3D GIS

3D Symbology and Labeling

Symbology is the visual representation of geographic data using colors, shapes, sizes, textures, and other visual variables. In 3D visualization, symbology becomes significantly more complex and powerful, allowing us to represent multiple dimensions of data simultaneously. This module covers how to symbolize terrain data, vector features, point clouds, and voxel data in 3D space. We'll explore both realistic and thematic visualization approaches, procedural generation techniques, and best practices for 3D labeling. Understanding effective 3D symbology is essential for creating compelling and interpretable geographic visualizations.

Module Topics

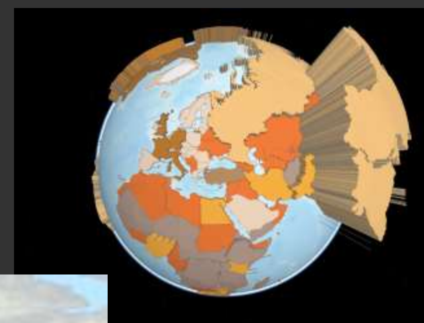
1. LODs
2. Terrain visualization
3. 2D vector data in 3D
4. Extrusion
5. 3D symbols for points, lines, and polygons
6. Picture fill
7. UV mapping
8. Procedural generation
9. Point cloud symbology and visualization
10. Voxel and multidimensional raster visualization
11. Labeling in 3D

2

These are the topics covered in this module.

Realistic -- Mixed Reality -- Thematic

- Symbology can very from:
  - Fully photorealistic
  - A mix of photorealistic and thematic
  - Fully thematic
- Best option depends on use case



3

Symbology approaches range along a spectrum from fully photorealistic to fully thematic.

Fully photorealistic visualization aims to depict the real world as accurately as possible, using realistic textures, lighting, and colors. This approach is excellent for communication and public engagement but can be computationally expensive and may obscure thematic data patterns.

Fully thematic visualization uses abstract colors and symbols to represent data values, following cartographic conventions. This approach is excellent for scientific analysis and pattern recognition but may appear less intuitive to general audiences.

Effective visualizations often use a mix of photorealistic and thematic elements, balancing visual appeal with data clarity. The best option depends on your specific use case, audience, and intended message.

LODs

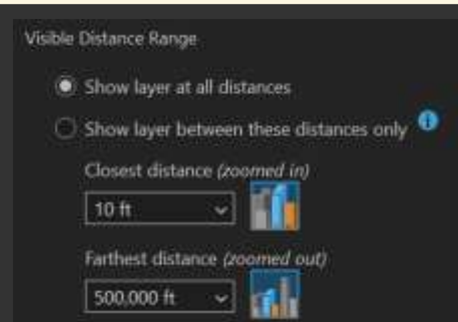
- Levels of Detail
- Different version of the same data used at different scales
- Zoom Out = Lower polygon count/Less detailed mesh
- Zoom In = Higher polygon count/More detailed mesh

4

Levels of detail, or LODs, are different versions of the same geographic data used at different scales. As you zoom out, lower polygon count and less detailed mesh versions are displayed for performance. As you zoom in, higher polygon count and more detailed mesh versions are displayed for visual quality. This approach balances performance and visual quality, allowing fast interaction at all scales. LODs are particularly important in 3D visualization where rendering complex geometry can be computationally expensive.

Visibility Scales/Distance-Based Rendering

- ❖Can display data at only certain ranges of scales
- ❖Can display data at only certain ranges of distances
- ❖Can use different symbols or LODs at different scales



5

Visibility can be controlled at different scales and/or distances. Data can be displayed only at certain ranges of scales or distances from the camera. Different symbols or LODs can be used at different scales. For example, buildings might appear as simple boxes when zoomed out but as detailed textured models when zoomed in. This visibility scaling improves both performance and clarity, preventing clutter and allowing progressive disclosure of information.

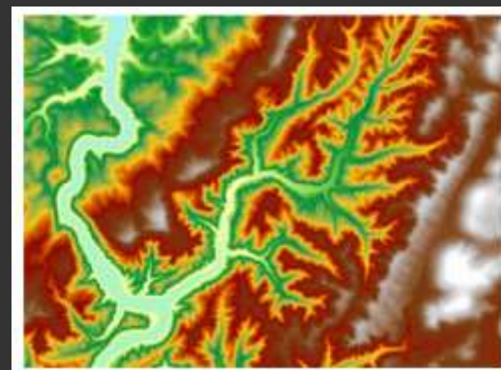
Terrain Visualization

6

Terrain visualization encompasses a variety of techniques for representing elevation data in 3D space. The following slides explore multiple approaches for symbolizing digital elevation models, each with different strengths and appropriate use cases.

Symbolizing DEMs

- Can use light-to-dark sequence that also transitions through related hues to provide more contrast
- Can be binned or stretched
- Light = low elevation
- Dark = high elevation
- Make sure DEM is of adequate resolution for map
  - May need to find better data
  - May need to smooth or generalize high spatial resolution data



7

Raster-based digital elevation models (DEMs) or digital terrain models (DTMs) provide a representation of the terrain surface where each cell holds an elevation measurement.

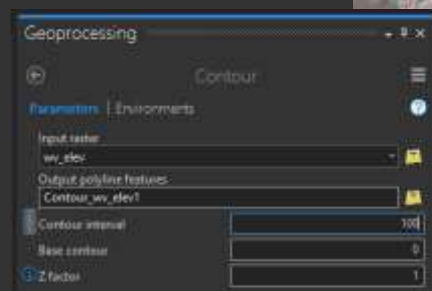
When selecting a color scheme for elevation data, it is generally recommended to use a light-to-dark sequence that also transitions through related hues to enhance contrast. A color ramp can be applied using a classification, also known as binning, or mapped continuously as a stretch.

It is recommended that lighter shades be used to show the lower elevations while darker shades are used to show the higher elevations when using a single hue and when not using a black-to-white stretch. When using black-to-white, low values should map to black and high values should map to white.

You should also consider the spatial resolution of the DEM. A coarse DEM may not be appropriate for a large scale map. On the other hand, detailed DEM data, such as those derived from LiDAR, may be too detailed and may require some smoothing or generalization.

Contours

- Raster → vector
- Must define an interval
- Make principal or index contours thicker
- Make intermediate contours thinner
- May require some smoothing

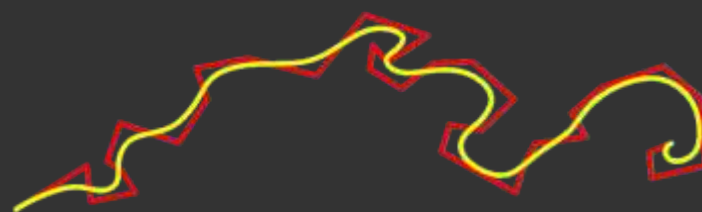


8

Contours, lines of constant elevation, can represent either raster or vector data. When creating contours, define an appropriate interval based on the terrain variability and desired detail. Make principal or index contours thicker and more prominent than intermediate contours to create visual hierarchy. Contours often require smoothing to create natural-looking lines. Combining contours with other elevation representations creates more informative visualizations.

Line Generalization

- Remove unnecessary vertices
- Smooth curves
- For cartographic generalization
- Simplify Line Tool
- Smooth Line Tool



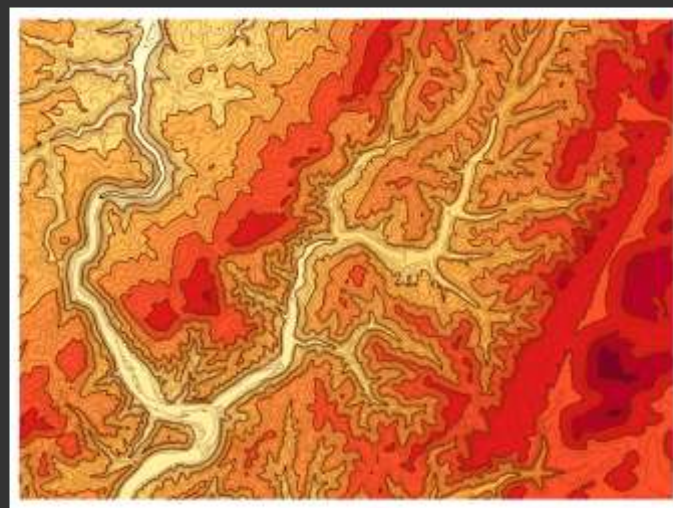
Available under Generalization Tools in the Cartography Toolbox

9

Line generalization is important for creating cartographically appropriate contour lines. Use the Simplify Line Tool to remove unnecessary vertices and simplify overly detailed contours. Use the Smooth Line Tool to smooth curves and create more aesthetic lines. These tools are available under Generalization Tools in the Cartography Toolbox and help create publication-quality contour maps. You will likely need to experiment with the available settings to obtain the desired product or level of generalization.

Combining DEMs and Contours

- Map color transitions with principal contours



10

It is common to combine contour lines and hypsometric tinting from a DEM to visualize the terrain surface. When these representations are combined, it is generally recommended that the color breaks align with the principal contours so that they reinforce one another. This makes for a very nice visualization, as demonstrated in the provided example.

Hillshade

- A visual representation of a terrain using shadows
- Shadows are cast over the landscape to created a shaded relief surface
- Azimuth = compass bearing of Sun
- Altitude = height of Sun above horizon (as an angle)
- Color ramps must run light to dark

11

Hillshades offer a means to visualize digital terrain data. In a hillshade, a sun or illuminating source is placed in the sky to model illumination. The result is a shaded relief surface. Bright areas represent areas of light exposure whereas dark areas have less light exposure.

The position of the illuminating source is defined using a compass bearing or direction, this setting is known as azimuth, and the angle of the source above the horizon, or altitude. Using azimuth and altitude, you can position the illuminating source anywhere in the local sky.

Imagine a dome over your head. If you would like to locate an object on this dome, you would need to specify the compass direction or bearing in which to look, or azimuth, and how high up to look as an angle, or altitude.

Sun position matters

Azimuth = 315, Altitude = 45

Azimuth = 135, Altitude = 45

12

The location of the illuminating source impacts the effectiveness and look of the hillshade. In ArcGIS Pro, the default settings are an azimuth of 315 and an altitude of 45. So, the illuminating source would be in a northwest position and halfway between the horizon and zenith, or top of the sky.

Since we generally see features illuminated from above or by the Sun, we expect shadows to be cast downward. This is mimicked with a Sun position in the northwest azimuth. If we move the Sun to a southeast azimuth, then features will be illuminated from below. This will result in a strange optical illusion where the topography looks inverted: ridges will be interpreted as valleys and valleys as ridges.

So, we tend to avoid using southeast azimuth positions.

Sun position matters

Azimuth = 315, Altitude = 45

Azimuth = 315, Altitude = 10

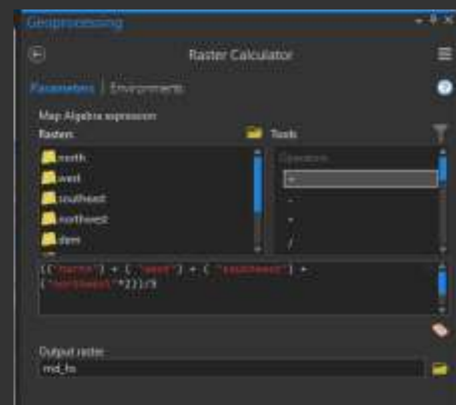
13

We also tend to avoid using low altitudes as this can create very contrasted hillshade images with very dark and bright slopes. Placing the illumination source very close to the zenith can make hillshade images look very "washed out" or low contrast.

I have generally found the default settings to work well. However, you may want to experiment with different settings. Different settings may be optimal in different terrains.

