A GIS Approach to Charting Terrain On Instrument Approach Procedures

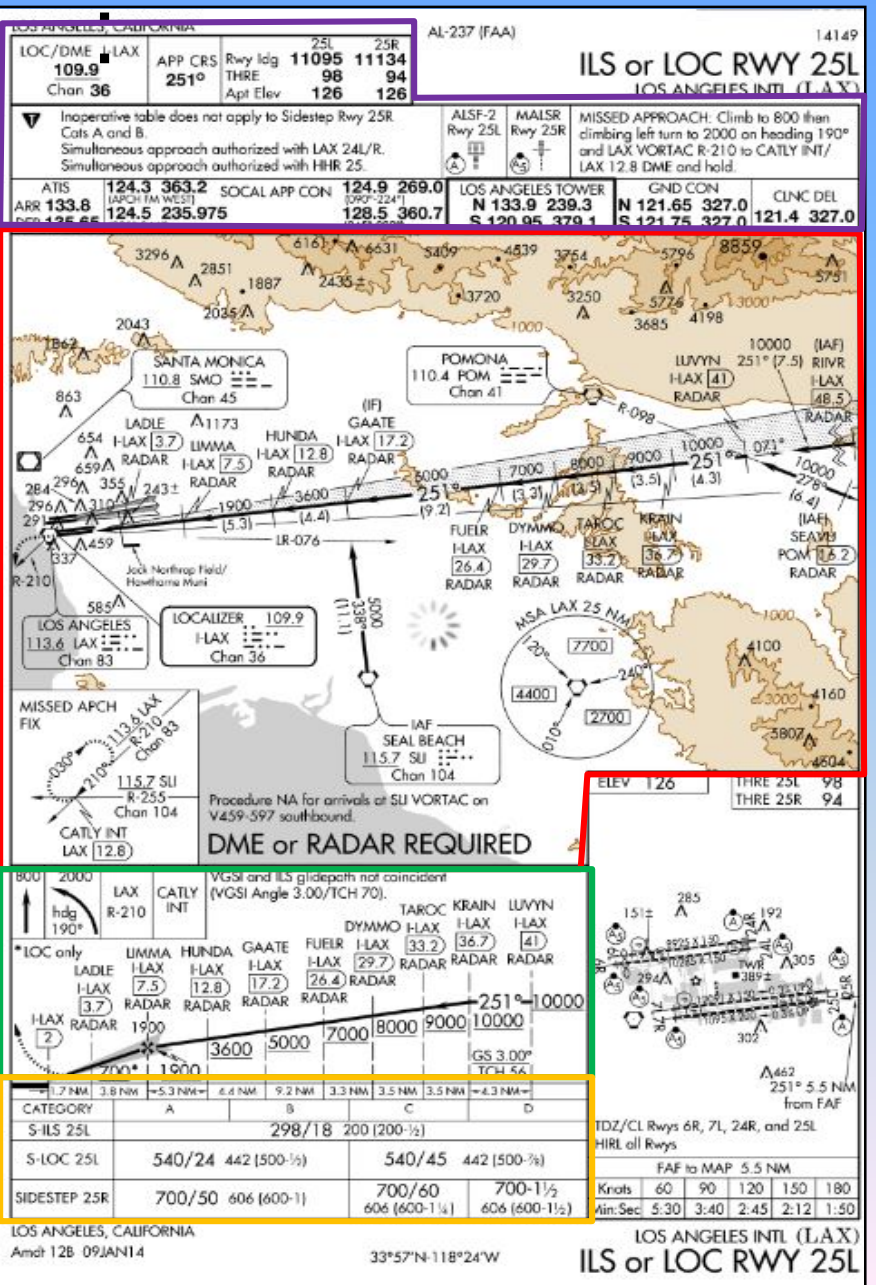
Brent M. Baumhardt

Esri Users Conference

July 23, 2015

Pennsylvania State University
Masters in Geographic Information Systems

Volpe Instrument Approach Procedure (IAP)



- Organizes information into logical flow
- Less time searching… more time flying
- Human factors study

Pilot Briefing Bar – Notes/Comm

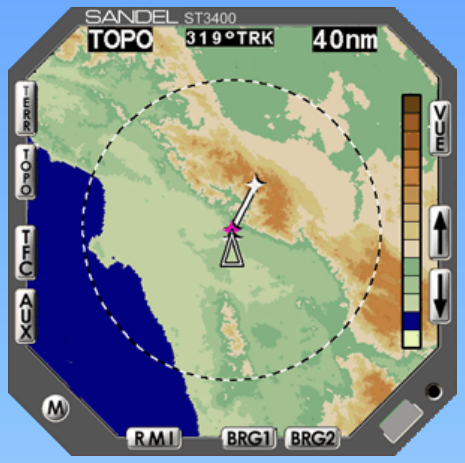
Planview – Complete IAP extent

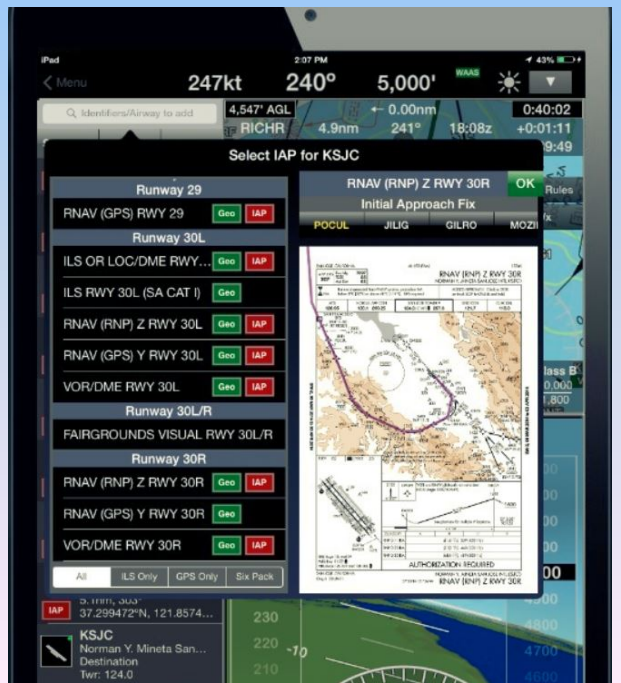
Profile – Altitude stepdowns

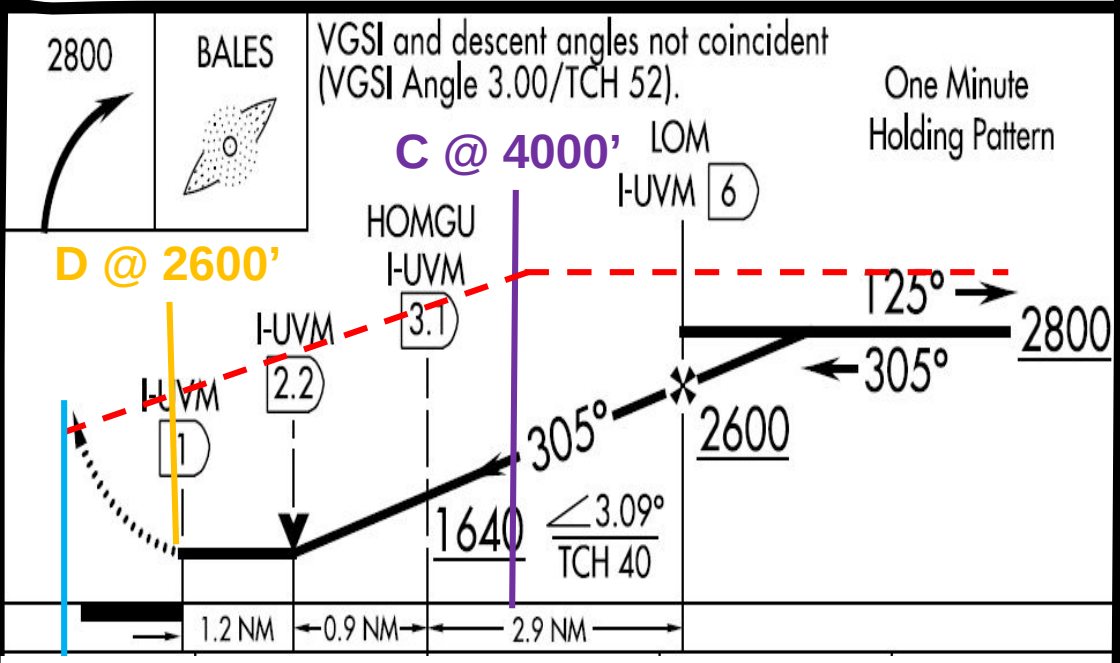
Minima – Defined minimum descent heights

