “Identify” behavior you can turn on a “drawing” capability by selecting that option from the “Identify or Draw” pullout of the sidebar:

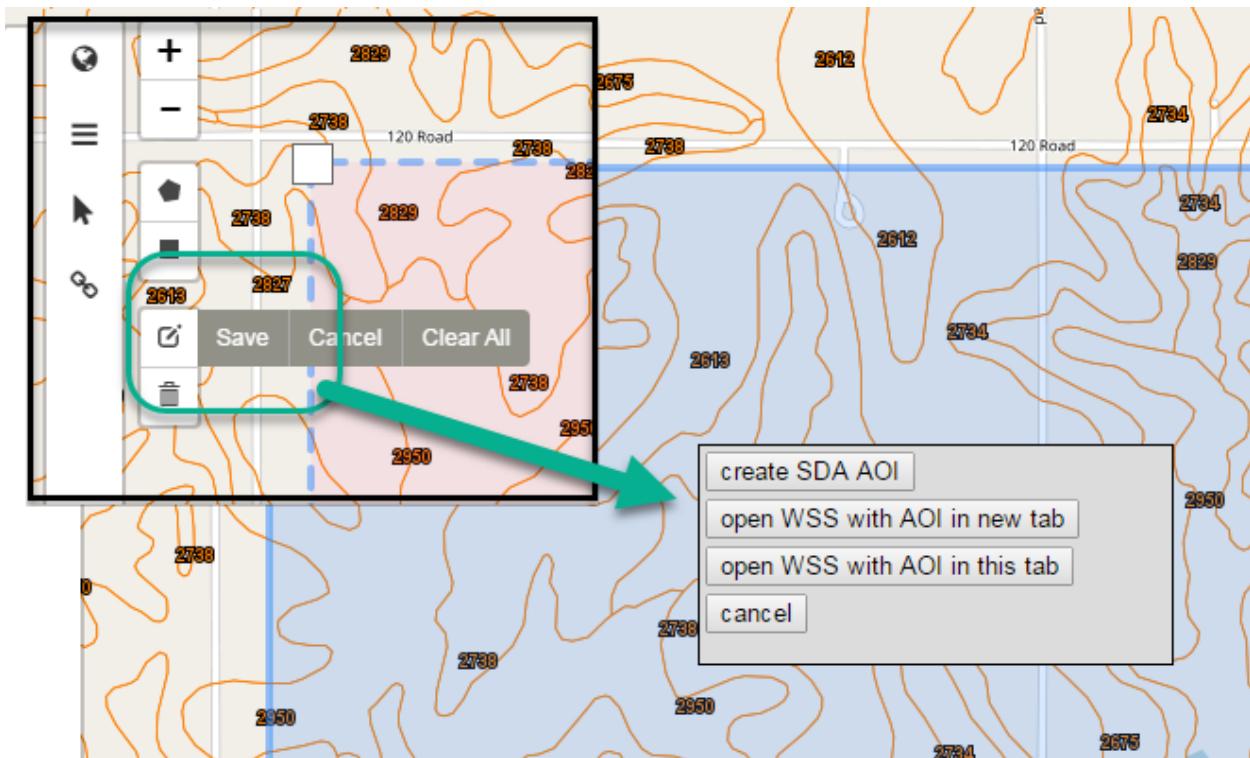


Try clicking on the “rectangle” symbol and using the mouse drag (or a touch drag) draw a rectangle on the map:

