

# Leveraging Python to Automate GIS Updates and Reporting in an Appraisal District

Keith R. Dailey; MS, GISP

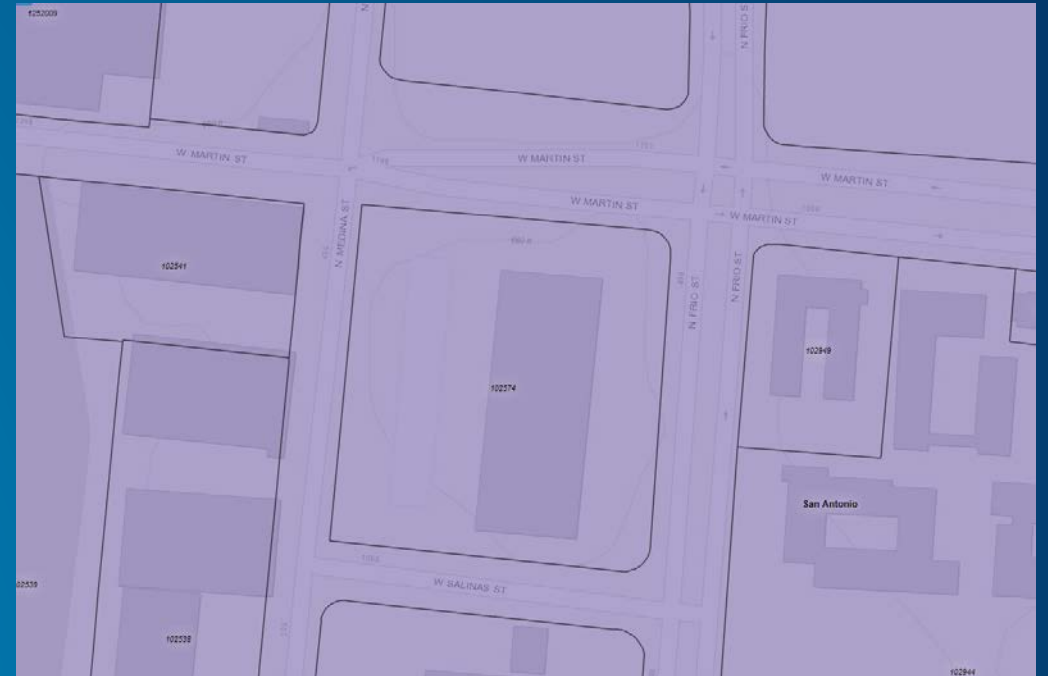
GIS Manager

Bexar Appraisal District



# Bexar Appraisal District

- Appraisal Districts in Texas started July 1, 1981
- GIS Department
  - Uses filed documents to determine ownership
  - Maintains mapping layers to aid in appraisal
  - Primary map layer is Parcel polygons



# Our Greatest Resource

- Excellent Staff
- TONS of data
- Computer Aided Mass Appraisal (CAMA)
  - Database of all our data used in appraising property
  - Current database goes back to accounts from 1998

PACS Appraisal - [Property List - 22 matches]