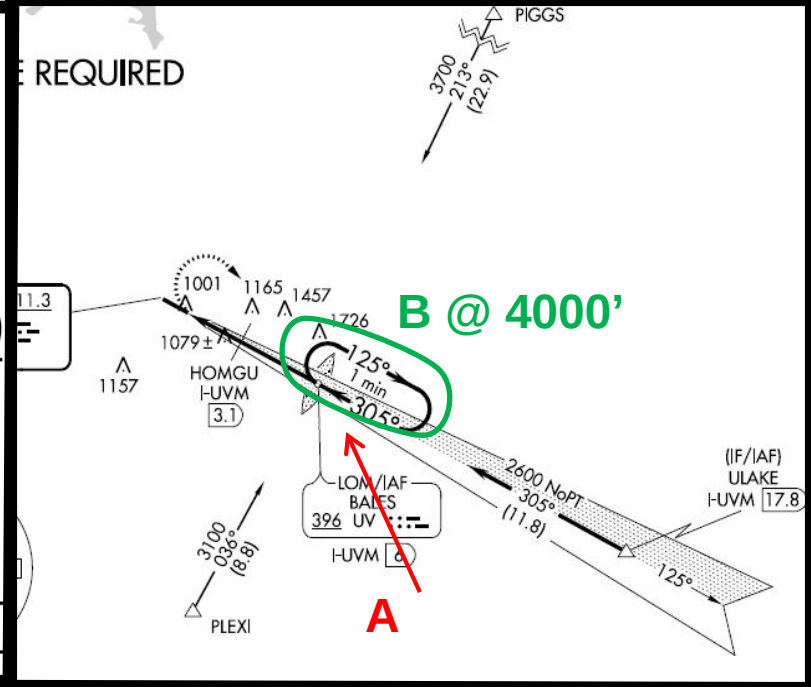
Technology navigational enhancements:

- Terrain Awareness and Warning System (TAWS)
- Electronic Flight Bags (EFB)
- FMS/GPS
- Improved ATC RADAR coverage

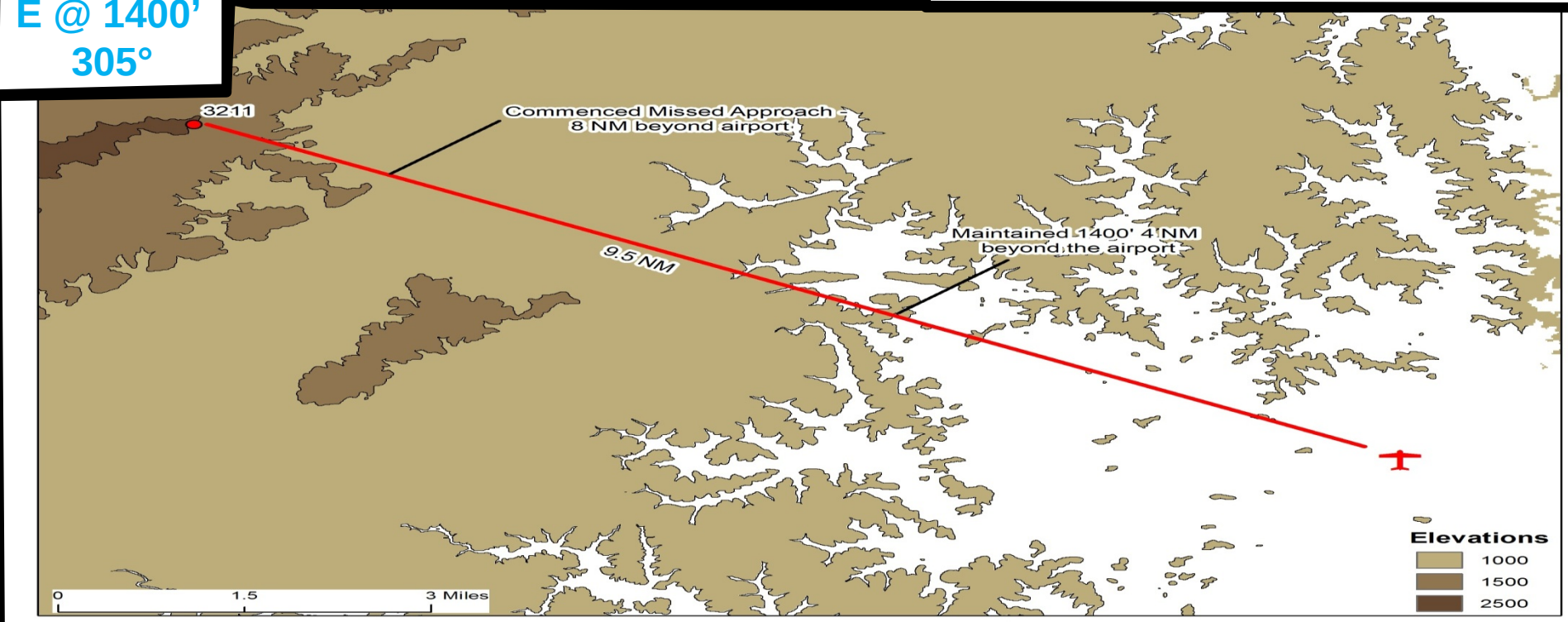





2800
BALES
VGSI and descent angles not coincident
(VGSI Angle 3.00/TCH 52).
One Minute
Holding Pattern
C @ 4000'
LOM
I-UVM 6
HOMGU
I-UVM
3.1
D @ 2600'
I-UVM
2.2
I-UVM
1
125°
2800
305°
305°
2600
1640
3.09°
TCH 40
1.2 NM
0.9 NM
2.9 NM


REQUIRED
PIGGS
3700
213°
(22.9)
1001
1165
1457
1726
B @ 4000'
1079±
1157
HOMGU
I-UVM
3.1
125°
1 min
305°
LOM/IAF
BALES
396 UV
I-UVM 6
2600 NoPT
305°
(11.8)
(IF/IAF)
ULAKE
I-UVM 17.8
125°
3100
036°
(8.8)
PLEXI
A

E @ 1400'
305°


3211
Commenced Missed Approach
8 NM beyond airport
9.5 NM
Maintained 1400' 4 NM
beyond the airport
Elevations
1000
1500
2500
0
1.5
3 Miles

Tool Objectives

- Improve IAP situational awareness
- Minimize interactions
  - User defines extent
  - ICAO Code and IAP name
- Let Python do the work…
  - Transform/import IAP
  - Clip elevation data
  - Mathematically calculate intervals
  - Manage permanent/intermediate data

