Developing Geoprocessing Service for Cropland Data Layer Thematic Map Creation

Weiguo Han, Liping Di
Center for Spatial Information Science and Systems
George Mason University
4400 University Drive, MS 6E1, Fairfax, VA 22030, USA
whan@gmu.edu, ldi@gmu.edu

Zhengwei Yang
US Department of Agriculture
National Agricultural Statistics Service
3251 Old Lee Highway, Fairfax, VA 22030, USA
zhengwei.yang@nass.usda.gov

***Abstract*—NASS Cropland Data Layer (CDL) data have been widely used in many business, research and academic areas. This valuable geospatial data has been accessible and disseminated through CropScape - a web-service based online geospatial application. Web geoprocessing services had been required and developed in CropScape to visualize, navigate, analyze, and download the CDL data in an open standard. Especially, an online thematic map generating function had been required by many users to produce, print or share CDL maps of an area of interest without using any mapping software. This paper is concerned with how to develop a new Web geoprocessing service, which uses API functions of Apache Axis2, Geospatial Data Abstraction Library and Cairo graphics library to generate the thematic map of CDL in PDF format with the preferred settings. The resulting geoprocessing service enabled CropScape to create the CDL thematic map interactively, significantly reduced operational resource and cost, and greatly improved efficiency and CDL data presentation.**

Keywords-Cropland Data Layer; CropScape; Thematic Map; Web Geoprocessing Service; Cairo

I. INTRODUCTION

National Agricultural Statistical Service (NASS) of US Department of Agriculture (USDA) has produced the Cropland Data Layer (CDL) data annually using mid-resolution satellite data and high quality ground truth data since 1997. The annual CDL data provide specific crop and non-crop land cover classifications over the Contiguous United States (CONUS) in geo-referenced raster files [1].

The CDL data (at the state level only) were previously distributed in traditional channels, such as paper maps, CD/DVD copies, or FTP downloading. For those users with limited or no GIS skills, it is not easy to retrieve directly the data with an area of interest (AOI), such as a county or Agricultural Statistical District (ASD) from the original data files, and to analyze the requested data to produce statistics report and graph using the desktop GIS software like ArcGIS. Obviously, a new approach is necessary to address the needs of visualizing, retrieving, and analyzing CDL data in an open and standard manner. To address this issue, CropScape - an online geospatial application (http://nassgeodata.gmu.edu/CropScape) has been built using the latest Web 2.0 technologies along with the Web service standards of World Wide Web Consortium (W3C) and Open Geospatial Consortium (OGC). CropScape was designed to navigate, customize, visualize, analyze and disseminate CDL data interactively through a common Web browser (e.g. Internet Explorer, Firefox, and Chrome) on any Internet connected computer [2].

The CDL data of 1997-2011, derived images as well as statistical tables can be retrieved from the Web user interface or CropScape's integrated geoprocessing services, or the deployed standard geospatial data services such as OGC Web Coverage Service (WCS) and Web Map Service (WMS). CropScape has greatly reduced NASS's operational expenses and resource needs on creating, printing, and shipping the traditional media for CDL data, and improved efficiency and effectiveness of the cropland related agricultural decision supports. CropScape has been used extensively in many research and applications by government agencies, non-profit organizations, educational institutions, and business firms around the world since its official release in January 2011 [3-6].

Besides the existing functions in CropScape, the needs for function for creating CDL AOI thematic map on-the-fly was ever-increasing from agricultural community and other geospatial communities. It was highly desired that the thematic map should be created in PDF format since this format allowed users easily to display, print, store, and share the thematic maps in all computer systems. CropScape was built in a Service Oriented Architecture (SOA). It is easy to address these new functional requirements by assimilating new processing services in the extensible framework of CropScape [2, 7]. In this paper, how to implement a new map generating Web geoprocessing service to create the online CDL thematic map within any AOI is presented. This Web geoprocessing service used the API functions of Apache Axis2, Geospatial Data Abstraction Library (GDAL), and Cairo graphic library. The geoprocessing service was integrated in CropScape Web portal to provide interactive and on-demand thematic map generation capability. It was also exposed to public along with other existing Web geoprocessing services which were deployed in Apache Axis2 for invocation by other applications.

