SDA2

Advanced Features: Context and Capabilities

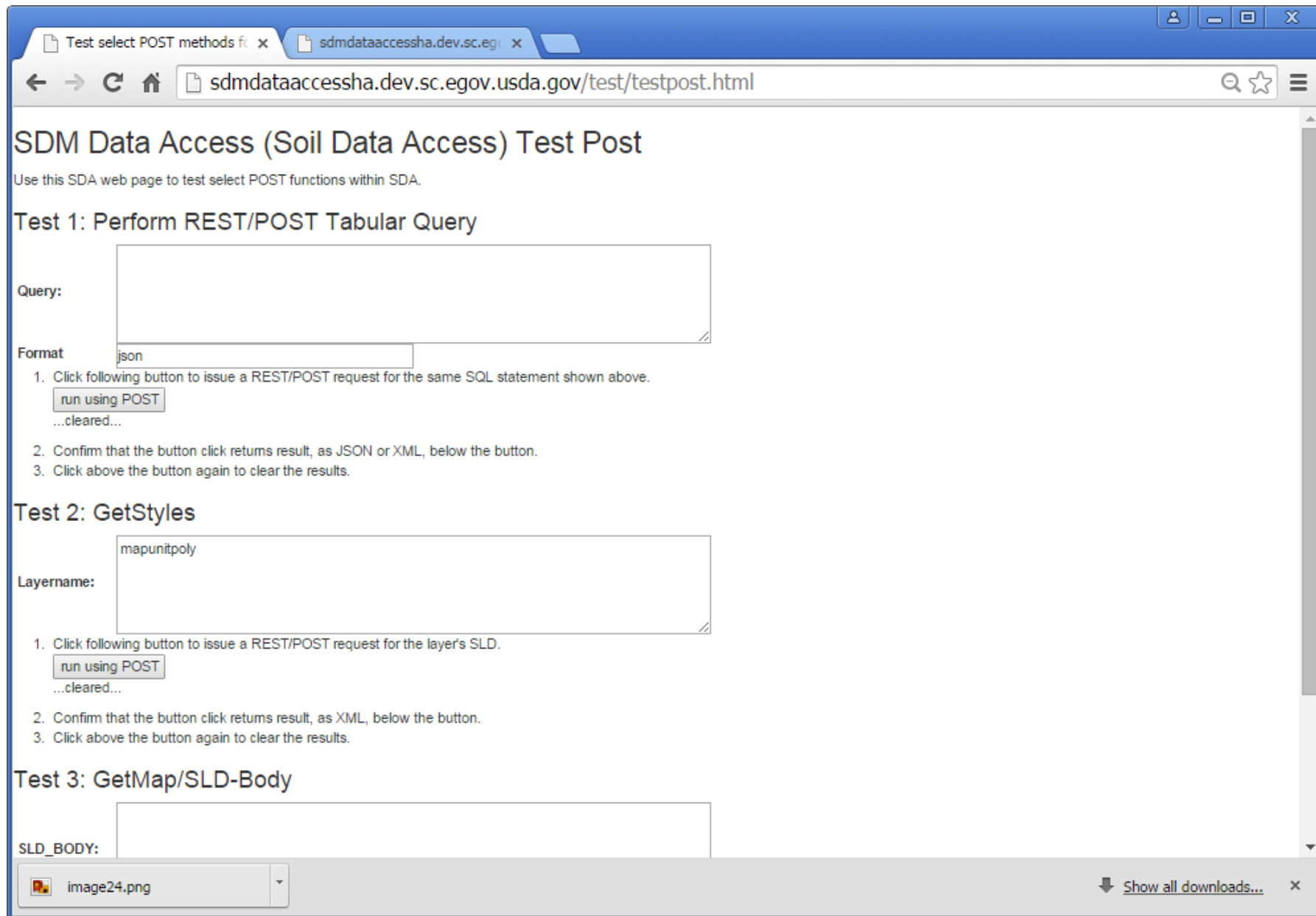Phil Anzel August 11, 2015

Agenda

1. Expressed needs
2. Typical application architectures using SDA
3. Related changes in SDM/WSS 3.0.
4. Pseudo-Web Mercator support
5. REST/POST and JSON support
6. Tabular-Spatial fusion in query context (the "Advanced" document)
7. WMS GetStyles and styled layer description
8. Future possibilities

Resources

- http://sdmdataaccess.sc.egov.usda.gov/ - SDA in production environment

- http://sdmdataaccess.nrcs.usda.gov/ - alias for above

- http://sdmdataaccessha.dev.sc.egov.usda.gov/ - SDA development version, only available on USDA backbone

- http://sdmdataaccessha.dev.sc.egov.usda.gov/documents/AdvancedSpatialDataAccess.pdf - advanced features document, to be copied to production late August

- http://sdmdataaccessha.dev.sc.egov.usda.gov/test/TestPost.html - Test page for trying JSON, GetStyles, SLD_BODY, to be copied to production late August

Advanced Spatial Data Access

Introduction

Tabular queries can access spatial data stored in the Soil Data Mart (SDM) database. Spatial data is stored in tables such as mupolygon, mupoint and muline as pairs of columns, one with the data in WGS84 geographic coordinates (epsg:4326) and one with the data in Web Mercator (WM, epsg:3857 – this is not quite accurate as true epsg:3857 uses a flattened sphere for coordinate projection, in the SDM database the WM data are actually projected using a perfect sphere, this pseudo-Web Mercator projection matches that used by common Web sources such as Google Maps and Microsoft Bing maps). The column names are formed as follows: <table-name><[geo|proj]>. For example in table "mupolygon", the WGS84 data are held in column "mupoygongeo".