Three Core Functions

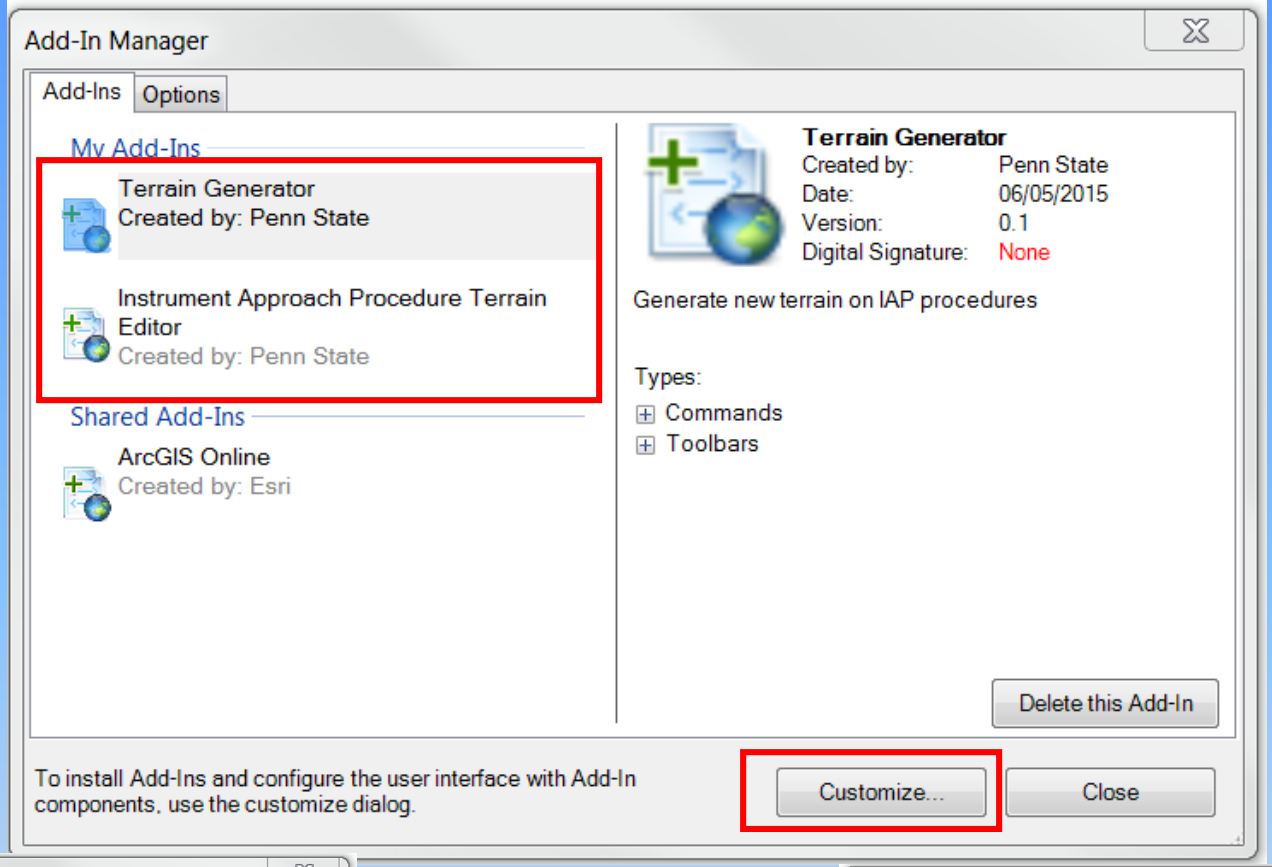
IAP Transformation

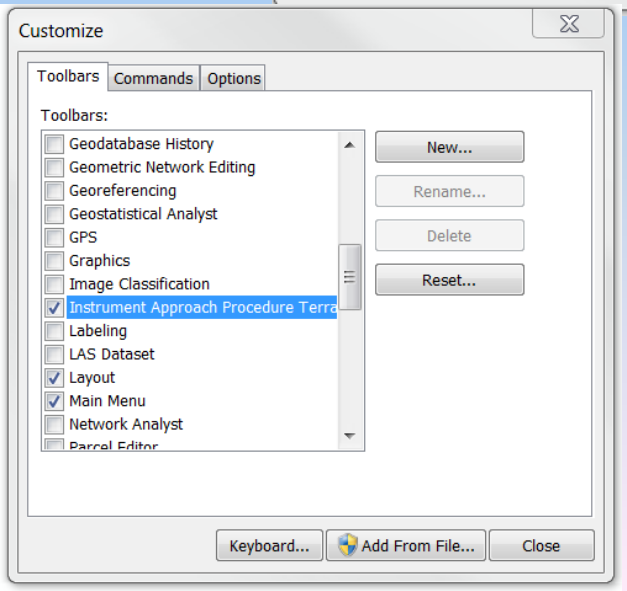
Terrain Manipulation

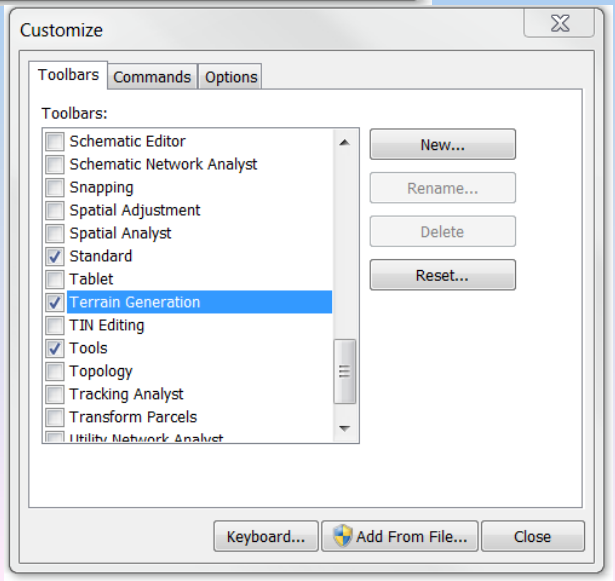
User Interactive Extents

File Geodatabase

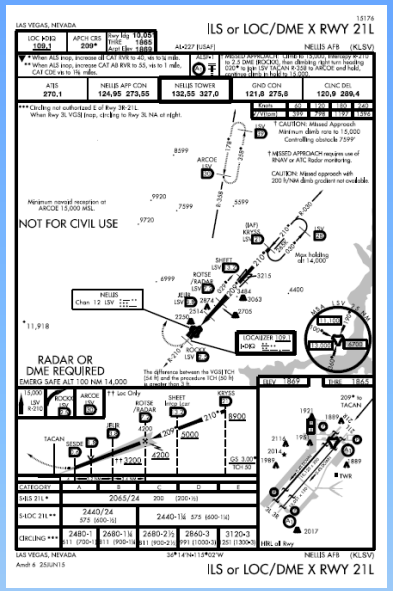
Data Source	Origination	Spatial Resolution	Spatial Reference	Format	Purpose
National Elevation Dataset	United States Geological Survey	1 arc second (approx 30 meters)	North America Datum 83 (NAD83) Geographic	GRID Float	Source to generate terrain shaded intervals
Global Airports	Global Hazards Information Network	N/A	World Geodetic System 1984 (WGS84)	Esri Shapefile	Source providing airfield elevation to mathmatically derive terrain shaded intervals


Add-In Manager
Add-Ins
Options
My Add-Ins
Terrain Generator
Created by: Penn State
Instrument Approach Procedure Terrain Editor
Created by: Penn State
Shared Add-Ins
ArcGIS Online
Created by: Esri
Terrain Generator
Created by: Penn State
Date: 06/05/2015
Version: 0.1
Digital Signature: None
Generate new terrain on IAP procedures
Types:
Commands
Toolbars
Delete this Add-In
To install Add-Ins and configure the user interface with Add-In components, use the customize dialog.
Customize...
Close


Customize
Toolbars
Commands
Options
Toolbars:
Geodatabase History
Geometric Network Editing
Georeferencing
Geostatistical Analyst
GPS
Graphics
Image Classification
Instrument Approach Procedure Terra
Labeling
LAS Dataset
Layout
Main Menu
Network Analyst
New...
Rename...
Delete
Reset...
Keyboard...
Add From File...
Close

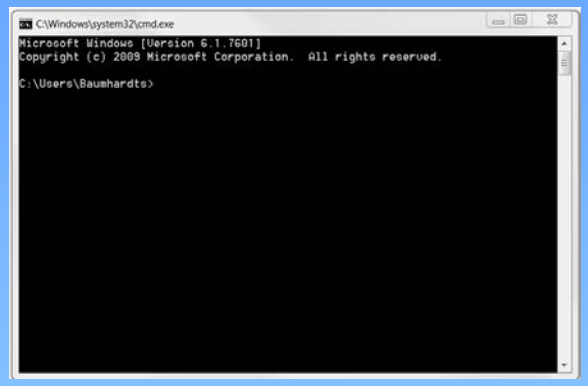

Customize
Toolbars
Commands
Options
Toolbars:
Schematic Editor
Schematic Network Analyst
Snapping
Spatial Adjustment
Spatial Analyst
Standard
Tablet
Terrain Generation
TIN Editing
Tools
Topology
Tracking Analyst
Transform Parcels
New...
Rename...
Delete
Reset...
Keyboard...
Add From File...
Close