The remainder this paper is organized as the following: the design and implementation of this Web processing service in CropScape are described in Section 2. The examples of service invocation in CropScape and other applications are given in Section 3. Finally, the future improvements are discussed and the conclusions are summarized in Section 4.

This project is supported by US Department of Agriculture, National Agricultural Statistics Service (Grant No.: 58-3AEU-0-0067, PI: Dr. Liping Di).

Authorized licensed use limited to: University of North Texas. Downloaded on December 18,2020 at 12:54:47 UTC from IEEE Xplore. Restrictions apply.

II. GEOPROCESSING SERVICE DESIGN AND IMPLMENTATION

Users can retrieve the CDL images via CropScape by sending standard WMS *GetMap* requests with the specified output format (e.g. PNG, JPEG, or PDF) and a given AOI to the CDL WMS Server or HTTP GET/POST or Simple Object Access Protocol (SOAP) request to the *GetCDLImage* operation of existing geoprocessing service named CDLService (http://nassgeodata.gmu.edu:8080/axis2/services/CDLService?wsdl) as the service requests listed in Table 1. However, the CDL thematic map with title, north, scale, legend, and logos were not available from these rendering services. This section will describe the design and implementation of on-demand thematic map generating geoprocessing service.

TABLE I. WEB SERVCE REQUEST EXAMPLES

Service Name	Operation	Request Example
WMS	*GetMap*	http://gis.csiss.gmu.edu/cgi-bin/wms_cdlall?service=wms&version=1.1.1&request=getmap&layers=cdl_2010new&srs=epsg:4326&bbox=-96,40,-90,43&width=800&height=400&format=image/png
CDLService	*GetCDLImage*	http://nassgeodata.gmu.edu:8080/axis2/services/CDLService/GetCDLImage?files=http://nassgeodata.gmu.edu/nass_data_cache/byfips/CDL_2010_19169.tif&format=png

2.1 *Materials and Methods*

The Web geoprocessing services enable users to leverage distributed geospatial data and computing resources over the network in their own applications and to automate geospatial data integration and analysis [8]. The CDLService in CropScape is deployed in the Apache Axis2 Web service runtime environment to support CDL data customization, visualization, and statistical and change analyses. The Axis2 plugins and Web Services Tools (WST) of the Eclipse Java Integrated Development Environment (IDE) was utilized to design, develop, deploy, and test Web services. Both the Top Down and the Bottom Up approaches are supported in Eclipse Java framework. In CropScape, the Top Down method is adopted. First, the Web service description (i.e. Web Service Description Language, WSDL) is created. Second, the Web service skeleton code is generated from WSDL. Then, the geoprocessing service code is developed and tested. Finally, the service is built and exposed. The general architecture of CDLService is shown in Fig. 1.

GDAL is an open source geospatial library that provides a set of API functions to read and write raster or vector files [9]. The common geospatial data formats such as GeoTIFF, JPEG, Arc/Info ASCII Grid, ERDAS Imagine, ESRI Shapefile, and GML can be translated and processed using command line utilities of GDAL or applications derived from GDAL. GDAL has been extensively used in open source and commercial GIS and remote sensing software. GDAL C API functions are called in the executable programs to generate images and statistical information from CDL data within a given AOI. These executable programs with multiple parameters will be invoked in the geoprocessing service.