A set of SQL tabular functions have been defined to simplify navigation between coordinates and SDM table data keys. For more advanced applications a series of macros have been included to provide extra functionality that may not normally be available through queries. In addition there is no restriction on referencing the spatial tables through normal queries.

Tabular Functions

These functions return a "table" so may be treated in SQL statements as a "FROM" table source.

Note that the WKT must follow the ISO standard definition (see the Wikipedia page at

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

Query:

Format json

1. Click following button to issue a REST/POST request for the same SQL statement shown above.

    run using POST
    ...cleared...

2. Confirm that the button click returns result, as JSON or XML, below the button.
3. Click above the button again to clear the results.

Test 2: GetStyles

Layername: mapunitpoly

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

    run using POST
    ...cleared...

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

Test 3: GetMap/SLD-Body

SLD_BODY:

1. (Informally-) Expressed needs

a. Dylan B.
b. Web Accessibility Teleconference
c. Department of Energy
d. Steve P.
e. Client Gateway
f. Comet-Farm
g. FarmOS/Cornell Soils Health
h. Current SDA users
i. NTT/RSS
j. Daily checks and development testing

1. Expressed needs

a. **Dylan B.**
b. Web Accessibility Teleconference
c. Department of Energy
d. Steve P.
e. Client Gateway
f. Comet-Farm
g. FarmOS/Cornell Soils Health
h. Current SDA users
i. NTT/RSS
j. Daily checks and development testing

10/25/2013 email:
- Need for mupolygon clipping to an AOI

1. Expressed needs

a. Dylan B.

b. Web Accessibility Teleconference

c. Department of Energy

d. Steve P.

e. Client Gateway

f. Comet-Farm

g. FarmOS/Cornell Soils Health

h. Current SDA users

i. NTT/RSS

j. Daily checks and development testing

SDA aspects of 12/5/2013 teleconference:
- access to spatial data via queries
- JSON/GeoJSON
- alternative to SOAP for queries
- GSSURGO data access joined to SDM data
- WFS SSURGO data by bounding box or other criteria

1. Expressed needs

a. Dylan B.
b. Web Accessibility Teleconference
c. **Department of Energy**
d. Steve P.
e. Client Gateway
f. Comet-Farm
g. FarmOS/Cornell Soils Health
h. Current SDA users
i. NTT/RSS
j. Daily checks and development testing

7/30/2014 email:
- Need customization of map appearance
- SDA accessed via OpenLayers (Web client Javascript library)

1. Expressed needs

a. Dylan B
b. Web Accessibility Teleconference.
c. Department of Energy
d. **Steve P.**
e. Client Gateway
f. Comet-Farm
g. FarmOS/Cornell Soils Health
h. Current SDA users
i. NTT/RSS
j. Daily checks and development testing

12/4/2014 email:
- Join spatial and tabular queries based upon AOI and mukey lookup
- Return clipped polygons

1. Expressed needs

a. Dylan B.
b. Web Accessibility Teleconference
c. Department of Energy
d. Steve P.
e. **Client Gateway**
f. Comet-Farm
g. FarmOS/Cornell Soils Health
h. Current SDA users
i. NTT/RSS
j. Daily checks and development testing

2/12/2014 SDA/WSS Demo for CG:
- WSS AOI multipolygons
- Thematic maps (and data?) required

1. Expressed needs

a. Dylan B.
b. Web Accessibility Teleconference
c. Department of Energy
d. Steve P.
e. Client Gateway
f. **Comet-Farm**
g. FarmOS/Cornell Soils Health
h. Current SDA users
i. NTT/RSS
j. Daily checks and development testing

7/2/2015-present email and telephone dialog
- Restore Web Mercator for WMS
- POST versus GET for WFS
- Polygon clipping to an AOI
- SDA accessed via server-based Mapserver acting as both WMS client and WMS server

1. Expressed needs

a. Dylan B.
b. Web Accessibility Teleconfer...
c. Department of Energy
d. Steve P.
e. Client Gateway
f. Comet-Farm

g. FarmOS/Cornell Soils Health

h. Current SDA users
i. NTT/RSS
j. Daily checks and development testing

7/21/2015-present email and Google Hangout dialog
- Restore Web Mercator for WMS
- Accessing SDM data (WFS or tabular/spatial query)
- SDA accessed via OpenLayers (Web client Javascript library)

1. Expressed needs

a. Dylan B.
b. Web Accessibility Teleconference
c. Department of Energy
d. Steve P.
e. Client Gateway
f. Comet-Farm
g. FarmOS/Cornell Soils Health
h. Current SDA users
i. NTT/RSS
j. Daily checks and development testing

Ongoing:
- Maintain backwards compatibility

1. Expressed needs

a. Dylan B.
b. Web Accessibility Teleconference
c. Department of Energy
d. Steve P.
e. Client Gateway
f. Comet-Farm
g. FarmOS/Cornell Soils Health
h. Current SDA users
i. NTT/RSS
j. Daily checks and development testing