Multidirectional Hillshades

- Combine multiple Sun positions
- North + Northwest + West + Southeast
- Extra weight for northwest
- Subtle detail on slopes that face different directions
- Useful in subtle terrain



14

Another option is to calculate multiple hillshades using different illuminating positions and then combine them to create a multidirectional hillshade. In the example, I have created hillshades using north, west, southeast, and northwest positions. I then use Raster Calculator to average them. I double-weight the northwestern result so that it has the largest impact on the output.

Using multiple directions generally allows for a reduction in deep shadows or extremely bright cells in complex topography. If a hillshade using the default illuminating position or a single position is not adequate for your visualization, it is worth experimenting with multidirectional hillshades as an alternative.

Multidirectional Hillshades

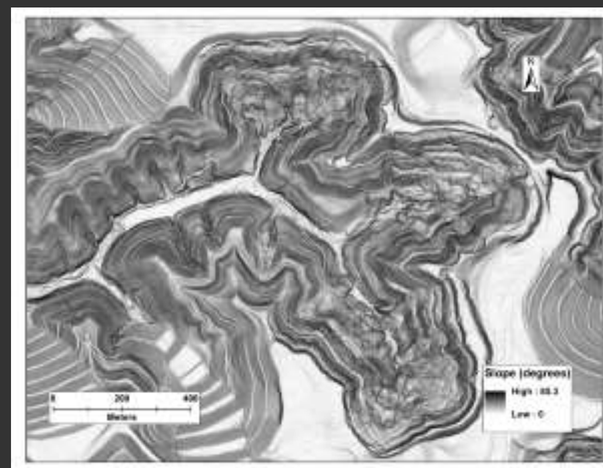
- Combine multiple Sun positions
- North + Northwest + West + Southeast
- Extra weight for northwest
- Subtle detail on slopes that face different directions
- Useful in subtle terrain

15

This slide shows the result of the multidirectional hillshade calculation.

Slopeshade

- ❖Display slope using a white-black stretch
- ❖White = shallow slopes
- ❖Black = steep slopes

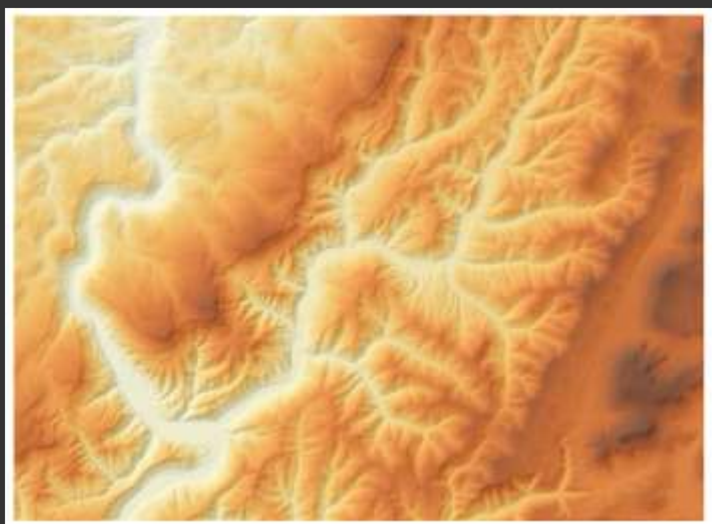


Map by Miles Reed

16

An alternative to the hillshade is the slopeshade. This is simply a topographic slope raster grid with a white-to-black color ramp applied. White shades indicate low slopes or flat areas while black indicates steep slopes. When quality, high spatial resolution DEM data are used, slopeshades can offer a very detailed terrain representation for interpretation. However, these visualizations are not commonly used in map layouts unless appropriate for the intended audience, such as geomorphologists.

Combining DEMs and Hillshades

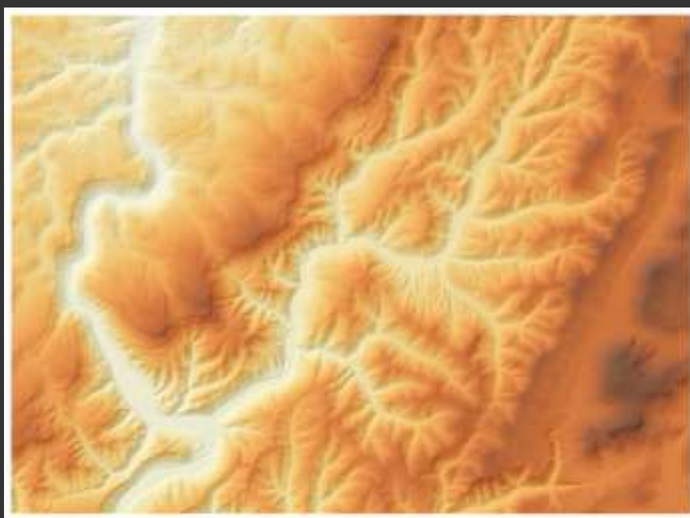


- Hillshade with 70% transparency
- DEM with 0% transparency

17

It is common to combine DEMs and hillshades to generate a textured and colorized terrain surface. These are often referred to as hypsometrically-tinted hillshades. Since both datasets are raster grids, this requires the use of transparency. In this example, the hillshade uses a black-to-white color ramp and has a 70% transparency applied. It is displayed above the DEM, which has an orange/red color ramp applied and no transparency. Determining the optimal amount of transparency will require some experimentation.

Combining DEMs and Hillshades



- DEM with 30% transparency
- Hillshade with 0% transparency

18

In this example, I have used the same color ramps but reversed the drawing order and applied a 30% transparency to the DEM data.

Combining DEMs, Hillshades, and Contours

In this example, I combine the DEM, hillside, and contour lines. The color binning applied to the DEM is designed to align with the principal contours, and transparency is used to add texture with the hillshade.

Surface Curvature

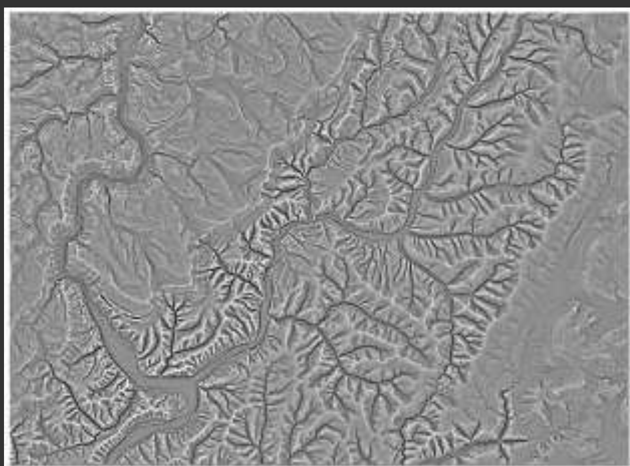
- Concavity/convexity of a surface
- Profile Curvature = curvature in direction of slope
- Planform Curvature = curvature perpendicular to direction of slope
- Can be used to highlight ridges vs. valleys
- Valley = dark
- Ridges = bright
- Black-to-white sequence

Negative = Concave
Positive = Convex
Near Zero = Flat

20

Surface curvature can also be applied to highlight ridges and valleys. Negative values indicate concavity, positive values indicate convexity, and zero indicates near flat terrain. So, applying a dark-to-bright color ramp will make valley bottoms appear dark and ridges appear bright.

Slope Position



- Elevation at cells compared to elevation of surrounding cells
- Must calculate mean using neighborhood analysis
- "DEM" – "Mean"
- Ridge vs. valley

21

As an alternative to a surface curvature data layer, slope position can be used to obtain the same effect. Here, the mean elevation value in a moving window is subtracted from the center value. This relies on a moving window analysis, so the shape and size of the window can have a large impact on the output. Larger window sizes will generate more generalized patterns whereas smaller window sizes will capture more local variability. The appropriate window size is case-specific and will require some experimentation. In the output, high values indicate locations where the center cell is higher than the average of its neighbors. These are generally topographic high points, like ridges. In contrast, low values indicate areas where cells are lower than their neighbors, and would represent topographic low points or valleys. So, the result is similar to curvature, but calculated differently.

Combining DEMs, Curvature, and Hillshades

- Curvature with 70% transparency
- Hillshade with 40% transparency
- DEM with 0% transparency

22

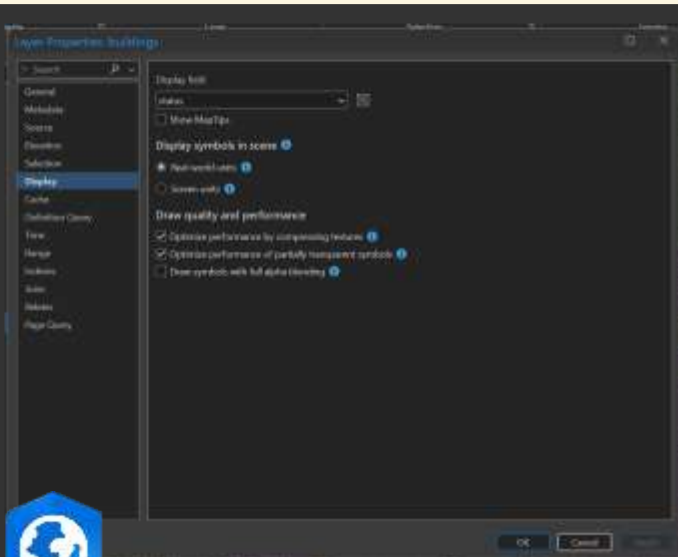
This last example combines surface curvature, a hillshade, and a DEM. Transparency is applied to both the curvature surface and hillshade. The result highlights ridges and valleys due to the incorporation of curvature.

2D Vector Data

23

Moving from terrain visualization, we now explore 3D symbology for 2D vector data: points, lines, and polygons. Vector features can be extruded into 3D space and symbolized using colors, sizes, and textures to represent data attributes.

Screen vs. Actual Size



- Real-world = use for known heights (m)
- Screen = relative to screen display (px)

24

When extruding vector features to 3D, size can be specified either in real-world coordinates or in screen-relative coordinates. Real-world size uses actual measurements in the same units as the map, making features part of the 3D scene. Screen size uses relative pixel measurements, making features appear consistent regardless of camera distance, but breaking the illusion of 3D space.

Extrusion in ArcGIS Pro

Method	Description
None	Features are not extruded. This is the default.
Min Height (line and polygon extrusion only)	Add extrusion to each feature's minimum height. A z-value is calculated by adding the extrusion height to the minimum z-value of each feature. They are extruded to a flat top at that value.
Max Height (line and polygon extrusion only)	Add extrusion to each feature's maximum height. A z-value is calculated by adding the extrusion height to the maximum z-value of each feature. They are extruded to a flat top at that value.
Base Height	Add extrusion to each vertex of each feature's base height. A z-value is calculated for each vertex of the feature's base and each feature is extruded to those z-values. In the case of polygon features, top and base faces are created to connect to the extruded walls to construct a closed shape.
Absolute Height	Each feature is extruded to the specified z-value from a field or expression as a flat top, regardless of the z-values of each feature.

https://pro.arcgis.com/en/pro-app/latest/help/mapping/layer-properties/extrude-features-to-3d-symbology.htm

25

ArcGIS Pro provides multiple extrusion methods for converting 2D vector data to 3D. None indicates not extruding features, which is the default. Min Height adds extrusion to each feature's minimum height. Max Height adds extrusion to each feature's maximum height. Base Height adds extrusion to each vertex's base height. Absolute Height extrudes to a specified z-value from a field or expression as a flat top. Each method is appropriate for different data types and visualization goals.

Points: Thematic

- Extrude to a height based on an attribute
- Change extruded symbol
- Color extruded symbol

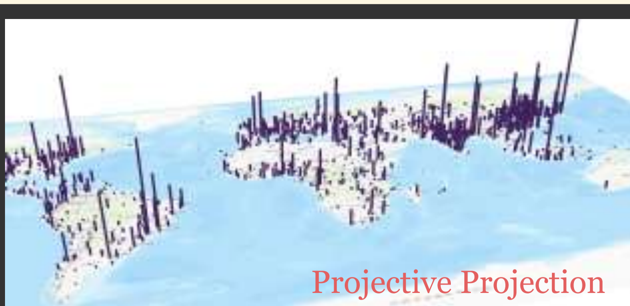
If using an attribute, may need to rescale.