File View Activities Reports Tools Machine Settings Window Help

/	ID	GEO ID	D	Taxpayer	Type	HS	Situs Address	Legal	Legal Acres	Effective Size
2019	102574	00261-077-0131		BEXAR APPRAISAL DISTRICT	Real	No	411 N FRIO ST SAN ANTONIO, TX 78207	NCB 261 BLK 77 LOT 13 (VISTA VERDE PROJECT TEX R-109)	0.0000	2.2785
2018	102574	00261-077-0131		BEXAR APPRAISAL DISTRICT	Real	No	411 N FRIO ST SAN ANTONIO, TX 78207	NCB 261 BLK 77 LOT 13 (VISTA VERDE PROJECT TEX R-109)	0.0000	2.2785
2017	102574	00261-077-0131		BEXAR APPRAISAL DISTRICT	Real	No	411 N FRIO ST SAN ANTONIO, TX 78207	NCB 261 BLK 77 LOT 13 (VISTA VERDE PROJECT TEX R-109)	0.0000	2.2785
2016	102574	00261-077-0131		BEXAR APPRAISAL DISTRICT	Real	No	411 N FRIO ST SAN ANTONIO, TX 78207	NCB 261 BLK 77 LOT W IRR 242.80 FT OF 13	0.0000	2.2785
2015	102574	00261-077-0131		BEXAR APPRAISAL DISTRICT	Real	No	411 N FRIO ST SAN ANTONIO, TX 78207	NCB 261 BLK 77 LOT W IRR 242.80 FT OF 13	0.0000	2.2785
2014	102574	00261-077-0131		BEXAR APPRAISAL DISTRICT	Real	No	411 N FRIO ST SAN ANTONIO, TX 78207	NCB 261 BLK 77 LOT W IRR 242.80 FT OF 13	0.0000	2.2785
2013	102574	00261-077-0131		BEXAR APPRAISAL DISTRICT	Real	No	411 N FRIO ST SAN ANTONIO, TX 78207	NCB 261 BLK 77 LOT W IRR 242.80 FT OF 13	0.0000	2.2785
2012	102574	00261-077-0131		BEXAR APPRAISAL DISTRICT	Real	No	411 N FRIO ST SAN ANTONIO, TX 78207	NCB 261 BLK 77 LOT W IRR 242.80 FT OF 13	0.0000	.0000
2011	102574	00261-077-0131		BEXAR APPRAISAL DISTRICT	Real	No	411 N FRIO ST SAN ANTONIO, TX 78207	NCB 261 BLK 77 LOT W IRR 242.80 FT OF 13	0.0000	.0000
2010	102574	00261-077-0131		BEXAR APPRAISAL DISTRICT	Real	No	411 N FRIO ST SAN ANTONIO, TX 78207	NCB 261 BLK 77 LOT W IRR 242.80 FT OF 13	0.0000	.0000
2009	102574	00261-077-0131		BEXAR APPRAISAL DISTRICT	Real	No	411 N FRIO ST SAN ANTONIO, TX 78207	NCB 261 BLK 77 LOT W IRR 242.80 FT OF 13	0.0000	.0000
2008	102574	00261-077-0131		BEXAR APPRAISAL DISTRICT	Real	No	411 N FRIO ST SAN ANTONIO, TX 78207	NCB 261 BLK 77 LOT W IRR 242.80 FT OF 13	0.0000	.0000
2007	102574	00261-077-0131		BEXAR APPRAISAL DISTRICT	Real	No	411 N FRIO ST SAN ANTONIO, TX 78207	NCB 261 BLK 77 LOT W IRR 242.80 FT OF 13	0.0000	.0000
2006	102574	00261-077-0131		BEXAR APPRAISAL DISTRICT	Real	No	411 N FRIO ST SAN ANTONIO, TX 78207	NCB 261 BLK 77 LOT W IRR 242.80 FT OF 13	0.0000	.0000
2005	102574	00261-077-0131		BEXAR APPRAISAL DISTRICT	Real	No	411 N FRIO ST SAN ANTONIO, TX 78207	NCB 261 BLK 77 LOT W IRR 242.80 FT OF 13	0.0000	.0000
2004	102574	00261-077-0131		BEXAR APPRAISAL DISTRICT	Real	No	411 N FRIO ST SAN ANTONIO, TX 78207	NCB 261 BLK 77 LOT W IRR 242.80 FT OF 13	0.0000	.0000
2003	102574	00261-077-0131		BEXAR APPRAISAL DISTRICT	Real	No	411 N FRIO ST SAN ANTONIO, TX 78207	NCB 261 BLK 77 LOT W IRR 242.80 FT OF 13	0.0000	.0000
2002	102574	00261-077-0131		BEXAR APPRAISAL DISTRICT	Real	No	411 N FRIO ST SAN ANTONIO, TX 78207	NCB 261 BLK 77 LOT W IRR 242.80 FT OF 13	0.0000	.0000
2001	102574	00261-077-0131		BEXAR APPRAISAL DISTRICT	Real	No	411 N FRIO ST SAN ANTONIO, TX 78207	NCB 261 BLK 77 LOT W IRR 242.80 FT OF 13	0.0000	.0000
2000	102574	00261-077-0131		WEST VISTA VERDE DEVELOPMENT	Real	No	411 N FRIO ST SAN ANTONIO, TX 78207	NCB 261 BLK 77 LOT W IRR 242.80 FT OF 13	0.0000	.0000
1999	102574	00261-077-0131		WEST VISTA VERDE DEVELOPMENT	Real	No	411 N FRIO ST SAN ANTONIO, TX 78207	NCB 261 BLK 77 LOT W IRR 242.80 FT OF 13	0.0000	.0000
1998	102574	00261-077-0131		TIOA MASTER LTD PTSHIP	Real	No	411 N FRIO ST SAN ANTONIO, TX 78207	NCB 261 BLK 77 LOT W IRR 242.80 FT OF 13	0.0000	.0000

Commercial Real Property: (00261-077-0131) Year - 2019 (Submitted to ARB)

**Summary**  
Legal  
Identification  
Entity-Exmpt  
Prop Codes  
Mtg-Permits  
Owner-Agent  
Deeds-Sales  
SplitMerge-Lnk  
Appr Summ  
Values  
Improvements  
Land  
Supp History  
Appraiser Info  
Roll History  
GIS  
Images  
RBack-ShProp  
Events  
ARB - Inquiry

**Property Info**  
PID: 102574  
00261-077-0131

**Owner Name**  
BEXAR APPRAISAL DISTRICT

**Legal Description**  
NCB 261 BLK 77 LOT 13 (VISTA VERDE PROJECT TEX R-109)

**Property**  
State Code: F1 Create Date:  
DBA: "EXEMPT" BEXAR APPRAISAL  
Nbhd: 10040 (NBHD code 10040)  
ECU: none Valuation Type: Cost

**Owner Name & Address [Pct: 100.0000000000%]**  
BEXAR APPRAISAL DISTRICT  
ATTN: MICHAEL AMEZQUITA  
PO BOX 830248  
SAN ANTONIO, TX 78283-0248