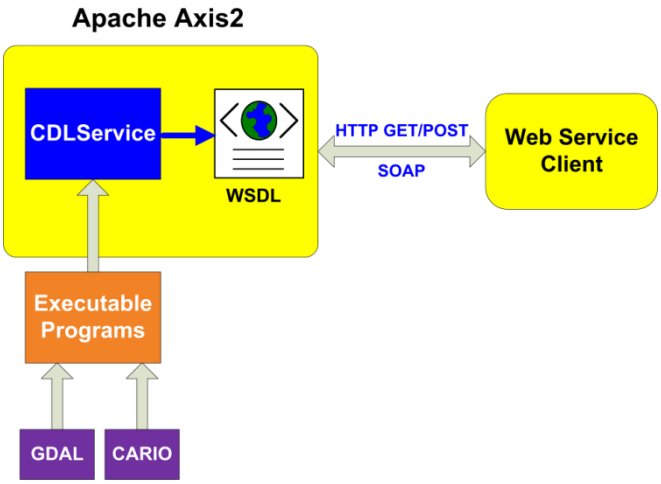


Figure 1. CDLService Architecture

Cairo is an open source 2D vector graphic library written in C language. It offers powerful drawing APIs, such as Graphics Device Interface (GDI) APIs on windows, to draw geometries, images, and texts with various styles and to perform affine transformations [10]. The output formats include PNG, PDF, Postscript, SVG, as well as contexts of Win32 GDI, OpenGL, etc. Cairo also has bindings available for other programming languages, including C++, C#, Java, Perl and Python. To implement the Web geoprocessing service for CDL map generating, Cairo C API functions are utilized in the executable program. This service creates the CDL thematic map from images and texts, and exports the drawing context in PDF format.

2.2 *Design and Implementation*

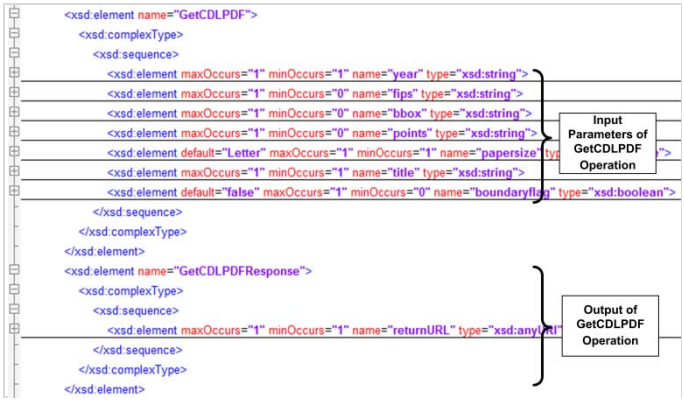


Figure 2. WSDL of GetCDLPDF Operation

As shown in Fig. 2, input parameters (crop year, FIPS code/bounding box/multiple points, paper size, title, and county boundary flag for CDL at the state level) and the output (URL of the PDF file) of the *GetCDLPDF* operation are added in the existing WSDL file of the CDLSevice. The skeleton code of this operation are generated using the updated WSDL in Eclipse IDE, and the executable programs derived from C APIs of GDAL and Cairo are to be invoked using Java *Process* and *Runtime* class in this operation to avoid redundant

Authorized licensed use limited to: University of North Texas. Downloaded on December 18,2020 at 12:54:47 UTC from IEEE Xplore. Restrictions apply.

programming and memory limit of Java Virtual Machine (JVM).

The whole process of creating the on-demand thematic map in PDF format is streamlined and automated as illustrated in Fig. 3.

