ArcSDE and ArcIMS Implementation: A Success Story

by
Neeraj Singh and Sharhonda Knight

Abstract

ArcSDE and ArcIMS 4.0 have been used to create an Intranet website called WebGIS for the Polk County Property Appraiser's Office (PAO). Seamless SDE Layers were created for parcels, roads, subdivisions and other important coverages relevant to property appraisal. These layers were then added to WebGIS. WebGIS was programmed to give users the ability to search for parcel records based on owner name, address, subdivision and other criteria. WebGIS also gives user the ability to dynamically classify the parcels based on tabular information. Scripting for WebGIS was done using ArcXML, Javascript, HTML and ASP. This paper examines the trails and tribulations in implementing WebGIS.

Background

Constitutionally, the Property Appraiser is charged with the duty and responsibility of placing a fair, equitable and JUST value on all real and tangible personal property. Real property comprises of land and all buildings as well as structures, and improvements to the land. Tangible personal property comprises of machinery and equipment, furniture, fixtures,and other items owned and used for business purposes. The appraisers at the Property appraiser office ensure the equality of valuations based on a system of property appraisal using modern, sound, and nationally accepted appraisal practices and principles. Modern techniques applied to property appraisal utilize Geographical Information Systems (GIS). A GIS showing the parcels linked with the CAMA (Computer Assisted Mass Appraisal) is a vital tool in property assessment.

GIS has been a integral part of PAO. New parcels are mapped and edited using COGO tools in Arc Info. The appraisers need to search and investigate parcels for appraisal purposes. This was made possible customizing software such as ArcView and linking parcel and CAMA information. However, the parcels were created and maintained using a tiling method based on section, township and range i.e the Public Land Survey System (PLSS). Thus appraisers could only examine parcels by sections and it was not seamless. This was a tedious method of investigating parcels. The Property Appraisers Office also needed to keep abreast of modern technologies and changes in the GIS Field. SDE (Spatial Database Engine) was a new ESRI (Environmental Systems Research Institute) technology that enabled coverages and shapefiles to be stored in a RDBMS (Relational Database Management System) thus bringing the two technologies GIS and DBMS closer. Hence the GIS Department decided to move towards this new technology.

Data creation is incomplete without data dissemination. Earlier, the internal applications using GIS and CAMA were created using ArcView. However, ArcView required the purchase and maintainence of licenses. Creating an application in ArcIMS (Arc Internet Map Server) helped solve the dilemma of data dissemination and financial savings. We were able to serve more users by creating an ArcIMS application called WebGIS. Also, it served as a single programming source. To update an application using ArcView required updates on every computer. However, with ArcIMS, we need to make changes only on the server.

Migration to SDE

Project Goals

This paper outlines the implementation of ArcSDE at the PAO. The primary goal of the ArcSDE project was the successful migration of existing PCPA geographic data into a geodatabase (spatial database) that utilizes ArcSDE as the gateway between the GIS and the RDBMS.

This goal had to be accomplished without hindering existing maintenance and conversion processes. It was important to relay to managers and users that initial conversion of GIS data into the geodatabase structure would not eliminate the need for the existing 'flat-file' data structure. The geodatabase is updated with information from the existing data structure and this will continue until maintenance and conversion routines have been developed utilizing the geodatabase schema. Complete migration from the existing data structure to full use of the GIS database is an on-going process that is closer to realization with the development of topology in ArcGIS 8.3. The next phase of our conversion will focus on taking advantage of true geodatabase functionality including the ability to model spatial features and their behavior with subtypes, domains, topology rules and relationships.

Existing Conditions

It is difficult to find your direction on a roadmap if you don't know where you are…with this in mind our team set out to discover exactly where we were. The discovery focused on five (in-house) areas in particular:

- Hardware
- Software
- Data / Work Flow
- Exisiting AMLs and Customized ArcView Programs
- Organizational Structure

In addition to these in-house issues, the PAO had to take a look at ESRI's stand on parcel based mapping. During the initial phases of our conversion to the geodatabase structure, ESRI had not yet resolved how to support polygon topology and a final or complete geodatabase model for parcel mapping was not available. As a result, we decided to move to the geodatabase to get a "feel" of the new environment and to basically familiarize ourselves with the new methodologies, so that we would be comfortable with the new software, able to maintain our investment, and prepared to move forward as the GIS industry evolved.