**Mortgage Co:**  
Market Value:  
Assessed Value:  
Living Area: 51,712 \$/SF:

**Exemptions: EX-XV**


**Agents**

**Entity** Frz Type Frz Year Frz Ceiling  
06  
08  
n9

**Print Appraisal Card** **View Plat Map**

**Situs**  
411 N FRIO ST SAN ANTONIO, TX 78207

**Recalculate** **<** **>** **OK** **Cancel** **Apply**



Leveraging Python to Automate GIS Updates and Reporting in an Appraisal District

# What does this data look like?

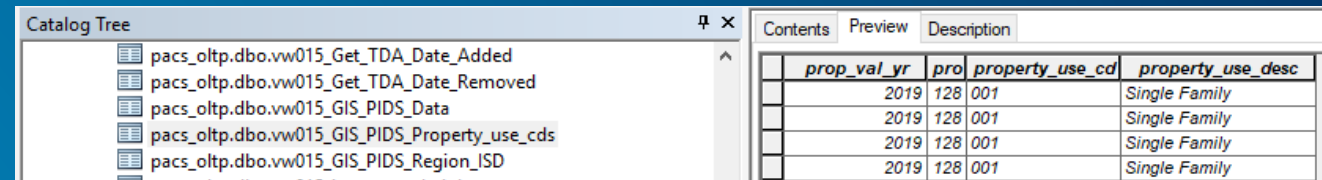
- We have a massive amount of data

- Static

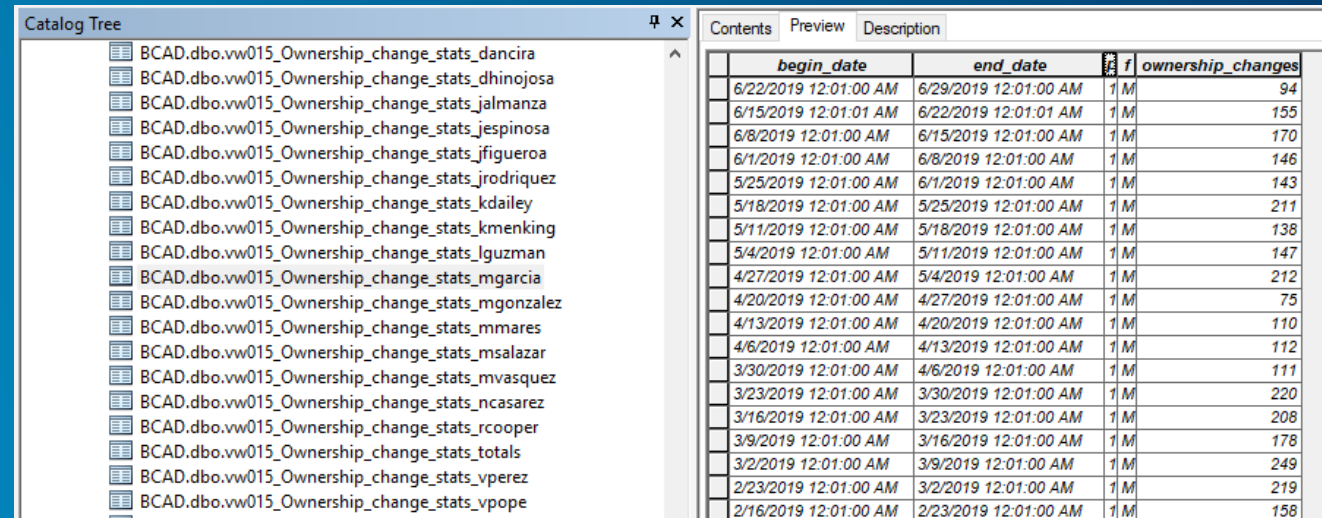
- Appraised value on certified rolls
    - Improvement specs
      - Year built
      - Square feet
      - Bedroom/bathroom/detachments
      - Etc
    - Land size

- Dynamic

- # of Deeds Processed
    - # of GIS Requests Processed
    - Daily/Weekly Production Reports



prop_val_yr	pro	property_use_cd	property_use_desc
2019	128	001	Single Family
2019	128	001	Single Family
2019	128	001	Single Family
2019	128	001	Single Family



begin_date	end_date	f	ownership_changes
6/22/2019 12:01:00 AM	6/29/2019 12:01:00 AM	1	94
6/15/2019 12:01:01 AM	6/22/2019 12:01:01 AM	1	155
6/8/2019 12:01:00 AM	6/15/2019 12:01:00 AM	1	170
6/1/2019 12:01:00 AM	6/8/2019 12:01:00 AM	1	146
5/25/2019 12:01:00 AM	6/1/2019 12:01:00 AM	1	143
5/18/2019 12:01:00 AM	5/25/2019 12:01:00 AM	1	211
5/11/2019 12:01:00 AM	5/18/2019 12:01:00 AM	1	138
5/4/2019 12:01:00 AM	5/11/2019 12:01:00 AM	1	147
4/27/2019 12:01:00 AM	5/4/2019 12:01:00 AM	1	212
4/20/2019 12:01:00 AM	4/27/2019 12:01:00 AM	1	75
4/13/2019 12:01:00 AM	4/20/2019 12:01:00 AM	1	110
4/6/2019 12:01:00 AM	4/13/2019 12:01:00 AM	1	112
3/30/2019 12:01:00 AM	4/6/2019 12:01:00 AM	1	111
3/23/2019 12:01:00 AM	3/30/2019 12:01:00 AM	1	220
3/16/2019 12:01:00 AM	3/23/2019 12:01:00 AM	1	208
3/9/2019 12:01:00 AM	3/16/2019 12:01:00 AM	1	178
3/2/2019 12:01:00 AM	3/9/2019 12:01:00 AM	1	249
2/23/2019 12:01:00 AM	3/2/2019 12:01:00 AM	1	219
2/16/2019 12:01:00 AM	2/23/2019 12:01:00 AM	1	158

