




ChatGPT x GIS

Exploring Conversational GIS Workflows

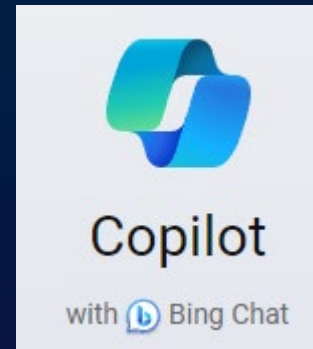
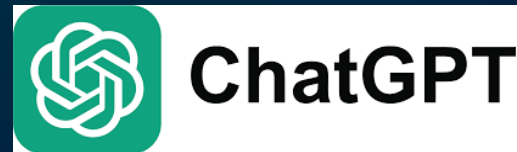
Zach Robinson | Sr GIS Analyst, County of Santa Clara Planning & Development

Steve Borgstrom | GIS Analyst, County of Santa Clara Planning & Development

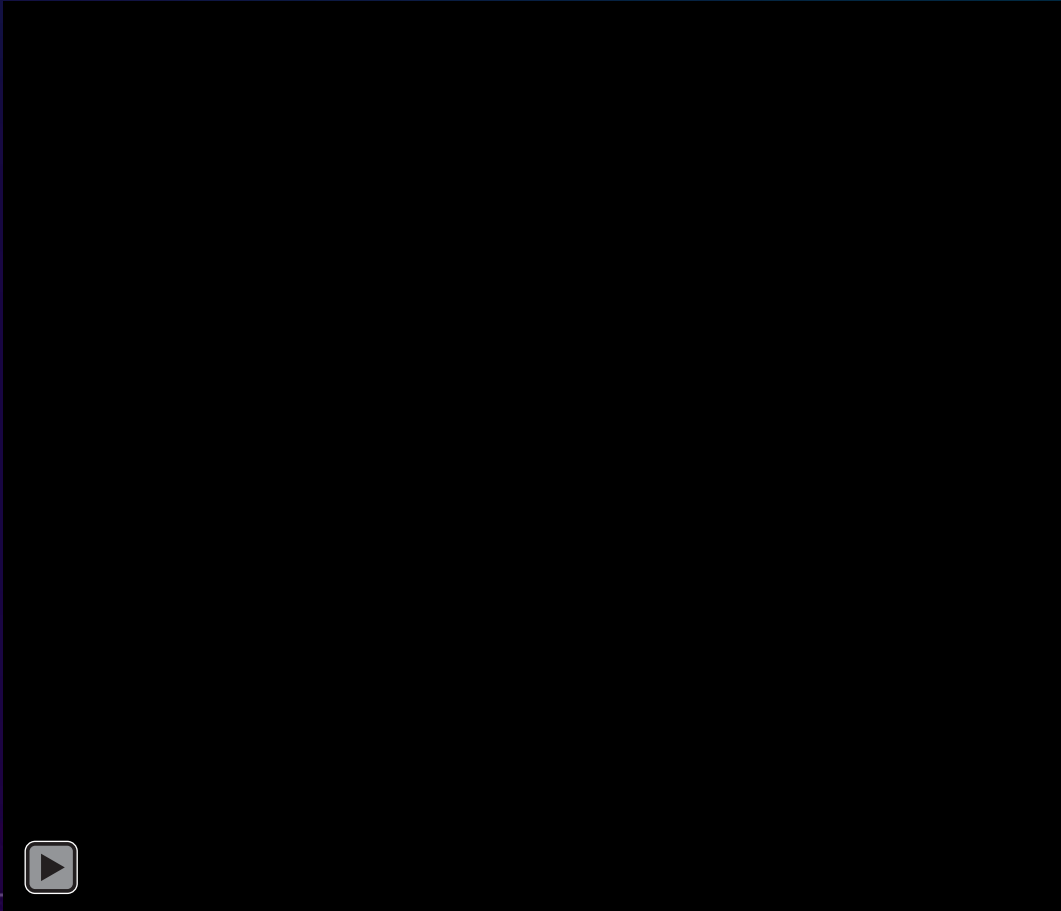


Conversational AI

- Not new – just an evolution
 - Early 90s chatbots
 - Voice assistants (Siri)
 - Customer Support chatbots
- Machine Learning and Natural Language Processing
 - Contextual understanding
 - Human-like responses
 - Learning on the fly
- Handful of major players today although most are familiar with ChatGPT