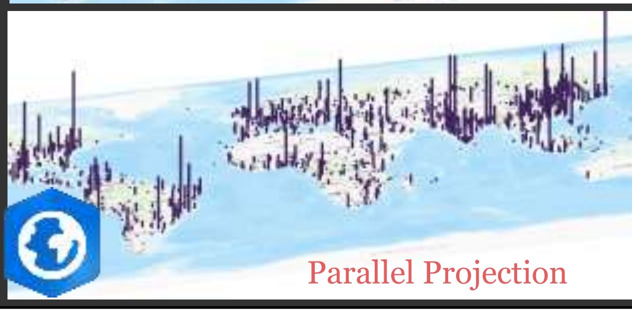
26

For point features, thematic extrusion uses a height attribute to represent data values. You can extrude to a height based on an attribute value, change the extruded symbol appearance, and color the extruded symbol based on another attribute. This technique allows representing multiple data dimensions simultaneously in 3D space.

Note that it is common to have to adjust the values to obtain a reasonable range of lengths.

Point: Thematic



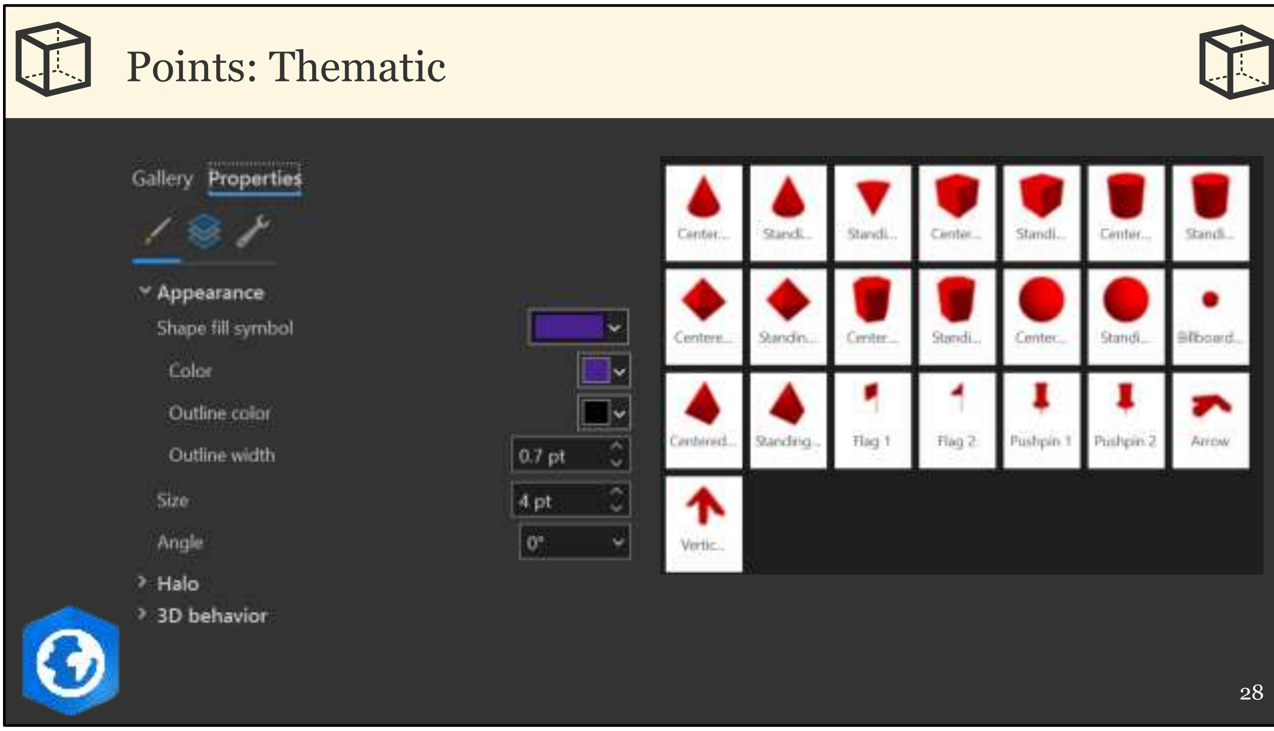


- Projective projection = objects of same physical size/height not rendered the same
- Parallel projection = objects of same physical size/height are rendered the same

27

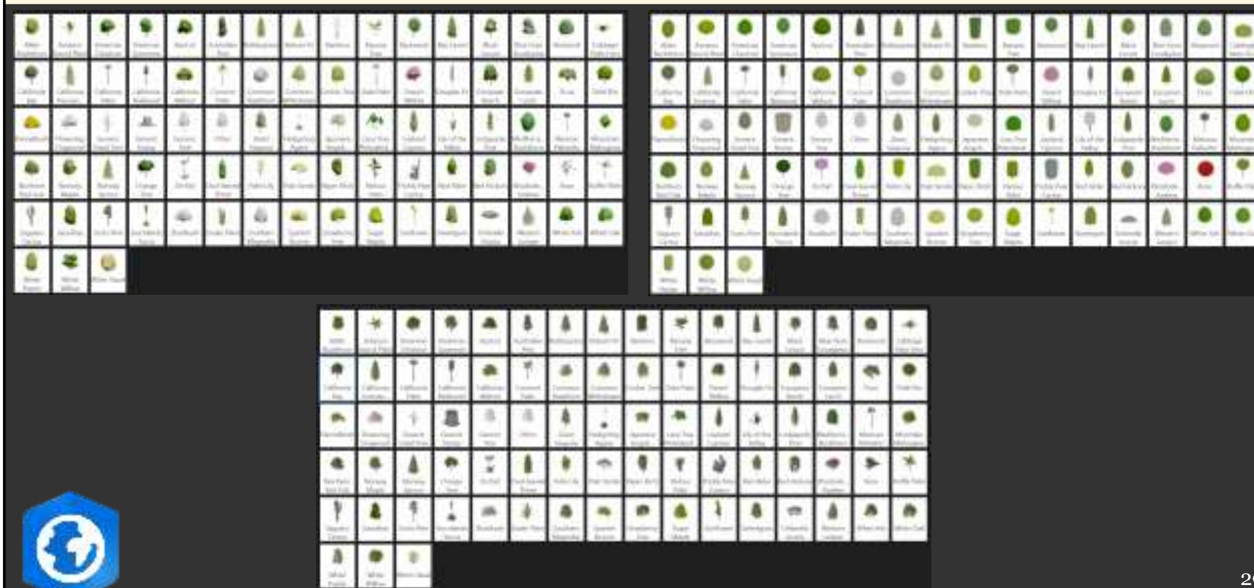
When symbolizing point features, the choice between projective and parallel projection matters. Projective projection makes objects of the same physical size appear different depending on distance from the camera, creating realistic perspective. Parallel projection makes objects of the same physical size appear the same size regardless of distance, eliminating perspective. The choice affects how data are perceived and interpreted.

Parallel may be more appropriate if the user needs to compare the relative height of features to appropriately interpret the values.



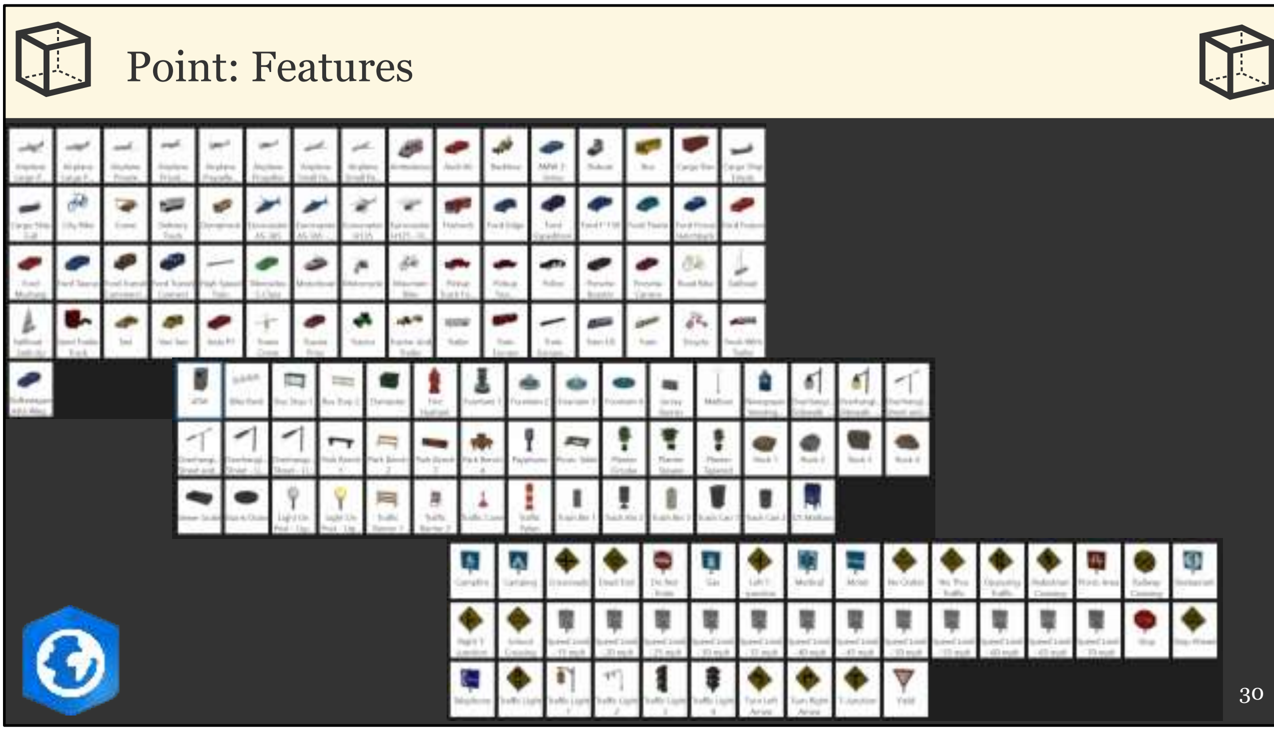
Point features can be symbolized thematically using various visual variables including color, size, and shape. The 3D visualization allows viewing from all angles and potentially creating more engaging visual representations than 2D mapping.

Point: Features



29

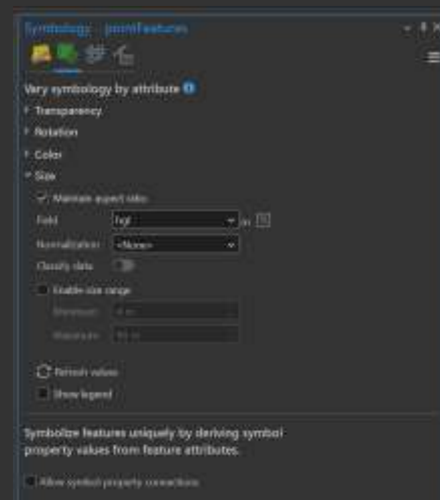
Point features can also be symbolized using categorical variations. Different symbols represent different feature classes, allowing multiple nominal variables to be displayed simultaneously in 3D space.



This slide shows some other 3D point symbol options available in ArcGIS Pro.