IAP GeoPDF to GeoTIFF Import:



IAP

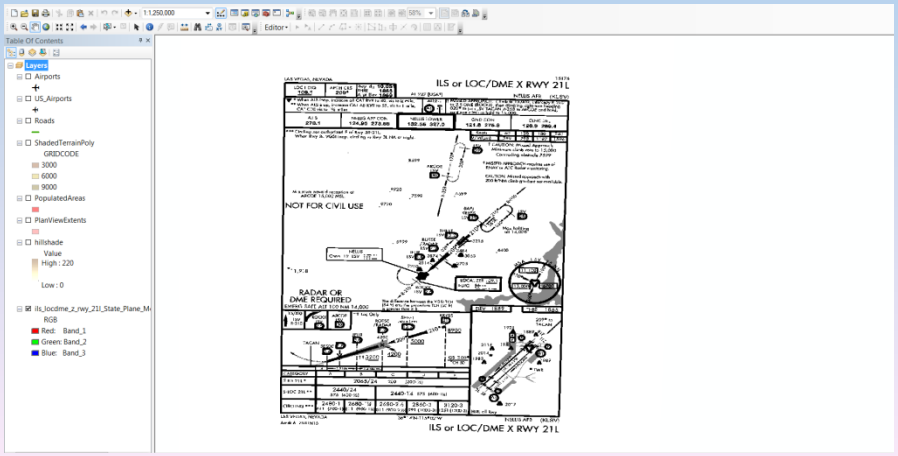
Raster
Translator



Python
Subprocess

GeoTIFF

- TerraGo functions costly
  - Required for Adobe
  - Required for ArcMap
- Open Source translator works well with numerous languages



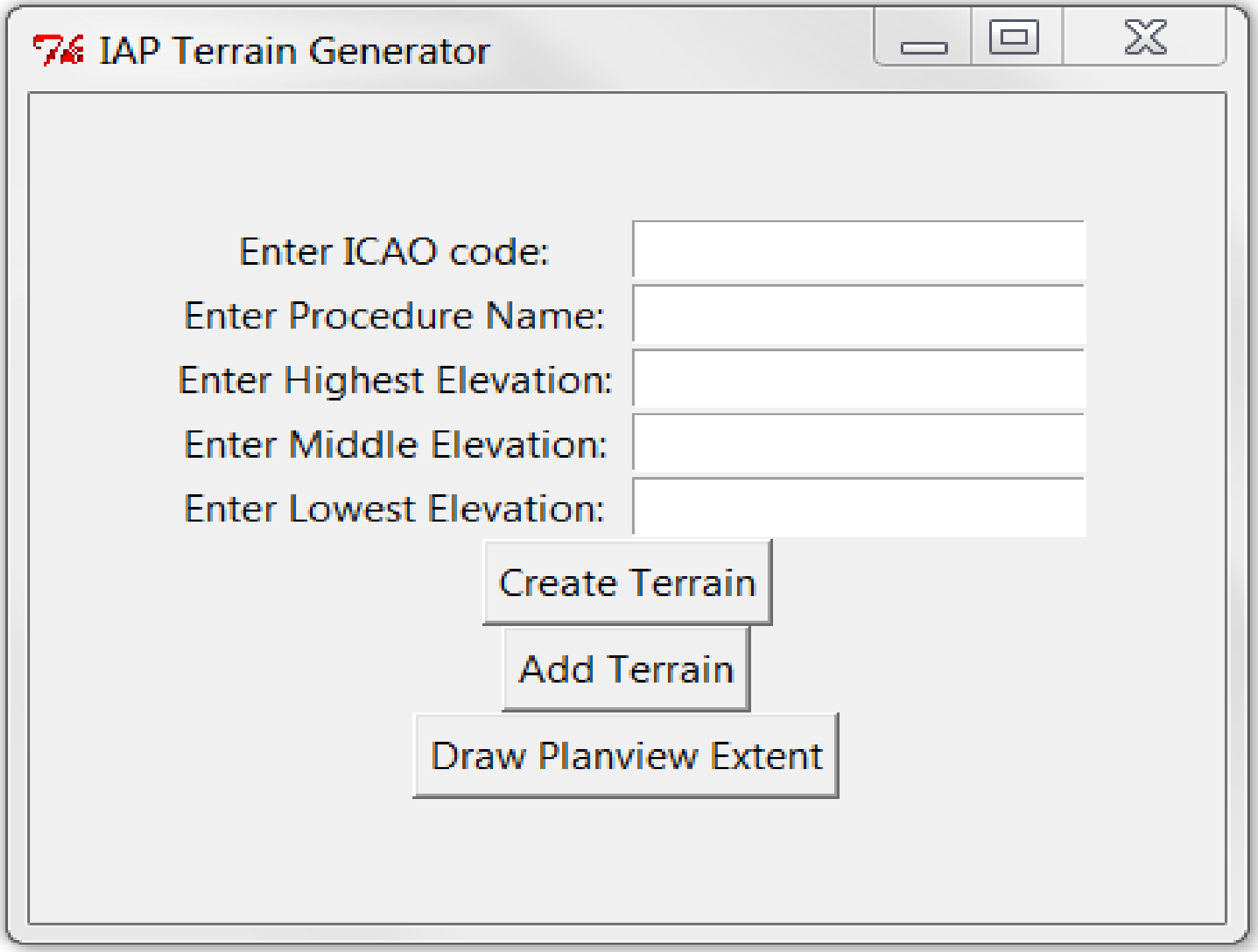
Terrain Generation

Import GeoPDF | ICAO Code | Procedure Name | Planview Extent | Terrain Generator

```
def __init__(self):
    self.enabled = True
    self.checked = False
def onClick(self):
    #Obtain the GeoPDF file to transform into a GeoTiff
    currentDir = os.getcwd()
    GeoPDFDir = currentDir + '\\GeoPDF'
    for fileName in os.listdir(GeoPDFDir):
        if fileName.endswith(".pdf"):
            inputFile = (fileName)
            #Define the parameters to pass to command line for GDAL
            sourceFile = currentDir + "\\GeoPDF\\" + inputFile
            outputFile = inputFile.split(".")[0]
            destinationFile = currentDir + "\\GeoPDF\\" + outputFile + ".tiff"
            #Run the GDAL command line
            subprocess.call(["gdal_translate", "-of", "GTiff", sourceFile, destinationFile])

            env.workspace = currentDir + '\\GeoPDF'

            #Make the raster a raster layer to be added to the dataframe through arcpy.mapping
            tempRaster = "rasterLayer"
            arcpy.MakeRasterLayer_management(destinationFile, tempRaster, "#", "", "#")

            #Add the transformed GeoTiff to the dataframe
            mxd = arcpy.mapping.MapDocument("Current")
            df = arcpy.mapping.ListDataFrames(mxd, "Layers")[0]
            addLayer = arcpy.mapping.Layer(tempRaster)
            arcpy.mapping.AddLayer(df, addLayer, "AUTO_ARRANGE")
            del mxd

    for dirpath, direname, files in os.walk(GeoPDFDir):
        if not files:
            pythonaddins.MessageBox("There is no PDF available in GeoPDF directory", "GeoPDF To GeoTiff")
```

Transforming with GDAL

Importing into MXD


IAP Terrain Generator
Enter ICAO code:
Enter Procedure Name:
Enter Highest Elevation:
Enter Middle Elevation:
Enter Lowest Elevation:
Create Terrain
Add Terrain
Draw Planview Extent

Terrain Generation

Import GeoPDF | ICAO Code | Procedure Name | Planview Extent | Terrain Generator

```
class EnterIcaoCode(object):
    """Implementation for AddNewTerrainPythonaddin_addin.IcaoCodeCbo (ComboBox)"""
    def __init__(self):
        self.items = []
        self.editable = True
        self.enabled = True
        self.dropdownWidth = 'WWWW'
        self.width = 'WWWW'

    def onEditChange(self, text):
        global icaoParameter
        icaoParameter = text
```

