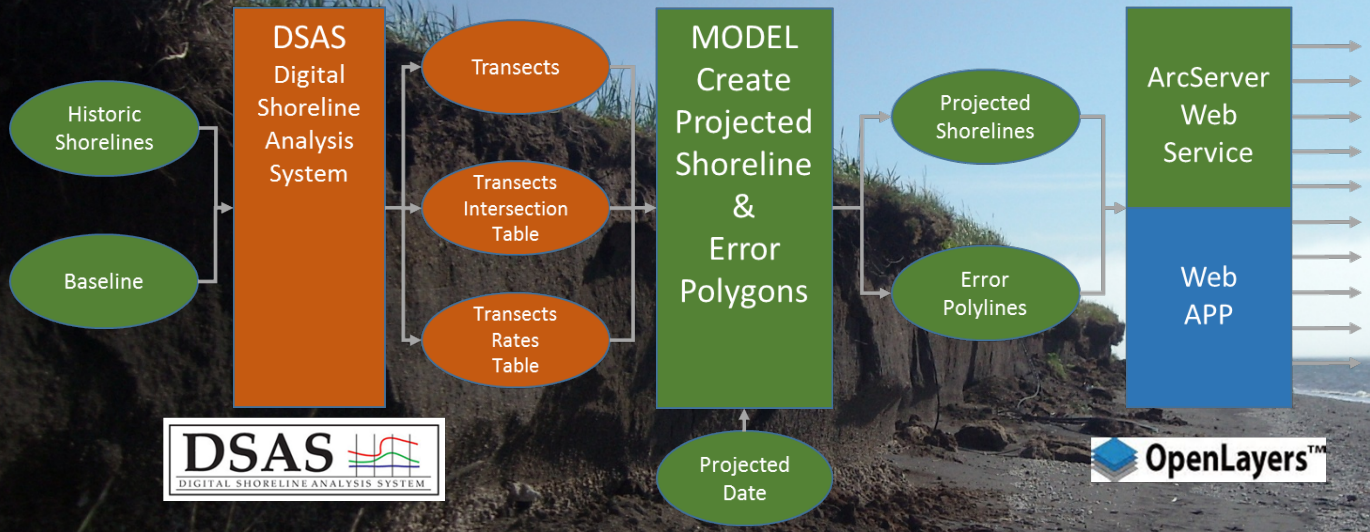
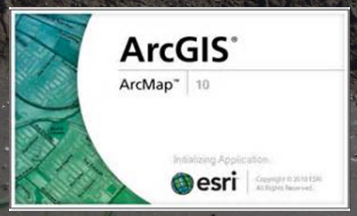
Creating and Serving Alaska Shoreline Change Predictions





Mike Hendricks
Geospatial Analyst
Alaska Division of Geological & Geophysical Surveys
3354 College Rd, Fairbanks AK 99709
mike.hendricks@alaska.gov

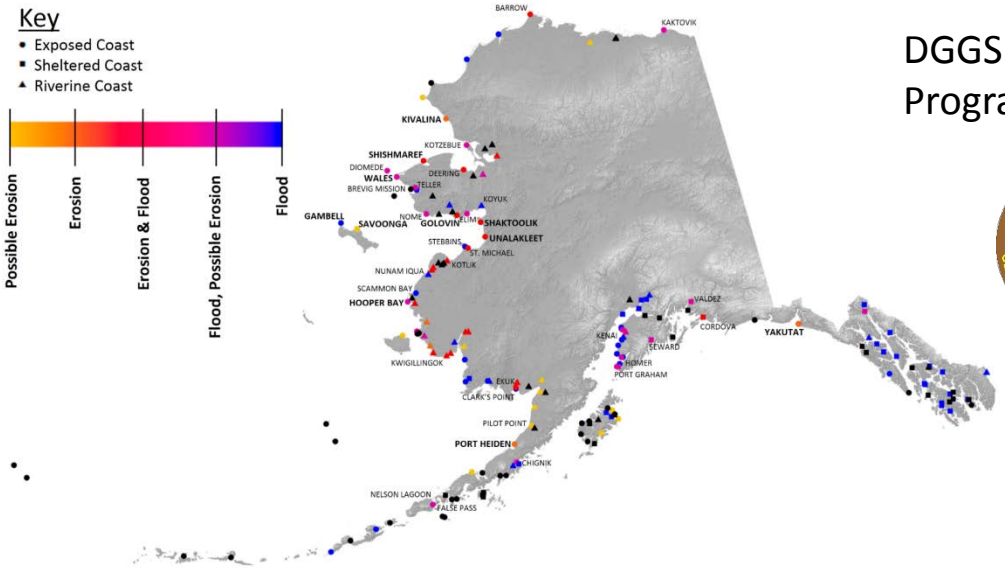
Agenda

- Coastal Program at DGGS
- Shoreline Change Prediction Requirements
- Model Builder & Python Scripting
- ArcGIS Models for Shoreline Change Prediction
- Creating GIS Web Services
- Creating Custom Web Applications

Coastal Program at DGGS

- Program launched in 2011 with seed funding from Coastal Impact Assistance Program
- Program objectives:
  - Increase quality/quantity of coastal baseline data
  - Provide shoreline change and coastal inundation tools
  - Encourage/develop coastal management resources for an Alaskan audience

Alaska Coastal Populations Vulnerable to Flooding and Erosion



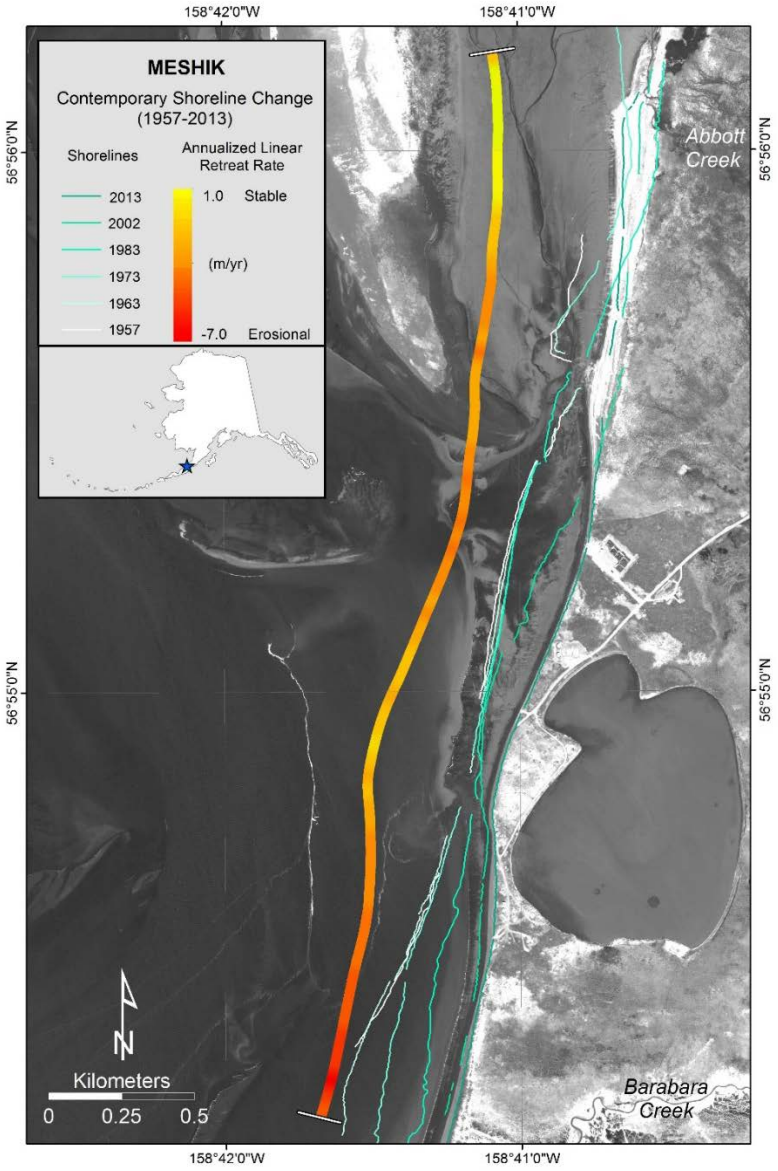
DGGS Coastal Hazards Program - Fairbanks, AK
Program Manager: Nicole Kinsman
nicole.kinsman@alaska.gov
907-451-5026