Point: Features

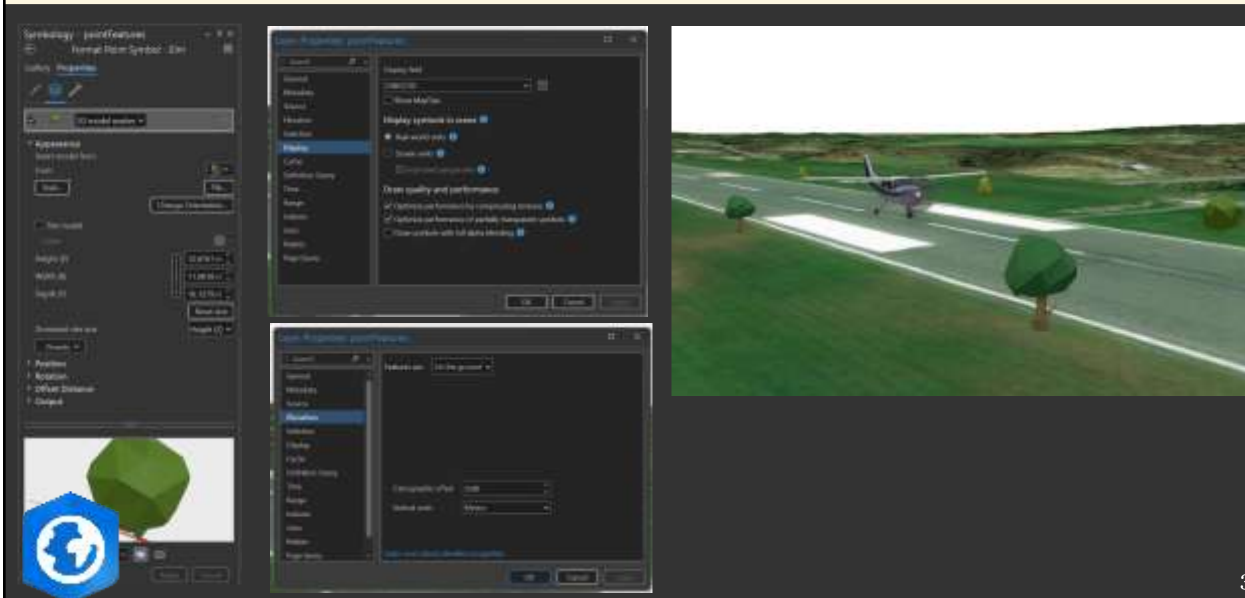
- ❖Same 3D symbol for all features
- ❖Different 3D symbol based on an attribute
- ❖Size, height, orientation, etc. of symbol collectively
- ❖Size, height, orientation, etc. of symbol based on an attribute



31

Point feature symbology can use a single symbol for all features, or different symbols based on attributes. Symbols can be sized, colored, or oriented based on attribute values, allowing effective representation of multivariate data in 3D space.

Points: Features



32

Symbols can generally be edited, such as there size, orientation, and offset along different directions. The examples on this slide show resizing features representing a tree.

Points: Features

In this example, I am symbolizing a point layer of car positions using 3D models of cars. Different cars are used depending on a nominal attribute.

A different set of point features is symbolized as trees. The tree type is selected based on a nominal attribute.

Billboards

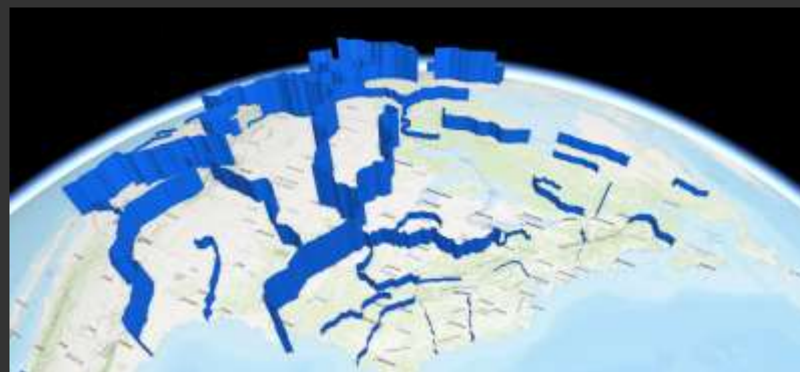
- Flat surface that rotates to face the camera

34

Billboards are flat surfaces that always face the camera, regardless of viewing angle. This technique is commonly used for point symbols, trees, and other features where maintaining a direct view of the symbol is important. Billboards create visual impact while maintaining clarity from all viewing angles.

Lines: Thematic

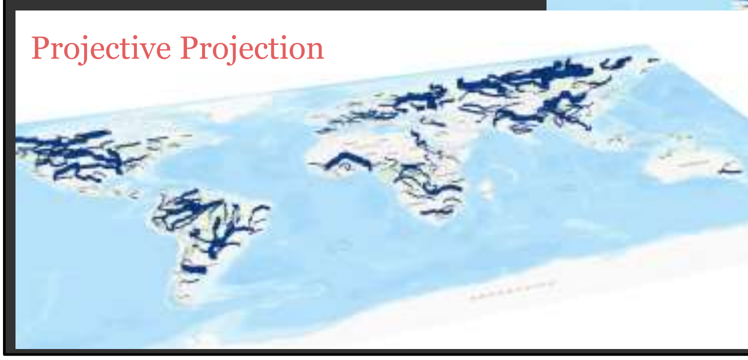
- Single symbol
- Nominal variable
  - Color
- Continuous variable
  - Color
  - Height



35

Lines can be symbolized using single symbols or thematic symbols. Thematic symbology uses color or height to represent data variables. Lines can represent nominal variables using different colors or continuous variables using color or height ramps. This allows representing one or two dimensions of data per line feature.

Lines: Thematic



36

Like points, line symbology can use projective or parallel projection. Projective projection creates realistic perspective while parallel projection maintains consistent appearance regardless of distance. The choice affects visual interpretation and how viewers interpret the symbolized data values and relative heights.

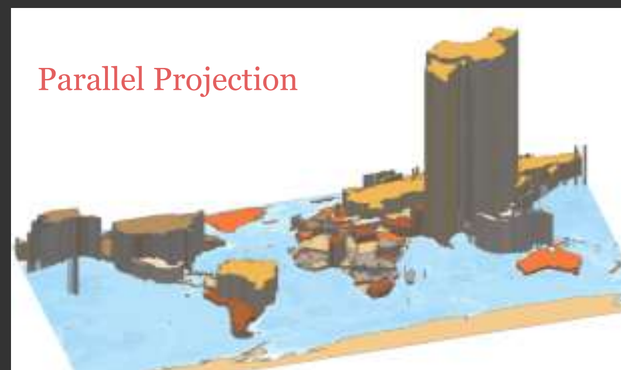
Polygons: Thematic

- Single symbol
- Nominal variable
  - Color
- Continuous variable
  - Color
  - Height

37

Polygons can be symbolized using single symbols or thematic symbols. Thematic symbology uses color or height to represent data variables. Polygons can represent nominal variables using different colors or continuous variables using color or height ramps.

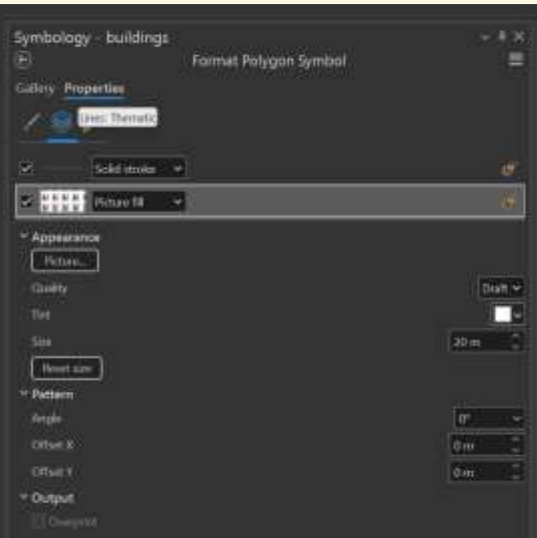
Polygons: Thematic

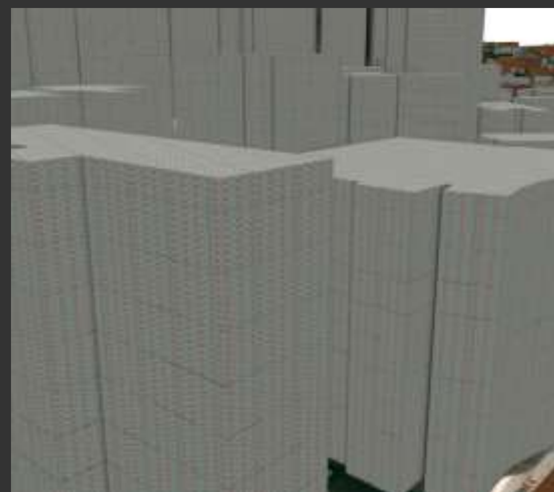


38

Polygons can use either projective or parallel projection depending on visualization goals. Projective projection creates realistic perspective while parallel projection maintains consistent appearance and is often better for scientific visualization and analysis. Parallel projections may be more appropriate if the user needs to interpret relative heights.

Picture Fill

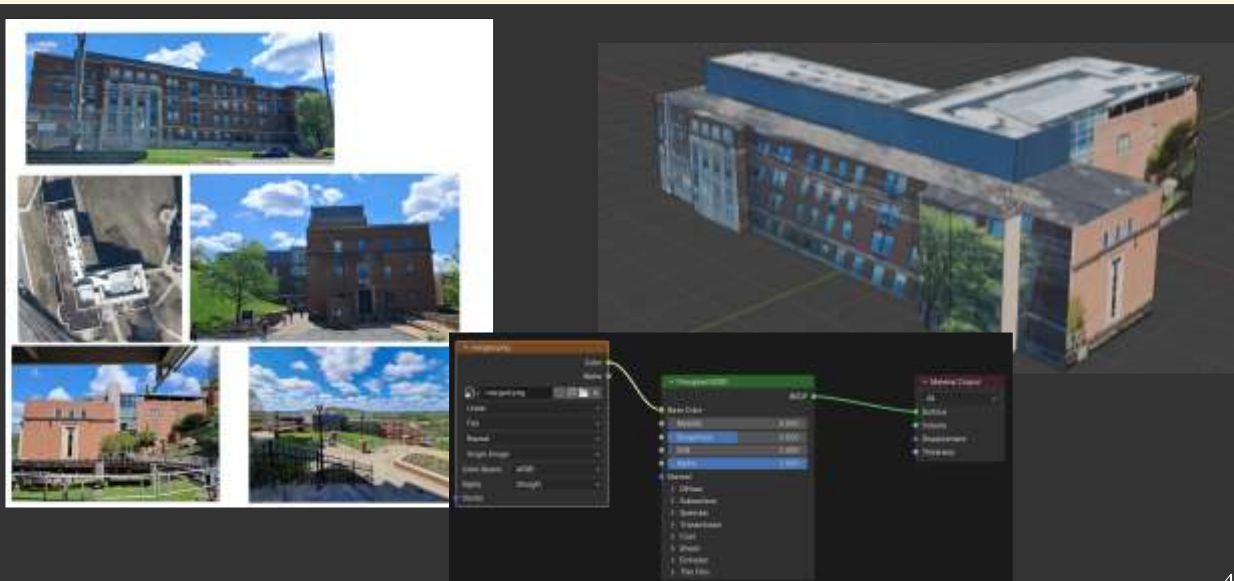




39

Picture fills allow applying photographic or graphical textures to polygon faces. This technique can add realism and detail, making visualizations more visually engaging. Textures can be repeated or stretched across polygon surfaces.