Global combobox work around

```
#-----------------------------------------------------------------------------
class EnterProcedureName(object):
    """Implementation for AddNewTerrainPythonaddin_addin.ProcedureNameCbo (ComboBox)"""
    def __init__(self):
        self.items = []
        self.editable = True
        self.enabled = True
        self.dropdownWidth = 'WWWWWWWWWWWWWWWWWW'
        self.width = 'WWWWWWWWWWWWWWWWWW'

    def onEditChange(self, text):
        global procNameParameter
        procNameParameter = text
```

Global combobox work around

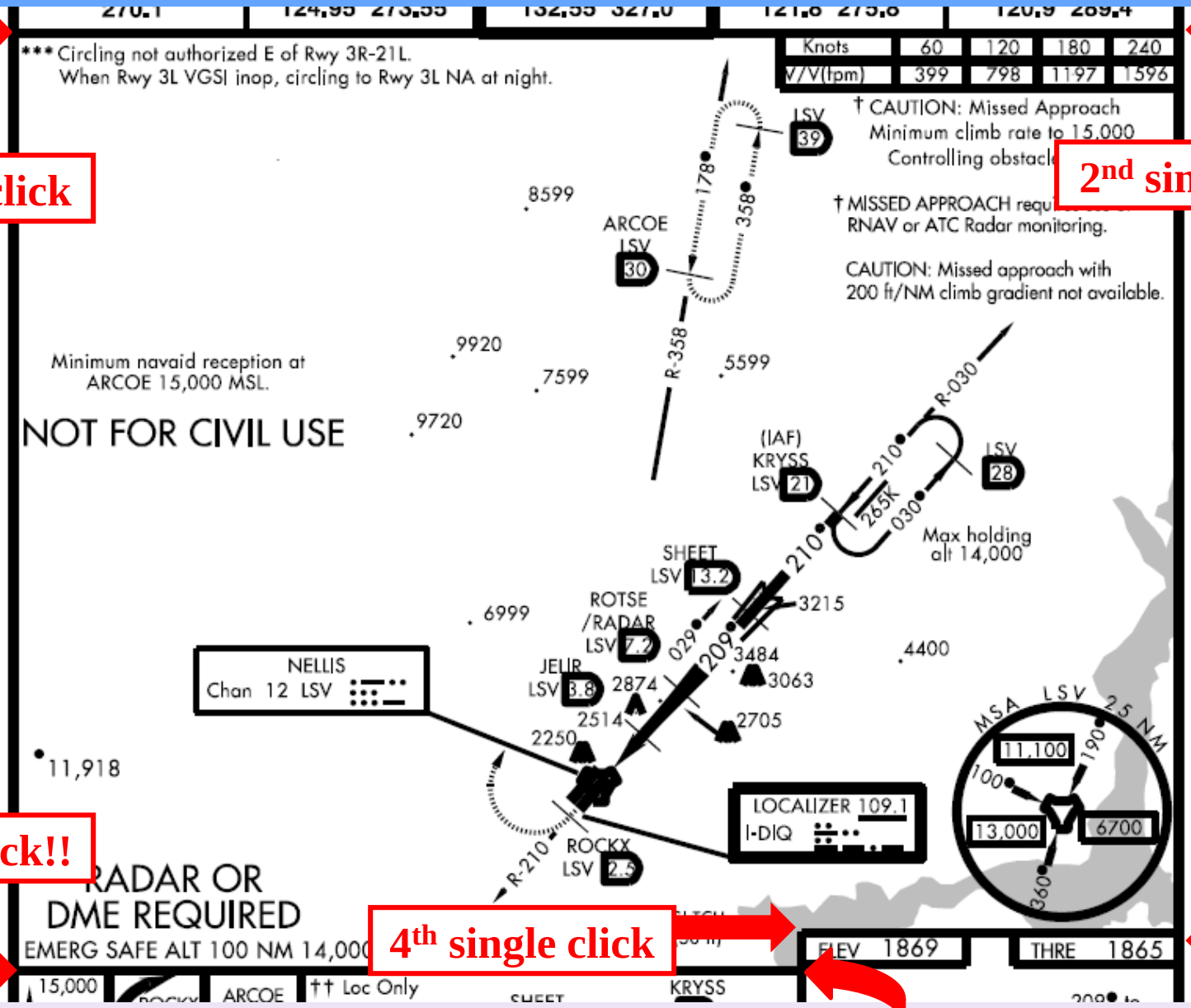
Terrain Generation

Import GeoPDF ICAO Code LSV | Procedure Name ILS OR LOC DME Z RWY 21L | Planview Extent | Terrain Generator

```
def __init__(self):
    self.enabled = True
    self.shape = "Line" # Can set to "Line", "Circle" or "Rectangle" for interactive shape drawing and to activate the
    self.cursor = 3
    self.width = 'WWWWWWWWW'

def onLine(self, line_geometry):
    #Check to see if ICAO Code parameter equals 3 characters, if not returns a message informing the user to change the
    if len(icaoParameter) == 3:
        array = arcpy.Array()
        part = line_geometry.getPart(0)
        for pt in  part:
            array.add(pt)
        array.add(line_geometry.firstPoint)
        closePoint = array[0]
        array.add(closePoint)

        #Define the feature class and insert cursor
        currentDir = os.getcwd()
        planViewExtents = currentDir + '\\Data\\TerrainGeneratorData.gdb\\PlanViewExtents'

        cursor = arcpy.da.InsertCursor(planViewExtents, ("SHAPE@", "ICAO", "PROC_NAME"))

        #Build the polygon extent from the array
        polygon = arcpy.Polygon(array)

        #Insert the polygon into the feature
        cursor.insertRow((polygon, icaoParameter, procNameParameter))

        del cursor

    else:
        pythonaddins.MessageBox("ICAO Code greater than 3 letters, enter in a 3 letter ICAO Code", "ICAO Code Error")
```

Assembly of points into array

Closes polygon via double click of mouse



1st single click

2nd single click

Double click!!

4th single click

5th single click

3rd single click

continue climb in hold to 15,000.

ATIS	NELLIS APP CON	NELLIS TOWER	GND CON	CLNC DEL
270.1	124.96 273.55	132.55 327.0	121.8 275.8	120.9 289.4

*** Circling not authorized E of Rwy 3R-21L.
When Rwy 3L VGSI inop, circling to Rwy 3L NA at night.

Knots	60	120	180	240
V/V(fpm)	399	798	1197	1596

† CAUTION: Missed Approach Minimum climb rate to 15,000 Controlling obstacle 7599'

† MISSED APPROACH requires use of RNAV or ATC Radar monitoring.

CAUTION: Missed approach with 200 ft/NM climb gradient not available

Minimum navaid reception at ARCOE 15,000 MSL.

NOT FOR CIVIL USE

LSV 39

ARCOF LSV 30

178° 358°

R-358

8599 9920 7599 5599 9720

(IAF) KRYSS LSV 21

210° 265° 030°

R-030

LSV 28

Max holding alt 14,000

SHEET LSV 3.2

210°

3215

ROTSE /RADAR LSV 2.2

029° 209°

3484 3063 4400 6999

IELIR LSV 3.8

2874 2514 2250 2705

NELLIS Chan 12 LSV

11,918

LOCALIZER 109.1 I-DIQ

ROCKX LSV 2.5

R-210

RADAR OR DME REQUIRED

EMERG SAFE ALT 100 NM 14,000

The difference between the VGSI TCH (54 ft) and the procedure TCH (50 ft) is greater than 3 ft