1/2/2015 - present email and face-to-face:
- Provide SOAP and better tabular query access
- Return polygons clipped to AOI
- Provide true geographic area calculation for clipped polygons
- Integrate AOI selection with arbitrary SQL queries
- SDA accessed via .NET/SOAP, tabular queries and possibly Leaflet (Web client Javascript library)

1. Expressed needs

a. Dylan B.
b. Web Accessibility Teleconference
c. Department of Energy
d. Steve P.
e. Client Gateway
f. Comet-Farm
g. FarmOS/Cornell Soils Health
h. Current SDA users
i. NTT/RSS
j. Daily checks and development testing

- Need quick-to-use HTTP interface for testing tabular and spatial services
- Need quick-to-develop single page to support PSO daily checks

2. Typical application architectures using SDA

- tabular – query.aspx
- tabular via soap -> Python/ArcMap, R?
- tabular via soap -> .NET server-side (NTT, RSS)
- tabular via rest/post –> web apps (NTT, RSS)
- WMS -> browser
- WMS -> ArcMap
- WMS -> OpenLayers/Leaflet in browser (e.g., farmOS/Cornell Soils Health, NTT)
- WMS -> intermediate mapserver -> CometFarm
- WFS -> browser
- WFS -> ArcMap (data interoperability?)
- WFS -> intermediate mapserver -> CometFarm

3. Related changes in SDM/WSS 3.0.

- NAD83 -> WGS84
- Parallel WGS84 and pseudo-Web Mercator data
- WSS map images changed to pseudo-Web Mercator

4. Pseudo-Web Mercator support

- Used for Bing and Google Maps.

- Introduced into SDM and WSS 3.0.

- Advertised in SDA and WSS as true Web Mercator (epsg:3857) - we're lying!

- SDA online documentation updated to indicate this bit of deception.

- Should we clean up our act?

- Different coordinate conversion math than that used by true Web Mercator.

reference: https://alastaira.wordpress.com/2011/01/23/the-google-maps-bing-maps-spherical-mercator-projection/

5. REST/POST and JSON support

- REST is easier to program than SOAP-based requests

- JSON is easier to consume in client-side Web applications

- JSON is more concise - for the query `select top 2 mukey from mapunit` as both XML and JSON:

<NewDataSet> <xs:schema id="NewDataSet" xmlns=""
xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns:msdata="urn:schemas-microsoft-
com:xml-msdata"> <xs:element name="NewDataSet" msdata:IsDataSet="true"
msdata:UseCurrentLocale="true"> <xs:complexType> <xs:choice minOccurs="0"
maxOccurs="unbounded"> <xs:element name="Table"> <xs:complexType>
<xs:sequence> <xs:element name="mukey" type="xs:string" minOccurs="0" />
</xs:sequence> </xs:complexType> </xs:element> </xs:choice> </xs:complexType>
</xs:element> </xs:schema> <Table> <mukey>49315</mukey> </Table> <Table>
<mukey>49316</mukey> </Table> </NewDataSet>
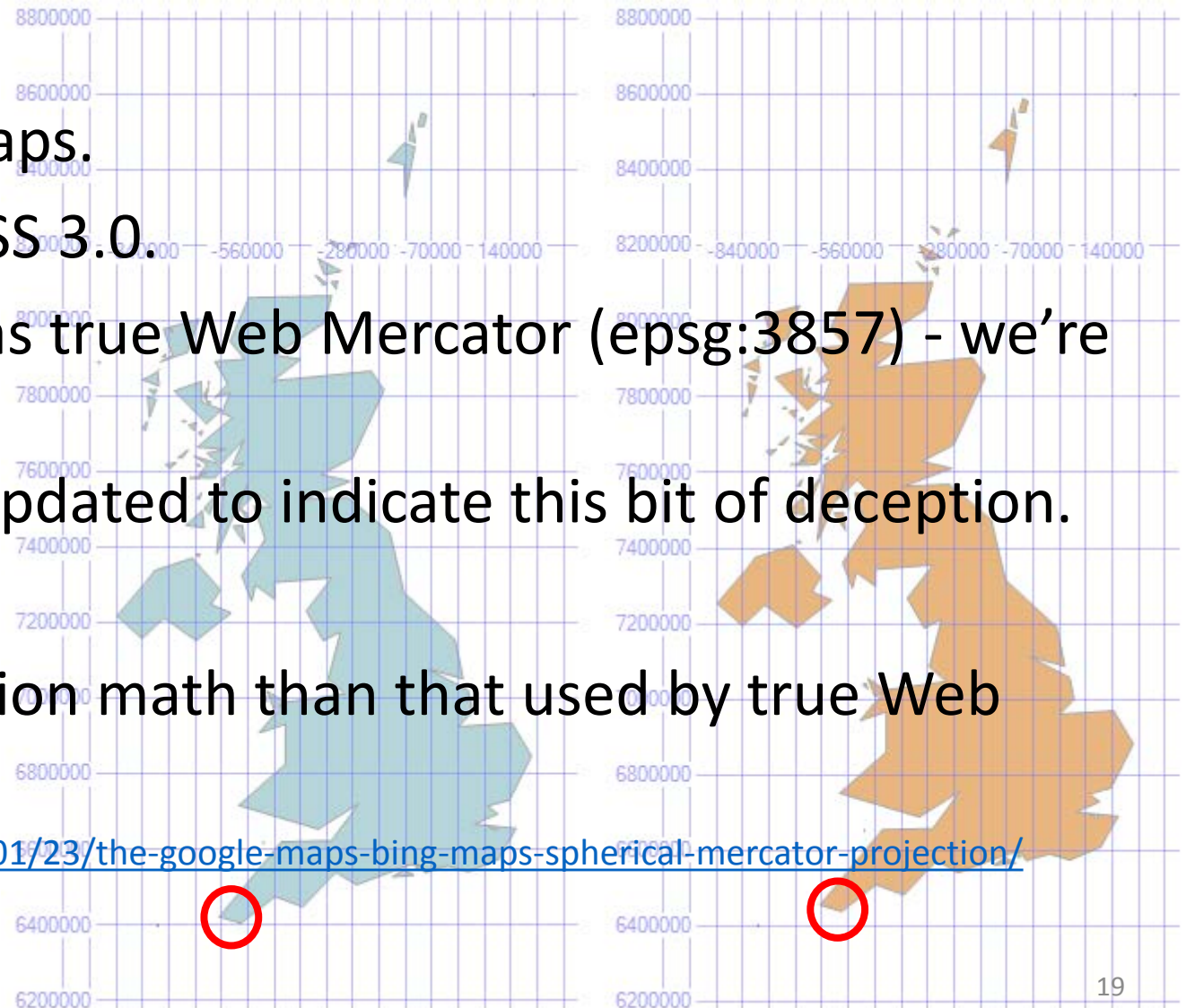
{"Table":[["49315"],["49316"]]}