Prevailing Shoreline Change Prediction Display Methodology

- Collect Historical Shoreline Data

  Proxy indicator of shoreline position typically bluff top/vegetation line

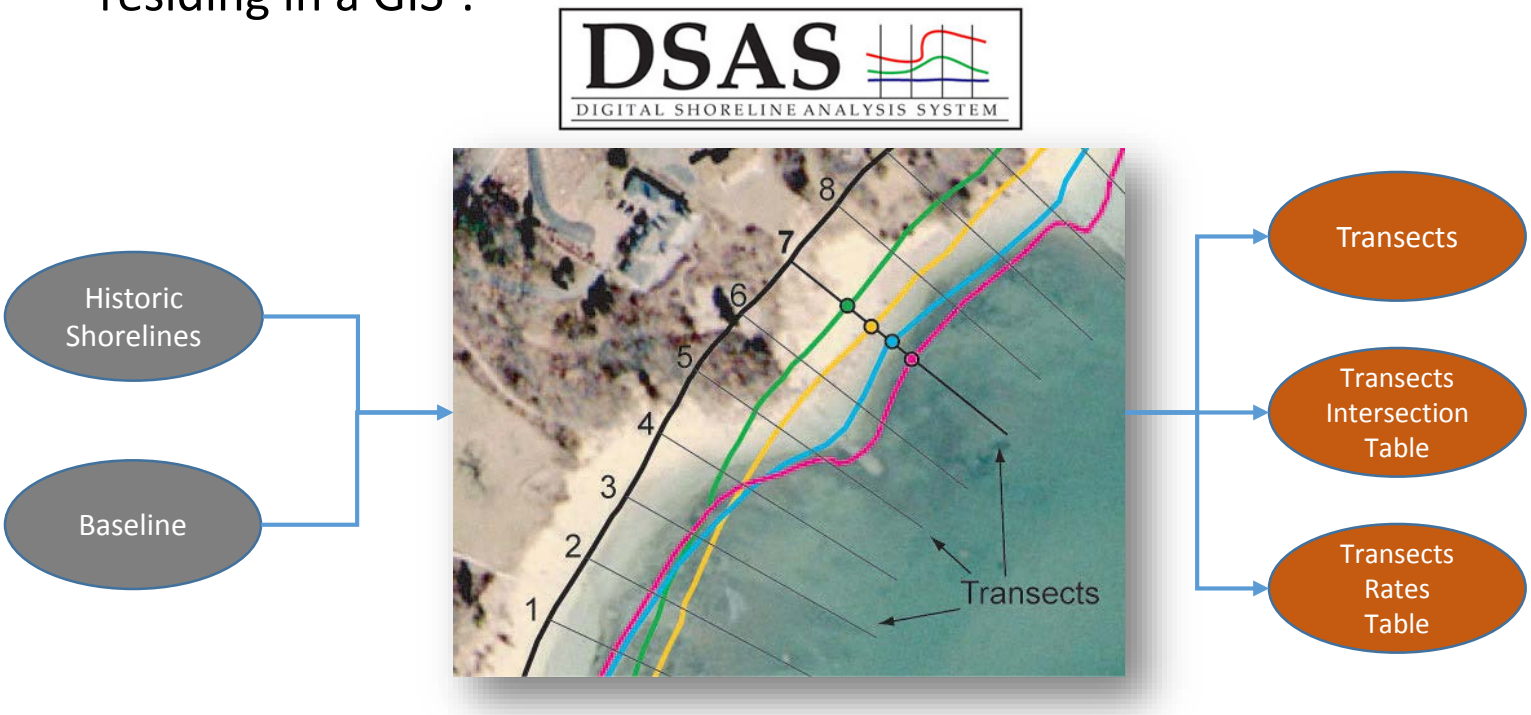
- Creation of change rates with Digital Shoreline Analysis System (DSAS) from USGS
- Display change rates along shoreline without explicitly mapping predicted shoreline at a future date.
- Difficult to effectively show prediction uncertainty along shoreline



Digital Shoreline Analysis System (DSAS)

DSAS version 4.3.4730 for ArcGIS 10

"The Digital Shoreline Analysis System (DSAS) is computer software that computes rate-of-change statistics from multiple historic shoreline positions residing in a GIS".



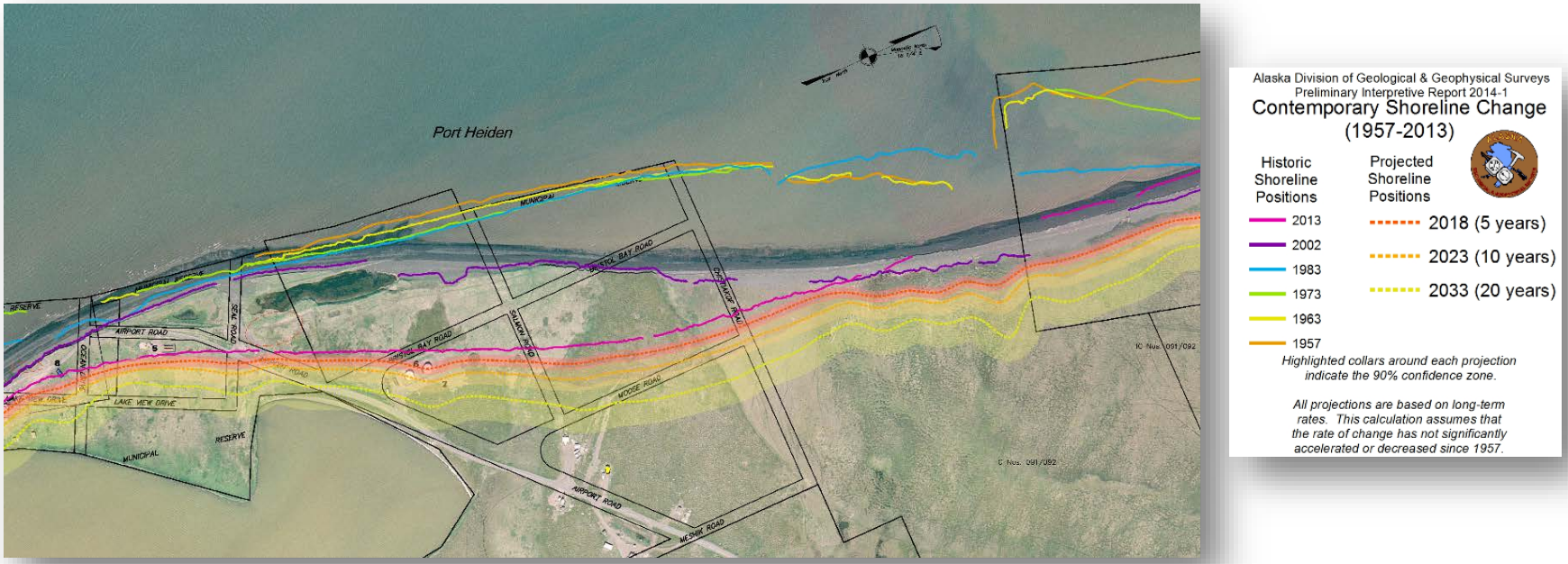
http://woodshole.er.usgs.gov/project-pages/DSAS/

Rate of Change Statistics provided by DSAS

- Simple End Point Rate
- **Linear Regression Rate**
  - Standard Error
  - Confidence Interval
  - R-Squared
- Weighted Linear Regression Rate
  - Standard Error
  - Confidence Interval
  - R-Squared
- Least Median of Squares

Method used in our web app

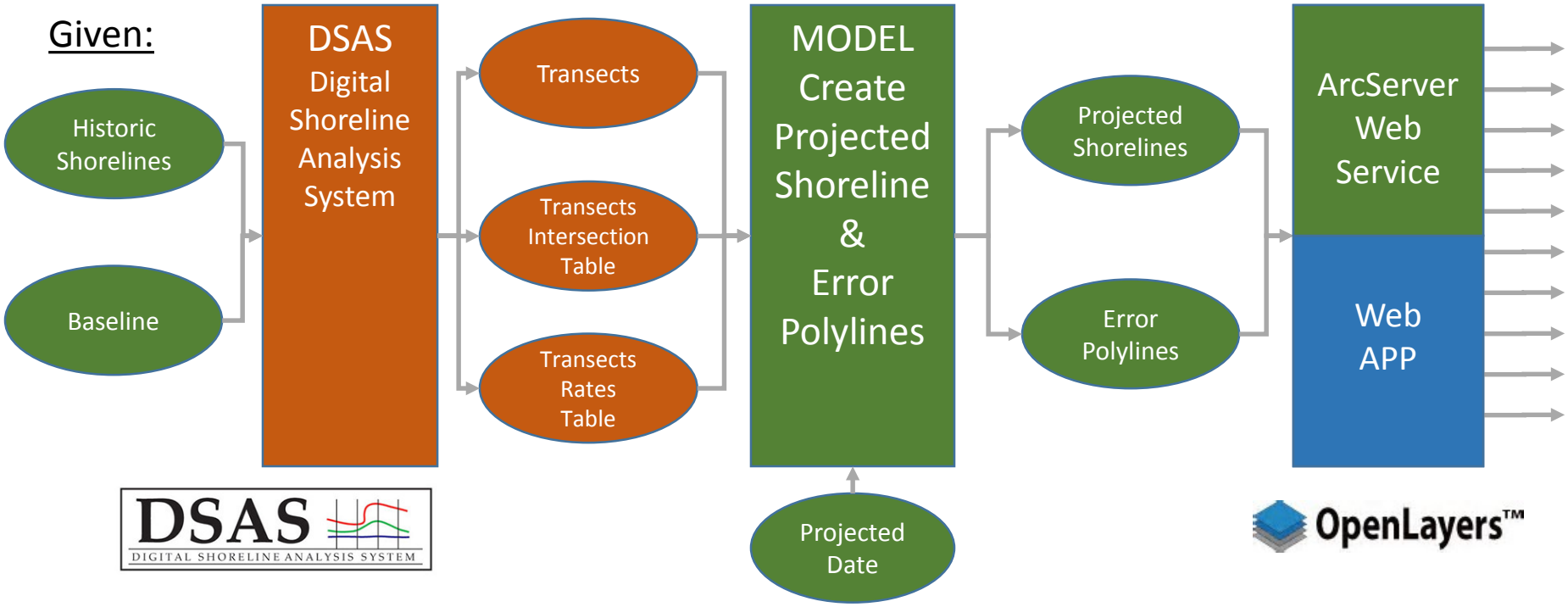
Goal: A set of Tools and Technologies to Calculate, Display, & Disseminate Projected Long Range Shoreline Positions



The previous manual GIS process to produce projected shorelines from DSAS rates of change output was labor intensive.