Data Migration

ArcSDE Migration Options

As we prepared to migrate to ArcSDE, two ArcSDE migration options presented themselves. Both options would allow the office to continue to maintain existing data creation and editing routines. This was extremely important as our team had decided that programming modifications would be tackled in a totally separate phase. Option 1 would utilize ArcSDE as a central data repository, while Option 2 would all the PAO to eliminate the existing coverage environment.

ArcSDE Migration Option 1: This option would allow the PAO to utilize ArcSDE as a central data repository while continuing to edit and maintain the existing set of coverages. With this option, our office was able to implement the daily update of the ArcSDE geodatabase and distribute the read-only geodatabase layers to multiple users via ArcIMS.

ArcSDE Migration Option 2: This option would require that existing coverages used by the PAO be replaced with data stored in ArcSDE. This would entail modification to exisiting AMLS to allow ArcInfo Workstation to interact with ArcSDE layers. Analyst would continue to create new coverages and import them into the geodatabase. Editing of existing coverages would involve querying the ArcSDE data and exporting the data to ArcInfo. Upon completion of editing the data would then be imported back into the geodatabase.

After significant discussion and research, our office decided to implement option 1. This option allowed us to immediately begin to use the ArcSDE geodatabase environment and eliminated modification of existing AMLs. In addition, we were able to immediately take advantage of the seamless parcel layer environment offered by the geodatabase structure. In addition, option 1 would allow us to get our feet wet and become more familiar with the Oracle RDBMS before jumping in head first. Finally, option 1 seemed less of an immediate shift in our GIS paradigm.

Oracle 8.1.7 Configuration

Prior to migrating to ArcSDE, the office had an Oracle 8.1.5 database with stand-alone GIS tables that were extracts of our larger CAMA database. We decided to implement a second Oracle 8.1.7 home so that the Oracle 8.1.5 database could continue uninterrupted, this decision later caused multiple issues that had to be resolved. The installation completed successfully, but nothing actually worked after the successful installation. We finally narrowed this issue down to problems with file permissions. The Oracle Installation guide gave specific instructions to set permission for most folders to (644).
We eventually had to modify some of these folders to (755) so that various applications would run properly. In addition, the Oracle 8.1.5 listener.ora file had to be modified to accommodate the new Oracle 8.1.7 database. Once these changes had been made the database was up and running.

Configuring Oracle for ArcSDE

The steps to configure Oracle for ArcSDE were straight-forward and easy to accomplish, a very general overview of the steps are listed below:

- Full backup of existing database files (control, system, tools, init<db>.ora)
- Placed database in archivelog mode and created a destination home for archived data.
- Set compatible parameter to 8.1.0 (Notes indicated that SDE needed this to run properly)
- Created 4 control file locations on different disks.
- Moved database files to minimize disk contention.
- Reconfigured the SGA
- Created mirrors for online redo log members
- Created SDE tablespace and user
- Created tablespaces for GIS data and indexes

After all configuration steps had been completed, we found that SDE is built as a 32-bit application on 32-bit OS. As a result, the 64-bit Oracle which we installed would not work with SDE unless we used Direct Connect options only. We decided to uninstall the 64-bit version and re-install the 32-bit version for Oracle / SDE compatibility. Once this was completed the new database had to be configured to support ArcSDE.

ArcSDE Installation

Prior to installing ArcSDE the SDE user, account, and profile were created. The installation process went very quickly. Post installation included granting execute on dbms_pipe and dbms_lock to the public role and running the sdesetupora8i to create ArcSDE metadata tables and stored procedures. Once the new ESRI keycode with the correct licensing information had been loaded and the ESRI FlexLM License Manager started we used the sdemon –o start command to start the ArcSDE services.

Loading Data

Importing 815 tables into 817 database: We used standard export / import Oracle utilities to complete the process of loading the 8.1.5 Oracle data into the 8.1.7 GIS tablespace. Once the data was loaded the environment variable for both oracle homes had to be added to access SQL*Plus from the Windows environment.