*Leveraging Python to Automate GIS Updates and Reporting in an Appraisal District*



```
#EYOC
print 'Dissolving EYOC'
arcpy.AddMessage ("Dissolving EYOC")
arcpy.env.outputCoordinateSystem = arcpy.SpatialReference("Coordinate Systems/Projected
arcpy.Dissolve_management(SelectedPIDS, EYOC_shp, "EYOC", "", "MULTI_PART", "DISSOLVE_LINES")
print 'Dissolved EYOC'
arcpy.AddMessage ("Dissolved EYOC")

#Roof
print 'Dissolving Roof'
arcpy.AddMessage ("Dissolving Roof")
arcpy.env.outputCoordinateSystem = arcpy.SpatialReference("Coordinate Systems/Projected
arcpy.Dissolve_management(SelectedPIDS, Roof_shp, "Roof", "", "MULTI_PART", "DISSOLVE_LINES")
print 'Dissolved Roof'
arcpy.AddMessage ("Dissolved Roof")

#ExtWall
print 'Dissolving ExtWall'
arcpy.AddMessage ("Dissolving ExtWall")
arcpy.env.outputCoordinateSystem = arcpy.SpatialReference("Coordinate Systems/Projected
arcpy.Dissolve_management(SelectedPIDS, ExtWall_shp, "ExtWall", "", "MULTI_PART", "DISSOLVE_LINES")
print 'Dissolved ExtWall'
arcpy.AddMessage ("Dissolved ExtWall")

#FirePlace
print 'Dissolving FirePlace'
arcpy.AddMessage ("Dissolving FirePlace")
arcpy.env.outputCoordinateSystem = arcpy.SpatialReference("Coordinate Systems/Projected
arcpy.Dissolve_management(SelectedPIDS, FirePlace_shp, "FirePlace", "", "MULTI_PART", "DISSOLVE_LINES")
print 'Dissolved FirePlace'
arcpy.AddMessage ("Dissolved FirePlace")

#Heat
print 'Dissolving Heat'
arcpy.AddMessage ("Dissolving Heat")
arcpy.env.outputCoordinateSystem = arcpy.SpatialReference("Coordinate Systems/Projected
arcpy.Dissolve_management(SelectedPIDS, Heat_shp, "Heat", "", "MULTI_PART", "DISSOLVE_LINES")
print 'Dissolved Heat'
arcpy.AddMessage ("Dissolved Heat")

#AC
print 'Dissolving AC'
arcpy.AddMessage ("Dissolving AC")
arcpy.env.outputCoordinateSystem = arcpy.SpatialReference("Coordinate Systems/Projected
arcpy.Dissolve_management(SelectedPIDS, AC_shp, "AC", "", "MULTI_PART", "DISSOLVE_LINES")
print 'Dissolved AC'
arcpy.AddMessage ("Dissolved AC")

#WholeBaths
print 'Dissolving WholeBaths'
arcpy.AddMessage ("Dissolving WholeBaths")
arcpy.env.outputCoordinateSystem = arcpy.SpatialReference("Coordinate Systems/Projected
arcpy.Dissolve_management(SelectedPIDS, WholeBaths_shp, "WholeBaths", "", "MULTI_PART", "DISSOLVE_LINES")
print 'Dissolved WholeBaths'
arcpy.AddMessage ("Dissolved WholeBaths")

#HalfBaths
print 'Dissolving HalfBaths'
arcpy.AddMessage ("Dissolving HalfBaths")
arcpy.env.outputCoordinateSystem = arcpy.SpatialReference("Coordinate Systems/Projected
arcpy.Dissolve_management(SelectedPIDS, HalfBaths_shp, "HalfBaths", "", "MULTI_PART", "DISSOLVE_LINES")
print 'Dissolved HalfBaths'
arcpy.AddMessage ("Dissolved HalfBaths")
Dissolve_time=time.strftime('%I:%M_%S %p')
Dissolve_message=(": Dissolved Based on Attributes \n")
print Dissolve_time + ': Dissolved Based on Attributes Successful!'
arcpy.AddMessage ("Dissolved Based on Attributes Successful!")
```