UV Map an Image



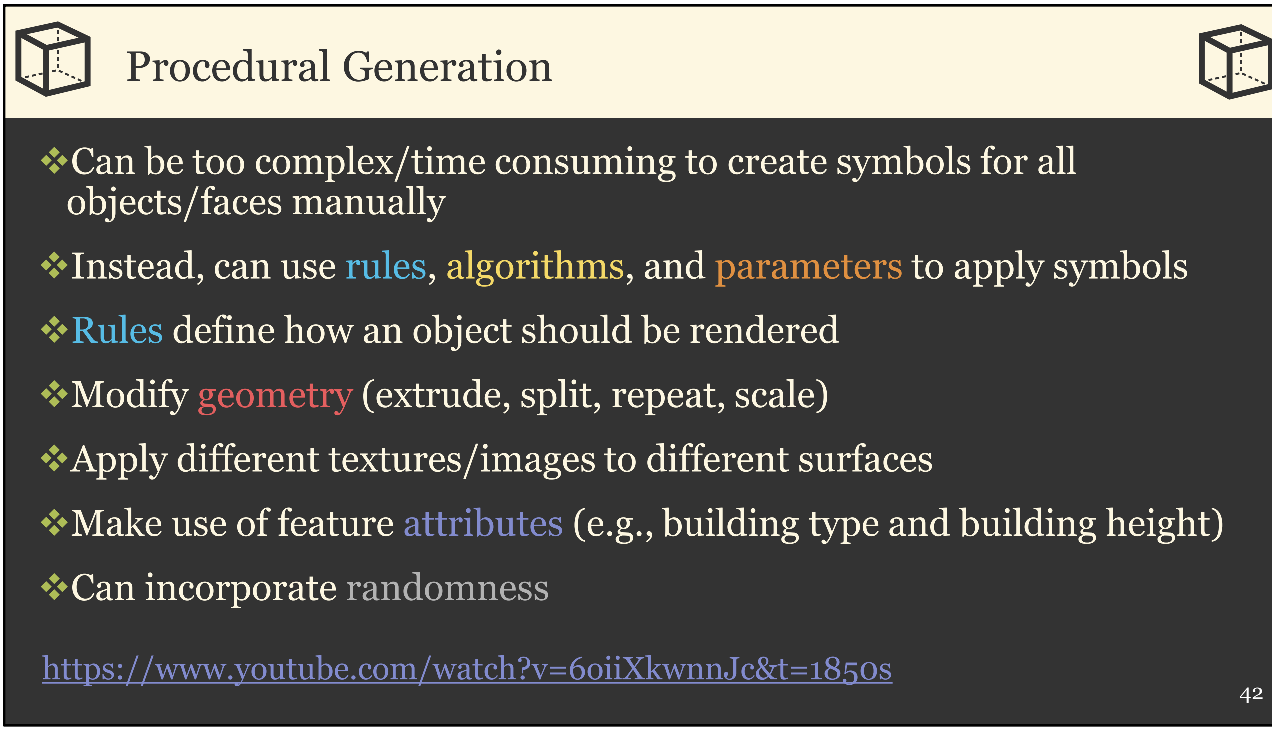
40

UV mapping allows mapping 2D images onto 3D polygon faces. Proper UV mapping ensures textures are oriented correctly and displayed without distortion.

Procedural Generation

41

Procedural generation uses rules, algorithms, and parameters to automatically apply symbols and generate geometry. Rather than manually creating symbols for every object, procedural techniques allow creating complex, varied geometry and textures efficiently.



Procedural generation is valuable when creating symbols for many objects would be time-consuming. Rules define how objects should be rendered. Algorithms modify geometry through extrusion, splitting, repeating, and scaling. Different textures and images can be applied to different surfaces based on rules. Feature attributes like building type and height can drive the generation process. Randomness can be incorporated to create natural variation.

ArcGIS CityEngine

- Procedurally generate 3D cities
- Computer Generated Architecture (CGA) rules
- Used by urban designers, planners, architects, and 3D artists

https://www.esri.com/en-us/arcgis/products/arcgis-cityengine/overview

43

ArcGIS CityEngine is a specialized tool for procedurally generating 3D cities. CityEngine uses Computer Generated Architecture rules to automatically create buildings and urban environments. It's used by urban designers, planners, architects, and 3D artists for creating realistic 3D city models.

Rule Packages

- RPK (zipped file)
- CPA = Computer Generated Architecture
- Rule Package = Rules (CGA) + Assets (geospatial data, textures)
- Can make it so users can or cannot see code
- Written in CityEngine
- Can be used in ArcGIS Pro and shared with AGOL
- An be used for symbology and also within analysis workflows

44

Rule packages are RPK files that bundle CGA rules with assets. They can include geospatial data and textures needed for procedural generation. Rule packages can be created in CityEngine and shared via ArcGIS Online. They allow users to apply procedural generation in ArcGIS Pro and in analysis workflows. You can control whether users can see the underlying code.

CGA

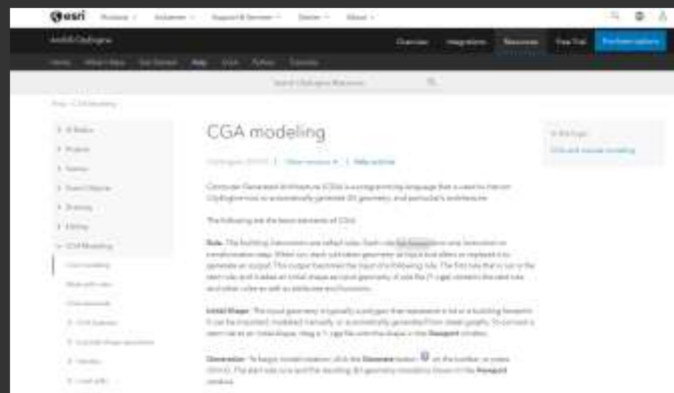
- Shape grammar programming language
- Functional programming language
- Scripting for shapes
- Process 3D geometries
- Shape = scope and geometry
  - Scope = bounding box
  - Geometry = actual geometry
- Can modify scope and geometry
- Rule = sequence of shape operations

45

CGA is a shape grammar programming language used in CityEngine. It's a functional programming language for scripting shapes and processing 3D geometries. In CGA, shape refers to both scope and geometry. Scope is the bounding box, while geometry is the actual shape. Rules are sequences of shape operations that modify scope and geometry.

CGA

- Types of modifications:
  - Move
  - Scale
  - Split
  - Extrude
  - Repeat
  - Texturize
  - Etc.

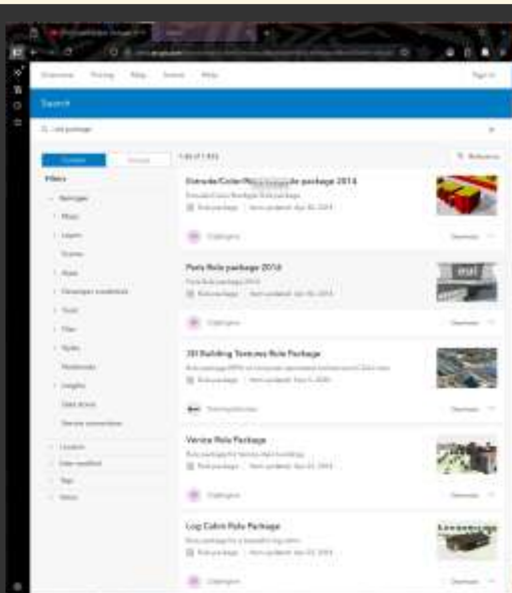


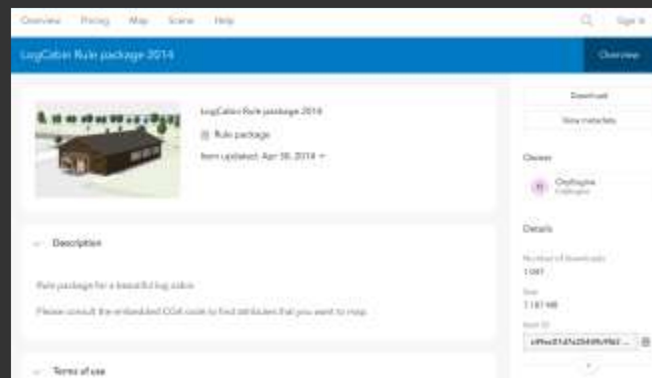
https://doc.arcgis.com/en/cityengine/2024.0/help/help-cga-modeling-overview.htm

46

CGA operations include move, scale, split, extrude, repeat, texturize, and many others. These operations can be combined in various ways to create complex 3D geometry and textures. The programming approach allows creating flexible, reusable rules for a wide variety of building types and urban patterns.

Using Rule Packages in ArcGIS Pro





https://www.arcgis.com/

47

Using rule packages in ArcGIS Pro involves accessing the 3D visualization tools and symbology options. Rule packages allow bringing procedurally generated content into ArcGIS Pro for visualization and analysis alongside other geospatial data.

Using Rule Packages in ArcGIS Pro

Symbology → Single Symbol → Properties → Procedural Fill

48

To use rule packages in ArcGIS Pro, access Symbology options and select Single Symbol or Procedural Fill. This allows applying procedural rules to features and generating complex 3D geometry automatically.

City Generator

https://superhivemarket.com/products/the-city-generator

49

City Generator is a third-party tool available through the Superhive Market that provides procedural building generation for Blender. It allows generating diverse, realistic buildings without manual modeling.

Nature Generator

https://superhivemarket.com/products/the-nature-generator

50

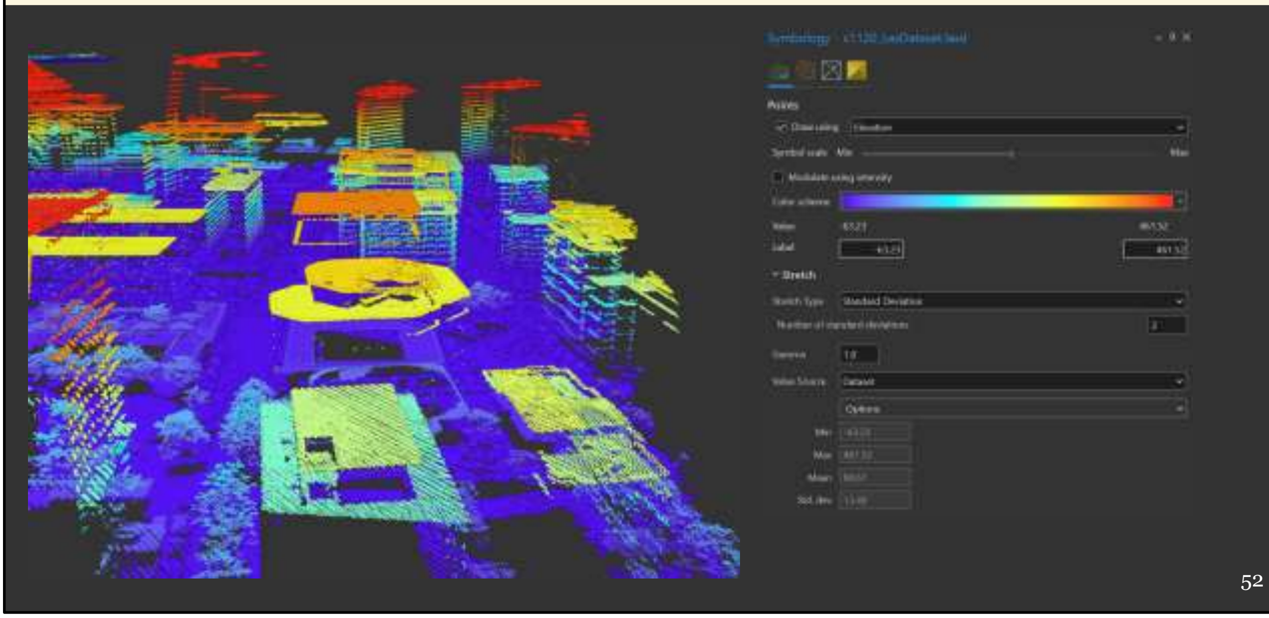
Nature Generator is another third-party tool available through the Superhive Market for procedurally generating trees, plants, and other natural features in Blender. These tools extend procedural generation capabilities for environmental visualization.

Point Clouds

51

Point clouds are sets of points in 3D space, typically generated from lidar or photogrammetry. Point clouds can represent terrain, buildings, vegetation, and other features with dense spatial information. Point cloud visualization requires specialized techniques due to the large number of points.