MSA LSV 25 NM

11,100 190° 100° 13,000 6700 260°

15,000 LSV R-210

ROCKX LSV 2.5

ARCOF LSV 30

†† Loc Only

ROTSE /RADAR 2.2

SHEET Intcp locr 13.2

KRYSS 21

210° 8900

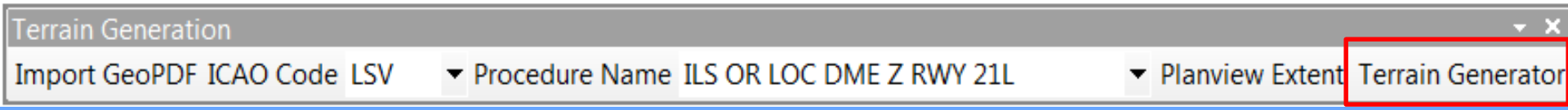
JELIR

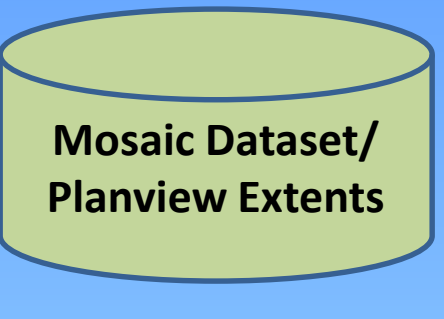
ELEV 1869

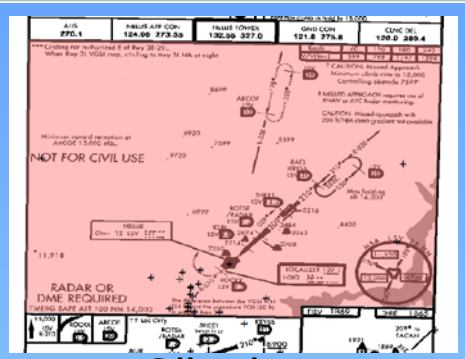
THRE 1865

209° to TACAN

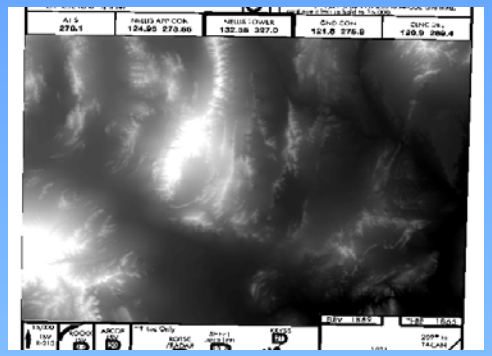
1921 1889

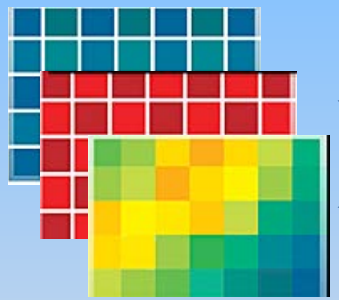




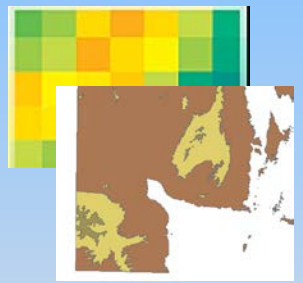


Clipping

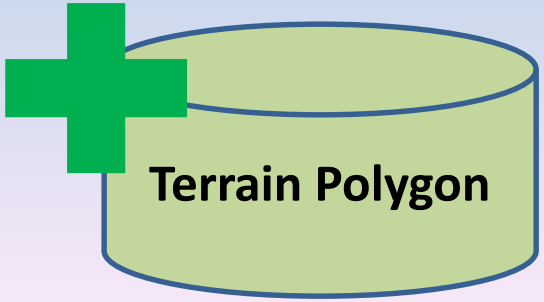


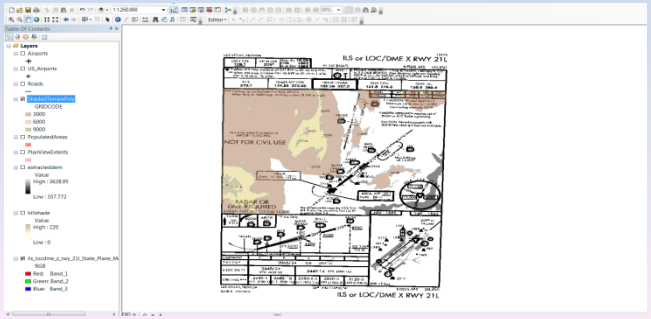


```
#Extracting the field elevation of the airport
airports = currentDir + "\\Data\\TerrainGeneratorData.gdb\\US_Airports"
locIDField = "LOCID"
elevField = "ELEV"
airfieldCursor = arcpy.da.SearchCursor(airports, elevField, '"' + locIDField +
for row in airfieldCursor:
    fieldElevVal = int(row[0])
    print "Extracted field elevation is : " + str(fieldElevVal)
    #Defining the terrain interval range
lowElevBand = int(math.ceil(fieldElevVal/1000.0))*1000.0 + 1000.0
diffLowElevMaxElevCalc = int(rasterElevMaxVal - lowElevBand)
bandIntervalCalc = int(round(diffLowElevMaxElevCalc/3,-3))
midElevBand = lowElevBand + bandIntervalCalc
topElevBand = midElevBand + bandIntervalCalc
print "High elevation interval is: " + str(topElevBand) + "ft"
print "Mid elevation interval is: " + str(midElevBand) + "ft"
print "Low elevation interval is: " + str(lowElevBand) + "ft"
    #Define remapping for terrain intervals
terrainRemap = RemapRange([[lowElevBand, midElevBand, int(lowElevBand)], [midEl
```