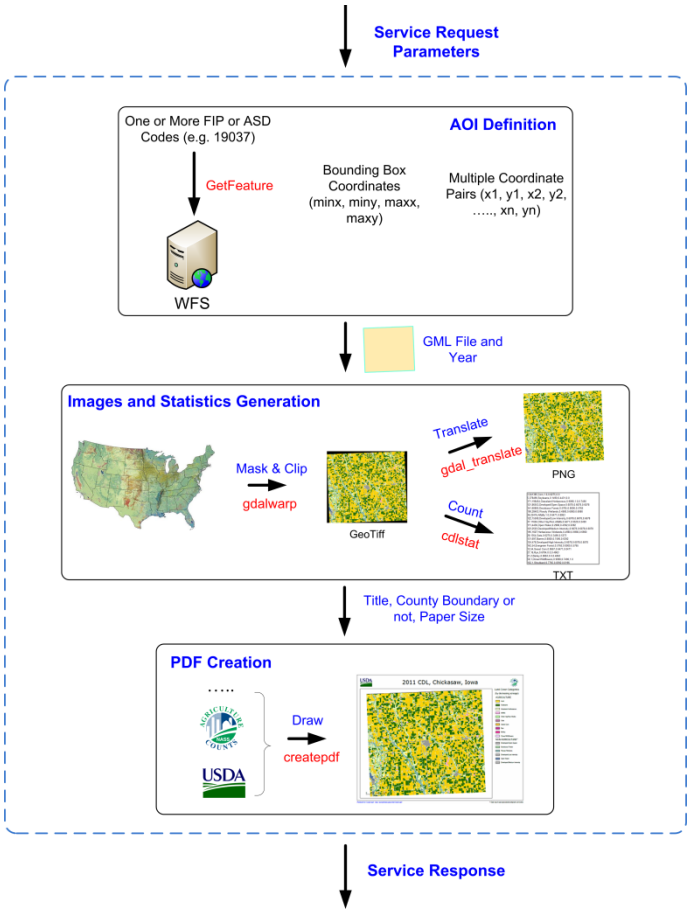


Figure 3. Data Process in GetCDLPDF Operation

The parameter which contains the AOI information is first processed. The *GetCDLPDF* operation supports three types of AOI definition: (1) by specified Federal Information Processing Standards (FIPS) code(s) of state or county or ASD code(s); (2) by bounding box; and (3) by polygon. For option 1, a GML file for the AOI boundary will be kept using filename of the code through sending *GetFeature* request with property filter to the WFS server and obtaining the boundary information from the response. If this GML file exists in the cache directory, this step will be skipped. For option 2 and 3, the coordinates retrieved from the requests will be stored as the geometry feature in a GML file with unique name.

Next, The GML files obtained will be used as a mask to subset and rasterize the CDL file within AOI for the given year. GDAL's *gdalwarp* utility is invoked to generate the required CDL file in GeoTIFF format. Then the image in PNG format is generated from this CDL file using GDAL's *gdal_translate* utility, and the agricultural and non-agricultural category information within AOI is counted pixel-by-pixel, sorted by decreasing acreage, and stored in a text file using the *cdlstat* program. *cdlstat* is built from GDAL APIs of opening/closing raster file, fetching raster band and reading raster data. Similarly, if these files with filename of FIPS or ASD code exist, the corresponding operation will be skipped.

Finally, the map elements, such as the CDL image, the sorted category information (for drawing legend), the title, the pre-existing USDA and NASS logos, along with scale bar will be drawn on the thematic map surface with the program of *createpdf* which utilizes Cairo. A set of map templates for various paper sizes (Letter, A4, A3, A2, A1, and A0) are designed with above mentioned elements in their appropriate positions (as shown in Fig. 4). The detailed rendering process is described as follows:

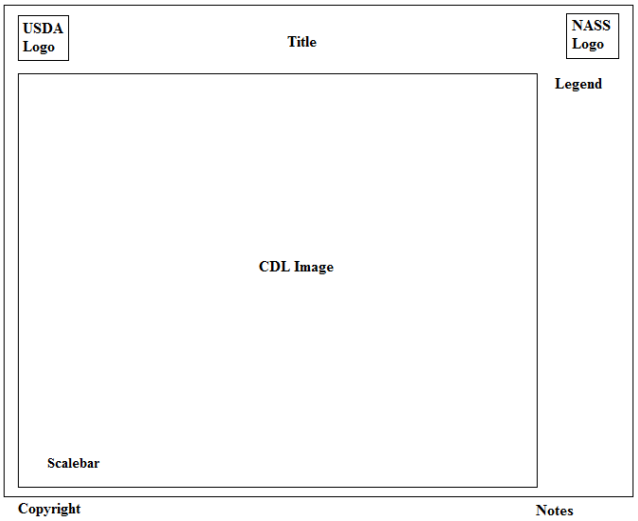


Figure 4. Thematic Map Template

Step 1: A PDF surface with white background is created as the drawing context to contain all above map elements. The context size depends on the input paper size. The sample code is listed as:

```
surface = cairo_pdf_surface_create(pdffilename, sizex, sizey);
cr = cairo_create(surface);
cairo_set_source_rgb(cr, 0, 0, 0);
```

Step 2: A valid rectangle drawing area is defined and rendered using the following code:

```
cairo_set_line_width(cr, 0.5);
cairo_rectangle(cr, 0.7*marginx, 2.1*marginy, sizex-5*marginx, sizey-2.9*marginy);
cairo_stroke(cr);
```