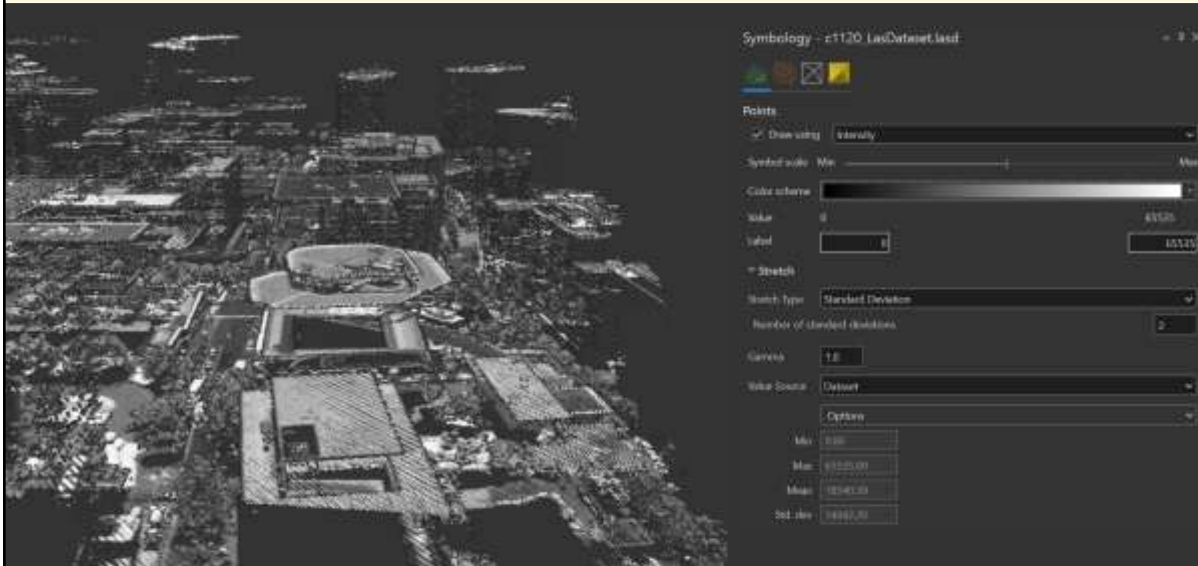
Point Style: Elevation



52

Point clouds can be styled by elevation, showing height variation through color ramps. This is particularly useful for identifying terrain features, buildings, and vegetation at different elevations.

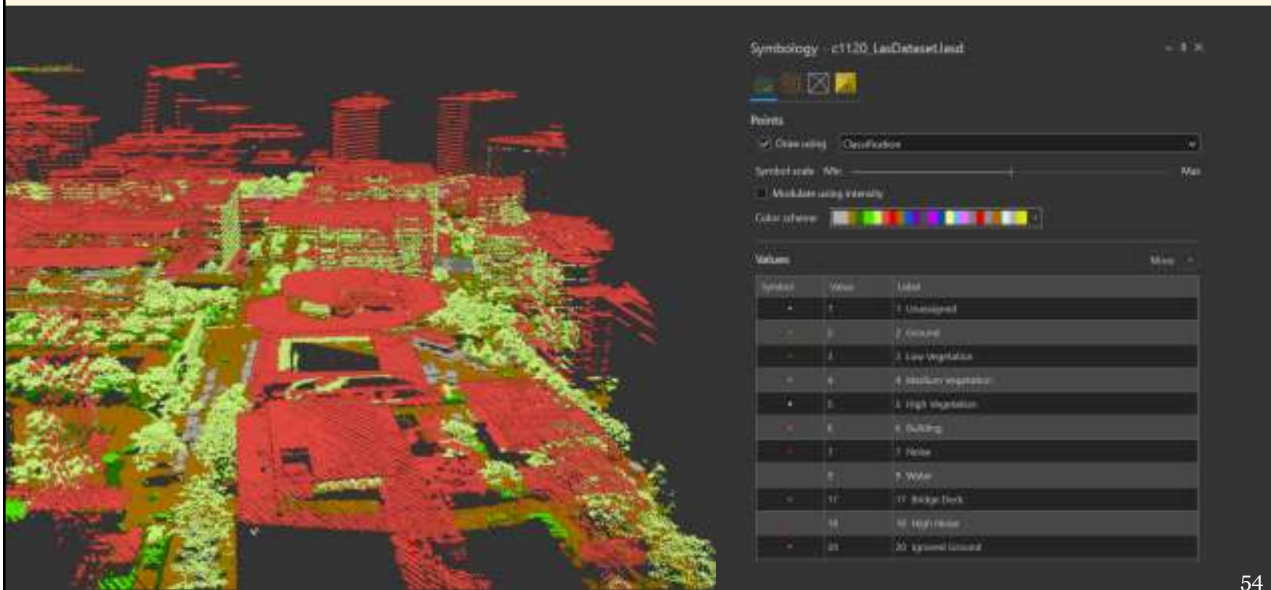
Point Style: Intensity



53

Point clouds can be styled by intensity, representing the intensity of the returned lidar signal. This indicates surface reflectivity and can help identify different materials and surface characteristics.

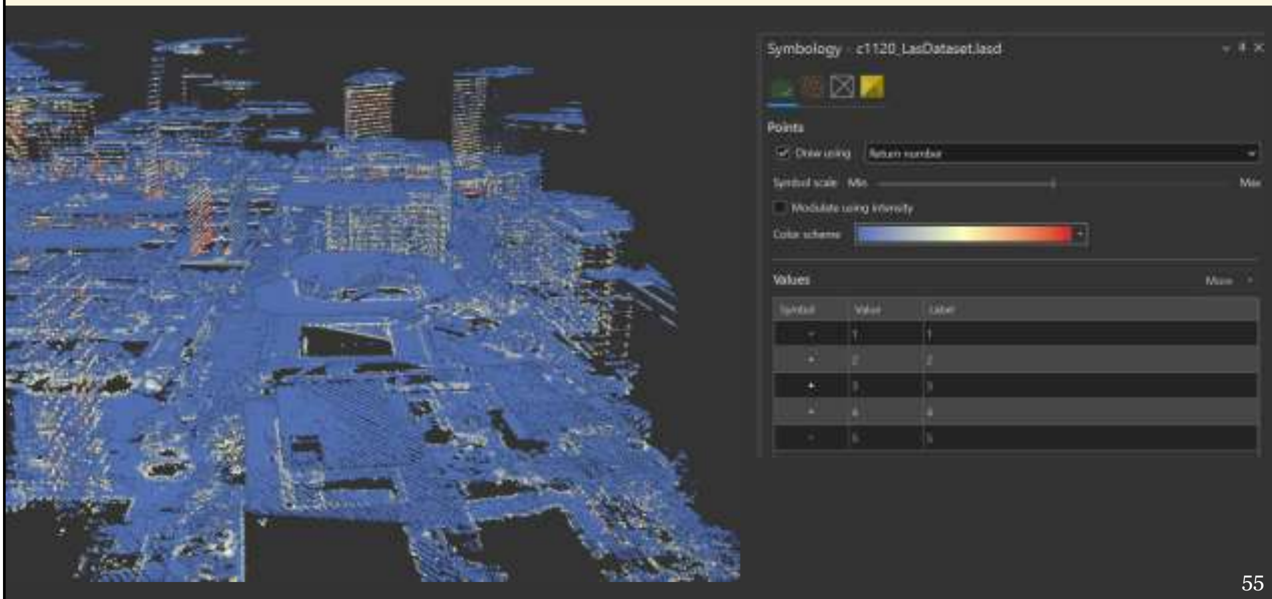
Point Style: Classification



54

Point clouds can be styled by classification, showing different feature types. Standard ASPRS classification includes ground, vegetation, buildings, water, and other classes. Classification visualization can allow distinguishing between different land cover types.

Point Style: Return Number

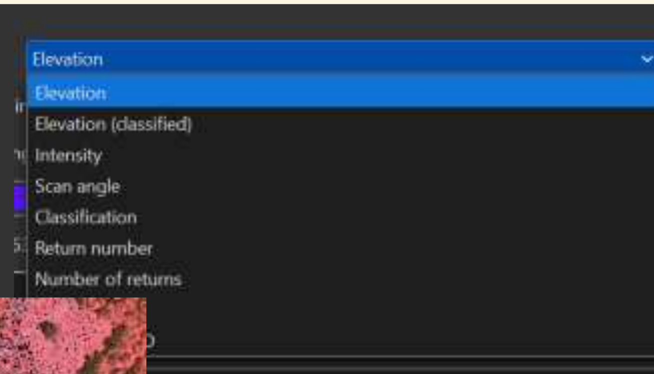


55

Point clouds can be styled by return number, showing which return in a multi-return lidar sequence each point represents. First returns typically represent canopy tops while later returns represent ground or intermediate heights.

Other Attributes

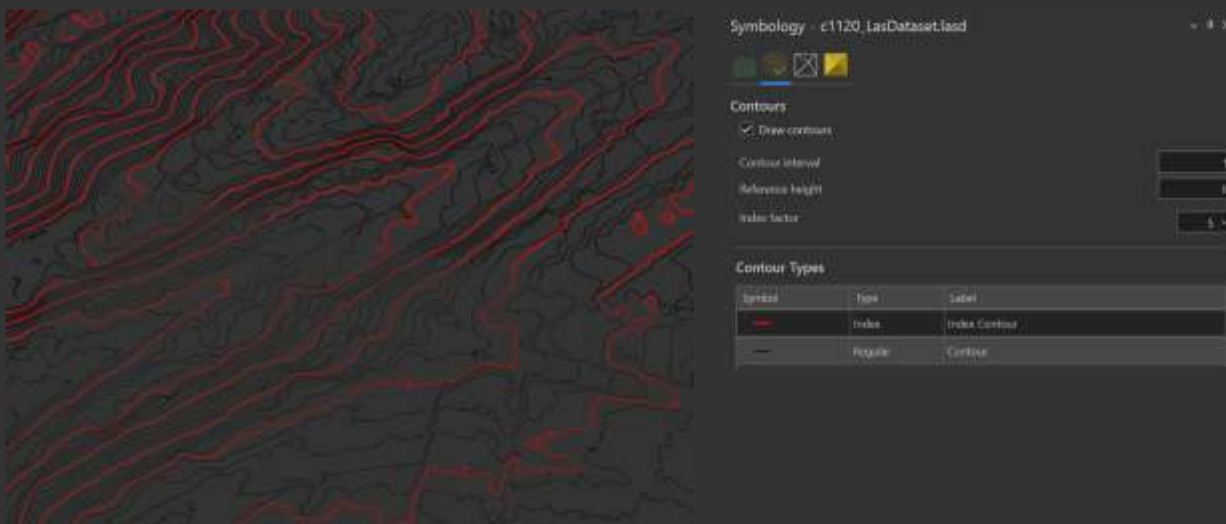
- Scan angle
- Number of returns
- RGB
- RGBN



56

Additional point cloud attributes include scan angle, number of returns, RGB color from imagery, and RGBN including near-infrared. These attributes can be used for analysis and visualization.