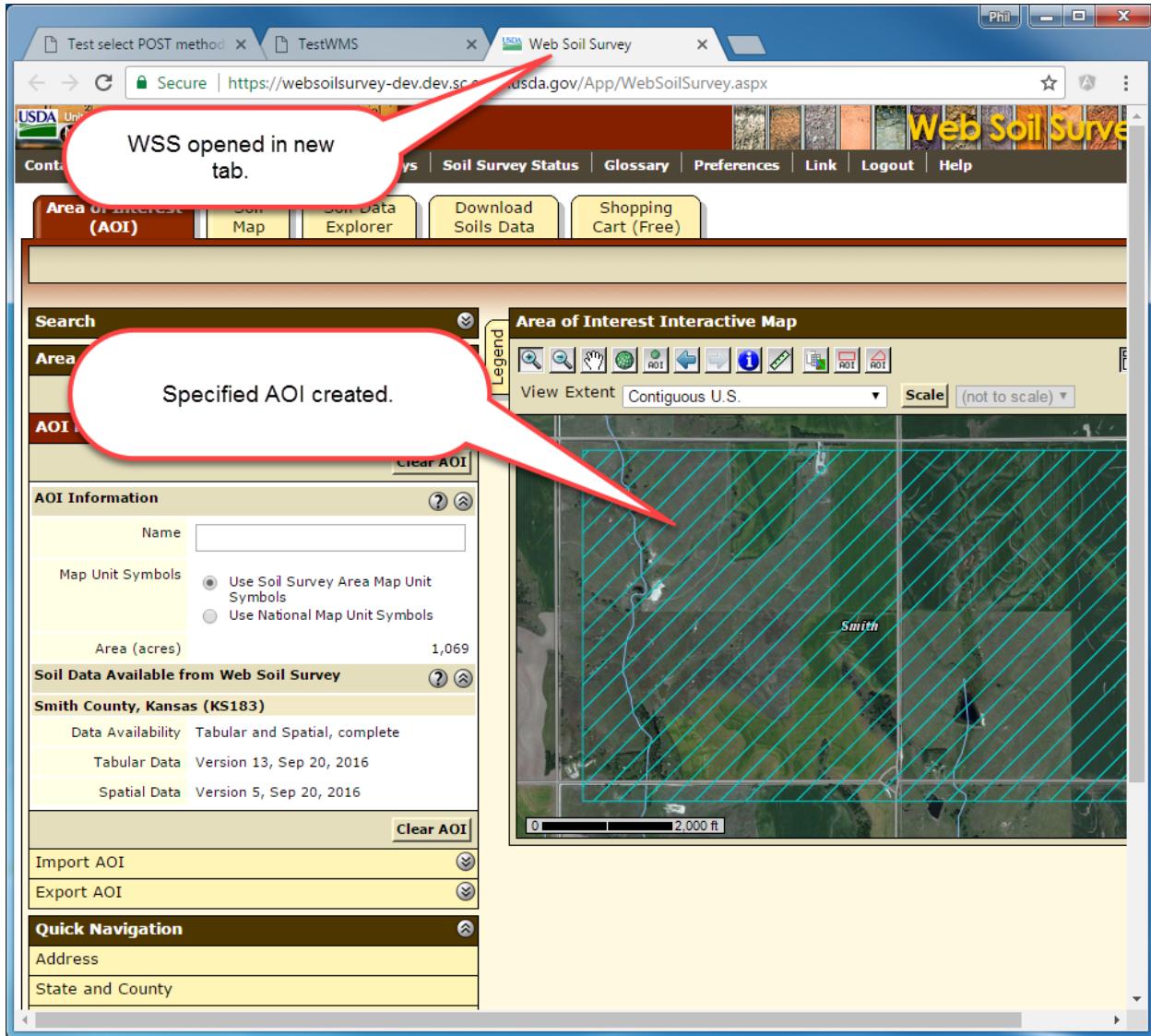


Clicking on the trash can allows you to remove the drawn rectangle (“Clear All”, “Save” is not implemented here). Don’t do that at this point unless you want to play with drawing other rectangles or polygons.

