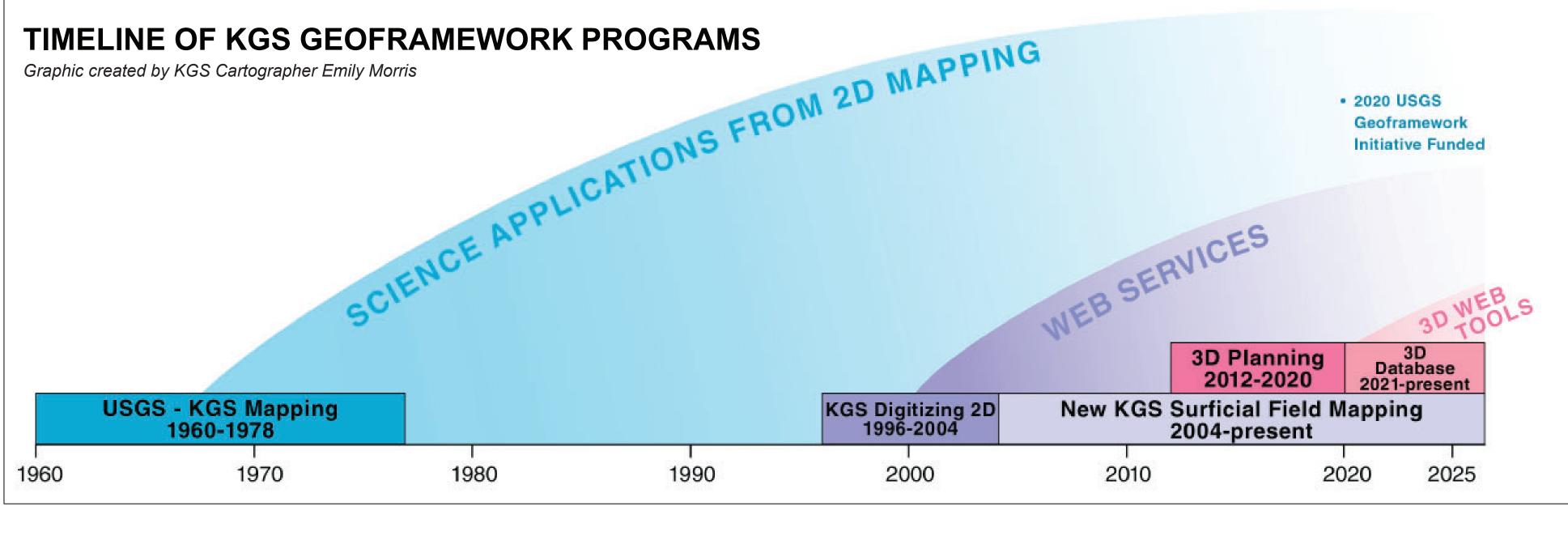
Kentucky Geological Survey

Abstract:

The Kentucky Geological Survey continues its efforts to develop 3D data elements from the best available 2D resources while documenting best practices for these processes and for the dissemination of available products to researchers and stakeholders. Digitized structure contours, geologic map units, and Digital Elevation Models were utilized within ArcGIS Pro to construct 3D surfaces for shale units in central and eastern Kentucky as well as the Jackson Purchase, with interpolative iterations being performed until each 3D object best represented current geologic understanding. Automated 3D surface interpolation using Jupyter Notebooks in ArcGIS Pro can be an efficient and effective method for creating 3D multipatch objects which closely represent geologic features that are otherwise difficult to visualize. For shales, 3D geologic units were produced for the Cumberland Fm, the Kope/Clays Ferry Fm, the Osgood Fm, the Crab Orchard Fm, the Borden Fm, and Pennsylvanian coal-baring marine shales. Regionally, the interpolated 3D shale units reflect the expected Cincinnati Arch structure when vertically exaggerated and will be useful in future landslide modeling and analysis. An in-depth inventory of data for surface interpolation in the Jackson Purchase was also conducted and datasets containing relevant structure contours were geo-referenced and digitized before being converted to 3D surfaces. Tops in the Jackson Purchase were modeled for the New Albany Shale, Paleozoic-Era bedrock, the McNairy Fm, the Porters Creek Clay, and the Wilcox Fm. These Jackson Purchase units represent some of the only remaining post-Paleozoic-Era strata in Kentucky and will be useful for studying erosional processes and seismic activity within continental deposits. In addition to stratigraphic horizons, a predictive 3D depth to bedrock model for the Ohio River Valley in Kentucky was interpolated using predictive models by Shangguan, et al. (2016) and USACE Hydrographic Surveys, showing river alluvium and bathymetric information, with geographic context. An elevation profile of the modeled Ohio River Valley's 3D bedrock surface closely aligned with the 1957 USGS Water Supply Paper #1411's Plate 3 deep channel bedrock profile, demonstrating the accuracy these methods can lend for modeling geologic surfaces in 3D.



Acknowledgemnts:

Funding for this project:

The U.S. Geoframework Initiative through the USGS-NCGMP Statemap Program

Global depth-to-bedrock framework:

Shangguan, W., T. Hengl, J. Mendes de Jesus, H. Yuan, and Y. Dai (2017), Mapping the global depth to bedrock for land surface modeling, J. Adv. Model. Earth Syst., 9, 65–88, doi:10.1002/2016MS000686.

Ohio River Hydrologic Survey Data:

U.S. Army Corps of Engineers (USACE) at https://www.arcgis.com/apps/dashboards/4b8f2ba307684cf597617bf1b6d2f85d

This project is built upon the multi-decadal effort and resulting data at the Kentucky Geological Survey.

A special shoutout to Anna Pearson at the University of Washington for helping to develop our framework for 3D interpolation.

Software is provided through the ESRI enterprise license at the University of Kentucky.