Basemap Data: Initially our goal was to simply get the basemap data into the ArcSDE database. We did not take advantage of feature datasets, domains and/or subtypes, and topology rules did not exist at that time. Each dataset outlined below was loaded as an individual feature class. The steps taken to load basemap data into the ArcSDE geodatabase are outlined below:

I. Database Design / Naming Conventions (20 separate layers)

a. Parcel Coverage

i. basemap.parcel_poly (parcel polygons)
ii. basemap.parcel_dim (parcel dimensions: $level 20)
iii. basemap.parcel_num (parcel numbers: $level 16)

b. Lot Coverage

i. basemap.lot_line (lot lines: dxf-layer 17)
ii. basemap.lot_dim (lot dimensions: $level 15)
iii. basemap.lot_num (lot numbers: $level 18, 29, 38)

c. City Coverage

i. basemap.city_line (city lines: dxf-layer 3)
ii. basemap.city_anno (city annotation: $level 4)

d. Hydro Coverage

i. basemap.hydro_poly (hydro polygons)
ii. basemap.hydro_line (hydro lines: dxf-layer 5, 50, 51)
iii. basemap.hydro_anno (hydro annotation: $level 6)

e. Subdiv Coverage

i. basemap.subdiv_poly (subdiv polygons)
ii. basemap.subdiv_anno (subdiv annotation: $level 30, 32)

f. Problem Coverage

i. basemap.prob_line (problem lines: dxf-layer 60)
ii. basemap.prob_anno (problem annotation: $level 60)

g. Geodetic Coverage

i. basemap.geodetic_line (geodetic lines: dxf-layer 12, 13, 14)
ii. basemap.geodetic_anno (geodetic annotation: $level 12, 13)

h. Misc Coverage

i. basemap.misc_line (misc lines: dxf-layer 21, 23, 27, 28, 40, 35, 57)
ii. basemap.misc_rdname (misc road names: $level 2, 22)
iii. basemap.misc_anno (misc annotation: $level 24, 26, 36, 37, 56)

II. Template coverages were created and then used to write AMLs to extract only those elements needed for view/query in a read-only ArcSDE data repository. The templates were used to create empty feature classes for our actual data to be loaded into. Once the templates coverages comprised all necessary elements and had been loaded into the geodatabase, the sdetable –o truncate command was used to prepare the empty feature classes for initial data load.

III. ArcSDE Update AML Scripts were written to initially load all data (~700 sections with 7 – 10 coverages in each) and maintain the ArcSDE database daily.

Oracle / ArcSDE Maintenance

Once Oracle and ArcSDE had been installed and the data had been successfully migrated to the ArcSDE geodatabase, the final step was to provide an environment conducive to on-going update and maintenance. A few things that we have implemented to facilitate update / maintenance are briefly mentioned below:

· Backup / Recovery: Currently, the database utilizes archive logging along with nightly exports of all data and cold backups once a week. With the move to Oracle 9i we will implement Oracle's Recovery Manager (RMAN) Utility.

· System Configuration: We have been very careful to minimize disk contention with proper / recommended placement of database files across the various disk drives. We are currently using raid5 but will be moving to raid1+0 for optimum performance and redundancy of an Oracle database. Oracle documentation on Striping and Mirroring Everything (SAME) has been very helpful as we prepare to move to a new SUN Array in the very near future.

· Database Tuning: At this point, our database is not used for edits and is basically a read-only repository of data. The tuning under such circumstances mainly involves keeping the Oracle DBMS statistics updated. This is accomplished with PL/SQL scripts that are run nightly. In addition, columns that are frequently used for reselecting / deleting features during nightly updates have been indexed to optimize performance. All indexes are re-built nightly.

· Image Data: At present our image data is accessed by ArcIMS and other applications as TIFF or MrSID images We have not loaded the raster files into ArcSDE because doing so would dramatically increase the amount of disk storage required. With the next phase of our migration we will move all significant raster files into SDE. Early tests have shown that the keys to successfully bulk-loading rasters include NOT building pyramids, calculating statistics or analyzing tables until ALL raster files are loaded.

System design considerations for ArcIMS

System design plays a big role in ArcIMS. Careful attention has to be paid to compatiblity issues. When we first implemented ArcIMS, we were installing ArcIMS 3.1. We selected the following as it was completely compatible with ArcIMS 3.1 according the ESRI website.

Microsoft Windows 2000 Advanced Server 5.00.2195 Service Pack 2
Intel Pentium III CPU family 1266 MHz 2096 MB RAM
Servlet Connector: JRUN 3.0 Admin Server
Microsoft Internet Information Server 5.0

Table 1: System Info

Installation of ArcIMS was very smooth and upgrade to ArcIMS 4.0 did not pose any problems. However when we did find out that ArcIMS 4.0.1 does not support JRUN 3.0 and hence we have to buy a completely new JRUN 4.0 with free upgrades version as we did not have the free upgrade version for JRUN 3.0.