6. Tabular-Spatial fusion in query context (the "Advanced" document)

a. UDF table sources

b. Access to all spatial tables using T-SQL queries

c. Macros

    i. Defense against SQL injection and server abuse

    ii. Shorthand for complicated or obscure SQL statements

    iii. Optimized/specialized tasks

    iv. Support polygon clipping to AOI and clipped area determination

    v. Can be updated without requiring a full SDA release

6. Tabular-Spatial fusion in query context (the "Advanced" document)

a. UDF table sources

b. Access to all spatial tables using T-SQL queries

c. Macros

i. Defenses against injection and server abuse

ii. Shorthand for complicated or obscure SQL statements

iii. Optimized/specialized tasks

iv. Support polygon composition, QA analysis, area determination

v. Can be updated without re-securing entire data source

Syntactically these functions return a "table" so may be treated in SQL statements as a "FROM" table source. For example:

```
select S.mukey, M.musym
from SDA_Get_Mukey_from_intersection_with_WktWm(
    'polygon((
        -13555610.9782664 4490483.16765171,
        -13555610.9782664 4491391.31746713,
        -13554135.9341103 4491391.31746713,
        -13554135.9341103 4490483.16765171,
        -13555610.9782664 4490483.16765171))') as S,
    legend as L, mapunit M
where M.mukey = S.mukey and M.lkey = L.lkey and L.areasymbol = 'CA646'
```

6. Tabular-Spatial fusion in query context (the "Advanced" document)

a. UDF table sources

b. Access to all spatial tables using T-SQL queries

c. Macros

i. Defense against SQL injection and server abuse

ii. Shorthand for complicated or obscure SQL statements

iii. Optimized/performant tasks

iv. Support indirect clients in ACL-antioned area determination

v. Can be ...

For example:

```
select mukey, sum(muareaacres)
from mupolygon
where mukey in (49315,49316,49317,49318,49319)
group by mukey
```

Standard spatial functions can also be used:

```
select L.areasymbol, P.nationalmusym
from mupolygon P, mapunit M, legend L
where P.mupolygonproj.STIntersects(
    geometry::STGeomFromText('point(-13555610.9782664
4490483.16765171)', 3857)) = 1
and P.mukey = M.mukey and M.lkey = L.lkey
```

6. Tabular-Spatial fusion in query context (the "Advanced" document)

a. UDF table sources

b. Access to all spatial tables using T-SQL queries

c. Macros

i. Defense against SQL injection and server abuse

ii. Shorthand for complicated or obscure SQL statements

iii. Optimize frequently used tasks

iv. Support polygon clipping to AOI and clipped area determination

v. Can be updated without requiring a full SDA release

"a single instruction that expands automatically into a set of instructions to perform a particular task."

In SDA, tabular queries macros are surrounded by the "tilde" character ("~") and any macro arguments must follow strict syntax rules.

Macros may be used in any SDA query context – in the Query.aspx page for both immediate and queued queries, in SOAP requests and in REST/POST requests.

6. Tabular-Spatial fusion in query context (the "Advanced" document)

a. UDF table sources

b. Access to all spatial tables using T-SQL queries

c. Macros

i. Defense against SQL injection and server abuse

ii. Shorthand for complicated or obscure SQL statements

iii. Optimized/specialized tasks

iv. Support polygon clipping to AOI and clipped area determination

v. Can be updated without requiring a full SDA release

Macros enable certain "bad words" (such as "declare") to be simulated in a strictly-controlled fashion within the query, protecting SDA from "SQL Injection attacks" and other potential exploits.