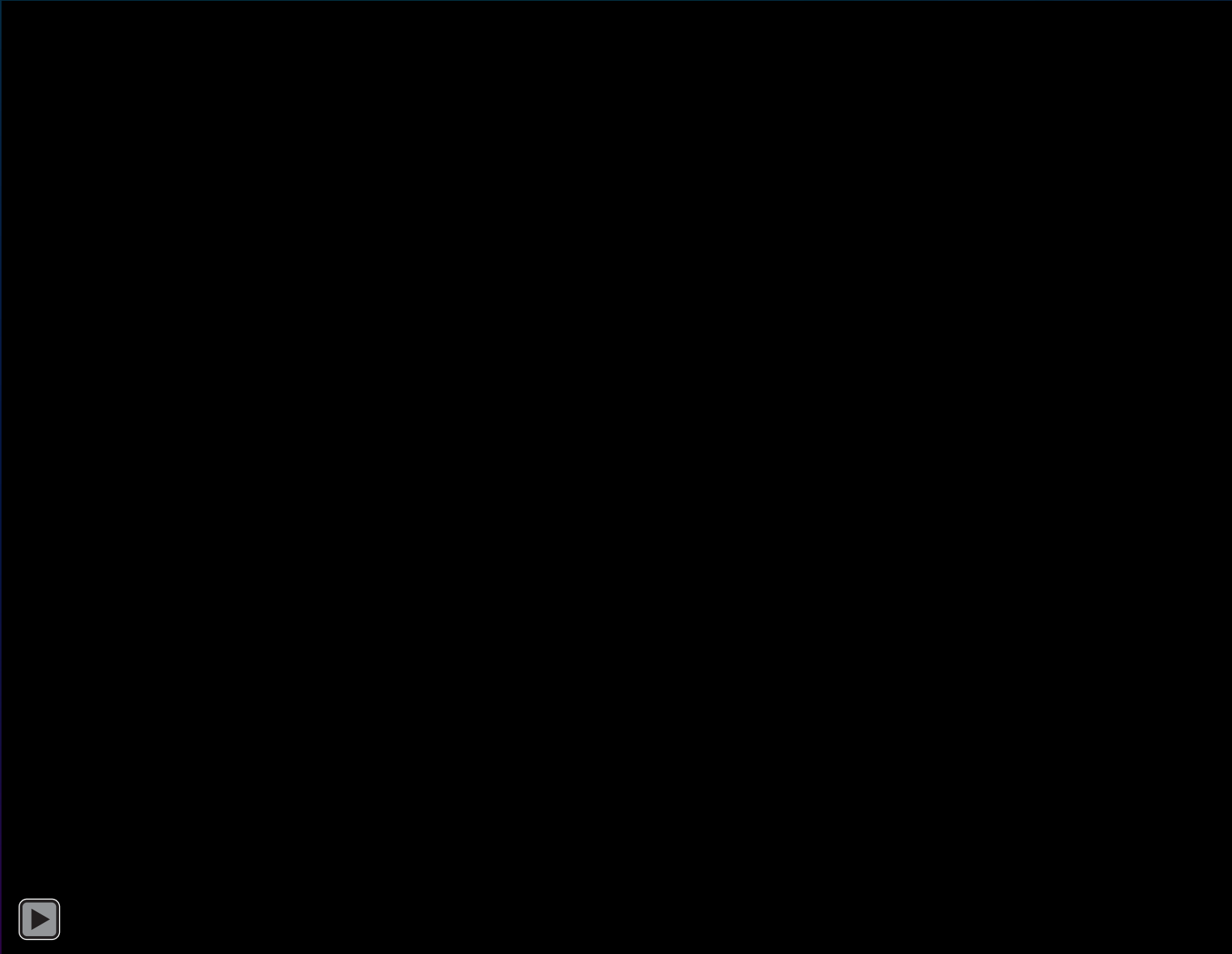
ChatGPT



- Launched Nov 2022
- Fastest growing software company in 2023
- Use cases
 - Research content
 - Code/debug programs
 - Compose music, write content

Potential Use Cases for GIS

Understand
Concepts



Potential Use Cases for GIS

Understand
Concepts

Suggesting
Workflows



Potential Use Cases for GIS

Understand
Concepts

Suggesting
Workflows

Troubleshoot
Issues



Potential Use Cases for GIS

Understand
Concepts

Suggesting
Workflows

Troubleshoot
Issues

Writing &
Debugging
Code



DEMOS



Address Correction and Standardization



Please correct and verify the following addresses, delivering the results in CSV format. Add information where necessary