A very useful whitepaper on system design strategies is at
http://downloads.esri.com/support/whitepapers/ao_/435sysdesig.pdf

Designing WebGIS

The main purpose of WebGIS was to create a user friendly search and analyse application for the office. WebGIS had to mimick the customized ArcView applications.

A standard out-of-the-box application created using ArcIMS looks as shown in Figure 1.

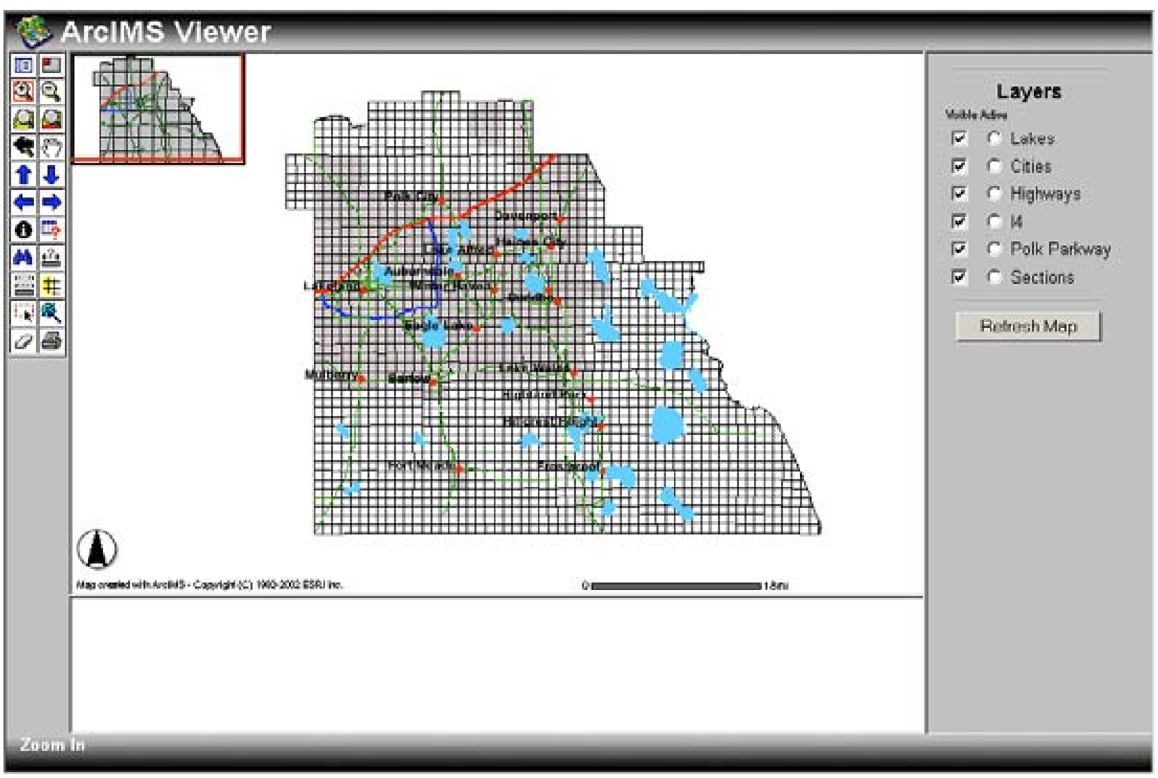

Figure 1. An out-of-the-box ArcIMS Application

However, most of the tools in the application were not intuitive to the users. Hence the whole interface was redesigned. The tools were modified and customized to suit the PCPA requirements. Figure 2 shows the newly designed interface.