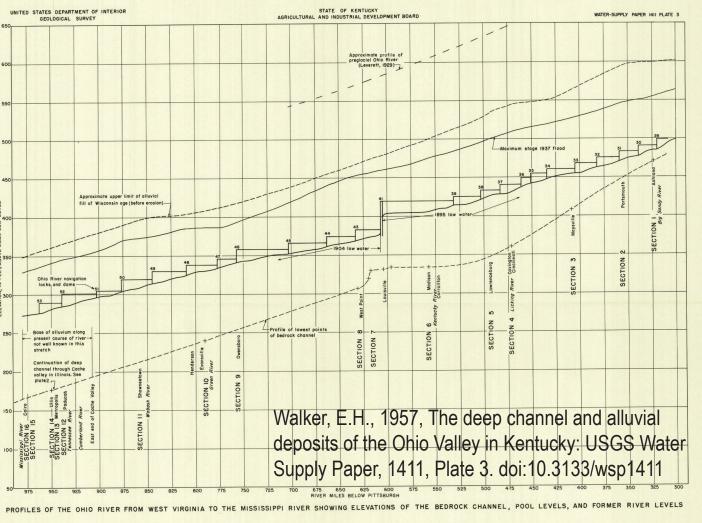
DEVELOPING A 3D GEOFRAMEWORK MODEL FOR KENTUCKY USING BEST AVAILIBLE DATA AND PRACTICES Robinson, Devan M., Andrews, William M., Hickman, John B.



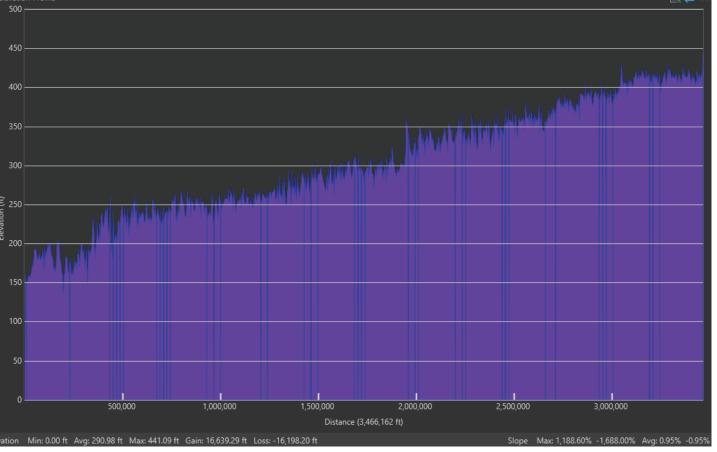
Data Structure:

	nts v 4 ×		
4 💼	Project	(€) () () Project ∨ Databases ∨	ت •
Þ 🗑	Maps	Name Type	
Þ 🧧	Toolboxes	R Contours.gdb File Geo	odatabase
6	Notebooks	🗇 Data.gdb File Ger	odatabase
4 🕞	Databases	Extents.gdb File Ger	odatabase
	🕞 Contours.gdb	Grids.gdb File Ge	odatabase
	B GeologicMap	-	odatabase
	Data.gdb		odatabase
	Extents.gdb	-	odatabase
	B GeologicMap		
	Grids.gdb		odatabase
	cells10m	Voxels.gdb File Ger	odatabase
	cells25m		
	Elliszon		
	 Eliszon Eliszon Cells500m 		
	▷ 🖽 cells1000m		
	▷ 📾 r1230300		
	▷ 🕮 r1240300		
	▷ 🖽 r1240500		
	▷ 🕮 r1250500		
	▷ 🕮 r3410300		
	▷ 🕮 r12303001		
	▷ 🖽 r12303002		
	▷ 🕮 r12403001		
	▷ 🕮 r12403002		
	▷ 🖽 r12405001		
	▷ 🖽 r33725001		
4	Isopachs.gdb		
	B GeologicMap		
	ConlineVisuals.gdb		
	Points.gdb		
	PointsVals		
	Uncertainty.gdb		
	B GeologicMap		
		9 Items 0 Items Selected	
	Voxels.gdb	9 Items 0 Items Selected	
		9 Items 0 Items Selected	
Jackson			
1 Jackson	Voxels.gdb		
Insert	Voxels.gdb	olygons 🔲 CreatingOnlineVisuals X 🔲 ExportSLPK 📵 ScannedMapDIgitizer 💽 JP_Contours 🏦 JP_Contours_3D	
Insert	Voxels.gdb		
Insert	Voxels.gdb ✓ Purchase Il SamplingFromPo Cell Help	olygons 🗓 CreatingOnlineVisuals X 🗓 ExportSLPK 📵 ScannedMapDigitizer 💽 JP_Contours 🏦 JP_Contours_3D	
Insert 5 1 2	 Voxels.gdb ✓ Purchase I SamplingFromPo Cell Help ↑ ↓ ▶ Run C #Takes in raster grids and 	Alygons II CreatingOnlineVisuals X II ExportSLPK II ScannedMapDigitizer II JP_Contours A JP_Contours_3D	
Insert	Voxels.gdb ✓ Purchase ① SamplingFromPo Cell Help ↑ ↓ ▶ Run C #Takes in raster grids an #Import necessary modules import arcpy	Alygons II CreatingOnlineVisuals X II ExportSLPK II ScannedMapDigitizer II JP_Contours A JP_Contours_3D	
Insert	 Voxels.gdb Purchase [] SamplingFromPo Cell Help ↑ ↓ ▶ Run Co #Takes in raster grids an #Import necessary modules import arcpy import pandas as pd 	Alygons II CreatingOnlineVisuals X II ExportSLPK II ScannedMapDigitizer II JP_Contours II JP_Contours_3D	
Insert 1 1 2 3 4 5 6 7	 Voxels.gdb Purchase Cell Help ▲ ▲ ▶ Run Cell Help Cell Help ▲ ▲ ▶ Run Cell Help Cell Help Takes in raster grids and an arcpy import arcpy import metadation 	Alygons II CreatingOnlineVisuals X II ExportSLPK II ScannedMapDigitizer II JP_Contours II JP_Contours_3D	
Insert 1 2 3 4 5 6 7 8	Voxels.gdb ✓ Purchase I SamplingFromPor Cell Help ↑ ↓ ▶ Run C #Takes in raster grids au #Toport necessary modules import necy import nectadat from arcpy import metadat	Alygons II CreatingOnlineVisuals X II ExportSLPK II ScannedMapDigitizer II JP_Contours II JP_Contours_3D	
Insert 1 I 2 3 4 5 6 7 8 9 10	Voxels.gdb ✓ Purchase II SamplingFromPoc Cell Help ↑ ↓ ▶ Run C #Takes in raster grids an #Import necessary modules import pandas as pd from arcpy import metadati import pdb from arcpy import env	Alygons II CreatingOnlineVisuals X II ExportSLPK II ScannedMapDigitizer II JP_Contours II JP_Contours_3D	
Insert Insert I I I I I I I I I I I I I	Voxels.gdb ✓ Purchase ① SamplingFromPoc Cell Help ↑ ↓ ▶ Run Co #Takes in raster grids an #Import necessary modules import pandas as pd from arcpy import metadati import polds as pd from arcpy import env from arcpy import env from arcpy.sa import * import os #Read in source text file	blygons II CreatingOnlineVisuals X II ExportSLPK II ScannedMapDigitizer II JP_Contours II JP_Contours_3D	
Insert 1 I 2 I 3 4 5 6 7 8 9 9 10 11 12 13	Voxels.gdb ✓ Purchase ① SamplingFromPoc Cell Help ↑ ↓ ▶ Run Co #Takes in raster grids an #Import necessary modules import pandas as pd from arcpy import metadati import polds as pd from arcpy import env from arcpy import env from arcpy.sa import * import os #Read in source text file	blygons CreatingOnlineVisuals X II ExportSLPK II ScannedMapDigitizer II JP_Contours A JP_Contours_3D	e text file
Insert 1 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15	<pre> Voxels.gdb Purchase SamplingFromPo Cell Help</pre>	blygons CreatingOnlineVisuals X ExportSLPK ExportSLPK ScannedMapDigitizer JP_Contours A JP_Contours_3D	e text file
Insert 1 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 6 17	Voxels.gdb ✓ Purchase □ SamplingFromPore Cell Help ★ ↓ ▶ Run C #Takes in raster grids and #Takes in raster grids and #Takes in raster grids and #Toport arcpy import netadati import pandas as pd from arcpy import env from arcpy import env from arcpy.sa import * import os #Read in source text fild file = open("A:\\kyGeofrailines = file.readlines() file.close() path=str(lines[0].split('pathMetadata=lines[2].split()	blygons CreatingOnlineVisuals X C ExportSLPK C ScannedMapDigitizer JP_Contours A JP_Contours_3D tode Code Code Code Code Code Code Code Code	e text file
Insert 1 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 6 17	Voxels.gdb ✓ Purchase [] SamplingFromPoc Cell Help ★ ↓ ▶ Run CC #Takes in raster grids an #Import necessary modules import pandas as pd from arcpy import metadati import pandas as pd from arcpy import env from arcpy.sa import env from arcpy.sa import * import os #Read in source text file file = open("A:\\KyGeofra lines - file.readlines() file.close() path=str(lines[0].split(")	blygons CreatingOnlineVisuals X C ExportSLPK C ScannedMapDigitizer JP_Contours A JP_Contours_3D tode Code Code Code Code Code Code Code Code	e text file