# What Can We Do with this Data?

## Automatically Update Map Layers

## Routine Update of Map Layers

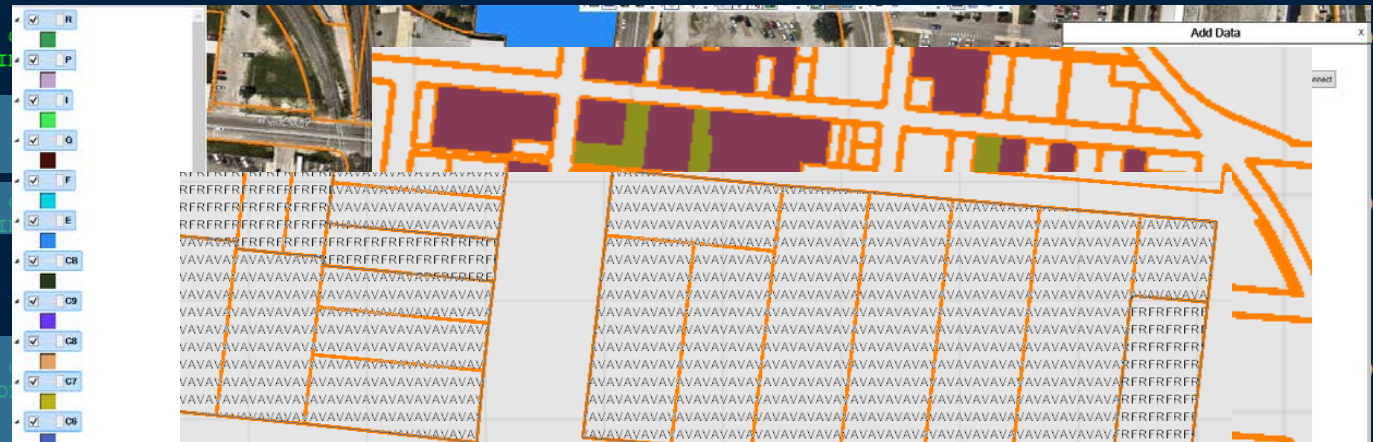
## CAMA

## FTP

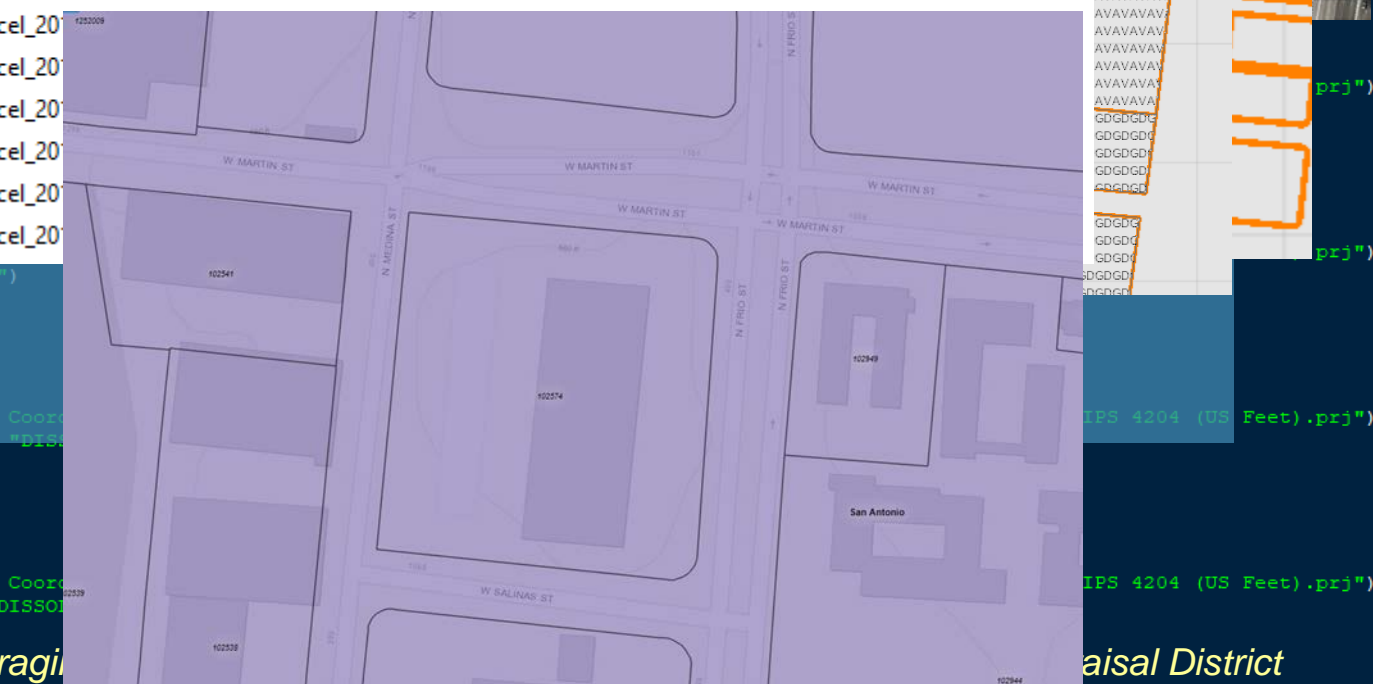
## Public website

Leveraging

aisal District



Name	Date modified	Type	Size
Parcel_2019_06_30.zip	6/30/2019 5:26 PM	Compressed (zipp...	204,746 KB
Parcel_2019_06_23.zip	6/23/2019 5:26 PM	Compressed (zipp...	204,721 KB
Parcel_2019_06_16.zip	6/16/2019 5:25 PM	Compressed (zipp...	204,557 KB



```
while counter <= Sortcount:
    percent = int((float(counter)/float(Sortcount))*100)
    prevrowPID = rowPID
    prevrowZONE = rowZONE
    rowPID = row.getValue("BCADParcelID")
    rowZONE = row.getValue("Zoning")
    if counter == Sortcount:
        print "Processing %s of %s" % (str("(:,g)".format(Sortcount)) + ' (' + str("(:.4f)".format(calc)) + "%) Num of PIDs with No Zone = " + str(NoZoneCount),
        del cursor
        del row
    else:
        row = cursor.next()
        rowPIDtest = row.getValue("BCADParcelID")
        rowZONetest = row.getValue("Zoning")
        if rowPID <> rowPIDtest:
            if rowZONE == rowZONetest:
                if PID_NO_ZONE == rowZONE:
                    PID_NO_ZONE = rowZONE
                    NoZoneCount = 1
                else:
                    PID_NO_ZONE = PID_NO_ZONE + ", " + rowZONE
                    NoZoneCount = NoZoneCount + 1
            else:
                AddRows = arcpy.InsertCursor(OutTable)
                AddRow = AddRows.newRow()
                AddRow.setValue ("PID", rowPID)
                AddRow.setValue ("ZONE", rowZONE)
                AddRows.insertRow(AddRow)
                del AddRow
                del AddRows
        elif rowPID == rowPIDtest and rowZONE == rowZONetest:
            A=1
        elif rowPID == rowPIDtest and rowZONE <> rowZONetest:
            while Name
            print "Processing %s of %s" % (str("(:,g)".format(Sortcount)) + ' (' + str("(:.4f)".format(calc)) + "%) Num of PIDs with No Zone = " + str(NoZoneCount),
            del cursor
            del row
        counter = counter + 1
    printpercent = int((float(counter)/float(Sortcount))*100)
    #Time = time.strftime('%I:%M:%S %p')
    calc = (float(counter)/float(Sortcount))*100
    print '\r' + 'Processing ' + str("(:,g)".format(counter)) + ' of ' + str("(:,g)".format(Sortcount)) + ' (' + str("(:.4f)".format(calc)) + "%) Num of PIDs with No Zone = " + str(NoZoneCount),
    del cursor
    del row
```

What Can We Do with this Data

Automatically Update CAMA

Routine Update from External Data

From FTP to CAMA

Updating of Zoning from City of

	OBJECTID *	BCA	Zoning	Base
	1	110	RM-4	RM-4
	2	110	RM-4	RM-4
	3	110	C-1	C-1
	4	110	RM-4	RM-4
	5	110	RM-4	RM-4
	6	110	RM-4	RM-4
	7	110	RM-4	RM-4
	8	110	RM-4	RM-4
	9	110	RM-4	RM-4
	10	110	RM-4	RM-4
	11	110	RM-4	RM-4
	12	110	RM-4	RM-4
	13	110	RM-4 CD	RM-4
	14	110	RM-4 CD	RM-4
	15	556	R-6	R-6
	16	110	RM-4	RM-4
	17	556	R-6	R-6
	18	113	I-2	I-2
	19	113	MF-33	MF-33
	20	113	MF-33	MF-33
	21	113	MF-33	MF-33
	22	113	MF-33	MF-33
	23	113	MF-33	MF-33
	24	113	MF-33	MF-33
	25	113	MF-33	MF-33
	26	113	MF-33	MF-33
	27	113	MF-33	MF-33
	28	113	I-2	I-2
	29	113	I-2	I-2
	30	113	I-2	I-2
	31	113	I-2	I-2
	32	113	MF-33	MF-33
	33	113	R-6 CD	R-6
	34	113	MF-33	MF-33
	35	113	MF-33	MF-33
	36	113	I-2	I-2
	37	113	I-2	I-2