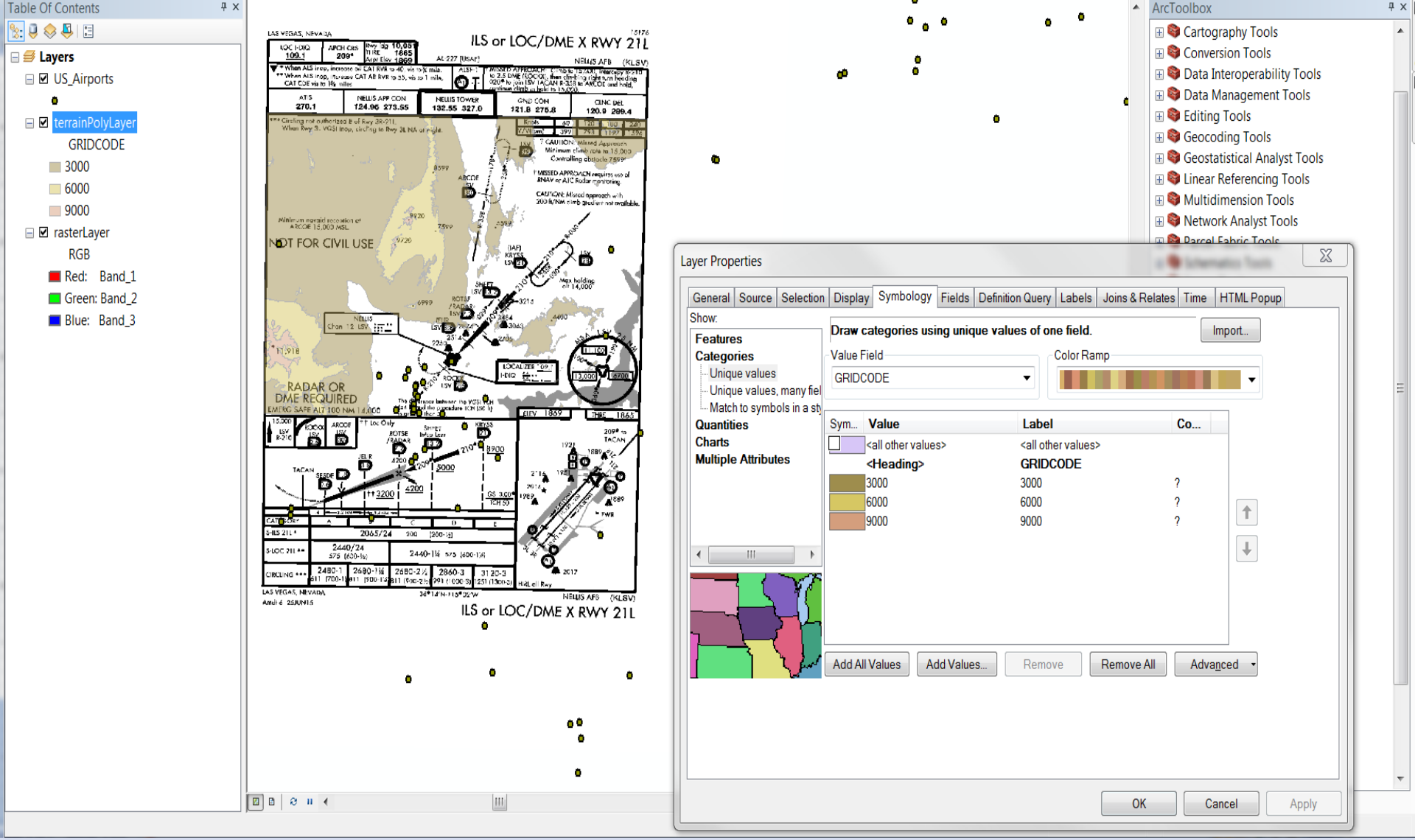


Mathematical Reclassification Remap

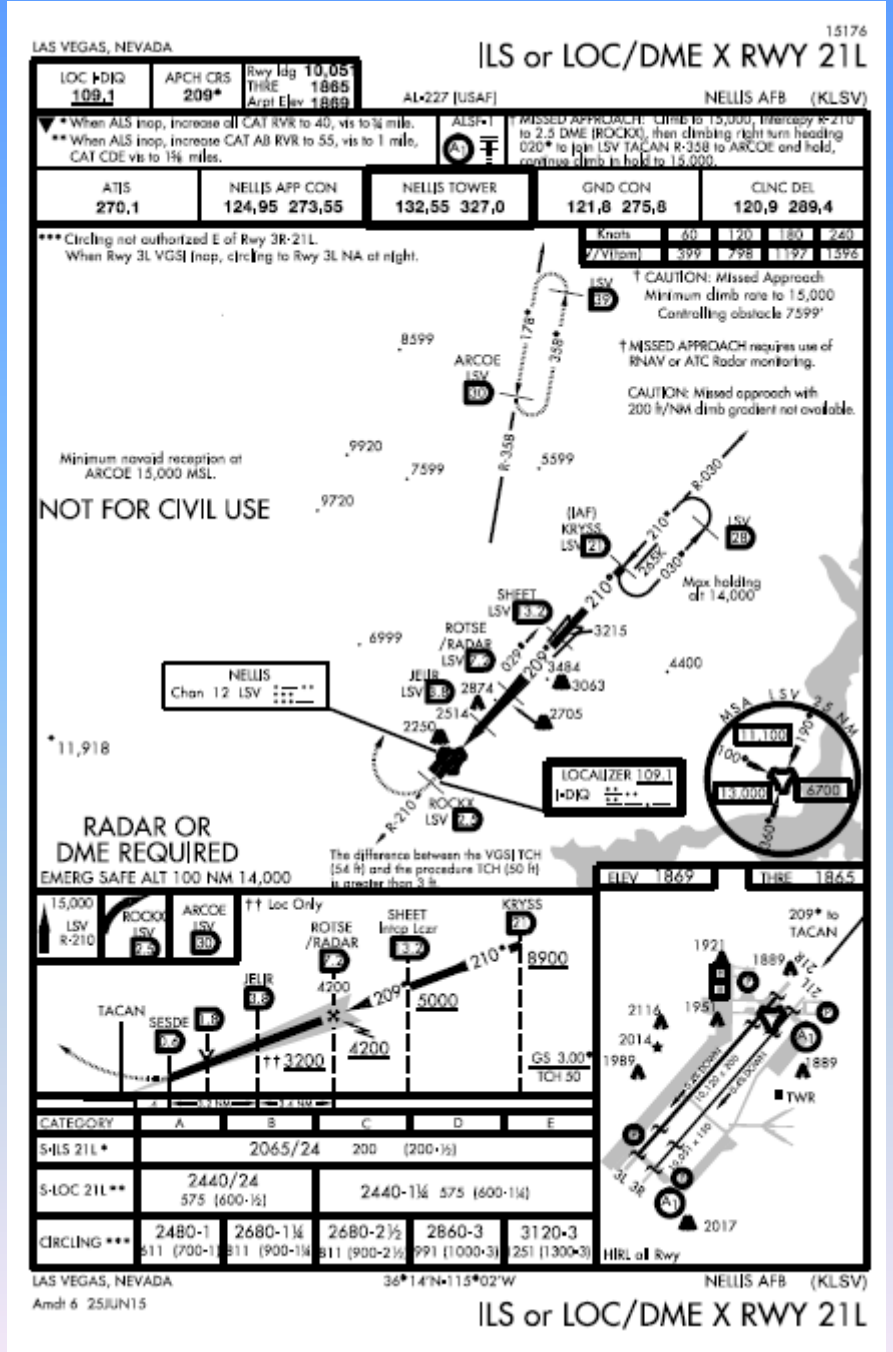




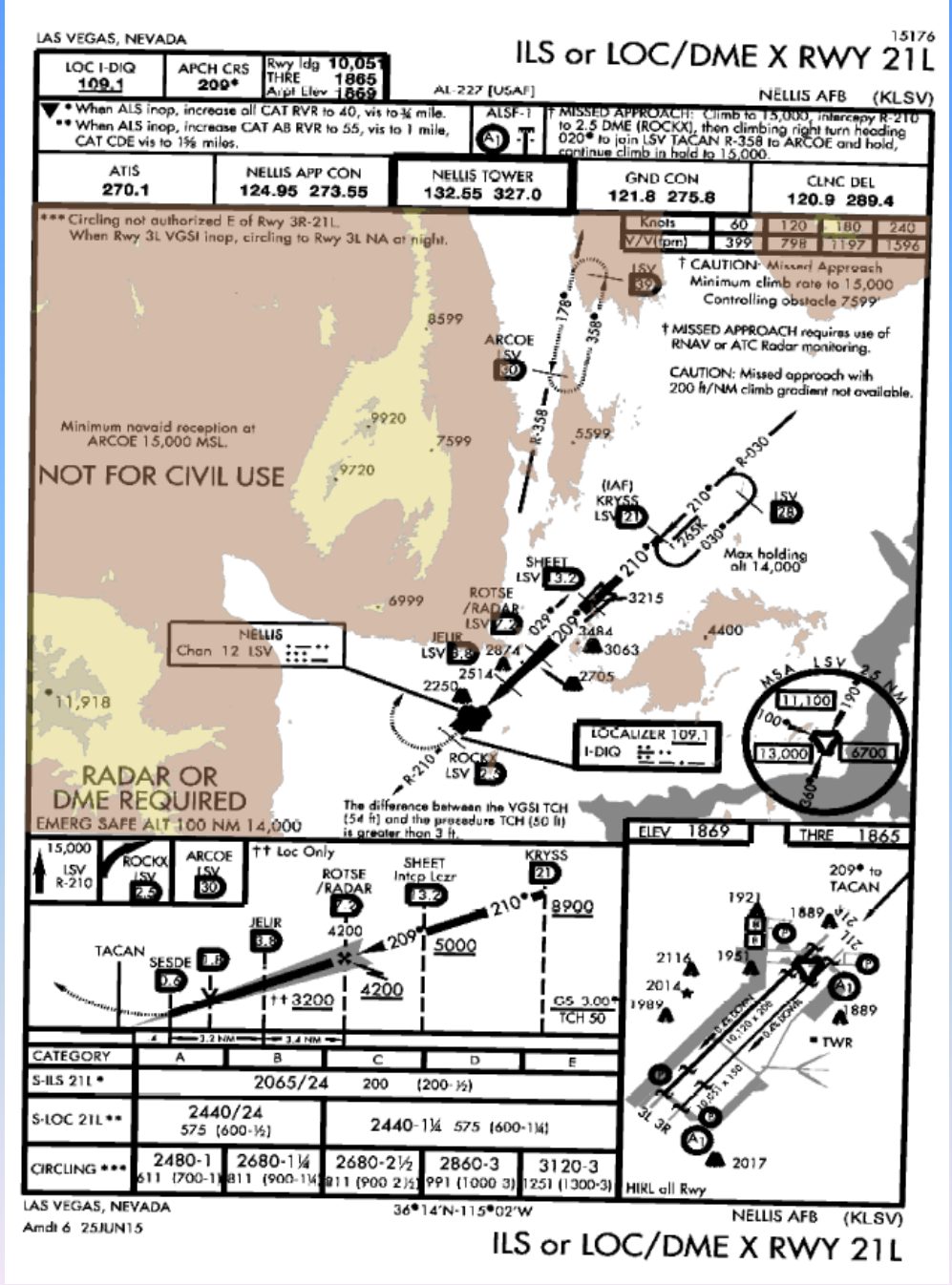
Field Calculations



- Symbolize with standard layer properties
- Export map to PDF



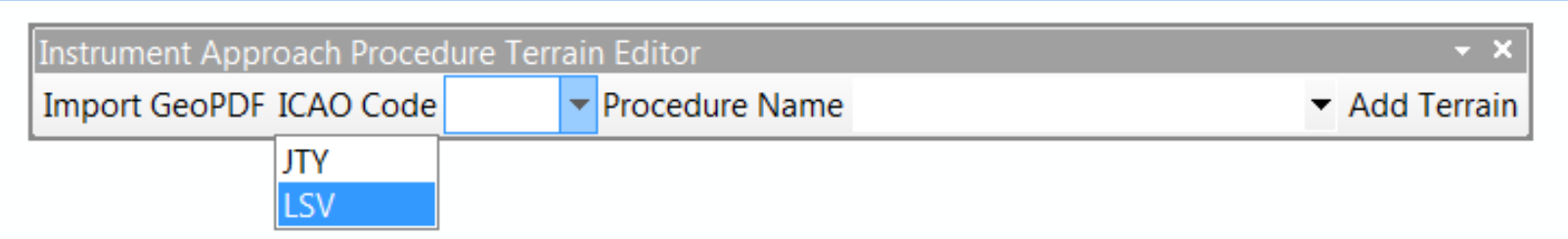
BEFORE

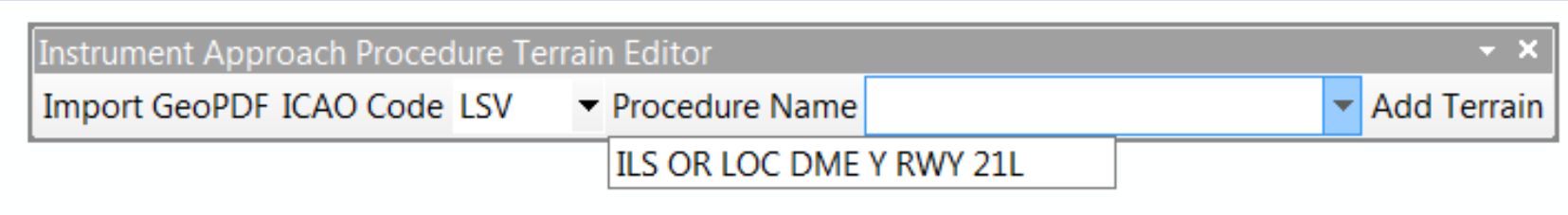


AFTER

IAP Terrain Editor

- Quickly update revised IAPs
  - Re-import procedure
  - Pre-populated combo boxes
  - Retrieves stored terrain





Future Enhancements:

- Consumption within electronic flight bags
- Managed within web service
  - User accounts
  - Independent of unique software
- Topographical additions:
  - Transportation
  - Populated Areas
  - Landcover

LAS VEGAS, NEVADA

15176

ILS or LOC/DME X RWY 21L