ArcGIS models and Python scripting allows automation and documentation of the process

The GIS based Task to Achieve Goal



Desired GIS based output:

- Predicted Shoreline for a user defined future date as a polyline
- Display variable location uncertainty along this predicted shoreline
- Deliver this information to the public with a web service & app

ArcGIS Model Builder

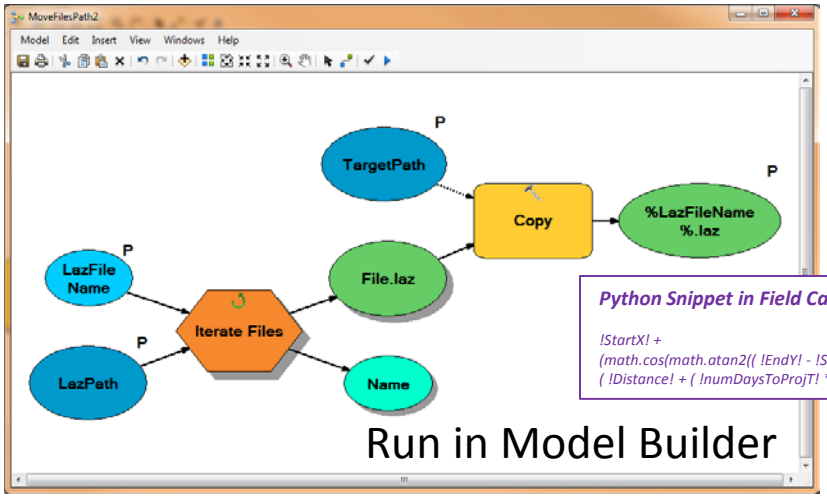
- Model Builder is an application you use to create, edit, and manage models.
- Models are workflows that string together sequences of geoprocessing tools, feeding the output of one tool into another tool as input.
- Model Builder can also be thought of as a visual programming language for building workflows.

ArcGIS Online Help

Models built in Model Builder are not meant to be the fast, efficient, or compact code, but… the visual process-flow interface is very compelling to many ArcGIS users.

ArcGIS Model Builder & Python Basics

Design Process Flow in Model Builder



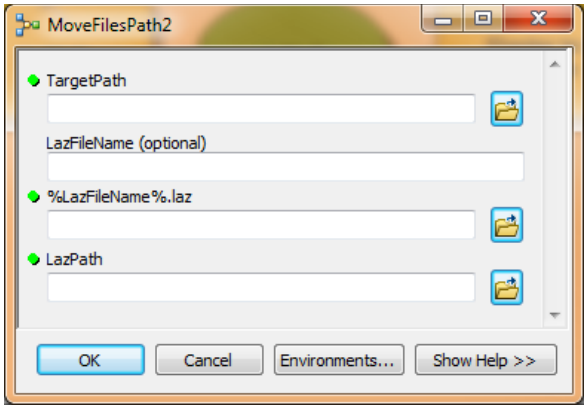
Run in Model Builder

Python Snippet in Field Calculations

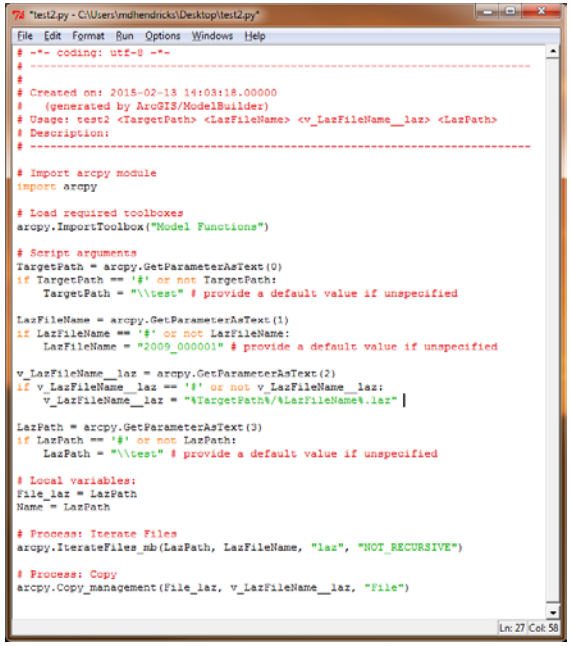
!StartX! +
*(math.cos(math.atan2((!EndY! - !StartY!),(!EndX! - !StartX!))) *
*(!Distance! + (!numDaysToProjT! * !Rate!/360)))*

or

Run from Dialog



Optionally: Export as Python Script



Alternate Work Flow:

- *Use Python exclusively to build tool*

Modeling Subtasks

1. Find Projected Shoreline Intersection for each Transect

   Given DSAS rate of change output data, calculate the location along a transect at some given future date

2. Create Shoreline from these points

   Given a set of predicted point locations along a set of transects, create a smoothed polyline representing the predicted shoreline at the given future date

3. Create Error Representation along Shoreline

   Given the location along the transect at some given future date and the Error Data from DSAS for each transect, create a geometric based error representation along the predicted shoreline

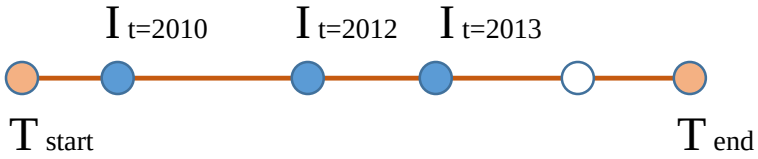
SubTask 1:

Calculate the location along a transect at some given future date

Given:

- transect
- historic shoreline intersections
- predicted rate of change along the transect from DSAS

Create: The location along the transect at some given future date



Python Snippet Used in Model:

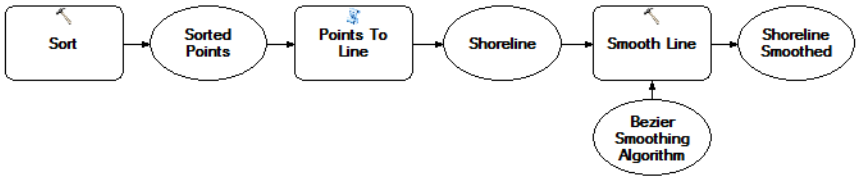
```
Calculate X Coordinate
!StartX! +
(math.cos(math.atan2(( !EndY! - !StartY! ),
( !EndX! - !StartX! ) )) *
( !Distance! + ( !numDaysToProjT! * !Rate!/360) ))

Calculate Y Coordinate
!StartY! +
 (math.sin(math.atan2(( !EndY! - !StartY! ),
( !EndX! - !StartX! ) )) *
( !Distance! + ( !numDaysToProjT! * !Rate!/360) ))
```

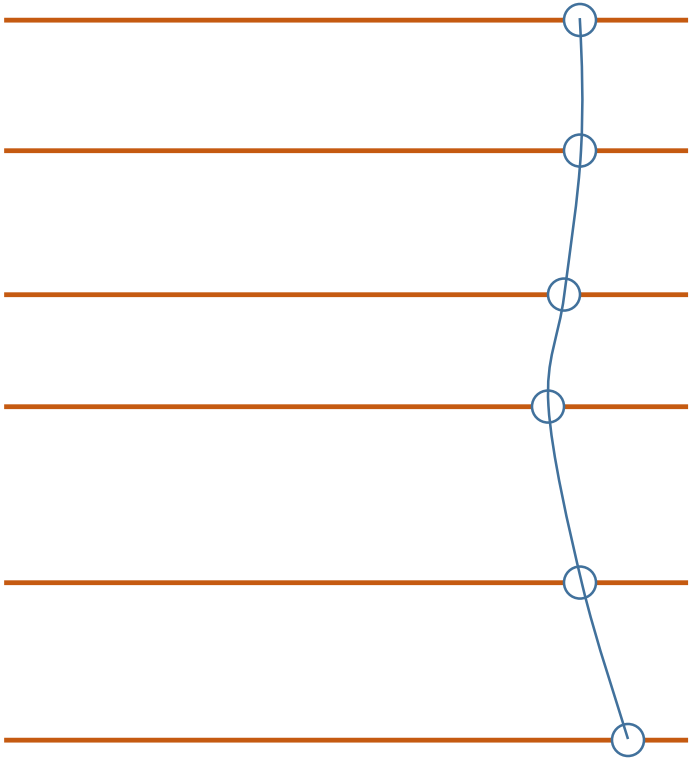
SubTask 2:

Create a smoothed polyline representing the predicted shoreline at the given future date