Here are the corrected and verified addresses in CSV format:

plaintext

Copy code

```
Address, City, State, ZIP Code
773 NEWELL RD, PALO ALTO, CA, 94303-2850
392 WAVERLEY ST, SUNNYVALE, CA, 94086-6055
1367 RAMON DR, SUNNYVALE, CA, 94087
181 RADFORD DR, CAMPBELL, CA, 95008
722 PURITAN CT, SAN JOSE, CA, 95123-4641
370 RUCKER AVE, MORGAN HILL, CA, 95020-9251
```

Coordinate Conversion



Convert the following coordinates into decimal degrees.
Deliver the results as CSV

34° 04' 25" N, 118° 14' 39" W
37° 46' 30" N, 122° 25' 11" W
33° 45' 02" N, 118° 23' 40" W
36° 36' 43" N, 121° 54' 35" W
38° 25' 13" N, 122° 42' 59" W
34° 24' 56" N, 119° 42' 01" W
40° 44' 25" N, 124° 05' 29" W
32° 43' 18" N, 117° 09' 24" W
35° 17' 47" N, 120° 39' 33" W
39° 16' 18" N, 120° 40' 38" W



Here are the given coordinates converted into decimal degrees and delivered in CSV format:

Latitude,Longitude
34.0736,-118.2442
37.7750,-122.4197
33.7506,-118.3945
36.6120,-121.9097
38.4203,-122.7164
34.4156,-119.7003
40.7403,-124.0914
32.7217,-117.1567
35.2964,-120.6592
39.2717,-120.6772

Symbology: Colors



I need a palette of five colors, preferably pastels, that maximize the differences between each. These colors are to be used for symbology on a map. Deliver the colors with names and hex codes.

1. Lavender Blush

• Hex Code: #F7E7F7



2. Mint Green

• Hex Code: #A1E09E



3. Peach Puff

• Hex Code: #FFDAB9



4. Sky Blue

• Hex Code: #87CEEB



5. Lilac

• Hex Code: #C8A2C8



*“Great job.
Can I have
another
palette?”*

1. Rose Quartz

• Hex Code: #F7CAC9



2. Aqua

• Hex Code: #00FFFF



3. Lemon Chiffon

• Hex Code: #FFFACD



4. Lavender

• Hex Code: #E6E6FA



5. Pale Green

• Hex Code: #98FB98



Symbology: Colors



I need a palette of six colors for use in symbolizing features on a map. They should be easy for color blind readers to differentiate.



Creating a palette that is easily distinguishable for color-blind readers is essential for inclusive design. Here's a palette of six colors that are distinct and should be easy to differentiate for people with color vision deficiencies:



1. **Blue**

• Hex Code: #3498DB



2. **Green**

• Hex Code: #2ECC71



3. **Orange**

• Hex Code: #E67E22



4. **Purple**

• Hex Code: #9B59B6



5. **Red**

• Hex Code: #E74C3C



6. **Teal**

• Hex Code: #1ABC9C



This palette includes a variety of colors, and their differences in hue and lightness should make them easily distinguishable for color-blind readers.

Symbology: Colors

```
import arcpy
import random

# Get the current map and the "Cities" layer
aprx = arcpy.mp.ArcGISProject("CURRENT")
map = aprx.listMaps()[0] # Assuming the first map in the project
cities_layer = map.listLayers("Cities")[0] # Assuming "Cities" is the name of the layer

# Get a list of unique values in the "NAME" field
unique_values = set(row[0] for row in arcpy.da.SearchCursor(cities_layer, "NAME"))

# Create a symbol dictionary with random fill colors and black outline
symbol_dict = {}
for value in unique_values:
    symbol = arcpy.Symbology.Symbol()
    symbol.color = [random.random(), random.random(), random.random(), 1] # Random RGBA color
    symbol.outlineColor = [0, 0, 0, 1] # Black outline
    symbol.outlineWidth = 0.5 # 0.5 points outline width
    symbol_dict[value] = symbol

# Apply symbology to the "Cities" layer
arcpy.management.ApplySymbologyFromLayer(cities_layer, "Path\\to\\your\\LayerFile.lyrx") # Update the layer file path

# Update the layer's symbology with the symbol dictionary
for value in unique_values:
    cities_layer.symbology.setRenderer("UniqueValueRenderer")
    cities_layer.symbology.addUniqueValueInfo(value, "NAME", symbol_dict[value])

# Refresh the display
arcpy.RefreshActiveView()
```

Limitations & Considerations

- Can and DOES make mistakes
 - Easy to interpret as fact
 - Overconfident
- Should only be used as a supplement, never a replacement
- Ethical Considerations
 - Current discussion in many industries
 - School/employer restrictions
- Privacy Considerations
- Free ChatGPT (v3.5) limited to knowledge up to January 2022