LOC I-DIQ	APCH CRS	Rwy ldg / THRE / Arpt Elev
109.1	209°	10,051 / 1865 / 1869

AL-227 [USAF]

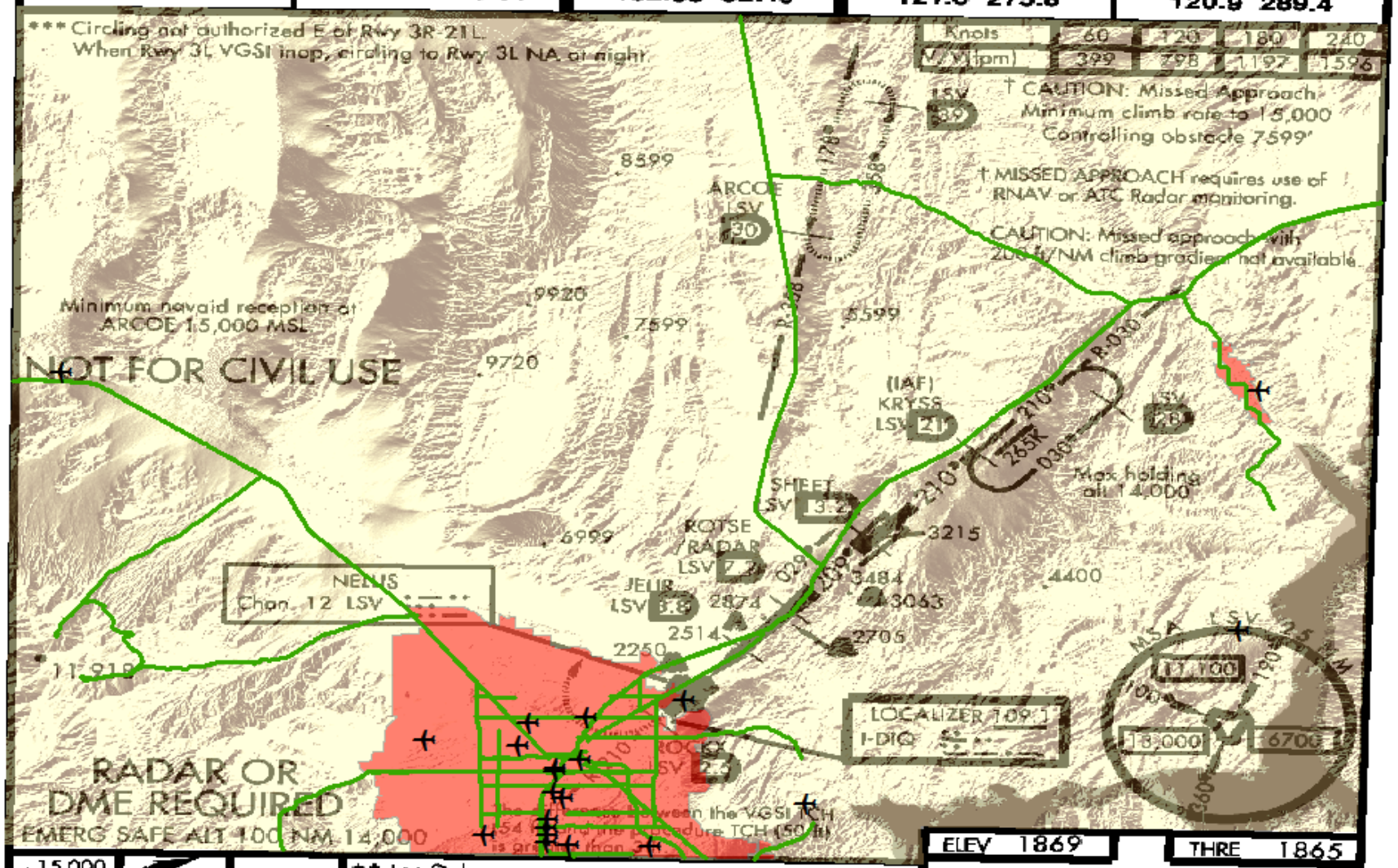
NELLIS AFB (KLSV)

▼ * When ALS inop, increase all CAT RVR to 40, vis to ¾ mile.
** When ALS inop, increase CAT AB RVR to 55, vis to 1 mile, CAT CDE vis to 1⅝ miles.

ALSF-1

† MISSED APPROACH: Climb to 15,000, intercept R-210 to 2.5 DME (ROCKX), then climbing right turn heading 020° to join LSV TACAN R-358 to ARCOE and hold, continue climb in hold to 15,000.

ATIS	NELLIS APP CON	NELLIS TOWER	GND CON	CLNC DEL
270.1	124.95 273.55	132.55 327.0	121.8 275.8	120.9 289.4



Summary:

- Performance based navigation will skirt the landscape
- Personalizes navigational situational awareness
- Alternative to expensive avionic systems