Given: The projected intersection points for a set of transects ○

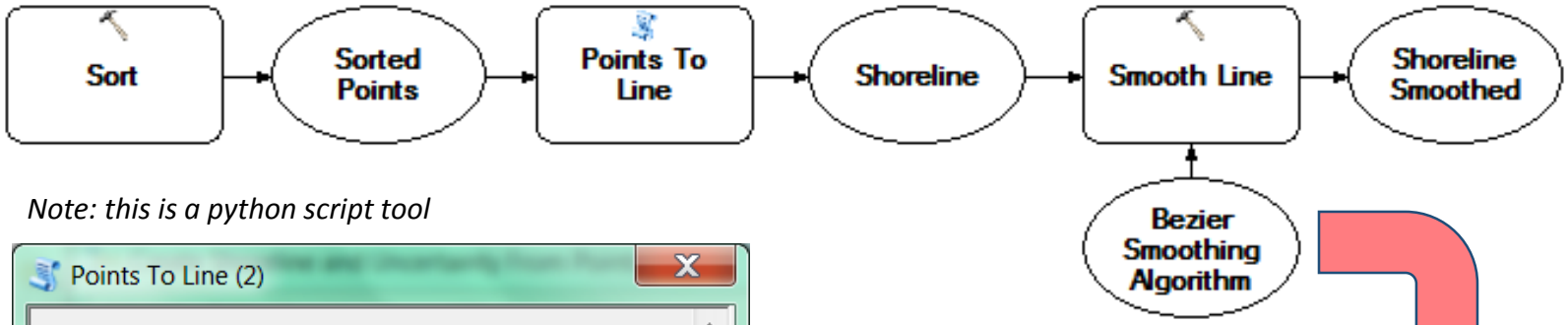
Create: The smoothed polyline representing the predicted shoreline at the given future date ⌒



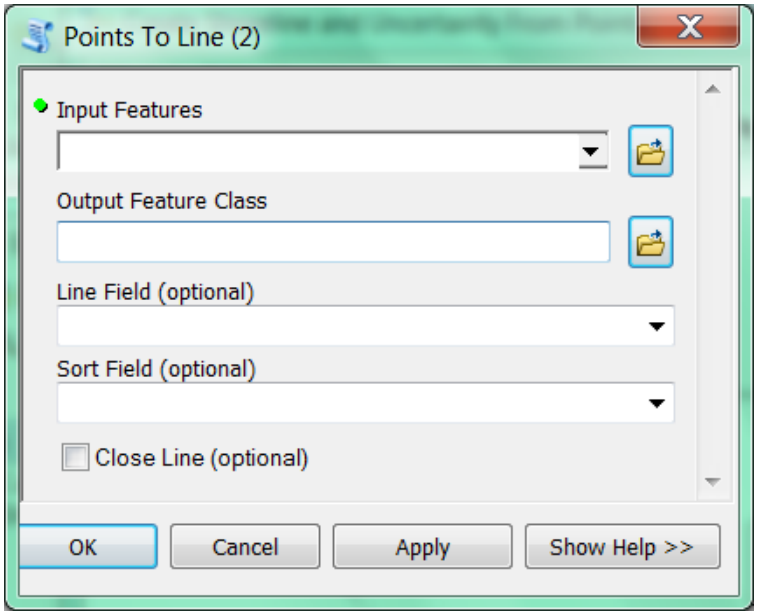


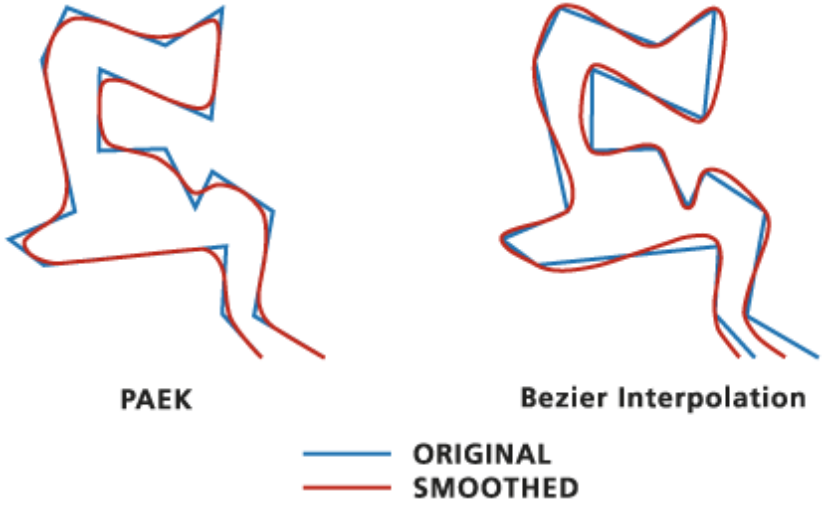
SubTask 2:

Create a smoothed polyline representing the predicted shoreline at the given future date



Note: this is a python script tool





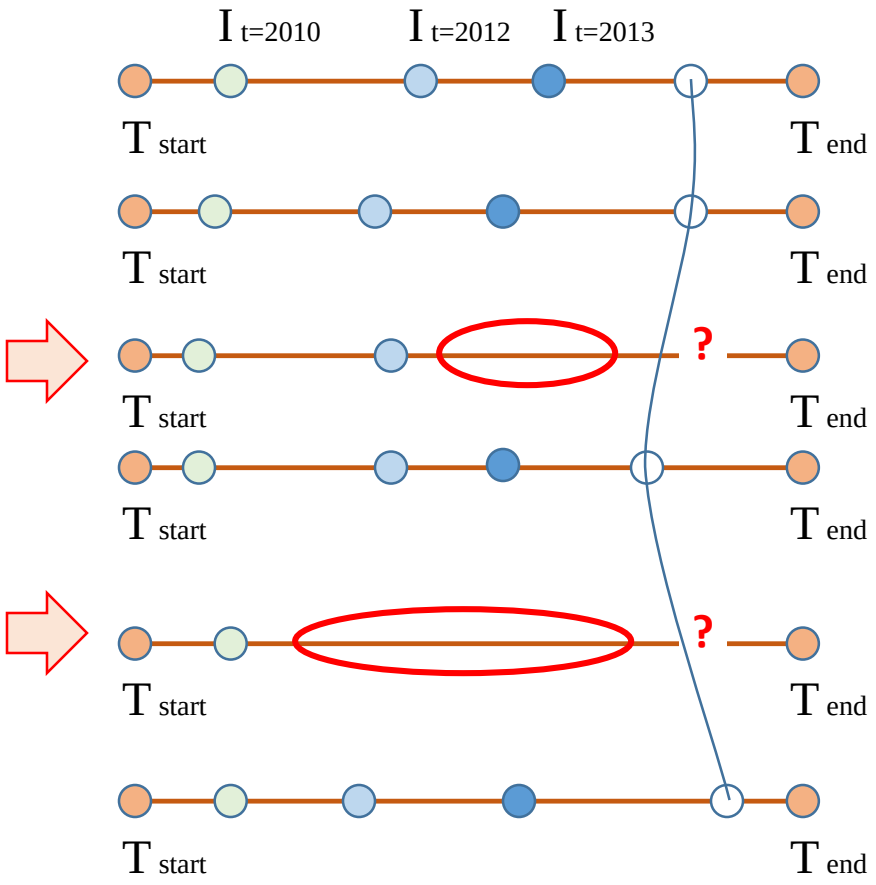
PAEK (Polynomial Approximation with Exponential Kernel)

Issue:

How should we project shoreline along a transect if recent historic shoreline data is not available?

- How current should the most recent shoreline be established to base calculated rate of change?

- Current Decision: We do not use projected points along a transect if the most recent date is older than user provided variable, but instead shoreline is smoothed over this transect



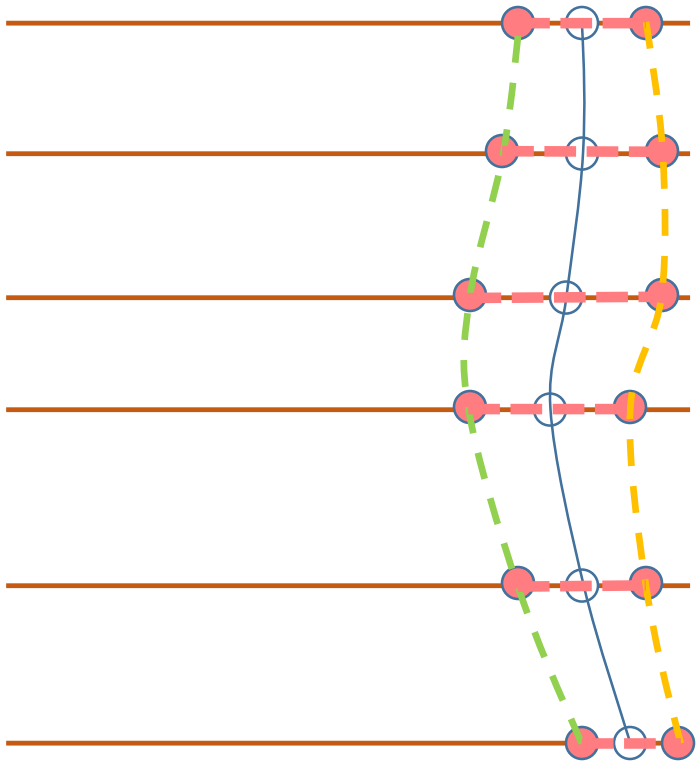
SubTask 3:

Create a geometric based error representation along the predicted shoreline

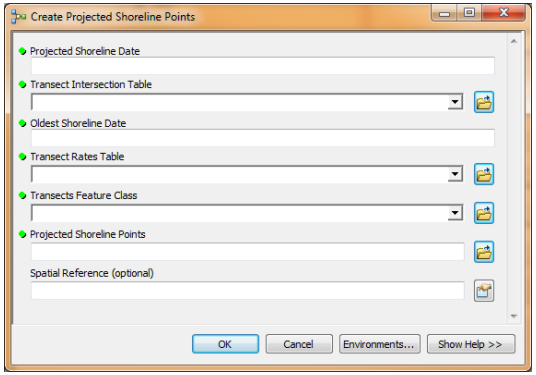
Given: The location along the transect at some given future date and the error data from DSAS for each transect

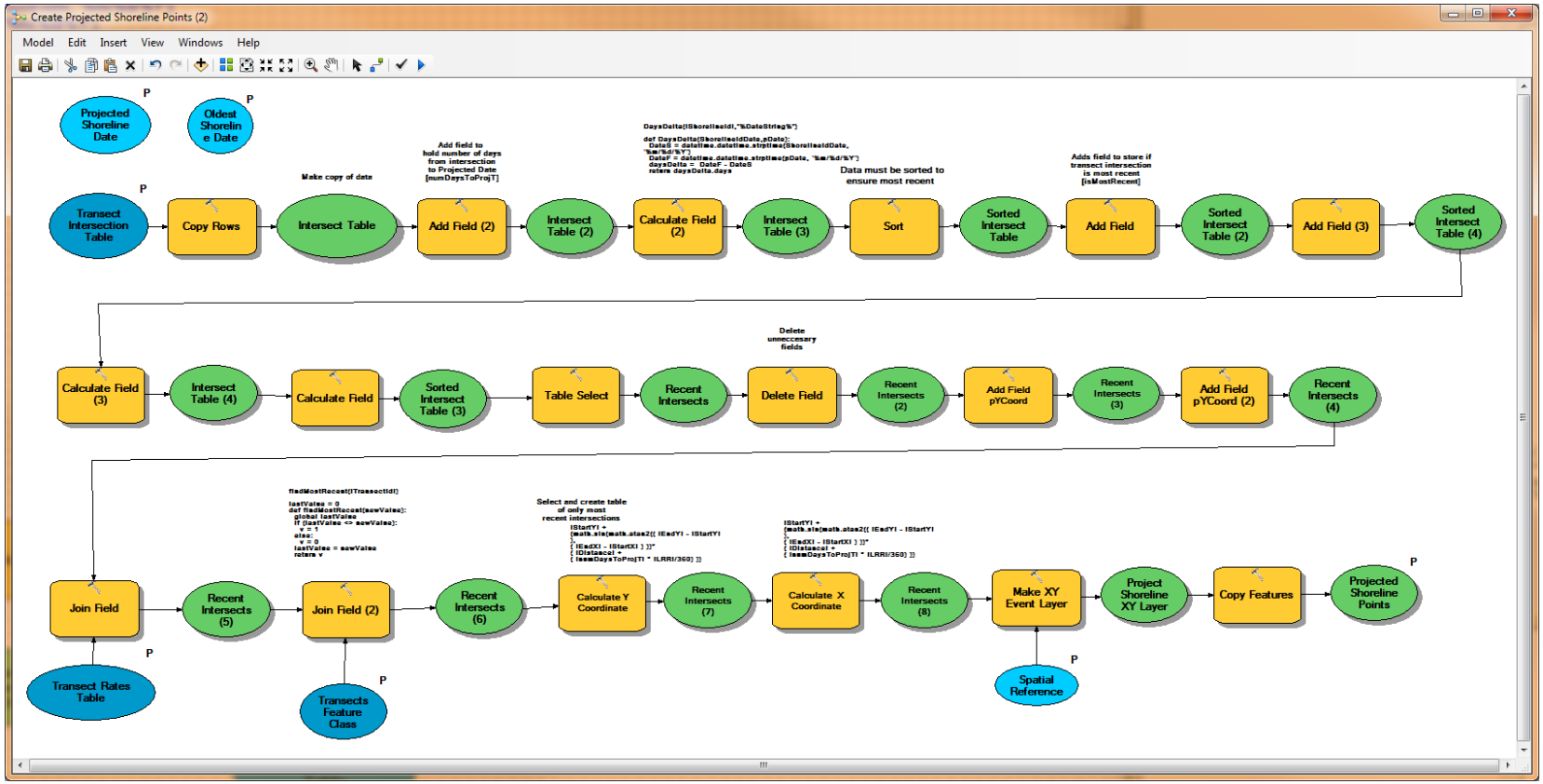
Create: A geometric based error representation along the predicted shoreline

Current Decision: Create Polyline (as opposed to polygon) representation of uncertainty

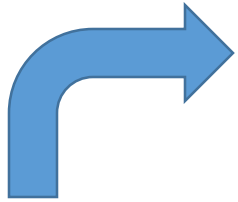


Create Projected Shoreline Points





Key lines on Python Script exported from model

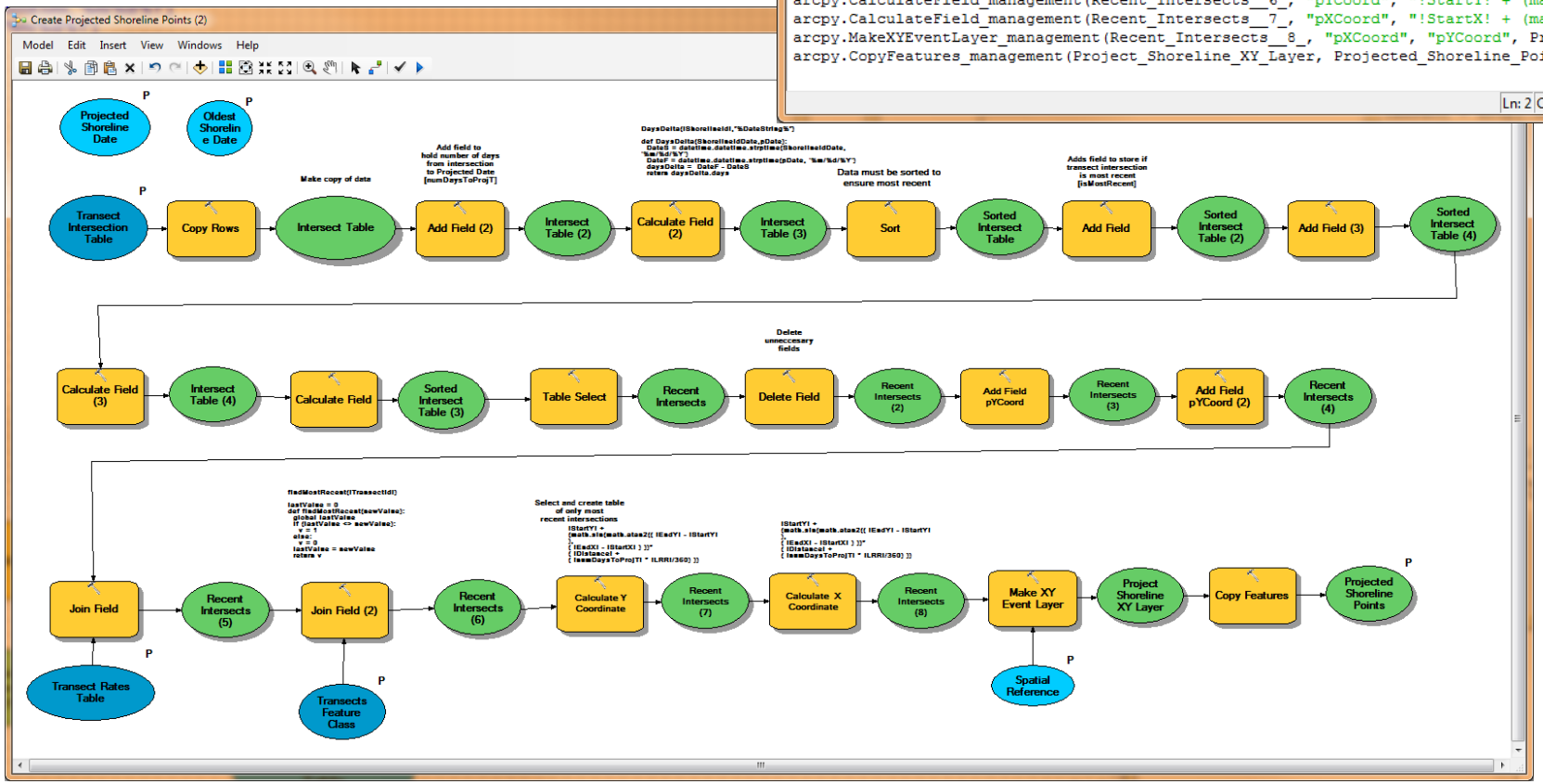


t.py - C:\Users\mdhendricks\Desktop\t.py

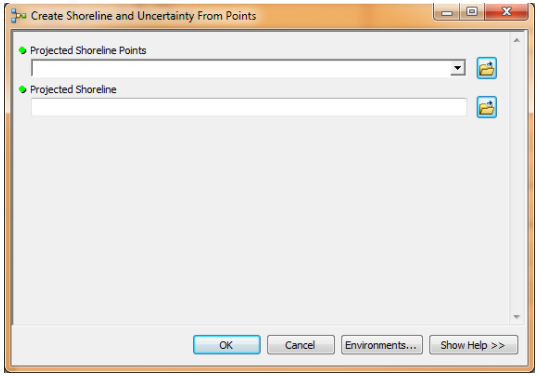
File Edit Format Run Options Windows Help