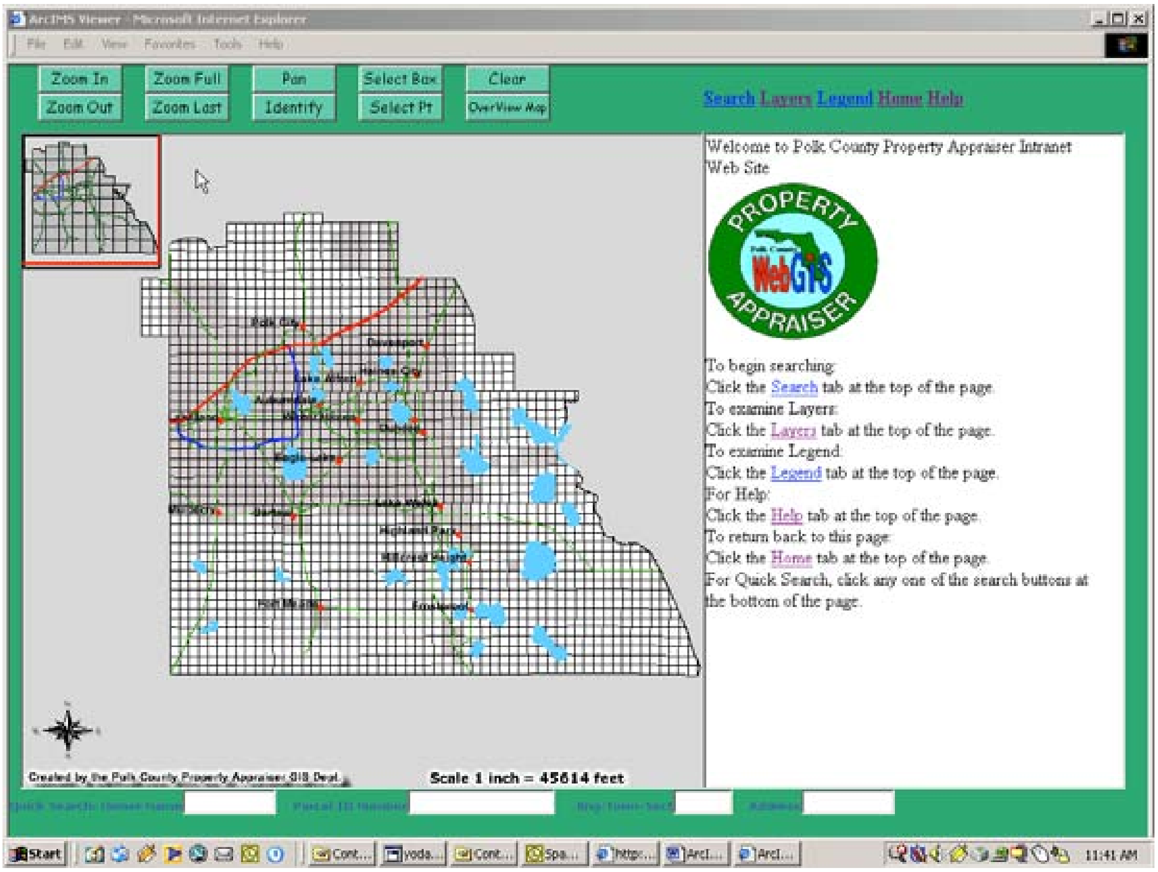

Figure 2: Newly designed interface for WebGIS

To accommodate various needs of Property Appraisal, we added more tools. Some of the major functionality added to WebGIS includes:

- Ability to search for parcels based on Owner Name, Parcel ID Number, RTS(Range-Township-Section) and Address using Quick Search.
- Ability to search for parcels using subdivision number.
- Examine deedlog entries.
- Search for engineered plats and display them using the Tiff Viewer.
- Dynamic Labelling of parcels based on tabular information.
- Dynamic Classification of parcels based on tabular information.
- Dynamic Querying of parcels and export the results to excel spreadsheet.
- Create 8.5”*11”, 11”*17” and 17”*22” print maps.
- Ability to select parcels by clicking (point and click)
- Add user defined text and lines.
- Zoom to scale functionality.
- Print Map to scale.
- Measure and show the area of the measurements.

The Search for parcels based on owner name and other criteria has been accomplished using ASP (Active Server Pages). We are using the GetRows method to get the results back quickly the client. The GetRows method is faster method if a large number of rows are returned by the query. Dynamic classification, labelling and querying is achieved by creating ArcXML requests and passing it to the server. Javascript is used for classification, labelling and querying, creating user defined text and lines and select by point. Images are stored as MrSid files and accessed from the server.