For example, to declare a "varchar(max)" variable named "@abc" in the batch (or "script") of T-SQL statements, the keyword "declare" may not be used. The following macro statement provides the same declaration:

```
~DeclareVarchar(@abc,max)~
```

For example, the macro statement:

```
  ~GetClippedMapunits(@aoi,polygon,geo,@intersectedPolygonGeometries)~
is expanded into
  -- ~GetClippedMapunits(@aoi,polygon,geo,@intersectedPolygonGeometries)~
  -- begin macro substitution
  insert into @intersectedPolygonGeometries
  SELECT id, geom.STGeometryN(Numbers.n).MakeValid() as geom
  FROM
   (select mukey as id, mupolygongeo.STIntersection(@aoi) as geom
    from mupolygon
    where mupolygongeo.Filter(@aoi) = 1) as G
   JOIN #_Numbers_ as Numbers
   ON Numbers.n <= G.geom.STNumGeometries();
  -- end macro substitution
```

a. UDF table...

b. Access to...

c. Macros

i. Defense against SQL injection and server abuse

ii. Shorthand for complicated or obscure SQL statements

iii. Optimized/specialized tasks

iv. Support polygon clipping to AOI and clipped area determination

v. Can be updated without requiring a full SDA release

6. Tabular-Spatial fusion in query context (the "Advanced" document)

a. UDF table sources

b. Access to all spatial tables using T-SQL queries

c. Macros

i. Defer logic to runtime

ii. Shorthand for complicated or obscure SQL statements

iii. Optimized/specialized tasks

iv. Support polygon clipping to AOI and clipped area determination

v. Can be updated without requiring a full SDA release

The previously-cited macro, `GetClippedMapunits`, illustrates both optimization and a specialized task. A non-obvious approach is used to perform a set-based operation upon all geometries contained within each "geom" object without requiring a "cursor" or "while" loop, both of which are expensive (and also "bad words"). The isolation of clipped polygons is a task required by a number of applications dependent upon SDA.

6. Tabular-Spatial fusion in query context (the "Advanced" document)

a. UDF table sources

b. Access to all spatial tables using T-SQL queries

c. Macros

i. Defense against SQL injection and server abuse

ii. Shorthand for complicated or obscure SQL statements

iii. Optimized/specialized tasks

iv. Support polygon clipping to AOI and clipped area determination

v. Can be updated without requiring a full SDA release

The "Advanced" document provides a fully-worked example of these tasks.

6. Tabular-Spatial fusion in query context (the "Advanced" document)

a. UDF table sources

b. Access to all spatial tables using T-SQL queries

c. Macros

i. Def...

ii. Sho...

iii. Op...

iv. Support polygon clipping to AOI and clipped area determination

> The macros are defined in a "static configuration text file" and as such may be updated in production by an application support request. The update does not trigger the need for a QA cycle nor a full deployment of SDA. The update may be performed during the day, requiring a file replacement and "IIS application pool recycling", minimizing impact upon users.

v. Can be updated without requiring a full SDA release

7. WMS GetStyles and styled layer description

a. Custom map generation

b. Ad-hoc thematic map creation with WSS comparison

To-date SDA's Web Map Service (WMS) largely provided a one-size-fits all painting of maps. There existed (and still exists) some rudimentary control of transparency, background color, and polygon fill. There has also been a "SYTLES" specifier that was never implemented in SDA. The painting has proven unsatisfactory for some users.

SDA uses the "OSGeo Mapserver" libraries to satisfy WMS (and Web Feature Service, WFS) requests. For a number of years Mapserver has offered client-side painting control via a "Styled Layer Description" (SLD) specification, a short fragment of XML akin to a CSS specification used in painting Web pages. SLD specifications for a layer may be retrieved with the GetStyles request, and an edited SLD used to control map painting via the SLD_BODY of a GetMap request.