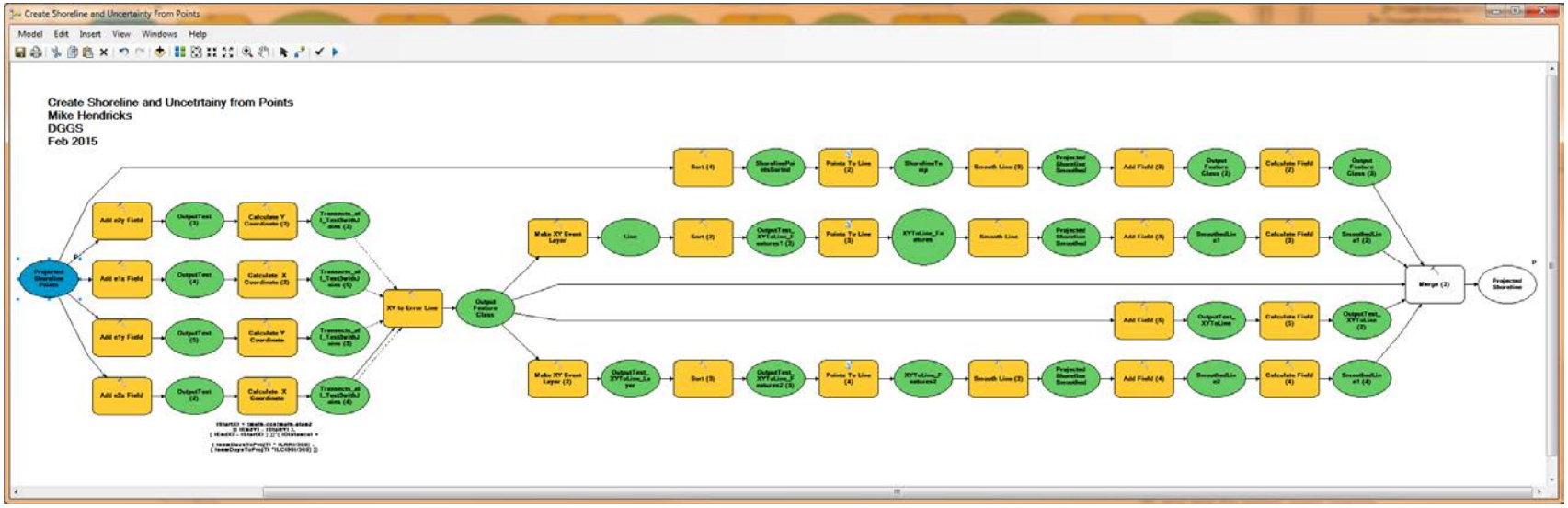
```
# Import arcpy module
import arcpy
# Script arguments REMOVED FOR DEMO
# Local variables REMOVED FOR DEMO
# Process: Copy Rows
arcpy.CopyRows_management(Transect_Intersection_Table, Intersect_Table, "")
arcpy.AddField_management(Intersect_Table, "numDaysToProjT", "LONG", "", "", "",
arcpy.CalculateField_management(Intersect_Table__2_, "numDaysToProjT", "DaysDelta
arcpy.Sort_management(Intersect_Table__3_, Sorted_Intersect_Table, "TransectId AS
arcpy.AddField_management(Sorted_Intersect_Table, "isMostRecent", "SHORT", "", ""
arcpy.AddField_management(Sorted_Intersect_Table__2_, "DaysFromOldestShorelineDat
arcpy.CalculateField_management(Sorted_Intersect_Table__4_, "DaysFromOldestShorel
arcpy.CalculateField_management(Intersect_Table__4_, "isMostRecent", "findMostRec
arcpy.TableSelect_analysis(Sorted_Intersect_Table__3_, Recent_Intersects, "isMost
arcpy.DeleteField_management(Recent_Intersects, "BaselineId;IntersectX;IntersectY
arcpy.AddField_management(Recent_Intersects__2_, "pYCoord", "DOUBLE", "", "", "",
arcpy.AddField_management(Recent_Intersects__3_, "pXCoord", "DOUBLE", "", "", "",
arcpy.JoinField_management(Recent_Intersects__4_, "TransectId", Transect_Rates_Ta
arcpy.JoinField_management(Recent_Intersects__5_, "TransectId", Transects_Feature
arcpy.CalculateField_management(Recent_Intersects__6_, "pYCoord", "!StartY! + (ma
arcpy.CalculateField_management(Recent_Intersects__7_, "pXCoord", "!StartX! + (ma
arcpy.MakeXYEventLayer_management(Recent_Intersects__8_, "pXCoord", "pYCoord", Pr
arcpy.CopyFeatures_management(Project_Shoreline_XY_Layer, Projected_Shoreline_Poi
```