We have also created a daily recreate_services.bat batch file for refreshing the services daily at 6.00 am. This helps keep the data fresh. Contents of the recreate_services.bat file are follows:

```
@Echo Off

Net Stop "ArcIMS Tasker 4.0"

Net Stop "ArcIMS Monitor 4.0"

Net Stop "ArcIMS Application Server 4.0"

del "D:\Program Files\ArcGIS\ArcIMS\AppServer\*.sez"

Echo Deleted *.sez files from D:\Program Files\ArcGIS\ArcIMS\AppServer

del "D:\Program Files\ArcGIS\ArcIMS\AppServer\*.tmp"

Echo Deleted *.tmp files from D:\Program Files\ArcGIS\ArcIMS\AppServer

Echo ArcIMS Services are Stopped.

Net Start "ArcIMS Application Server 4.0"

Net Start "ArcIMS Monitor 4.0"

Net Start "ArcIMS Tasker 4.0"

Echo ArcIMS Services are Started.

java com.esri.aims.admincore.cmd.Exec http://yoda admin pcpagis file D:\arcims\axl\recreateservices.xml

Exit
```

The file above accesses the recreateservices.xml file the contents of which are as follows:

```
<?xml version="1.0"?>
<ADMINCMD version="1.0">
<SERVICES>
<SERVICE type="add" name="ArcMap"
axl="D:\ArcIMS\AXL\arcims.mxd"
```

```
vsname="ImageServerArcMap1"
imgloc="d:\ArcIMS\output"
imgurl="http://yoda/output"
imagememory="4"
cleanup="20"
imgtype="JPG" />

<SERVICE type="start" name="ArcMap" />

<SERVICE type="add" name="overview"
axl="D:\ArcIMS\AXL\overview.axl"
vsname="ImageServer1"
imgloc="d:\ArcIMS\output"
imgurl="http://yoda/output"
imagememory="21"
cleanup="60"
imgtype="PNG" />

<SERVICE type="start" name="overview" />

<SERVICE type="add" name="sdetest"
axl="D:\ArcIMS\AXL\sdetest.axl"
vsname="ImageServer1"
imgloc="d:\ArcIMS\output"
imgurl="http://yoda/output"
imagememory="21"
cleanup="60"
imgtype="PNG" />

<SERVICE type="start" name="sdetest" />

</SERVICES>
</ADMINCMD>
```

The recreateservices.xml recreates the services daily.

In addition to daily recreation of the ArcIMS Services, we also update the tabular information from CAMA daily to keep the data current with the CAMA database.

Table 2 shows the layers added to WebGIS.

LayerName
Sections
Sections0
Sections1
Sections2
Sections3
Sections4
Polk Parkway
I4

Highways
Cities
Lakes
Quad Map
Soils
Bartow LandUse
Lakeland LandUse
Polk City LandUse
Winter Haven LandUse
County LandUse
Millage
Ft Meade Images
Two Foot Images (2002)
One Foot Images (2001)
MrSid Archive (1995,1997,1998)
Wetlands
Contours
2002 DEM Contours
FloodZones
Parcels
2002 Hydro Poly
Hydro Poly
Parcel Boundary
Transparent Parcels
Base Area Parcels
Class Parcels
Decade Built Parcels
DOR Use Code Parcels
Land Type Parcels
Hydro Line
PLSS
Subdivision
Lot
City
Roads
Miscellaneous
Notes
Parcel Number
Parcel Dim
Layer Name
Hydro Anno
PLSS Anno
Subdivision Anno
RoadName
Lot Number
Lot Dim
City Anno
Misc Anno
Notes Anno
Parcel Number1
Parcel Dim1

Hydro Anno1
PLSS Anno1
Subdivision Anno1
RoadName1
Lot Number1
Lot Dim1
City Anno1
Misc Anno1
Notes Anno1
Parcel Number0
Parcel Dim0
Hydro Anno0
PLSS Anno0
Subdivision Anno0
RoadName0
Lot Number0
Lot Dim0
City Anno0
Misc Anno0
Notes Anno0
Section Line
Transparent Parcel Number
Transparent Parcel Dim
Transparent Parcel Number1
Transparent Parcel Dim1
Transparent Parcel Number0
Transparent Parcel Dim0
PolkZipCode
DOTBillboard

Table 2: Layers used in WebGIS

Conclusions

Employees at the PAO rely heavily on WebGIS for research and analysis of properties. Customer service uses WebGIS to help the public and creates printed maps for them. WebGIS is able to handle many concurrent users because of a robust system design and ArcSDE database design. Classification of properties enables appraisers to quickly visualize outliers and data inconsistencies. Ag properties can also be researched using WebGIS. The layers within WebGIS like soils, landuse etc. are used for appraisal purposes. Buffer tools helps PAO determine property ownership adjacent to roads and railroads. Export to Excel functions in WebGIS enable users to export their queried data to an excel spreadsheet for further analysis. The functionality of WebGIS makes it an indispensable tool for PAO.