Name	Date modified	Type
CoSA_Zoning_2019_03_19.zip	3/19/2019 3:36 PM	Com
CoSAZoning_2019_04_18.zip	4/18/2019 3:14 PM	Com
CoSAZoning_2019_05_3.zip	5/3/2019 10:52 AM	Com
CoSAZoning_2019_05_31.zip	5/31/2019 2:39 PM	Com

Commercial Real Property: (00261-077-0131) Year - 2019 (Submitted to ARB)

Summary

Legal

Identification

Entity-Exmpt

Prop Codes

Mtg-Permits

Owner-Agent

Deeds-Sales

SplitMerge-Lnk

Appr Summ

Values

Improvements

Land

Supp History

Appraiser Info

Roll History

GIS

Images

RBack-ShProp

Events

ARB - Inquiry

Property Info

Owner Name

Legal Description

General

Other

TIF Zone:

Group Codes

Mobile Assignment Group:

Property Designation:

Recalculate

<

>

OK

Cancel

Apply

Integrate GIS Updates and Reporting in an Appraisal District

```
#Iterate through GIS Report by Work Areas to create separate report tables for each work area
print "Creating Work Area Specific Report Tables"
```

```
arcpy.env.overwriteOutput = True
```

# What Can We Do with this Data?

## Send Work Reports to Staff

```
AreaCodeSet = arcpy.da.SearchCursor(WorkAreaReport, AreaField)
strAreaCode = map(str,AreaCodeSet)
print strAreaCode
AreaCodeList = strAreaCode
for AreaCode in AreaCodeList:
```

```
print "Work Area " + AreaCode + " is no
```

```
AreaSQL = "Work_Area = {0}".format(AreaCode)
dbfName = "WorkArea"+AreaCode+"DBF.dbf"
```

```
csvName = "WorkArea"+AreaCode+".csv"
```

```
arcpy.TableToTable_conversion(WorkAreaReport, FileToCalc = CSVworkspace+"\\ "+dbfName
```

```
arcpy.TableToTable_conversion(FileToCalc, FileToSend = CSVworkspace+"\\ "+csvName
```

```
EmailTo = GISDict.get("WorkArea"+AreaCode)
dfl = pandas.read_csv(FileToSend)
```

```
dfl['Date_Assig'] = pandas.to_datetime(dfl['Date_Assig'])
dfl['Date_Assig'] = dfl['Date_Assig'].dt.strftime('%Y-%m-%d')
print(dfl)
```

```
Content=dfl.to_html()
```

```
text_subtype = 'html'
```

```
msg = MIMEText(Content, text_subtype)
```

```
msg ['Subject'] = 'Work Area Report for ' + AreaCode
```

```
msg ['From'] = sender
```

```
msg ['To'] = EmailTo
```

```
msg ['CC'] = CC_Email
```

```
s = smtplib.SMTP('emailserver')
```

```
s.sendmail(sender, [EmailTo, sender], msg.as_string())
```

```
s.quit()
```

Pending Plats for Residential For Week of: July 01, 2019

To Keith R. Dailey

Cc Keith R. Dailey

If there are problems with how this message is displayed, click here to view it in a web browser.

	OID	Date_Assigned	Account_Number	Year	Type	Vol_Page	Action	Count_of_Accts	Tile	PID	Work_Area
0	-1	04/18/2019	51930000413	2020	PL	20001/684-686	PLAT	61.0	4259	339484	6
1	-1	04/18/2019	40540130130	2020	PL	20001/687	PLAT	2.0	4123	175268	5
2	-1	04/18/2019	58860000040	2020	PL	20001/688	PLAT	2.0	4123	357905	5
3	-1	04/18/2019	109360040181	2020	PL	20001/696	PLAT	6.0	4202	469698	7
4	-1	04/18/2019	43850150032	2020	PL	20001/701	PLAT	3.0	4088	203828	2
5	-1	04/18/2019	36900280090	2020	PL	20001/702	PLAT	7.0	4299	150198	3
6	-1	04/18/2019	10840110132	2020	PL	20001/707	PLAT	4.0	4132	112154	5
7	-1	04/18/2019	58901080210	2020	PL	20001/708	PLAT	5.0	4123	358005	5
8	-1	04/18/2019	47090001182	2020	PL	20001/715	PLAT	9.0	4244	253130	4
9	-1	05/17/2019	50870000765	2020	PL	20001/736	PLAT	2.0	4259	1278641	6
10	-1	05/17/2019	10290020090	2020	PL	20001/907	PLAT	14.0	4232	111695	7

Leveraging Python to Automate GIS Updates and Reporting in an Appraisal District



# What Can We Do with this Data?

## Watch Daily Work Log in Real Time

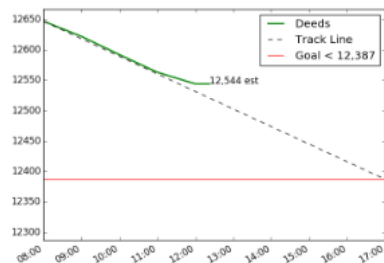
[Print This Report](#)

GIS Department Health Report for **July 05, 2019**  
Report Last Ran at 12:22:51 PM

Report runs automatically every hour between 8:00 AM and 5:00 PM M-F  
Webpage will auto-refresh every 15 minutes  
[Click Here to Request a Manual Update or Report an Issue](#)

### Deed Stats

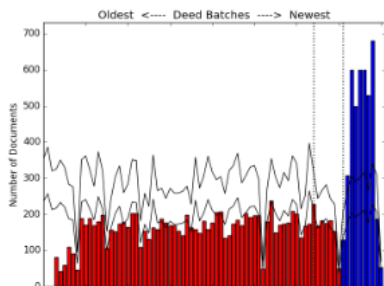
Within 30 Day Goal Date at Assigned Rate: **February 10, 2020** (220 Days from today)  
Within 30 Day Goal Date at Current Rate: **November 12, 2021** (861 Days from today)