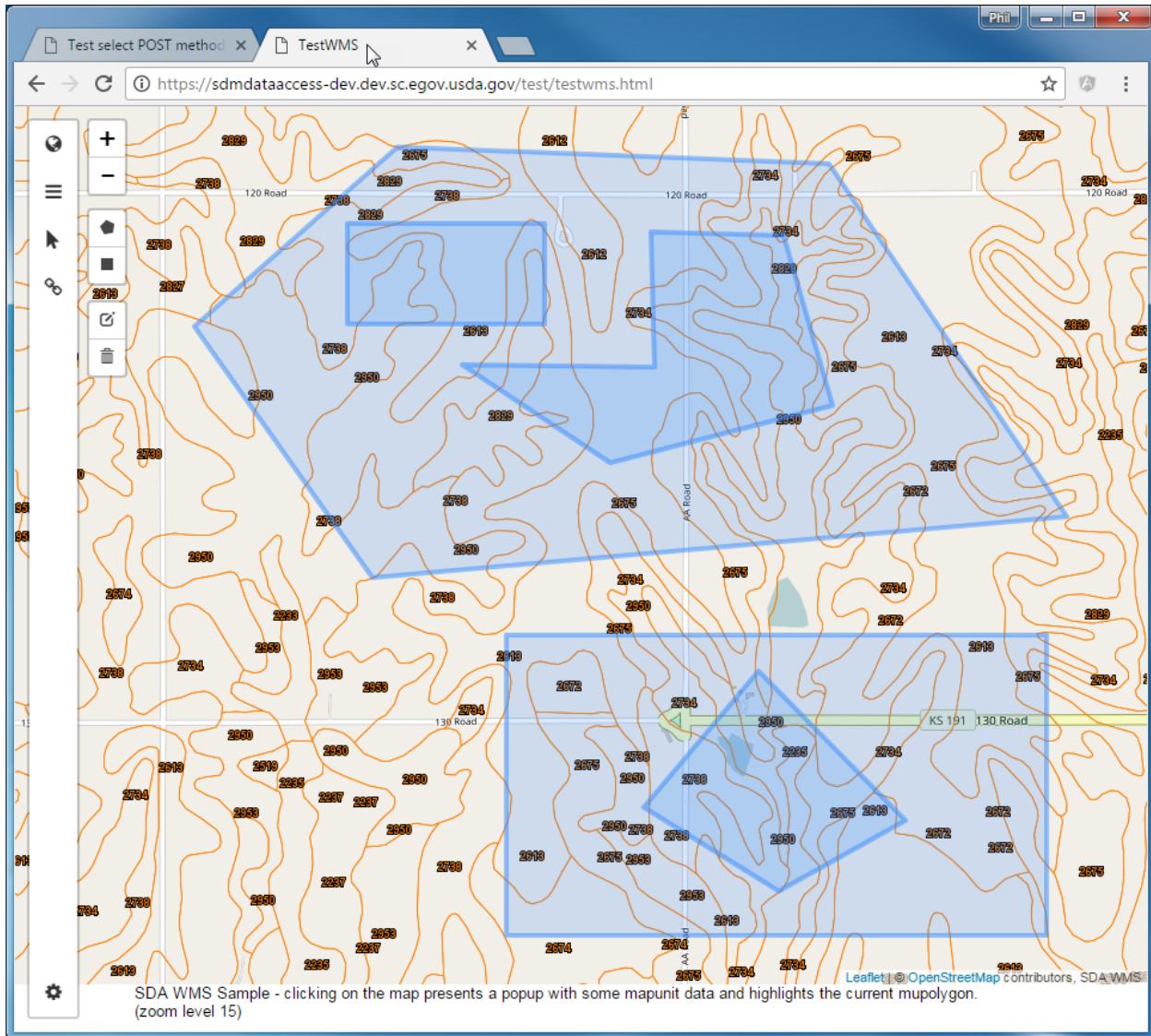
Clicking on the “Edit” button allows you to modify the rectangle(s) and/or polygon(s) that you’ve drawn. The “Save” option also brings up some additional choices:



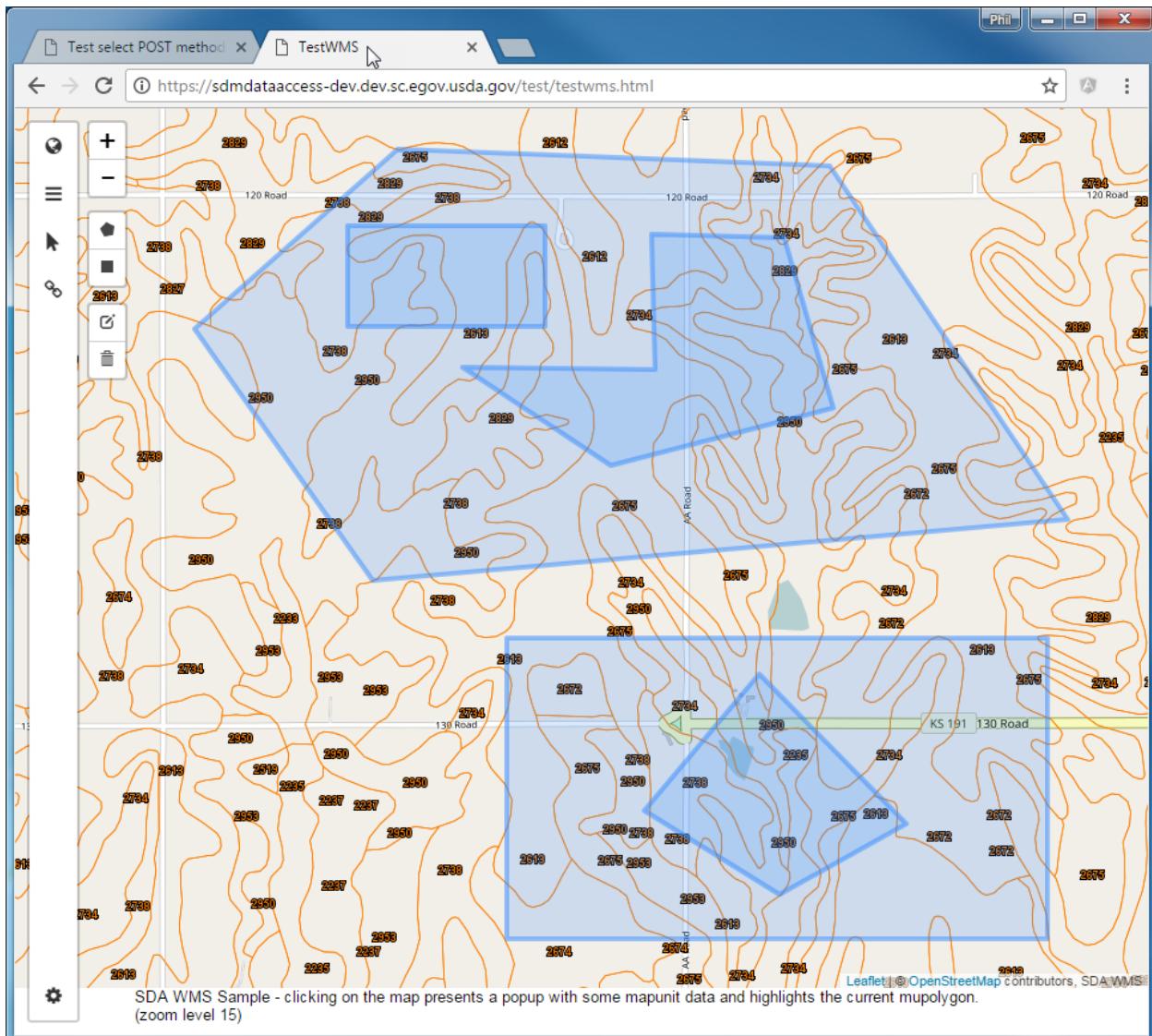
In the case of “create SDA AOI” a new pAOI is created and the map is moved to show the new AOI. Using “open WSS with AOI in new tab” does just what it claims (save for the previous “what could go wrong” notes):