7. WMS GetStyles and styled layer description

a. **Custom map generation**

b. Ad-hoc thematic map creation with WSS comparison

For the mapunit polygons shown in WSS as isolated by the AOI specfication:
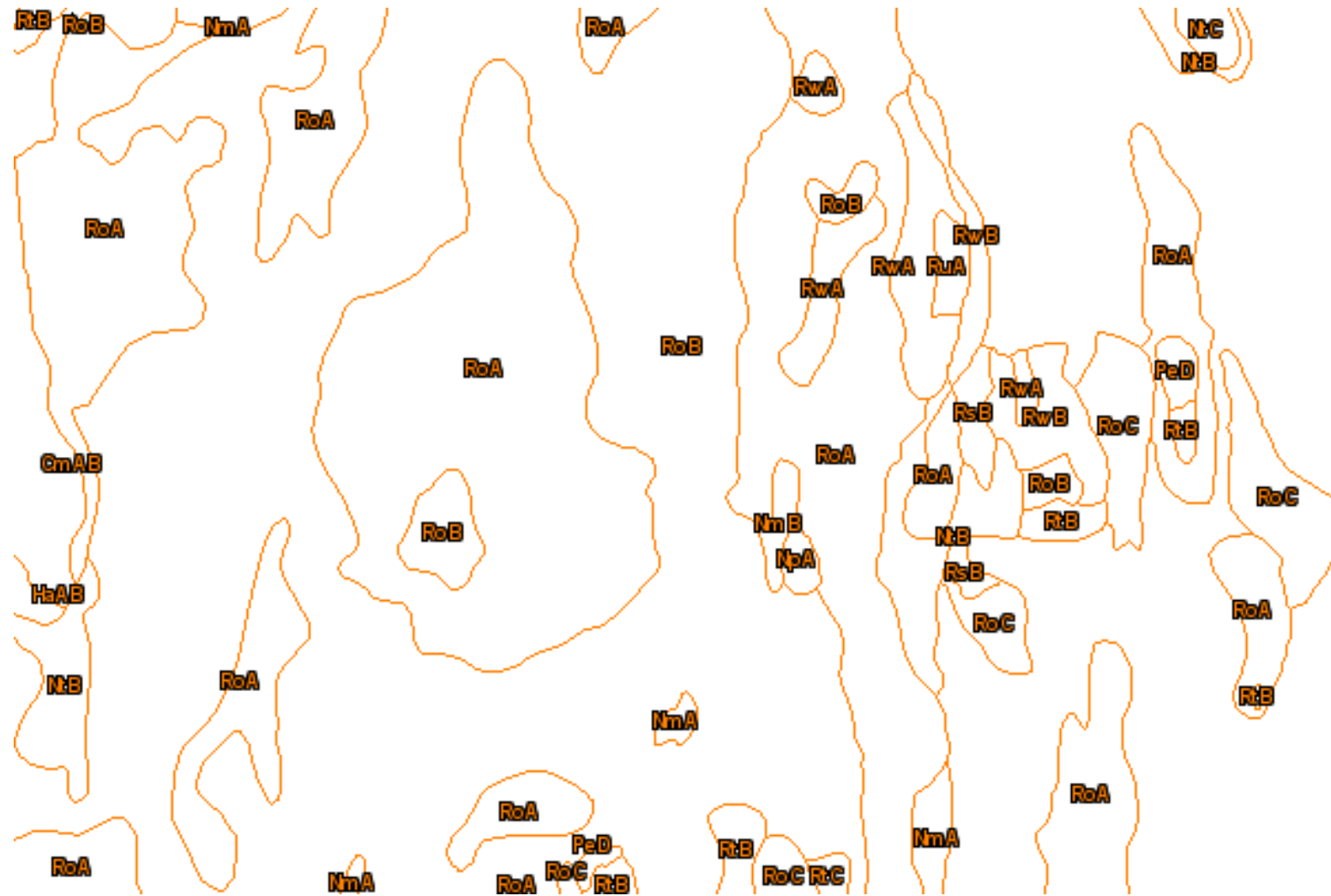```
?aoicoords=((-102.65253 38.18368,-102.65253 38.21139,-
102.59682 38.21139,-102.59682 38.18368,-102.65253
38.18368))
```
(demo)

?aoicoords=((-102.65253 38.18368,-102.65253 38.21139,-102.59682 38.21139,-102.59682 38.18368,-102.65253 38.18368))
Soil Map tab, All layers disabled except for soil map unit polygons.

# Default SLD, pretty-printed
## (use TestPost.html to fetch)

```xml
<StyledLayerDescriptor version="1.0.0"
    xmlns="http://www.opengis.net/sld"
    xmlns:gml="http://www.opengis.net/gml"
    xmlns:ogc="http://www.opengis.net/ogc"
    xmlns:xlink="http://www.w3.org/1999/xlink"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.opengis.net/sld
http://schemas.opengis.net/sld/1.0.0/StyledLayerDescriptor.xsd">
    <NamedLayer>
        <Name>mapunitpoly</Name>
        <UserStyle>
            <FeatureTypeStyle>
                <Rule>
                    <MinScaleDenominator>0.010000</MinScaleDenominator>
                    <MaxScaleDenominator>250000.000000</MaxScaleDenominator>
                    <PolygonSymbolizer>
                        <Stroke>
                            <CssParameter name="stroke">#ff8000</CssParameter>
                            <CssParameter name="stroke-width">0.75</CssParameter>
                        </Stroke>
                    </PolygonSymbolizer>
                    <TextSymbolizer>
                        <Label>musym</Label>
                        <Font>
            . . . remainder removed . . .
```

Change label from "musym" to "nationalmusym"

Paint only a few specific polygons(with mukey values 94860,94888,94861), color them magenta and label them with musym in big black letters