2018 Deeds Not Processed: 0,000  
2019 Deeds Not Processed: **15,323** .....10,922 older than 30 days  
Total Deeds Not Processed: **15,323**  
Deeds Loaded into PACS through June 11, 2019  
(24 days from current)

2018 SIPs: 0,076  
2019 SIPs: **0,088**  
Total SIPs: **0,164**

Ownership Transfer Events Last Week: **1,305**



419-GIS Hold Accounts: **14**  
[GIS Correction Form](#)

### Deed Bonus (✓ = completed)

All Loaded Deed Batches Are Cleaned  
Within 30 Days of Clerk's Office.....(95 Days)  
Oldest deed not processed from approx. April 01, 2019

### Mapping Stats

2019 GIS List Pending: 0,002  
2020 GIS List Pending: **4,474**  
Total GIS List Pending: **4,476**  
Accounts from Plats: 3,940

Accounts needing QC: **1,618**  
2020 Total Accounts Worked: **1,938**

Number of CR's pending: 261

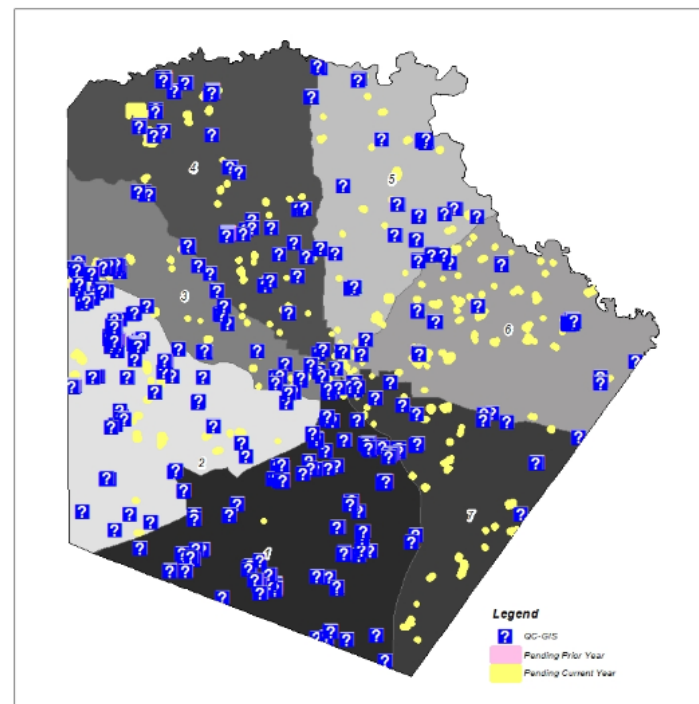
CR's Older than 150 Days: **0**  
Oldest CR is **116** days old  
[Click for List of CR's Older Than 150 Days](#)

COM Accts with NO-G Hold: 0  
RES Accts with NO-G Hold: 0  
TOT Accts with NO-G Hold: 0

[Click for List of Pending Residential Plats](#)  
[Click for List of Pending Commercial Plats](#)  
[Click for List of Pending Plat Images](#)

### Mapping Bonus (✓ = completed)

All Plats with more than 50 Accounts Worked  
All Plats with more than 100 Accounts Worked  
All Plats with more than 150 Accounts Worked



### Current Department Priorities

1. Clean the new deed batches  
[Check for Missed Deeds Here](#)  
[Check for Deeds Already Processed](#)
2. Work oldest deeds first
3. SIPs  
[Research SIPs here](#)

1. Research received from Deed Techs
2. Plats with more than 50 accounts
3. Oldest plats (after the large plats are completed)
4. Oldest splits, merges, resurveys, etc...

[CR Priority List](#)  
[All Pending GIS Types \(postpend list\)](#)  
[Research Situs Addresses](#)



[illegible]

```
arcpy.Delete_management (template2)
arcpy.Delete_management (styleSheet2)
print "Template2 Deleted"
arcpy.Copy_management (template, template2)
arcpy.Copy_management (styleSheet, styleSheet2)
print "Template2 recreated"

htmlSiteWrite = open(template2).read().format(HprintDate=printDate,HgoalDate=goalDate, HintCumuDAYS=intCumuDAYS,
HstartTime=startTime,
HDeedLogEmail_png=DeedLogEmail_png,
HDeedBatchEmail_png=DeedBatchEmail_png,
HDeedpriorYr=stx(DeedpriorYr), HintPriorYrDeeds2="{:04,g}".format(intPriorYr),
HintNotProc302="{:04,g}".format(intNotProc302),
HDeednowYr=stx(DeednowYr), HFindCurYrDeeds="{:04,g}".format(FindCurYrDeeds),
HintDeedsLoadNotProc2="{:04,g}".format(intDeedsLoadNotProc2),
HEndDate=EndDateReport, HEndDaysAway=EndDaysAway,
HintCumuDAYS=HintCumuDAYS, HintCumuDAYS=intCumuDAYS,
HpriorYr=stx(priorYr), HnowYr=stx(nowYr),
HintSIP_prior2="{:04,g}".format(intSIP_prior2),
HintSIP_cur2="{:04,g}".format(intSIP_cur2),
HtotSIP="{:04,g}".format(TotSIP), HSIPRate=SIPRate,
HintTransferEventsLAST2="{:04,g}".format(intTransferEventsLAST2), HEventRate=EventRate,
HintG419HoldCount2=intG419HoldCount2,
HcheckMarkCleaned=checkMarkCleaned,
HcheckMarkDays=checkMarkDays, HDaysAway=DaysAway,
HStartDateReport=StartDateReport,
HMapCR150Email=MapCR150Email, HMAXdaysPrint=MAXdaysPrint,
HintFindp_SUM GIS2="{:04,g}".format(intFindp_SUM GIS2),
HfutureYr=stx(futureYr), HintFindc_SUM GIS2="{:04,g}".format(intFindc_SUM GIS2),
HTotalGIS="{:04,g}".format(TotalGIS), HGISPendRate=GISPendRate,HintQCTBLCount2=HintQCTBLCount2,
HintTotPlatAcctNum2="{:04,g}".format(intTotPlatAcctNum2),
HintTotCRNum2="{:04,g}".format(intTotCRNum2),
HCRI150Days=CR150Days,
HintComHoldCount2=intComHoldCount2, HintResHoldCount2=intResHoldCount2, HHintTotPlatNum2="{:04,g}".format(intTotPlatNum2),
HPercentGISToLock="{:04,g}".format(PercentGISToLock),
HGISWorkCur="{:04,g}".format(GISWorkCur),
HcheckMarkAcctNum50=checkMarkAcctNum50,
HcheckMarkAcctNum100=checkMarkAcctNum100,
HcheckMarkAcctNum150=checkMarkAcctNum150,
HHealthRating=HealthRating,
Hgraydeed_p=graydeed_p,
Hgraydeed_c=graydeed_c,
Hgrayclean=grayclean,
Hcleanstrike=cleanstrike,
Hmapgray50=mapgray50,
Hmapgray100=mapgray100,
Hmapgray150=mapgray150,
HgrayGIS_p=grayGIS_p)

htmlSite=open(template2, "w+")
htmlSite.write(htmlSiteWrite)
htmlSite.close()

lognameDate = today.strftime('%m%d%Y')
logname = stx(lognameDate)+"_"+stx(lognameTime)+"_html1"
newLog = ReportLogs+"\\"+logname
arcpy.Copy_management (template2, newLog)
```