Insert Insert	Voxels.gdb ✓ Purchase [] SamplingFromPoc Cell Help ★ ↓ ▶ Run C #Takes in raster grids an #Import necessary modules import pandas as pd from arcpy import metadati import padas as pd from arcpy import env from arcpy.sa import * import os #Read in source text file file = open("A:\\KyGeofra lines = file.readlines() file.close() path-str(lines[0].split('pathMetadata=lines[2].split #If it doesn't already end	blygons ① CreatingOnlineVisuals × ① ExportSLPK ② ScannedMapDlgitizer 〗 JP_Contours ⚠ JP_Contours_3D Node v ■ nd creates inputs to export to ArcOnLine s ta as md e (the same as used for BasicWorkflow) amework_2023\\ClayShaleUnits\\Testing\\Sampling\\SourceTextFile.txt", 'r') #Must replace with name of source '\n'')[0] ht("\n")[0] ht("\n")[0] xist, make TINFiles folder	e text file
Insert Insert	<pre> Voxels.gdb Purchase SamplingFromPo Cell Help Purchase SamplingFromPo Cell Help Purchase SamplingFromPo Cell Help Purchase SamplingFromPo Samport necessary modules Samport necessary mod</pre>	<pre>bygons @ CreatingOnlineVisuals × @ ExportSLPK @ ScannedMapDlgitizer @ JP_Contours A JP_Contours_3D bode</pre>	e text file
Insert I Section I S	<pre> Voxels.gdb Purchase SamplingFromPo Cell Help Toport necessary modules import necessary modules import pandas as pd from arcpy import metadati import pb from arcpy import env from arcpy.sa import * import os #Read in source text file file = open("A:\\KyGeofra lines = file.readlines() path=str(lines[0].split() pathSource=lines[6].split() #If it doesn't already en newpath = path + '\\TINFi </pre>	<pre>bygons @ CreatingOnlineVisuals × @ ExportSLPK @ ScannedMapDlgitizer @ JP_Contours A JP_Contours_3D bode</pre>	e text file
Insert Insert	<pre> Voxels.gdb Purchase SamplingFromPo Cell Help To Purchase Build A Purchase Build A</pre>	<pre>blygons ① CreatingOnlineVisuals × ② ExportSLPK ③ ScannedMapDlgitizer ③ JP_Contours ④ JP_Contours_3D iode v v d creates inputs to export to ArcOnLine s ta as md e (the same as used for BasicWorkflow) amework_2023\\ClayShaleUnits\\Testing\\Sampling\\SourceTextFile.txt", 'r') #Must replace with name of source '\n')[0] lit("\n')[0] t("\n')[0] trist, make TINFiles folder lies' upath):</pre>	e text file
Insert Insert	<pre> Voxels.gdb Voxels.gdb Voxels.gdb Voxels.gdb Voxels.</pre>	<pre>blygons @ CreatingOnlineVisuals × @ ExportSLPK @ ScannedMapDIgitizer P_Contours A P_Contours_3D bode</pre>	e text file
Insert b C C C C C C C C C C C C C	<pre> Voxels.gdb Purchase SamplingFromPo Cell Help Town Point Point #Takes in raster grids an #Import necessary modules import necessary modules import necessary modules import necessary modules import pandas as pd from arcpy import metadati import pb from arcpy import env from arcpy.sa import * import os #Read in source text file file = open("A:\\KyGeofra lines = file.readlines() file.close() path=str(lines[0].split() pathSource=lines[6].split() pathSource=lines[6].split() if not os.path.exists(nee os.makedirs(newpath) #Make multipatches arcpy.env.workspace = pat gdb=arcpy.env.workspace = pat gd</pre>	<pre>blygons ① CreatingOnlineVisuals × ② ExportSLPK ③ ScannedMapDlgitizer ③ JP_Contours ④ JP_Contours_3D iode v v d creates inputs to export to ArcOnLine s ta as md e (the same as used for BasicWorkflow) amework_2023\\ClayShaleUnits\\Testing\\Sampling\\SourceTextFile.txt", 'r') #Must replace with name of source '\n')[0] lit("\n')[0] t("\n')[0] trist, make TINFiles folder lies' upath):</pre>	e text file
Insert I 2 I I I I I I I I I I I I I	<pre> Voxels.gdb Voxels.gdb Voxels.gdb Purchase Cell Help A ↓ ▶ Run CC #Import necessary modules import arcpy import pandas as pd from arcpy import metadati import pandas as pd from arcpy import env from arcpy import env from arcpy import env from arcpy.sa import * import os #Read in source text fild file = open("A:\\KyGeofra lines = file.readlines() file.close() path=str(lines[0].split(' pathMetadata=lines[2].spl pathSource=lines[6].split(' pathMetadata=lines[2].spl pathSource=lines[6].split(' #If it doesn't already ee neupath = path + '\\TINFi if not os.path.exists(new os.makedirs(newpath) #Make multipatches arcpy.env.workspace = pat gdb=arcpy.env.workspace = pat gdb=arc</pre>	<pre>blygons @ CreatingOnlineVisuals × @ ExportSLPK @ ScannedMapDIgitizer P_Contours A P_Contours_3D bode</pre>	e text file
Insert Insert	<pre> Voxels.gdb Purchase SamplingFromPo Cell Help Takes in raster grids an #Import necessary modules import arcpy import pandas as pd from arcpy import metadati import pandas as pd from arcpy import env from arcpy.sa import env from arcpy.sa import * import os #Read in source text fild file = open("A:\\KyGeofra lines = file.readlines() file.close() path=str(lines[0].split(' pathMetadata=lines[2].spl pathSource=lines[6].split(' if not os.path.exists(new os.makedirs(newpath) #Make multipatches arcpy.env.workspace = pat gdb-arcpy.env.workspace = arcpy.Li listClasses=[] </pre>	<pre>blygons @ CreatingOnlineVisuals × @ ExportSLPK @ ScannedMapDIgitizer P_Contours A P_Contours_3D bode</pre>	e text file
Insert Insert	<pre> Voxels.gdb Purchase SamplingFromPo Cell Help Town Point I accessary modules import necessary modules import necessary modules import necessary modules import necessary modules import arcpy import pandas as pd from arcpy import env from arcpy import env from arcpy.sa import * import os #Read in source text fild file = open("A:\\KyGeofra lines = file.readlines() file.close() path=str(lines[0].split(' pathMetadata=lines[2].spl pathSource=lines[6].split(' pathMetadata=lines[2].spl pathSource=lines[6].split(' if not os.path.exists(new os.makedirs(newpath) #Make multipatches arcpy.env.workspace = pat gdb=arcpy.env.workspace = pat listClasses=[] print(featureClasses) for x in featureClasses: print(x) env.workspace = path </pre>	<pre>bygons CreatingOnlineVisuals X CreatingOnlineVisuals X CreatingOnlineVisuals X CreatingOnlineVisuals X CreatingOnlineVisuals X Creating ScannedMapDigitizer II JP_Contours ID code Control Creates inputs to export to ArcOnLine s ta as md e (the same as used for Basickonkflow) amework_2023\\ClayShaleUnits\\Testing\\Sampling\\SourceTextFile.txt", 'r') #Must replace with name of source '\n'')[0] itt('\n')[0] itt('\n')[0] itt('\n')[0] itt('\n')[0] itt th + "\\Grids.gdb" + "\\Grids.gdb"</pre>	e text file