```xml
<StyledLayerDescriptor version="1.0.0"
    xmlns="http://www.opengis.net/sld"
    xmlns:gml="http://www.opengis.net/gml"
    xmlns:ogc="http://www.opengis.net/ogc"
    xmlns:xlink="http://www.w3.org/1999/xlink"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.opengis.net/sld http://schemas.opengis.net/sld/1.0.0/StyledLayerDescriptor.xsd">
    <NamedLayer>
        <Name>mapunitpolyextended</Name>
        <UserStyle>
            <FeatureTypeStyle>
                <Rule>
                    <Name>SpecificMukeys</Name>
                    <Filter>
                        <Or>
                            <PropertyIsEqualTo>
                                <PropertyName>mukey</PropertyName>
                                <Literal>94860</Literal>
                            </PropertyIsEqualTo>
                            <PropertyIsEqualTo>
                                <PropertyName>mukey</PropertyName>
                                <Literal>94888</Literal>
                            </PropertyIsEqualTo>
                            <PropertyIsEqualTo>
                                <PropertyName>mukey</PropertyName>
                                <Literal>94861</Literal>
                            </PropertyIsEqualTo>
                        </Or>
                    </Filter>
                    <MinScaleDenominator>0.010000</MinScaleDenominator>
                    <MaxScaleDenominator>250000.000000</MaxScaleDenominator>
                    <PolygonSymbolizer>
                        <Stroke>
                            <CssParameter name="stroke">#fff8000</CssParameter>
                            <CssParameter name="stroke-width">0.75</CssParameter>
                        </Stroke>
                        <Fill>
                            <CssParameter name="fill">#aaaaff</CssParameter>
                        </Fill>
                    </PolygonSymbolizer>
                    <TextSymbolizer>
                        <Label>musym</Label>
                        <Font>
                            <CssParameter name="font-family">arial</CssParameter>
                            <CssParameter name="font-weight">bold</CssParameter>
                            <CssParameter name="font-size">14.00</CssParameter>
                        </Font>
                        <LabelPlacement>
                            <PointPlacement>
                                <AnchorPoint>
                                    <AnchorPointX>0.5</AnchorPointX>
                                    <AnchorPointY>0.5</AnchorPointY>
                                </AnchorPoint>
                            </PointPlacement>
                        </LabelPlacement>
                        <Fill>
                            <CssParameter name="fill">#000000</CssParameter>
                        </Fill>
                    </TextSymbolizer>
                </Rule>
            </FeatureTypeStyle>
        </UserStyle>
    </NamedLayer>
</StyledLayerDescriptor></StyledLayerDescriptor>
```

Paint only a few specific polygons(with mukey values 94860,94888,94861), color them magenta and label them with musym in big black letters

RtB
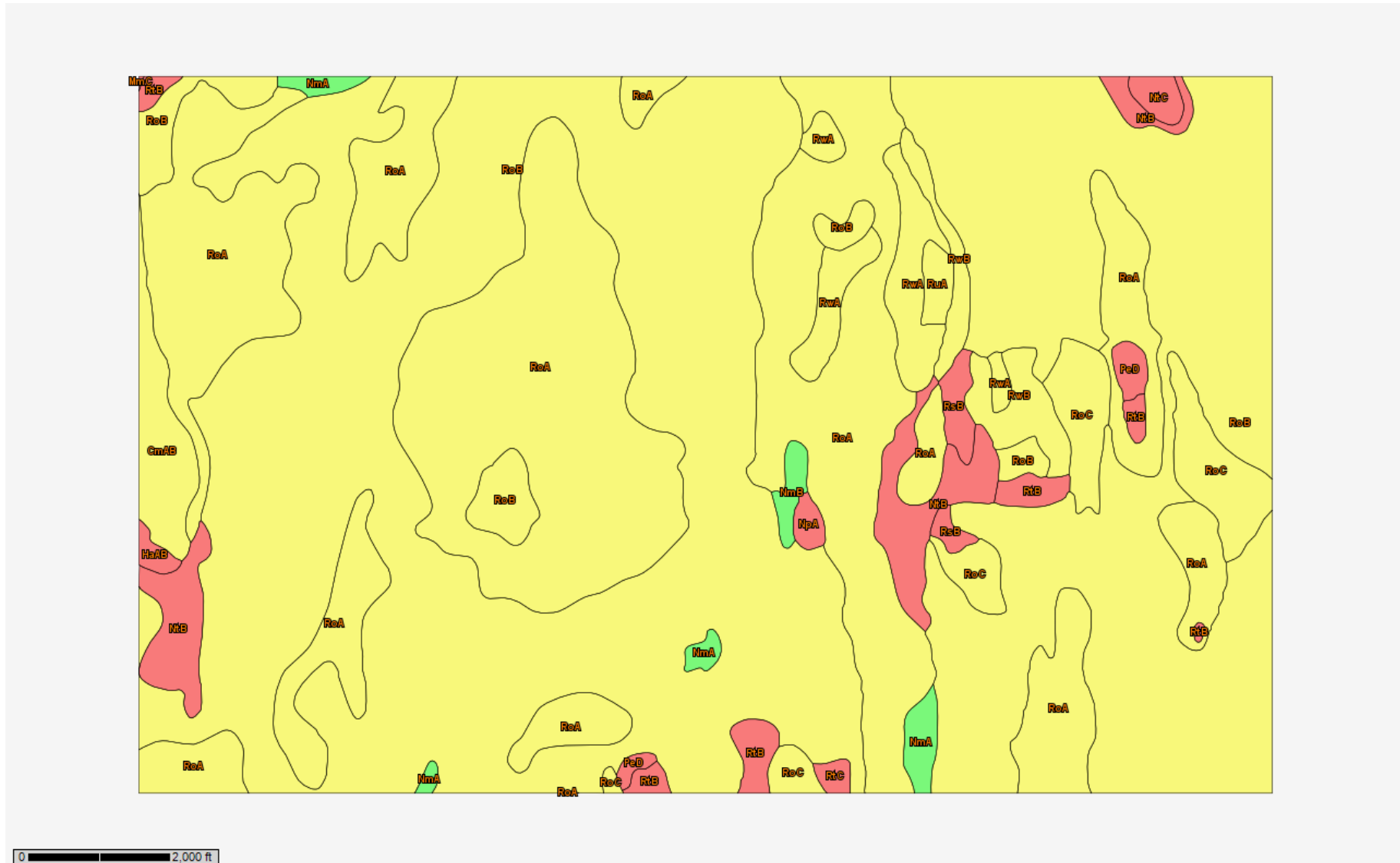
NtB

RtB

NtB RtB

RtB

NtB

RtB

RtB

RtB

7. WMS GetStyles and styled layer description

a. Custom map generation

b. Ad-hoc thematic map creation with WSS comparison

The SLD now allows a client application to create an ad-hoc "thematic map" provided that the requisite selection data are either in mapunitpolyextended or an mukey selection can be performed.
(demo)

Using mapunitpolyextended (a proxy for muaggatt), use engsldcp to color the polygons and determine outline color and width, use urbrecptwta (just for novelties' sake) to label only the "Somewhat limited" polygons (more novelty).