[Print This Report](#)

GIS Department Health Report for **{HprintDate}**  
Report Last Ran at {HstartTime}

Report runs automatically every hour between 8:00 AM and 5:00 PM M-F  
Webpage will auto-refresh every 15 minutes  
[Click Here to Request a Manual Update or Report an Issue](#)

### Deed Stats

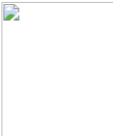
Within 30 Day Goal Date at Assigned Rate: **{HgoalDate}** ({HintCumuDAYS} Days from today)  
Within 30 Day Goal Date at Current Rate: **{HgoalDate\_cur}** ({HintCumuDAYS\_cur} Days from today)



**{HDeedpriorYr} Deeds Not Processed: {HintPriorYrDeeds2}**  
**{HDeednowYr} Deeds Not Processed: {HFindCurYrDeeds}**.....**{HintNotProc302}**  
older than 30 days  
Total Deeds Not Processed: **{HintDeedsLoadNotProc2}**  
Deeds Loaded into PACS through **{HEndDate}**  
{HEndDaysAway} days from current

**{HpriorYr} SIPs: {HintSIP\_prior2}**  
**{HnowYr} SIPs: {HintSIP\_cur2}**  
Total SIPs: **{HTotSIP}**

Ownership Transfer Events Last Week: **{HintTransferEventsLAST2}**



419-GIS Hold Accounts: **{HintG419HoldCount2}**  
[GIS Correction Form](#)

**Deed Bonus (✓ = completed)**  
**{HcheckMarkCleaned} All Loaded Deed Batches Are Cleaned**  
**{HcheckMarkDays} Within 30 Days of Clerk's Office**.....  
**{HDaysAway} Days**  
Oldest deed not processed from approx. **{HStartDateReport}**

### Mapping Stats

**{HnowYr} GIS List Pending: {HintFindp\_SUM GIS2}**  
**{HfutureYr} GIS List Pending: {HintFindc\_SUM GIS2}**  
Total GIS List Pending: **{HTotalGIS}**  
Accounts from Plats: {HintTotPlatAcctNum2}

Accounts needing QC: **{HintQCTBLCount2}**  
{HfutureYr} Total Accounts Worked: **{HGISWorkCur}**

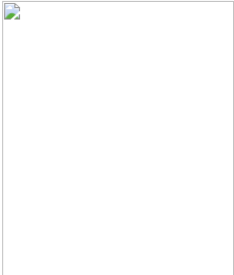
Number of CR's pending: {HintTotCRNum2}

CR's Older than 150 Days: **{HCRI150Days}**  
Oldest CR is **{HMAXdaysPrint}** days old  
[Click for List of CR's Older Than 150 Days](#)

COM Accts with NO-G Hold: {HintComHoldCount2}  
**RES Accts with NO-G Hold: {HintResHoldCount2}**  
TOT Accts with NO-G Hold: **{HTotalNOG}**

[Click for List of Pending Residential Plats](#)  
[Click for List of Pending Commercial Plats](#)  
[Click for List of Pending Plat Images](#)

**Mapping Bonus (✓ = completed)**  
{HcheckMarkAcctNum50} All Plats with more than 50 Accounts Worked  
{HcheckMarkAcctNum100} All Plats with more than 100 Accounts Worked  
{HcheckMarkAcctNum150} All Plats with more than 150 Accounts Worked



Leveraging Python to Automate GIS Updates and Reporting in an Appraisal District

## Our Newest Tool

## Appraisal District

# Tips and Tricks I've Learned

- Have good communication with other departments
  - What do they *need* to see from the data?
  - What do they *want* to see from the data?
- Start with 'What If...'
- E-Mail Status of Report
  - Windows Automated Tasks
  - Did it succeed?
- Try / Except is GREAT for automated tasks



# Acknowledgments

- **My amazing team of Deed Techs and Map Techs**
- **BCAD Information Services Department**
- **Dedicated Executive Leadership:**
  - **Michael Amezquita**, Chief Appraiser
  - **Scott Griscom**, Assistant Chief Appraiser
  - **Rogelio Sandoval**, Assistant Chief Appraiser

# Questions?

Section Subhead