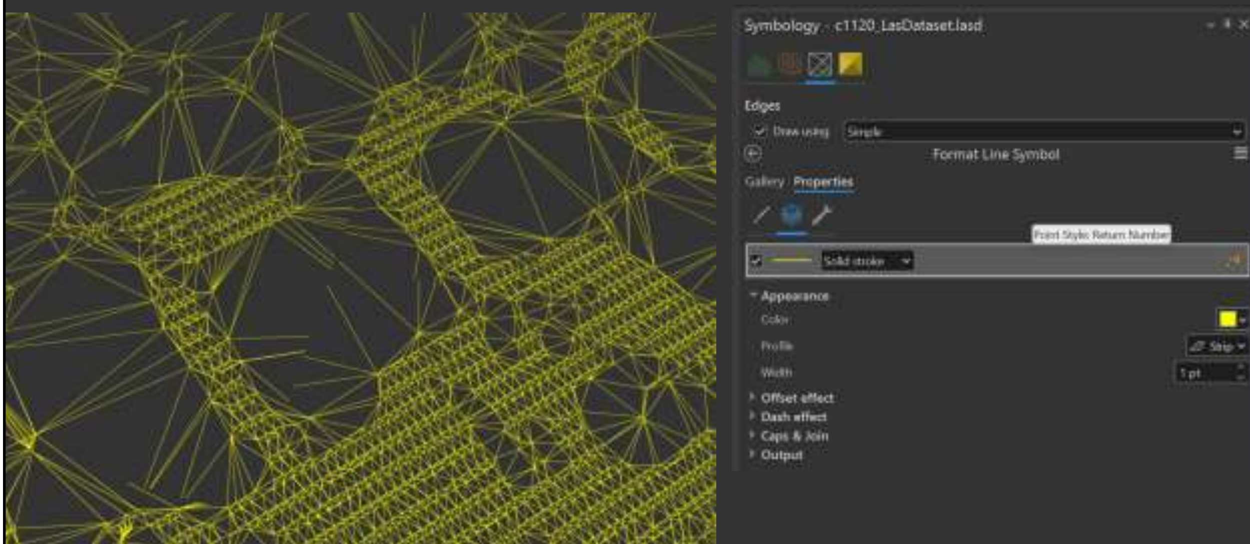
Isolines or Contours



57

Point clouds can be visualized as isolines or contours. This converts dense point data into more traditional contour maps.

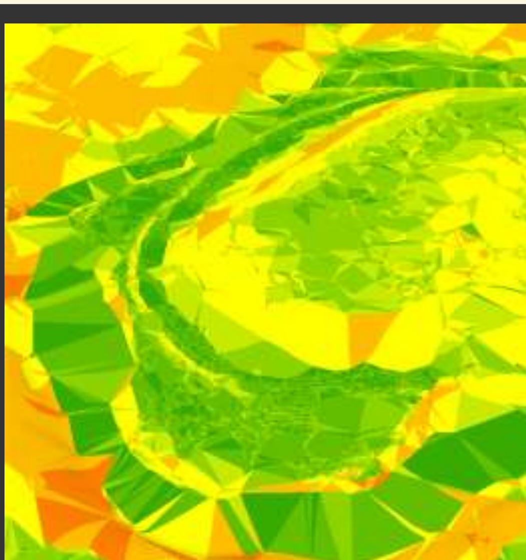
Edges



58

Point clouds can be visualized as edges, similar to a TIN surface.

As a Surface



- Elevation
- Slope
- Aspect

59

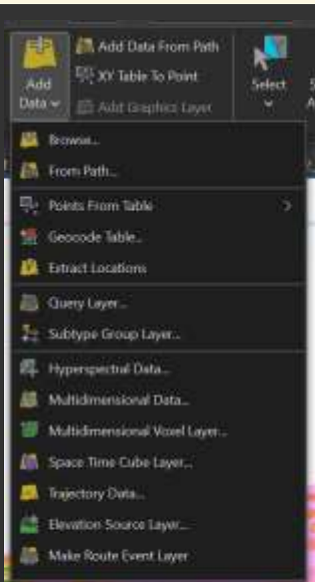
Point clouds can be visualized as surfaces representing elevation, slope, or aspect. Converting point cloud data to surfaces allows traditional raster analysis and visualization.

Voxels and Multidimensional Raster Grids

60

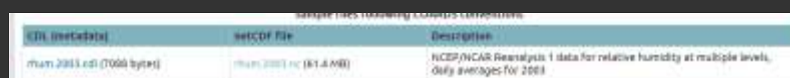
Voxels are three-dimensional pixels representing values assigned to volumes in a 3D grid. Voxel and multidimensional raster grids allow representing multivariate spatial data across three spatial dimensions and often time.

Adding Voxel Data in ArcGIS Pro



- ❖Multidimensional Data
- ❖Multidimensional Voxel Layer
- ❖Space Time Cube Layer

Example data used here from:
https://archive.unidata.ucar.edu/software/netcdf/examples/files.html



61

Adding voxel data in ArcGIS Pro involves creating multidimensional voxel layers or space-time cube layers. These layers allow visualizing complex, multidimensional datasets including climate, atmospheric, and oceanographic data.

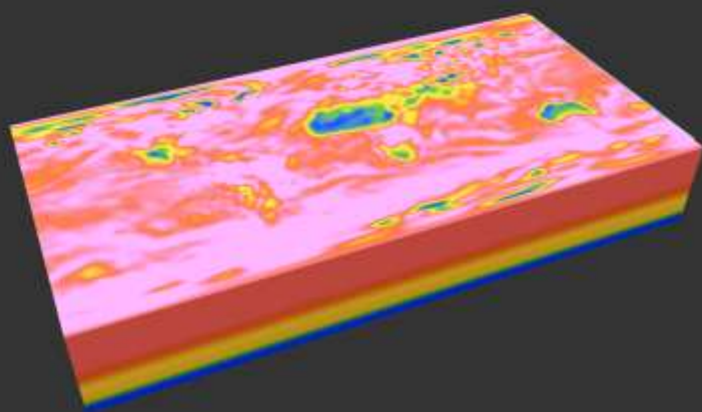
File Formats

- **NetCDF**
  - .nc
  - Dimensions = x, y, z, time, variable
  - Metadata = units, CRS, descriptions
  - Commonly used for climate, atmospheric, and oceanographic data
- **GRIB**
  - .grib or .grib2
  - Commonly used for weather and forecast data
- **HDF**
  - .hdf or .h5
  - Commonly used for Earth observation data
- **Cloud Optimized GeoTIFF**
  - .tif
  - Not fully multidimensional
- **Zarr**
  - Cloud native and large datasets
- **Cloud Raser Format**
  - .crf
  - ESRI’s cloud-native raster file format
  - Data stored in chunks or tiles
  - Fast read and parallel processing

62

Multiple file formats support multidimensional data. NetCDF format stores dimensions of x, y, z, time, and variables with metadata. GRIB format is commonly used for weather and forecast data. HDF is commonly used for Earth observation data. Cloud Optimized GeoTIFF represents a transition toward cloud-native rasters. Zarr provides cloud-native support for large datasets. Cloud Raster Format is ESRI's cloud-native raster format for storing data in chunks for efficient access.

As a Cube or Volume

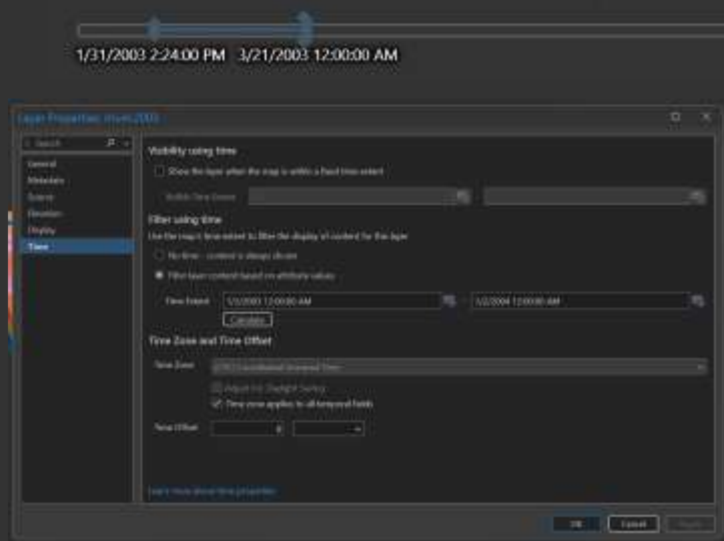


- *x*, *y*, *z* (depth or altitude)
- Variable mapped to color ramp
- Could be continuous or discrete

63

Voxel data can be visualized as a cube or volume where x, y, z coordinates define position and a variable is mapped to color. Variables can be continuous or discrete depending on the data characteristics.

Time Slices



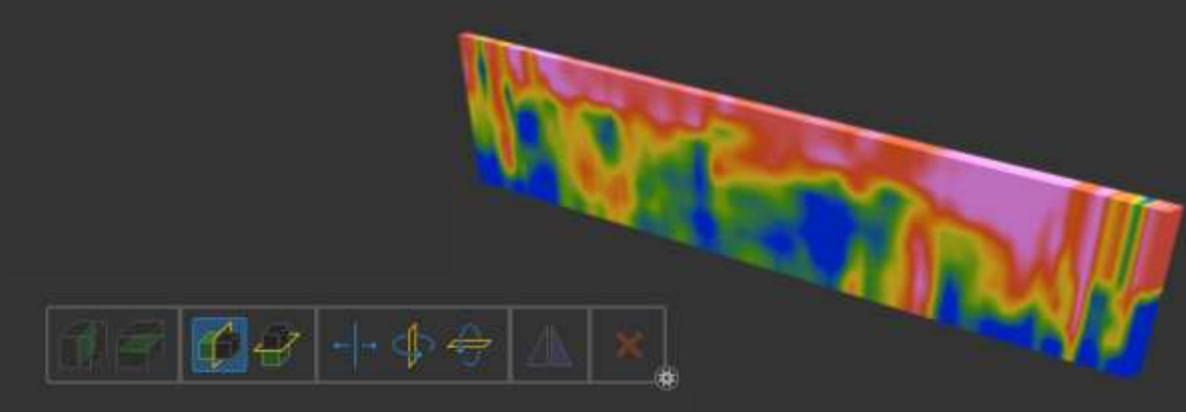
- User slider or animation to show change over time
- Can combine with time series graphs

64

Time slices allow viewing specific time steps from multidimensional data. Users can interact with sliders or animations to show how values change over time. Time series graphs can be combined with spatial visualizations.

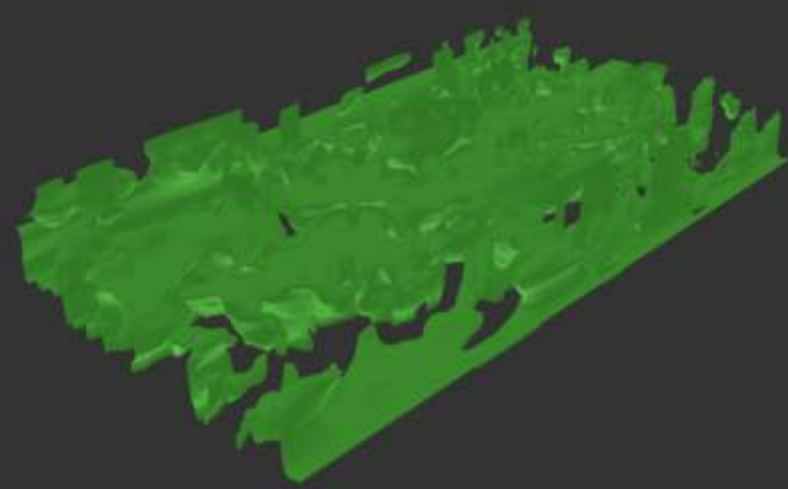
Spatial Slices

❖Can be vertical, horizonal, or oblique



Spatial slices show cross-sections through 3D data. Slices can be vertical, horizontal, or oblique depending on the analysis question. This technique helps visualize spatial patterns within 3D volumes.

Isosurface



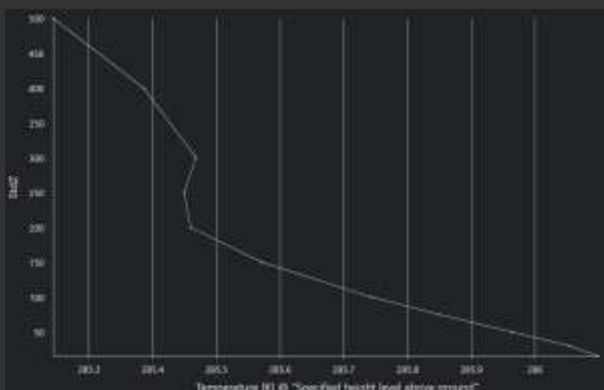
❖ Surface representing locations with the same value

66

Isosurfaces are surfaces representing locations with the same value. This technique allows visualizing 3D patterns and thresholds, making it easy to identify regions meeting specific criteria.