```xml
<StyledLayerDescriptor version="1.0.0"
    xmlns="http://www.opengis.net/sld"
    xmlns:gml="http://www.opengis.net/gml"
    xmlns:ogc="http://www.opengis.net/ogc"
    xmlns:xlink="http://www.w3.org/1999/xlink"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.opengis.net/sld http://schemas.opengis.net/sld/1.0.0/StyledLayerDescriptor.xsd">
  <NamedLayer>
    <Name>mapunitpolyextended</Name>
    <UserStyle>
      <FeatureTypeStyle>
        <Rule>
          <Name>VeryLimited</Name>
          <Filter>
            <PropertyIsEqualTo>
              <PropertyName>engsldcp</PropertyName>
              <Literal><![CDATA[Very limited]]></Literal>
            </PropertyIsEqualTo>
          </Filter>
          <MinScaleDenominator>0.010000</MinScaleDenominator>
          <MaxScaleDenominator>250000.000000</MaxScaleDenominator>
          <PolygonSymbolizer>
            <Stroke>
              <CssParameter name="stroke">#000000</CssParameter>
              <CssParameter name="stroke-width">1</CssParameter>
            </Stroke>
            <Fill>
              <CssParameter name="fill">#ff8888</CssParameter>
            </Fill>
          </PolygonSymbolizer>
        </Rule>

        <Rule>
          <Name>SomewhatLimited</Name>
          <Filter>
            <PropertyIsEqualTo>
              <PropertyName>engsldcp</PropertyName>
              <Literal><![CDATA[Somewhat limited]]></Literal>
            </PropertyIsEqualTo>
          </Filter>
          <MinScaleDenominator>0.010000</MinScaleDenominator>
          <MaxScaleDenominator>250000.000000</MaxScaleDenominator>
          <PolygonSymbolizer>
            <Stroke>
              <CssParameter name="stroke">#ff8000</CssParameter>
              <CssParameter name="stroke-width">0.75</CssParameter>
            </Stroke>
            <Fill>
              <CssParameter name="fill">#ffff88</CssParameter>
            </Fill>
          </PolygonSymbolizer>
          <TextSymbolizer>
            <Label>urbrecptwta</Label>
            <Font>
              <CssParameter name="font-family">arial</CssParameter>
              <CssParameter name="font-weight">bold</CssParameter>
              <CssParameter name="font-size">7.00</CssParameter>
            </Font>
            <LabelPlacement>
              <PointPlacement>
                <AnchorPoint>
                  <AnchorPointX>0.5</AnchorPointX>
                  <AnchorPointY>0.5</AnchorPointY>
                </AnchorPoint>
              </PointPlacement>
            </LabelPlacement>
            <Fill>
              <CssParameter name="fill">#ff8000</CssParameter>
            </Fill>
          </TextSymbolizer>
        </Rule>

        <Rule>
          <Name>NotLimited</Name>
          <Filter>
            <PropertyIsEqualTo>
              <PropertyName>engsldcp</PropertyName>
              <Literal><![CDATA[Not limited]]></Literal>
            </PropertyIsEqualTo>
          </Filter>
          <MinScaleDenominator>0.010000</MinScaleDenominator>
          <MaxScaleDenominator>250000.000000</MaxScaleDenominator>
          <PolygonSymbolizer>
            <Stroke>
              <CssParameter name="stroke">#000000</CssParameter>
              <CssParameter name="stroke-width">1</CssParameter>
            </Stroke>
            <Fill>
              <CssParameter name="fill">#88ff88</CssParameter>
            </Fill>
          </PolygonSymbolizer>
        </Rule>
      </FeatureTypeStyle>
    </UserStyle>
  </NamedLayer>
</StyledLayerDescriptor>
```

Using mapunitpolyextended (a proxy for muaggatt), use engsldcp to color the polygons and determine outline color and width, use urbrecptwta (just for novelties' sake) to label only the "Somewhat limited" polygons (more novelty).

8. Future possibilities

a. Rule evaluation and thematic maps via queries, WFS and WMS

b. Ad-hoc queries through WFS and WMS

c. SSA and mu-poly/point/line image pyramids for much faster WMS response (could also be used by WSS)

d. Integration of GGSURGO data into SDA

e. Branding of SDA versus WSS?