Insert 1 1 2 3 4 5 6 7 8 9 10 11 11 12 13 14 15 16 16 17 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 33 34 35	<pre> Voxels.gdb Purchase SamplingFromPo Cell Help Takes in raster grids an #Import necessary modules import arcpy import metadata import pandas as pd from arcpy import metadata import pdb from arcpy import env from arcpy.sa import * import os #Read in source text file file = open("A:\\KyGeofra ines = file.readlines() file.close() path=str(lines[0].split() pathSource=lines[6].split() pathSource=lines[6].split() #If it doesn't already en ewpath = path + '\\TINFi if not os.path.exists(new os.makedirs(newpath) #Make multipatches arcpy.env.workspace = pat gdb=arcpy.env.workspace = pat gdb=arcpy.env.workspace = pat print(featureClasses) for x in featureClasses: print(x) env.workspace = pat env.scratchWorkspace = pat </pre>	<pre>bygons ① CreatingOnlineVisuals × ② ExportSLPK ② ScannedMapDigitizer ② JP_Contours À JP_Contours_3D ind creates inputs to export to ArcOnline s ta as md e (the same as used for Basickkorkflow) amework_2023\\ClayShaleUnits\\Testing\\Sampling\\SourceTextFile.txt", 'r') #Must replace with name of source '\n' [0] it('\n')[0] it('\n')[0] it('\n')[0] xist, make TINFiles folder lles' upath): th + "\\Grids.gdb" istRasters('r*")+arcpy.ListRasters("k*")+arcpy.ListRasters("=*")</pre>	e text file
Insert 1 1 2 3 4 5 6 6 7 7 8 9 9 10 11 12 13 14 15 16 16 16 177 18 19 20 21 22 23 24 22 23 24 22 26 27 28 29 30 34 35 36	<pre> Voxels.gdb Purchase SamplingFromPo Cell Help To Purchase Build of the purchase Build of the purchase Build of the purchase Build of the purchase of t</pre>	<pre>bygons @ CreatingOnlineVisuals × @ ExportSLPK @ ScannedMapDigitizer @ JP_Contours_3D bode</pre>	e text file
Insert 21 1 23 4 5 6 7 8 9 10 112 13 14 15 156 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 322 33 34 35 36 7 38 36	<pre> Voxels.gdb Purchase SamplingFromPo Cell Help Town Point Point #Takes in raster grids an #Import necessary modules import arcpy import arcpy import metadati import pandas as pd from arcpy import env from arcpy import env from arcpy.sa import * import os #Read in source text file file = open("A:\\KyGeofra lines = file.readlines() file.close() pathAetadata=lines[2].split(' pathAetadata=lines[6].split(' pathAetadata=lines[6].split(' if it doesn't already en newpath = path + '\\TINFi if not os.path.exists(new os.makedirs(newpath) #Make multipatches arcpy.env.workspace = pat gdb=arcpy.env.workspace = pat gdb=arcpy.env.workspace = pat for x in featureClasses) for x in featureClasses; for x in featureClasses; for x in featureClasses; print(x) env.workspace = path env.scratchWorkspace = path env.scratc</pre>	<pre>bygons [] CreatingOnlineVisuals X [] ExportSLPK [] ScannedMapDigitizer [] /P_Contours]D bode v d creates inputs to export to ArcOnline s ta as md e (the same as used for BasicKorkfiow) amework_2023\\ClayShaleUnits\\Testing\\Sampling\\SourceTextFile.txt", 'r') #Must replace with name of source "\n")[0] iit('\</pre>	e text file
Insert 2 Image: Constraint of the second seco	<pre> Voxels.gdb Purchase SamplingFromPo Cell Help Takes in raster grids an #Import necessary modules import arcpy import pandas as pd from arcpy import metadat import os #Read in source text fild file = open("A:\\KyGeofra lines = file.readlines() path=str(lines[0].split(' pathMetadata=lines[2].spl pathSource-lines[0].split(' pathMetadata=lines[2].spl pathSource-lines[6].split(' #If it doesn't already ev newpath = path + '\\TINFi if not os.path.exists(new os.makedirs(newpath) #Make multipatches arcpy.env.workspace = pat gdb=arcpy.env.workspace = pat gdb=arcpy.env.workspace = pat env.workspace = path env.scratcNworkspace = path env.workspace = path e</pre>	<pre>bygons [] CreatingOnlineVisuals X [] ExportSLPK [] ScannedMapDigitizer [] /P_Contours]D bode v d creates inputs to export to ArcOnline s ta as md e (the same as used for BasicKorkfiow) amework_2023\\ClayShaleUnits\\Testing\\Sampling\\SourceTextFile.txt", 'r') #Must replace with name of source "\n")[0] iit('\</pre>	e text file
Insert 2	<pre> Voxels.gdb Purchase SamplingFromPo Cell Help Takes in raster grids an #Import necessary modules import arcpy import metadata import pandas as pd from arcpy import metadata import pdb from arcpy import env from arcpy.sa import * import os #Read in source text file file = open("A:\\KyGeofra lines = file.readlines() path=str(lines[0].split() pathetadata=lines[2].split() pathscource=lines[0].split() pathetadata=lines[2].split() arcpy.env.workspace = pat gdb=arcpy.env.workspace = pat for x in featureClasses: print(x) env.workspace = path env.scratchWorkspace = path en</pre>	<pre>bygons [] CreatingOnlineVisuals X [] ExportSLPK [] ScannedMapDigitizer [] /P_Contours]D bode v d creates inputs to export to ArcOnline s ta as md e (the same as used for BasicKorkfiow) amework_2023\\ClayShaleUnits\\Testing\\Sampling\\SourceTextFile.txt", 'r') #Must replace with name of source "\n")[0] iit('\</pre>	