Ln: 2 Col: 0

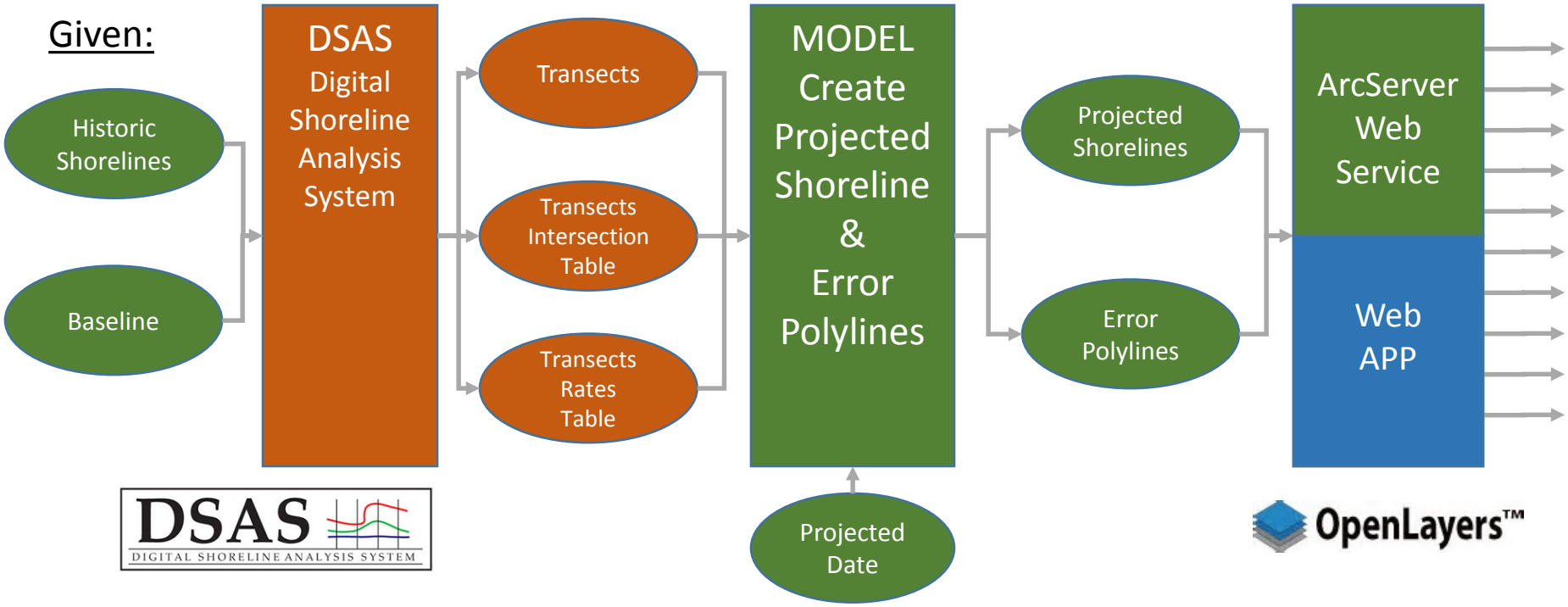


Create Shoreline and Uncertainty From Points





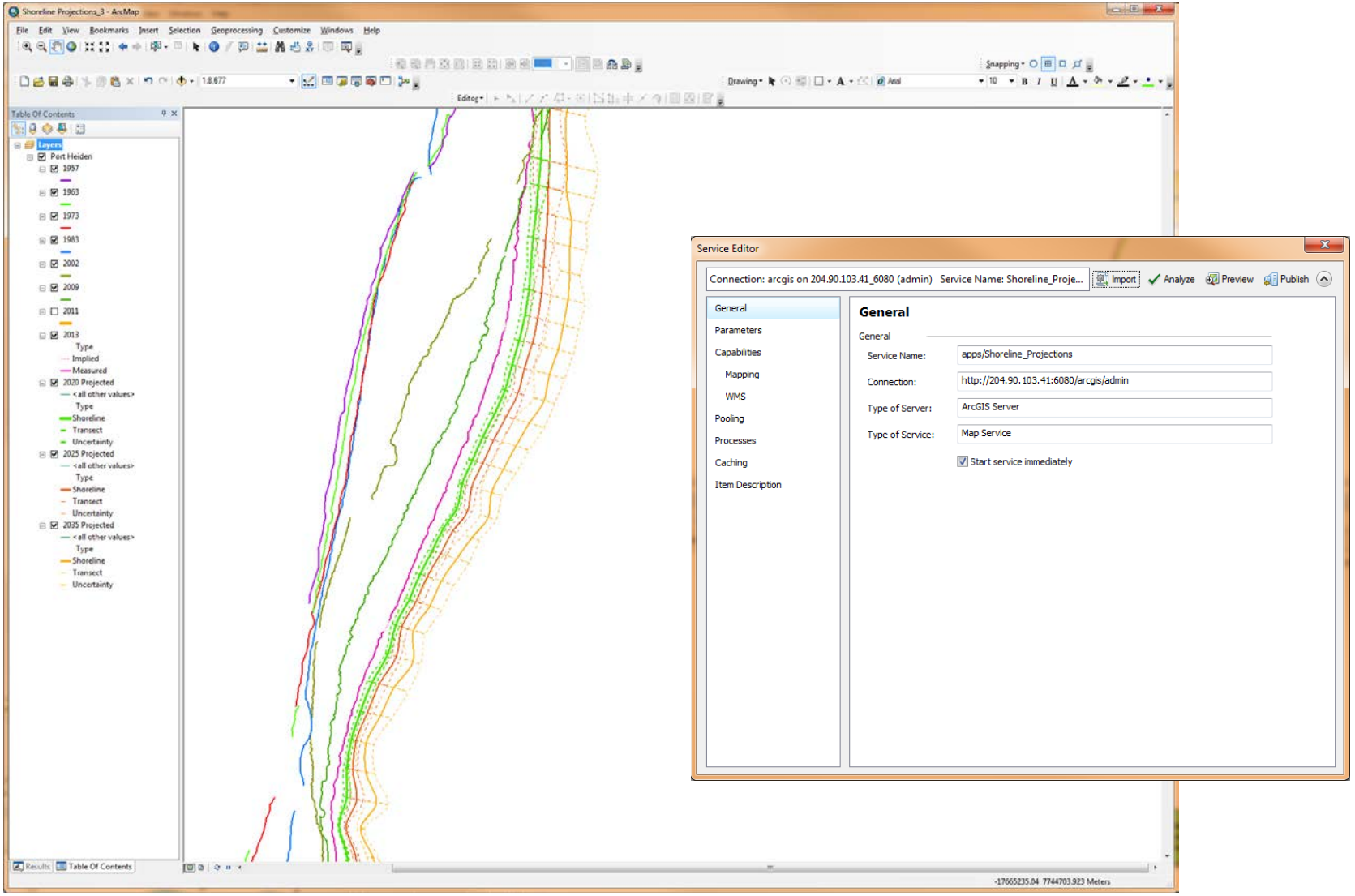
The GIS based Task to Achieve Goal



Desired GIS based output:

- Predicted Shoreline for a user defined future date as a polyline
- Display variable location uncertainty along this predicted shoreline
- Deliver this information to the public with a web service & app