Graphs

- **Profiles** with altitude or depth at a pixel location
- **Time series** of value at pixel location



67

Graphs can show profiles with altitude or depth at a specific pixel location, or time series of values at a pixel. These complementary visualizations provide both spatial context and detailed value information.

3D Labeling

68

3D labeling places text annotations in 3D space to identify and provide information about features. Effective 3D labeling requires careful consideration of placement, size, orientation, color, and visibility.

Type

- Type Family = group of type designs that reflect common design characteristics and share a common base name (Calibri, Arial, Comic Sans, Cambria, Courier New, Times New Roman, Verdana)
- Type Style = Roman (normal), bold, *italic* (Less common: condensed, expanded, light, and extra bold)
- Type Face = Family + Style (Gill Sans Ultra Bold)
- Type Size = height (11 pt, 14 pt, 18 pt)
- Font = a set of all alphanumeric and special characters of a particular family, style, and size (Times New Roman 12 pt)

69

Type fundamentals include understanding type families: groups of related typefaces sharing common characteristics. Type styles include roman, bold, and italic. A typeface combines family and style. Type size is measured in points. A font is the complete set of characters in a specific family, style, and size.

Case

- Title Case = lowercase letters with the first letter of each word set to upper case (except linking words like in, on, or, of, per, by, for, with, the, and, over, etc.)
- Sentence case = Only first word in the sentence is capitalized
- UPPERCASE = All caps

70

Text case affects readability and visual impact. Title case capitalizes the first letter of each word except linking words. Sentence case capitalizes only the first word. Uppercase uses all capitals. Case selection affects how text appears and how quickly it's processed by viewers.

Serif vs. Sans Serif

- Serif = has short extensions at the end of major letter strokes (This is a serif font).
- Sans serif = does not have serifs (This is a sans serif font).

71

Serif fonts have short extensions at the ends of letter strokes, often used for body text in printed documents. Sans serif fonts lack serifs and are often used for headers and screen display. The choice between serif and sans serif affects perceived formality and readability.

Spacing

- Letter Spacing = space between letters in a word

  Here I have changed the letter spacing.

- Word Spacing = spacing between words
- Kerning = variation of spacing between adjacent letters
- Leading = vertical spacing between lines

  Here I have changed the leading.
  Here I have changed the leading.
  Here I have changed the leading.
  Here I have changed the leading.

72

Spacing controls include letter spacing between individual letters, word spacing between words, kerning for variation between adjacent letters, and leading for vertical spacing between lines. Careful spacing adjustment improves readability and visual appeal.

In Scene vs. On Screen

- In Scene
  - Treated like object that consumes space in 3D view
  - Can be obstructed by other objects in the view
  - May use distance-based rendering or multiple LODs
  - Often attached or associated with objects

- On Screen
  - Separate layer of content
  - Does not interact with content in the view
  - May use distance-based rendering
  - Used to help with general navigation or when associated with features covering a wide area

73

3D text can be positioned in scene, where it consumes space and can be occluded by other objects, or on screen, where it's a separate layer and doesn't interact with scene content. In-scene text can use distance-based rendering or LODs. On-screen text helps with navigation and remains visible regardless of scene content.

Font

- Rules for 2D maps generally apply
- Match font to theme of the scene

74

When selecting fonts for 3D scenes, apply traditional 2D mapping font rules. Match the font to the theme of the scene. Choose legible fonts that work at the intended size and viewing distance.

Text Color

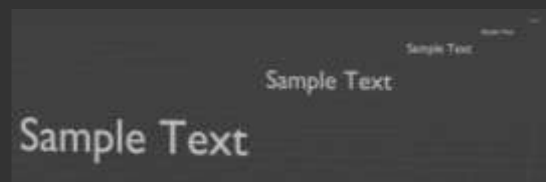
- Selecting an appropriate color can be tricky since the background may change as the user navigates
- When using realistic content and textures, bright colored text is preferred
- For thematic data, text color should contrast with the colors used to display the map features
- Can use halos or background fills to improve legibility
- Can apply transparency
- In scene text will be impacted by lighting configuration

75

Text color selection is tricky in 3D because background colors change as users navigate. With realistic content, bright colored text is preferred. With thematic data, text should contrast with displayed colors. Halos or background fills improve legibility. Transparency can be applied. In-scene text is affected by lighting configuration.

Text Size

- **Real-world size**
  - Get larger as user gets closer
  - Resizes with other objects in the scene
  - Can be hard to read at different distances
- **Screen-Space-Size**
  - Take up consistent area on screen
  - Same size with changing distance
  - Look artificial/lack of depth perception



76

Text size can be real-world size, getting larger as users approach and resizing with other scene objects, which can make text hard to read at different distances. Alternatively, screen-space-size maintains consistent screen area regardless of distance, but can look artificial and lack depth perception.

Placement

- Stand-Alone-Text
  - Treated as its own independent object
  - Positioned relative to a source feature
  - Often used for point features
  - Can be billboarded
- Embedded Text
  - Draws directly on a scene object
  - Text and object treated as one feature
  - Need surface area on which to add text

Text placement includes stand-alone text positioned relative to features, and embedded text drawn directly on scene objects. Stand-alone text is common for point features and can be billboarded. Embedded text requires surface area and is treated as part of the feature.

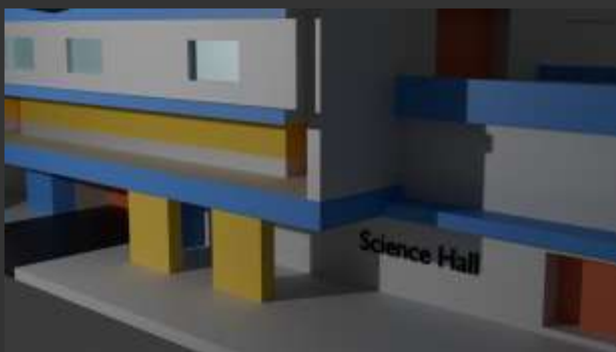
Stand-Alone Text

- When billboarding, use the same plane as the symbol
- Place above, on, or below symbol
- Can vertically offset to minimize overlap
- Can add leader line with increased vertical offset
- Set priorities to minimize conflict

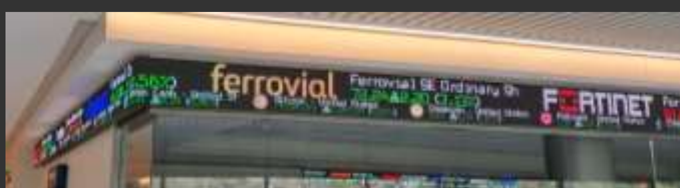
78

When using billboarded stand-alone text, use the same plane as the symbol. Place text above, on, or below the symbol. Vertical offset can minimize overlap. Leader lines with increased vertical offset clarify associations. Priorities minimize text conflicts.

Embed in Feature



- Reduces ambiguity
- Text does not add to space taken up by the feature
- Must have enough surface area
- Only readable from some angles
- Consider a ticker-tape effect



79

Embedding text in features reduces ambiguity and doesn't add to feature size. However, it requires sufficient surface area and is only readable from some angles. Consider ticker-tape effects for continuous text.

Embed in a Surface

- Best when
  1. Text physical at the location
  2. Only text used in scene
- Generally, more legible with flatter terrain

80

Embedding text in surfaces works best when text is physically at a location and is the only text in the scene. Generally, such text is more legible within flatter terrain.

Text Orientation and Direction

- Billboarding
  - Most common way to orient text in 3D
  - Can lock so rotation can only occur relative to certain axes
  - Works for real-world-size and screen-space-size
- Along a line
  - Sometimes used for linear features
  - Can use real-world size text that decreases in size with distance
  - Works for real-world-size
- Constrained to a virtual shape
  - Use spheres, cubes, or cylinders
  - Works for real-world-size (sometimes screen-space-size)

81

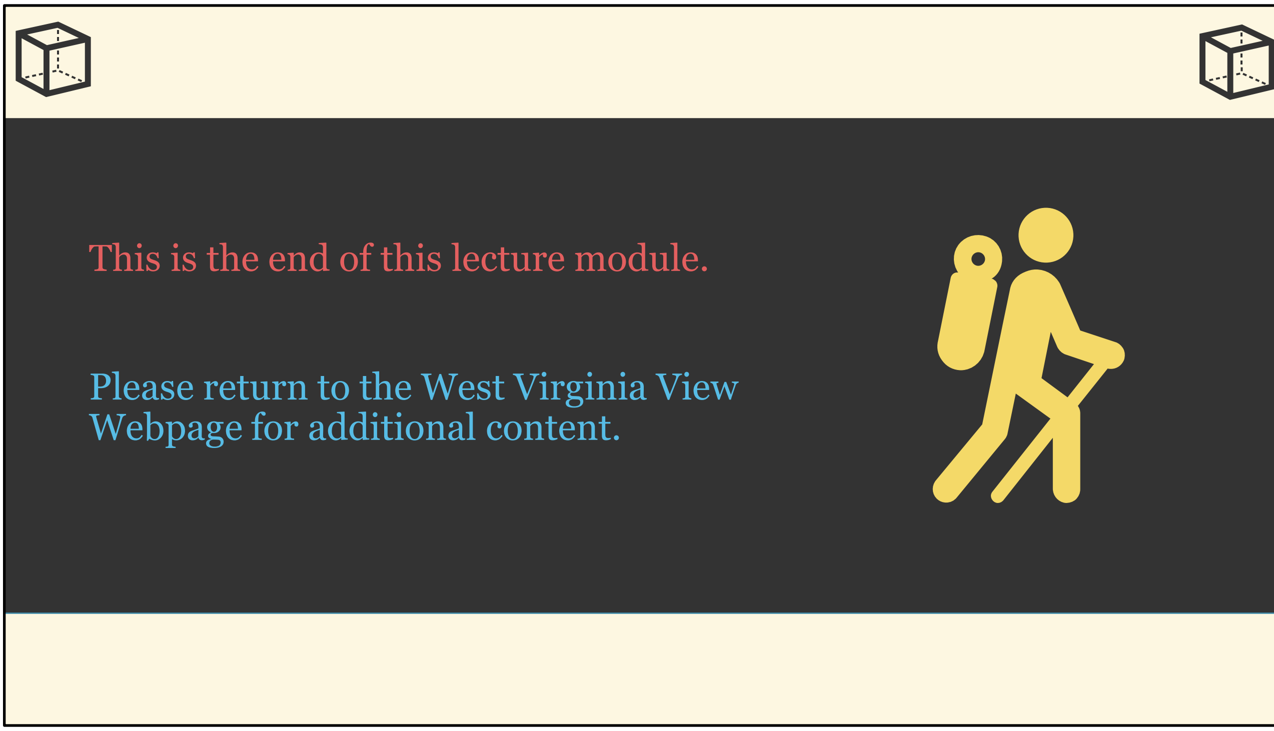
Text orientation options include billboarding, which can be locked to specific axes. Along-a-line orientation works for linear features with real-world-size text decreasing with distance. Constrained-to-shape orientation uses spheres, cubes, or cylinders.

As a 3D Object

- Volumetric
- Can be shaded
- Can cast shadows
- Can navigate through or around letters
- Cinematic
- Works for real-world-size

82

Text as 3D objects can be volumetric, shaded, cast shadows, and allow navigation through and around letters. This cinematic approach works for real-world-size text and can create a dramatic effect.



Thanks! Hope you found this useful.