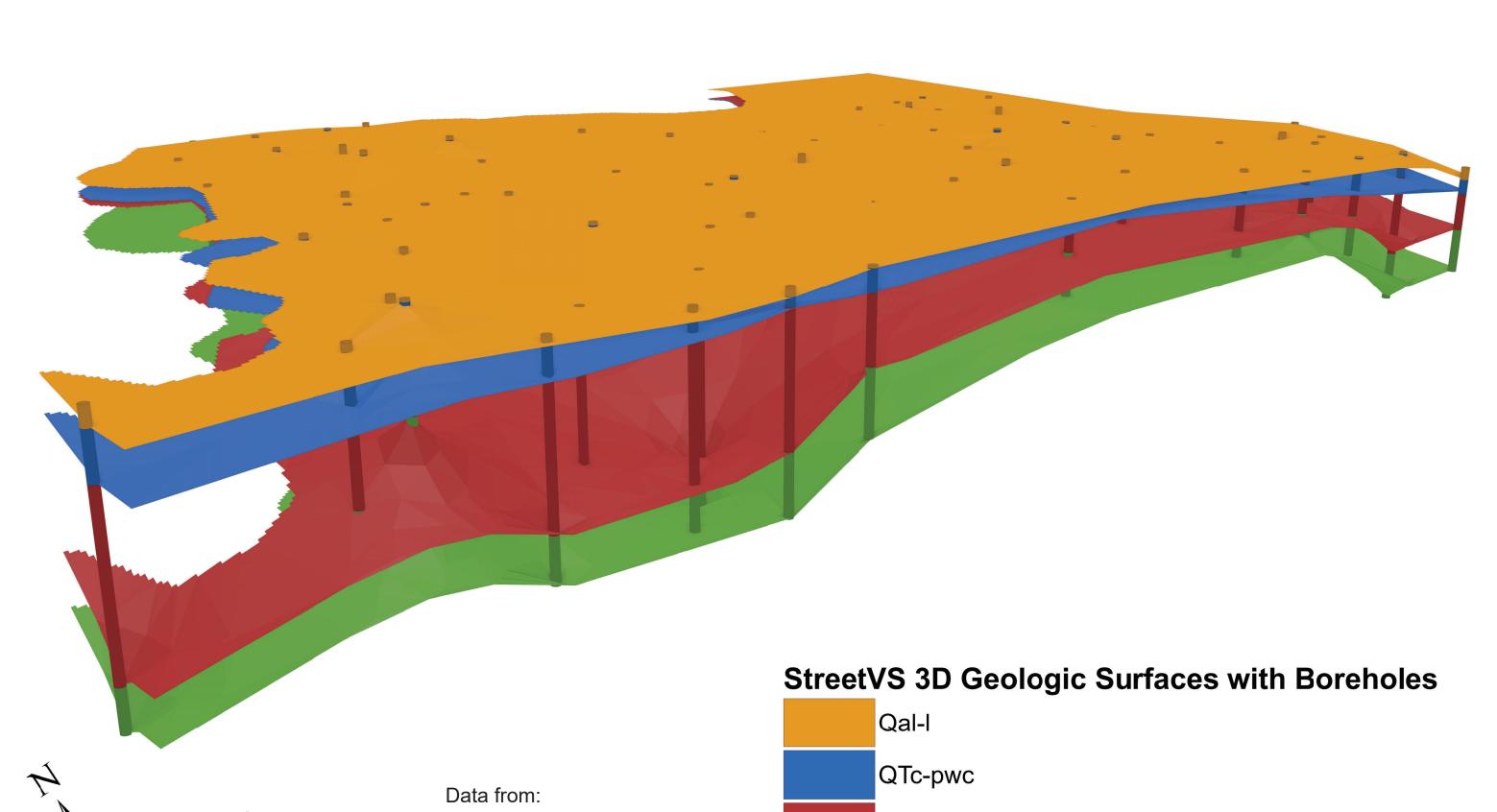
Model Validation:

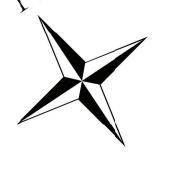


Profile of the Ohio River from West Virginia to the Mississippi River showing elevations of the bedrock channel, pool levels, and former river levels.



Elevation profile for the modeled Ohio River Valley deep channel bedrock surface. Measured in ArcGIS Pro, this closely agrees USGS Water-Supply Paper 1411 by Walker, 1957.

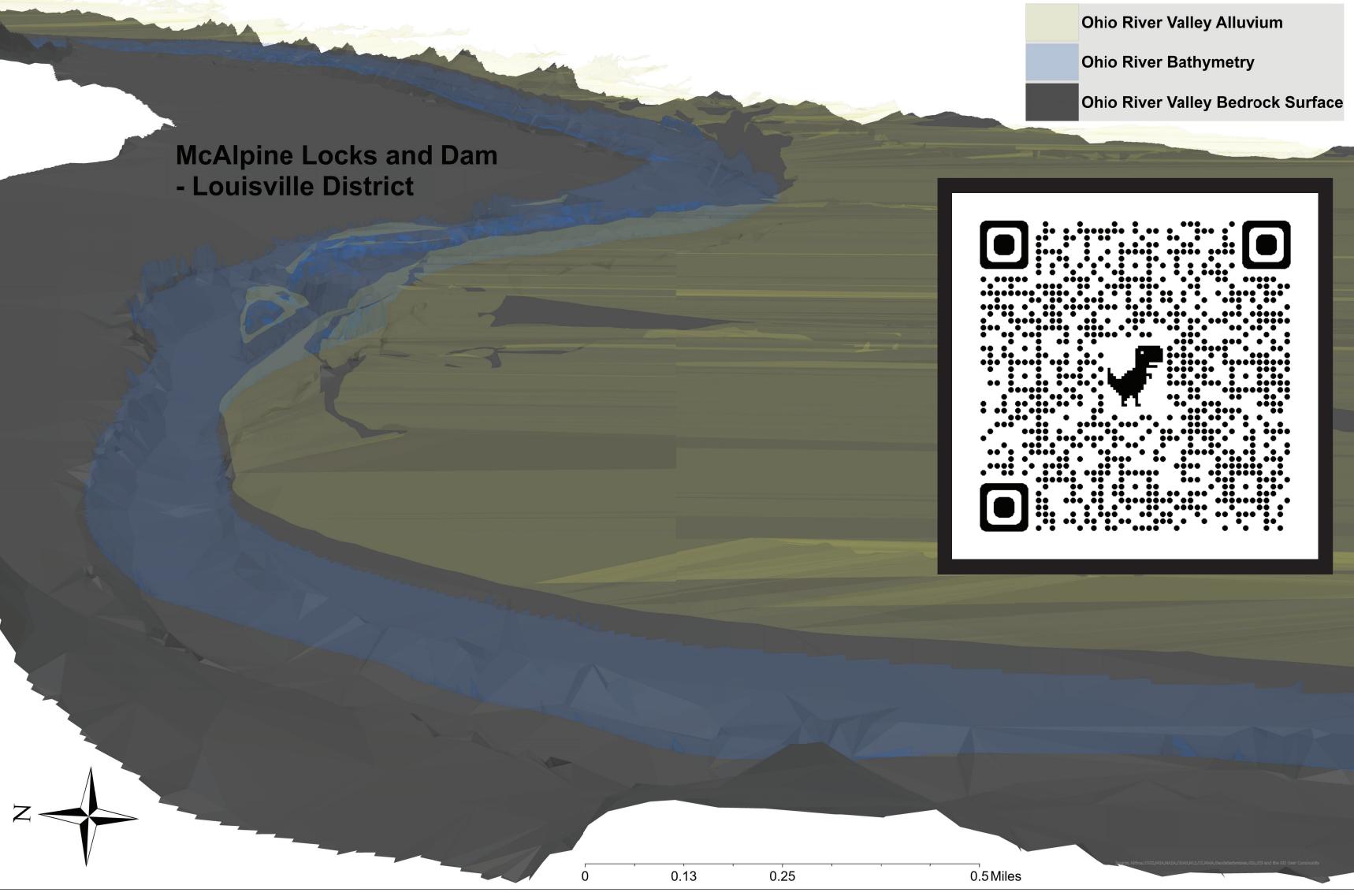




Street, R., Woolery, E., Wang, Z., and Harik, I.E., 1997. esponse spectra and seismic coefficients for building ovisions in western Kentucky: Engineering Geology, Vol. 46, Issues 3-4, 331-347 p. https://doi.org/10.1016/S0013-7952(97)00010-(

Scene displaying 3D interpolated surfaces for Alluvium, Tertiary and Quaternary continental deposits, Tertiary, and Cretaceous and Tertiary: Clayton and McNairy Formations in the Jackson Purchase region of Kentucky. Surfaces vertically exaggerated to 30x. Boreholes created using similar automation methods.

3D Ohio River Valley bedrock surface model (predicted):



Scene displaying sample of 3D Ohio River Valley Bedrock surface model, Ohio River Bathymetry, and River Valley Alluvium at Louisville KY. Scene vertically exaggerated to 10x to bring out 3D valley U-shape.

SE-GSA 2024: Poster 14-10, Booth No. 42

University of Kentucky, Kentucky Geological Survey

Scan QR codes with your phone and try them!

3D Jackson Purchase geologic surfaces:

Dwd TKcm



pint data measuring seismic wave velocity with correlation to Geologic units were used to make rasters from each point dataset. 3D boreholes were also made using the pre-formatted points with an automated script in Jupyter Notebooks within ArcGIS Pro. Features were then interpolated into 3D multipatches. (LEFT)

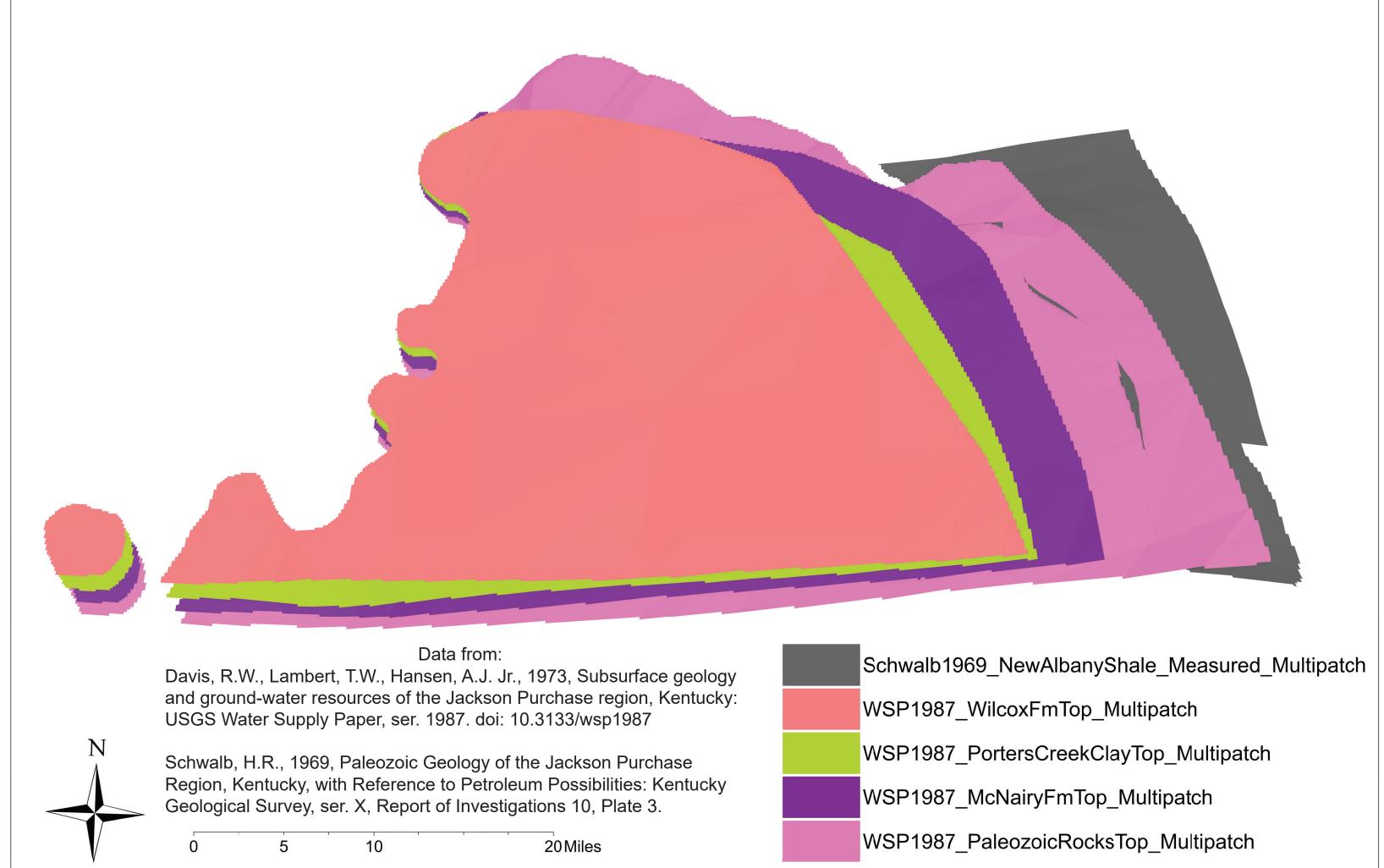
Structure contours were georeferenced and digitized in Arc Pro, converted to rasters, and then to 3D multipatches. Full dataset contains various partial and complete surfaces from the Geothermal project, the DVGQData project, the Solis project, and others. Contour data included a projected and measured set for the New Albany Shale. (RIGHT)