Step 3: All images, including logos, CDL image and county boundary image, are scaled to the appropriate size and drawn on the image surface as follows:

```
usdasurface = cairo_image_surface_create_from_png(usdafilename);
usdaw = cairo_image_surface_get_width(usdasurface);
usdah = cairo_image_surface_get_height(usdasurface);
cairo_save(cr);
ratio = usdaw/printareaw<usdah/printareah?usdaw/printareaw:usdah/printareah;
cairo_scale(cr, ratio, ratio);
cairo_set_source_surface(cr, usdasurface, 0.7*marginx/ratio, 0.5*marginy/ratio);
cairo_paint(cr);
cairo_restore(cr);
```

Authorized licensed use limited to: University of North Texas. Downloaded on December 18,2020 at 12:54:47 UTC from IEEE Xplore. Restrictions apply.

Step 4: The texts (like title, crop category name) and geometries (e.g. dashed line for scale bar, color rectangle for crop category) are added with the given font and style:

```
cairo_select_font_face(cr, "Sans", CAIRO_FONT_SLANT_NORMAL, CAIRO_FONT_WEIGHT_NORMAL);
cairo_set_font_size(cr, 25.0);
cairo_text_extents(cr, title, &extents);
titlex = sizex/2.0 - (extents.width/2.0 + extents.x_bearing);
titley = 1.3*marginy - (extents.height/2.0 + extents.y_bearing);
cairo_move_to(cr, titlex, titley);
cairo_show_text(cr, title);
```

Step 5: The drawing context is shown and saved in a PDF file and all surfaces are destroyed:

```
cairo_show_page(cr);
cairo_surface_destroy(surface);
cairo_surface_destroy(imagesurface);
```

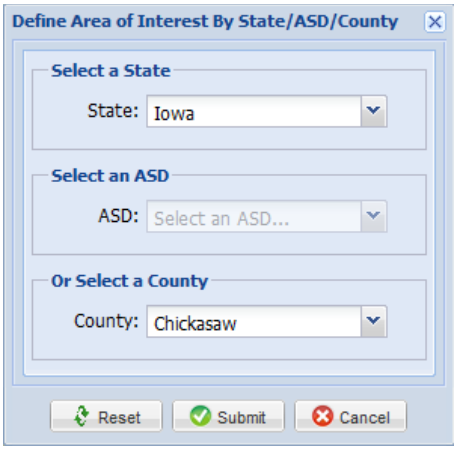
The link of this PDF file will finally be returned in response of the *GetCDLPDF* operation.

III. EXAMPLES OF SERVICE INVOCATION

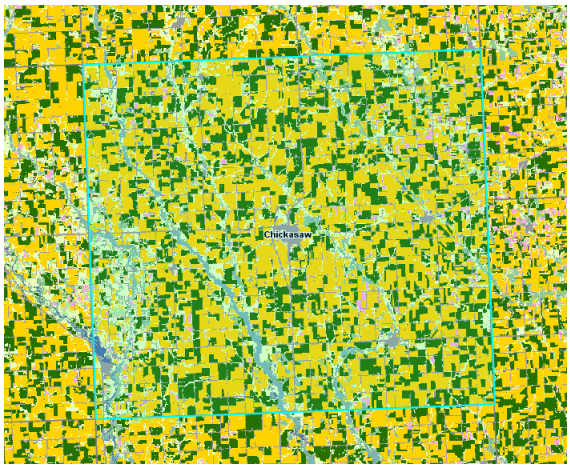
The geoprocessing service of creating on-demand CDL thematic map has been integrated in the user interface of CropScape for public use. Users can also consume this service in other applications using standard HTTP GET, HTTP POST, or SOAP request.