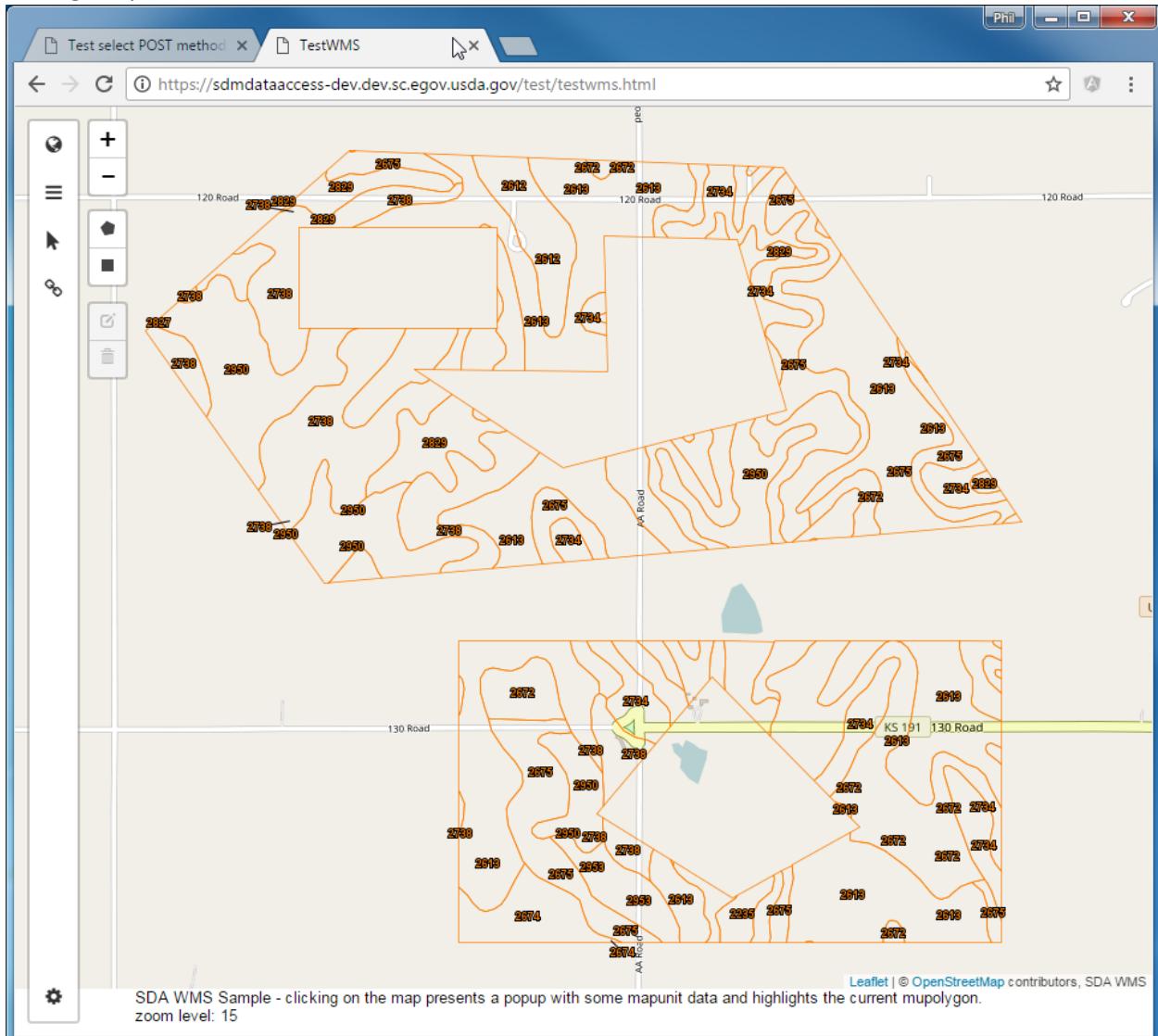
You may also draw multiple polygons and rectangles. Overlapped polygons can yield AOIs with interior voids:



In WSS this yields:



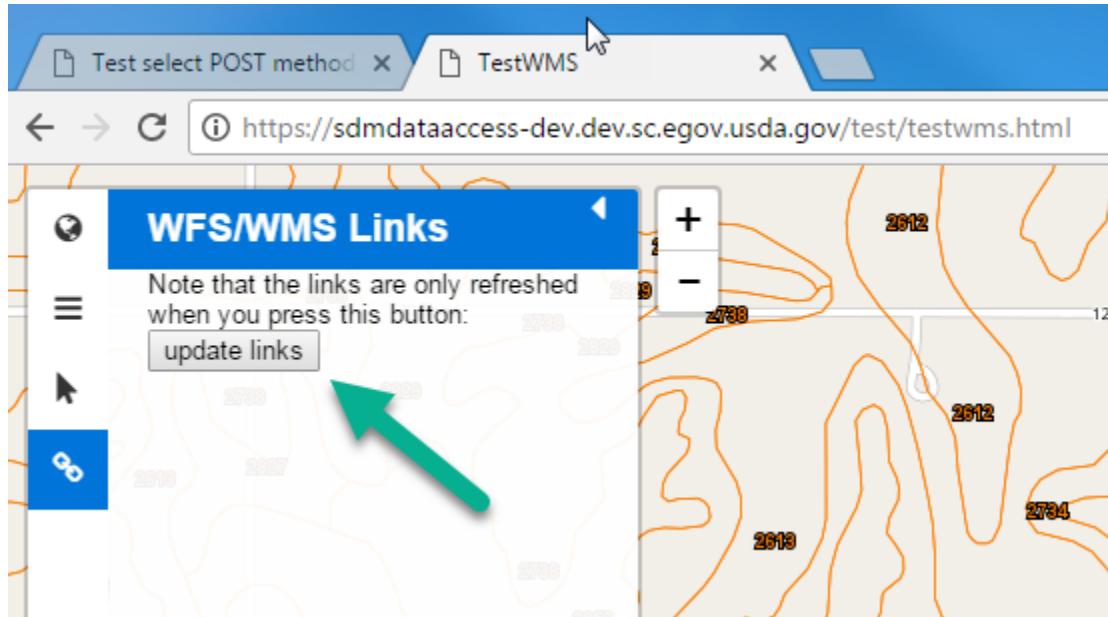
Analogously in SDA,



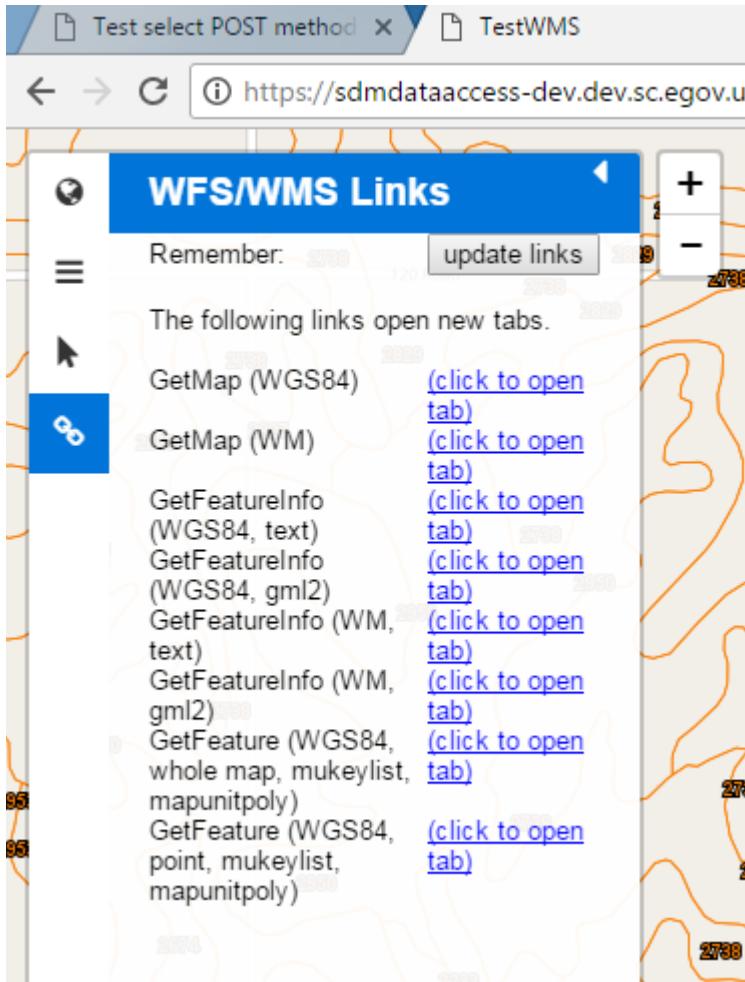
In this point in the evolution of this experimental facility the created AOI, whether in SDA or WSS, is a “non-multipart AOI”: while it may have multiple regions, they are treated as a whole.

## WMS and WFS service links

The “WFS/WMS Links” pullout demonstrates additional WFS and WMS SDA service use (the source code of the TestWMS page can also be mined for details of post.rest, WFS and WMS service use in support of the page). Start by opening the slide-out:



The links are not actively maintained while you work on the map, so you need to manually update the links before using them:



Each of these links opens a new Web browser tag. Your currently-selected layers and any AoId, SLD\_ID, thematicmapid specifications are used as with the links as appropriate to the service. Try them all.