A Ohio River center line, a river edge line. a 500ft buffer line, and a river valley boundary line were all merged, and points were generated along these lines every 50 ft. The KY DEM, bedrock depth rasters, and Ohio River depth raster were then extracted to the points. A field for the final bedrock elevation was added and used to assign a single elevation for each point of the predicted bedrock surface by subtracting the bedrock depth from the KY DEM, and then for the points on the river also subtracting the river depth. The 1km predicted bedrock depth raster was used were there existed holes in the 250m raster. These points were converted into a 50ft raster before being interpolated into a 3D multipatch surface. (LEFT)

Using the 'DVGQ 24K GeoSciML Poly' and 'KGS_Contacts_24K' layers from KGS's 1:24,000 Geologic Map Data for Kentucky, formations were meticulously extracted using their sort codes and run through two KGS automated workflows in Jupyter Notebooks for creating 3D surface features, and 3D multipatches were interpolated and sorted by formation. The Jessamine Dome is clearly visible with 30x vertical exaggeration. (RIGHT)

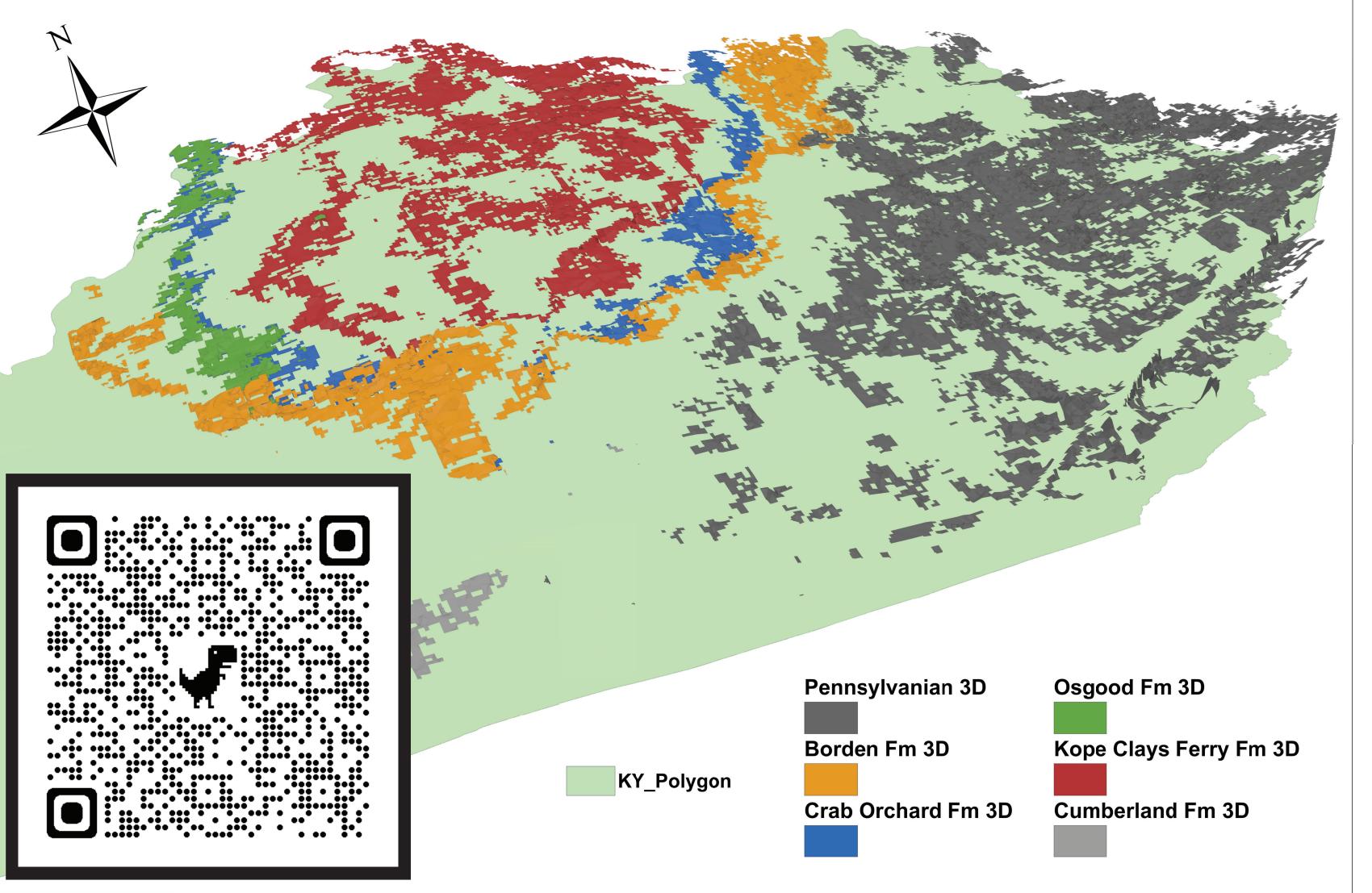


More 3D Jackson Purchase geologic surfaces:



Scene displaying 3D interpolated tops for the Paleozoic Rocks, McNairy Fm, Porters Creek Clay, Wilcox Fm, and the New Albany Shale in the Jackson Purchase region of Kentucky. Surfaces vertically exaggerated to 30x.

3D Clay Shale geologic surfaces:



Scene displaying 3D Clayshale surfaces for the Pennsylvanian Marine Coal-baring Shales, the Borden Fm, the Crab Orchard Fm, the Osgood Fm, the Kope Clays Ferry Fm, and the Cumberland Fm in Kentucky. Surfaces are vertically exaggerated to 30x. Data obtained from KGS's 1:24,000 Geologic Map Data for Kentucky.

Kentucky Geological Survey