3.1 CropScape Web Portal

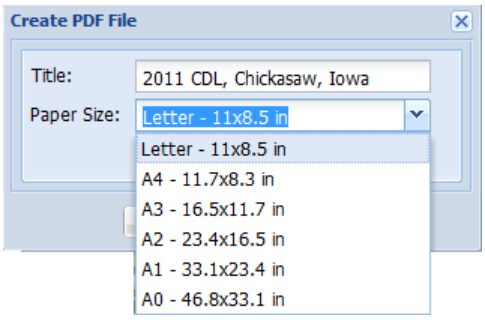
To demonstrate how to invoke the proposed new map generating service, an AOI such as administrative district, rectangle, circle, or polygon is first defined in CropScape. The state/ASD/county lists of CONUS are preloaded in the browser client for selection (as shown in Fig. 5-1). The returned GML file of the selected administrative district AOI will be drawn and highlighted in the map view (as shown in Fig. 5-2) in CropScape. The rectangle, circle, and polygon are to be dragged and drawn directly in the highlight color. The title and paper size of the PDF file can be specified in a popup window (as shown in Fig. 5-3). These parameter values will be sent to CDLService in a SOAP request. When the browser client receives the response from the service, the returned PDF file will be loaded in a window for display and downloading (as shown in Fig. 5-4).



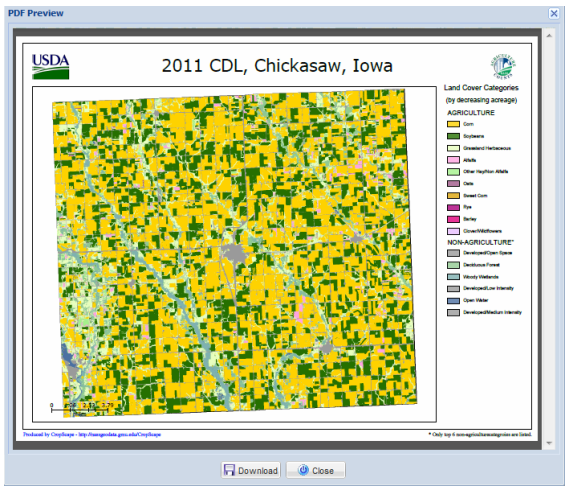
(1) Select Dsitrcit



(2) Display AOI



(3) Specify PDF Parameters



(4) Display and Download PDF

Figure 5. Create CDL Thematic Map in PDF Format in CropScape

3.2 GET Request

Users can use a standard HTTP *GET* request to obtain the CDL thematic map directly. The following GET/KVP (key-value pair) encoding requests give examples of using *Get* request to produce the CDL thematic map in PDF format within the district(s) of interest (use %20 for spaces in the title):

1) One FIPS or ASD code

http://nassgeodata.gmu.edu:8080/axis2/services/CDLService/GetCDLPDF?ye

Authorized licensed use limited to: University of North Texas. Downloaded on December 18,2020 at 12:54:47 UTC from IEEE Xplore. Restrictions apply.

ar=2011&fips=19&papersize=Letter&title=2011%20Iowa%20CDL&boundaryflag=true

2) Two or more FIPS or ASD codes (separated by ",")

http://nassgeodata.gmu.edu:8080/axis2/services/CDLService/GetCDLPDF?year=2011&fips=19003,19029&papersize=A4&title=2011%20CDL

3.3 POST Request

HTTP *POST request* can be sent from user's HTML Forms to CDLService to get and display the PDF file. The following example gives a *POST* example for requesting the PDF file by crop year, polygon, paper size and title.

Host:
http://nassgeodata.gmu.edu:8080/axis2/services/CDLService/GetCDLPDF
Content:
year=2011&points=175207,2219600,175207,2235525,213693,2235525,213693,2219600&papersize=Letter&title=2011%20CDL
Content Type:
application/x-www-form-urlencoded

3.4 SOAP Request

A CDL thematic map can also be obtained using a SOAP request. The code as shown in Fig. 6 gives an example of SOAP request and response of obtaining a 2011 CDL thematic map of Chickasaw County, Iowa (FIPS code: 19037) in a Letter size PDF file.

```
<?xml version='1.0' encoding='utf-8'?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <ns1:GetCDLPDF xmlns:ns1="http://cropscape.csiss.gmu.edu/CDLService/">
      <year>2011</year>
      <fips>19037</fips>
      <papersize>Letter</papersize>
      <title>2011 Chickasaw County, Iowa</title>
      <boundaryflag>true</boundaryflag>
    </ns1:GetCDLPDF>
  </soapenv:Body>
</soapenv:Envelope>
```

(1) SOAP Request

```
<?xml version='1.0' encoding='utf-8'?>
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <ns1:GetCDLPDFResponse xmlns:ns1="http://cropscape.csiss.gmu.edu/CDLService/">
      <returnURL>
        http://nassgeodata.gmu.edu/nass_data_cache/CDL_2011_19037_1936045126.pdf
      </returnURL>
    </ns1:GetCDLPDFResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

(2) SOAP Response

Figure 6. SOAP Example

IV. DISCUSSIONS AND CONCLUSIONS

This paper presented the design and implementation of a new geoprocessing service of creating on-demand CDL thematic map in details. This service extends the reach of CDL data and promotes sharing and utilization of geospatial cropland information in the agricultural related decision making. This new geoprocessing service has been proved to be very useful in NASS operation and utilized by many users in agricultural business. It significantly reduced operational resource and cost, and greatly improved efficiency and CDL data presentation. However, it was found that Cario library only supports images with less than 300 dpi resolutions, and doesn't provide API functions to read and render vector file (e.g. ESRI Shapefile) in the graphic context. In this application, the pre-generated boundary images are utilized to overlay with other images to overcome the issue. Moreover, the pre-defined templates provide a layout for automating the CDL thematic map generating while a WYSISYG (What-You-See-Is-What-You-Get) user interface will provide user more flexibility in creating a high-quality map with his/her own preferences. The issues to provide capabilities of supporting high-resolution image over 300 dpi and rendering vector file directly, as well as to develop a WYSIWYG user interface need to be further investigated in the future.

ACKNOWLEDGMENT

The authors would like to thank Ms. Lee Ebinger of USDA NASS for her valuable comments and suggestions in the map template design.

REFERENCES

[1] C. Boryan, Z. Yang, R. Mueller, and M. Craig, "Monitoring US agriculture: the US Department of Agriculture, National Agricultural Statistics Service, Cropland Data Layer Program", Geocarto International, Vol. 26, No. 5, pp. 341-358, 2011.

[2] W. Han, Z. Yang, L. Di, and R. Mueller, "CropScape: A Web service based application for exploring and disseminating US conterminous geospatial cropland data products for decision support", Computers and Electronics in Agriculture, Vol. 84, pp. 111–123, 2012.

[3] J. B. Belden, B. R. Hanson, S. T. McMurry, L. M. Smith, and D. A. Haukos, "Assessment of the effects of farming and conservation programs on pesticide deposition in high plains wetlands", Environmental Science & Technology, Vol. 46, No. 6, pp. 3424-3432, 2012.

[4] F. J. Kutz, J. M. Morgan, J. Monn, and C. P. Petrey, "Geospatial approaches to characterizing agriculture in the Chincoteague Bay Subbasin", Environmental Monitoring and Assessment, Vol. 184, No. 2, pp. 679-692, 2012.

[5] V. Chandola, R. R.Vatsavai, "A scalable gaussian process analysis algorithm for biomass monitoring", Statistical Analysis and Data Mining Vol. 4, No. 4, pp. 430-445, 2011.

[6] C. Wang, F. B. Fritschi, G. Stacey, and Z. Yang, "Phenology-Based Assessment of Perennial Energy Crops in North American Tallgrass Prairie", Annals of the Association of American Geographers, Vol. 10, No. 4, pp. 742-751, 2011.

[7] C. Granell, D. Díaz, and M. Gould, "Service-oriented applications for environmental models: reusable geospatial services", Environmental Modelling and Software, Vol. 25, No. 2, pp. 182-198, 2010.

[8] P. Zhao, T. Foerster, and P. Yue, "The geoprocessing web", Computers & Geosciences, http://dx.doi.org/10.1016/j.cageo.2012.04.021, in press, 2012.

[9] Open Source Geospatial Foundation, "GDAL - Geospatial Data Abstraction Library: Version 1.8.0", http://www.gdal.org, 2012.

[10] Cairo, "Cairo Tutorial", http://www.cairographics.org/, 2012.

Authorized licensed use limited to: University of North Texas. Downloaded on December 18,2020 at 12:54:47 UTC from IEEE Xplore. Restrictions apply.