Creating GIS Web Services

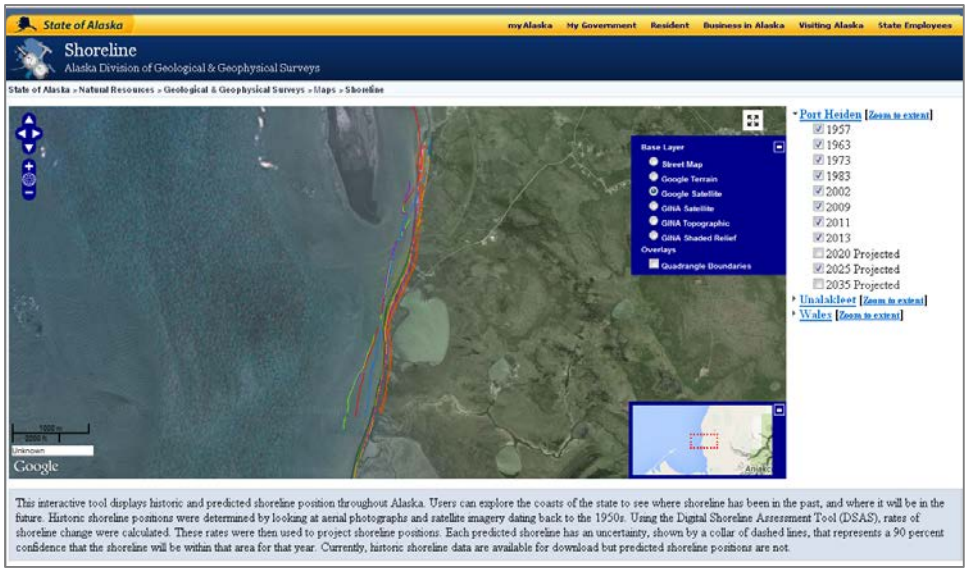


Creating the Shoreline Web App

Overlay
Shoreline
WMS Service

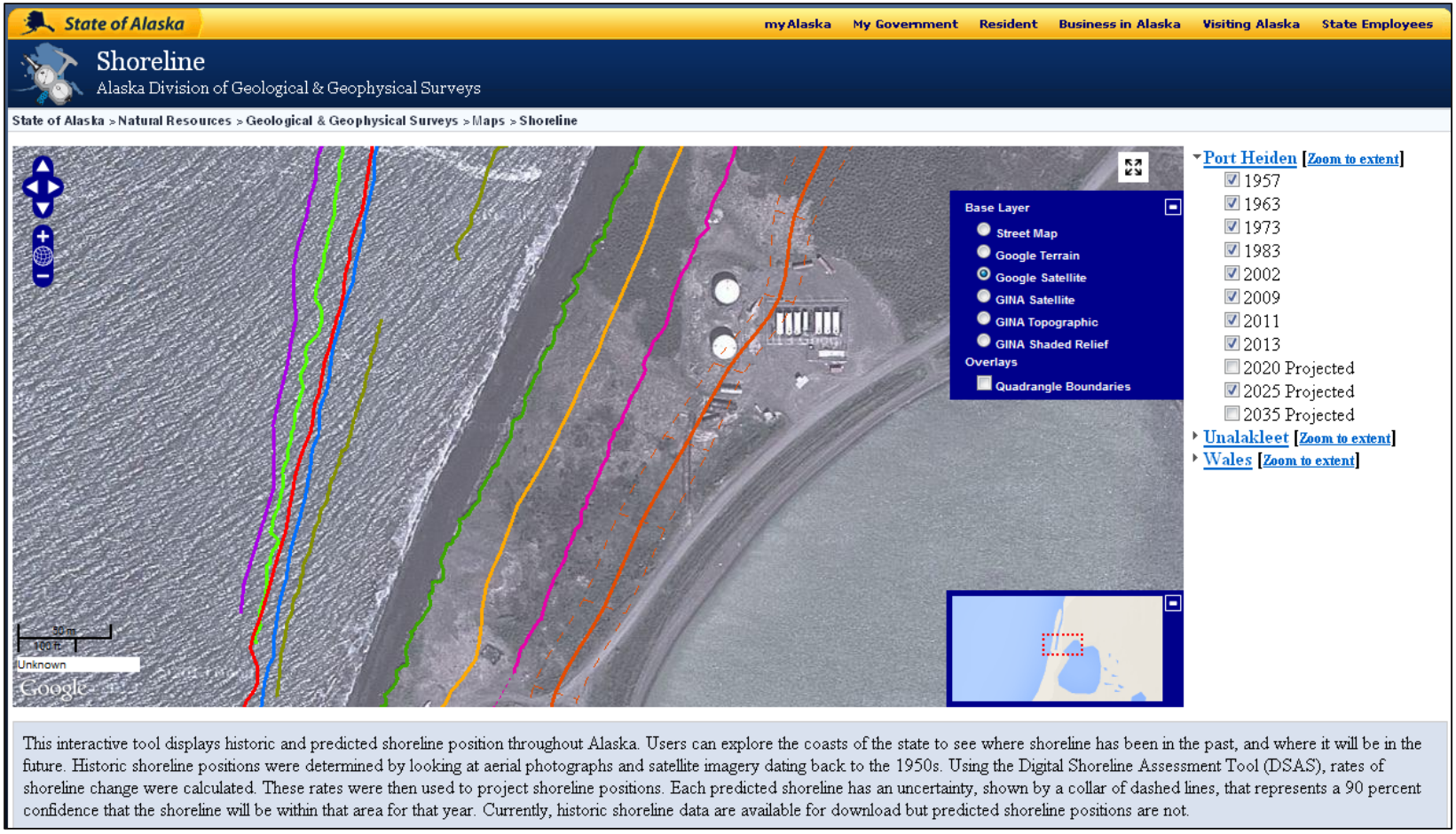
OpenLayers™

Basemap
WMS Services

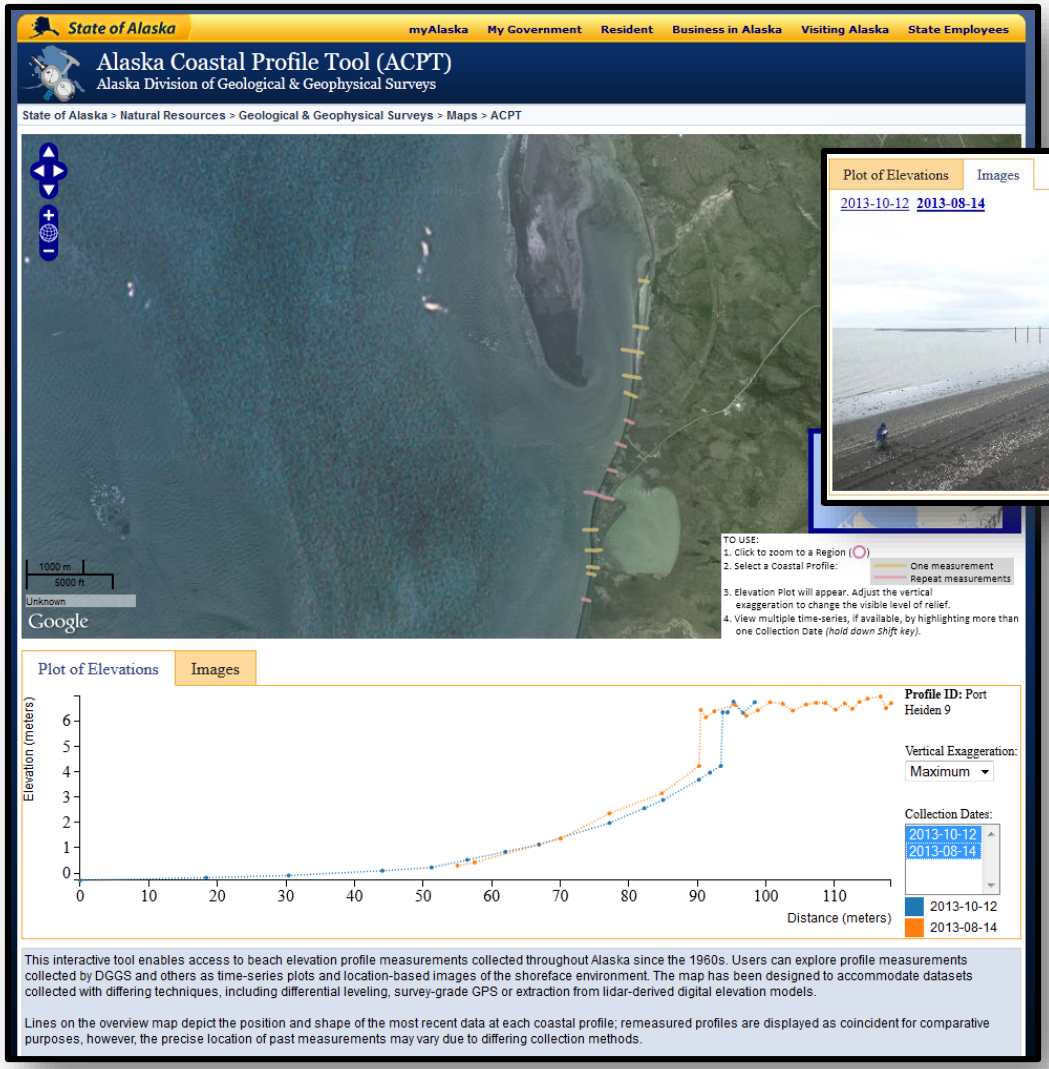


In final development

Creating the Shoreline Web App

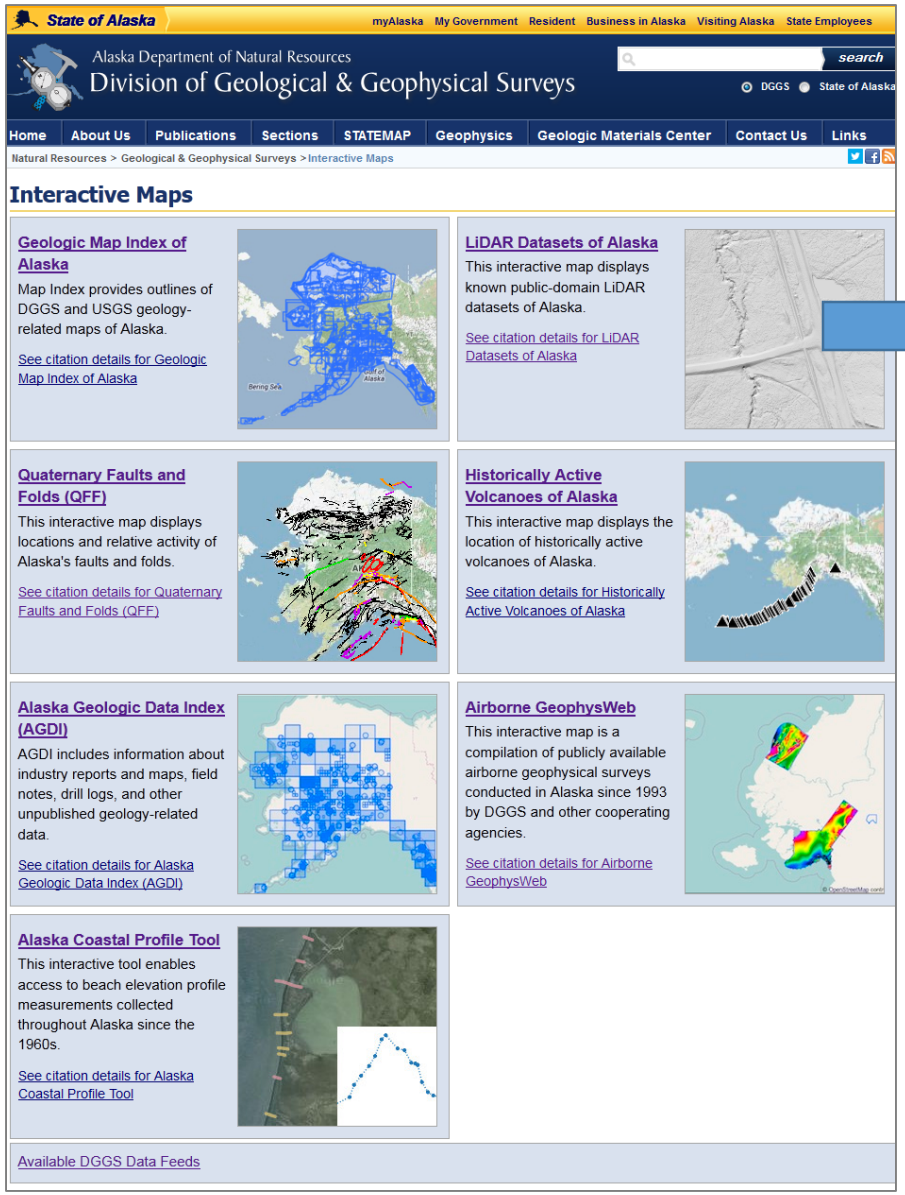


Related Coastal Web App



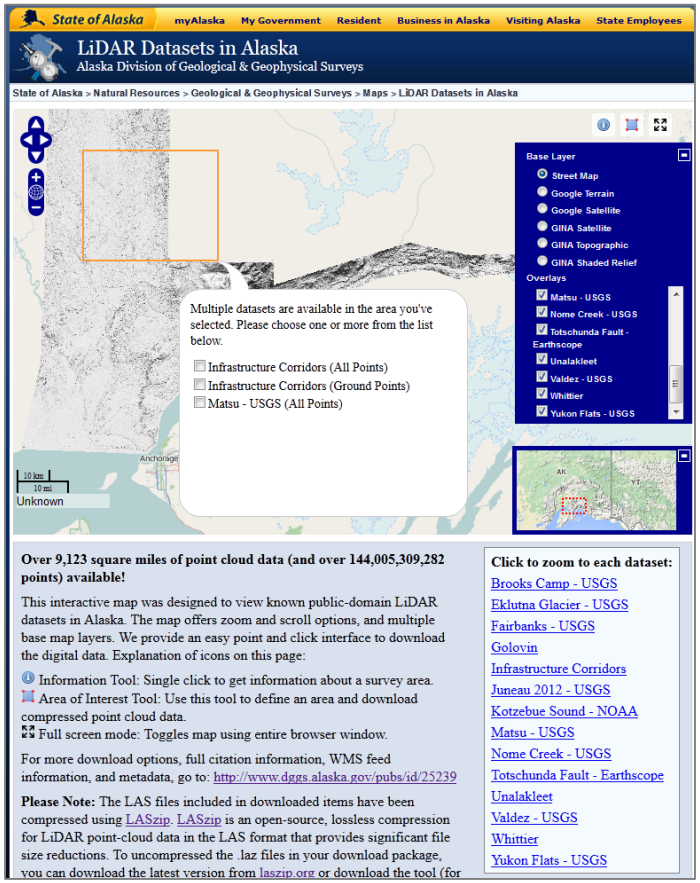
http://maps.dggs.alaska.gov/acpt

Other DGGS Web Apps



http://maps.dggs.alaska.gov

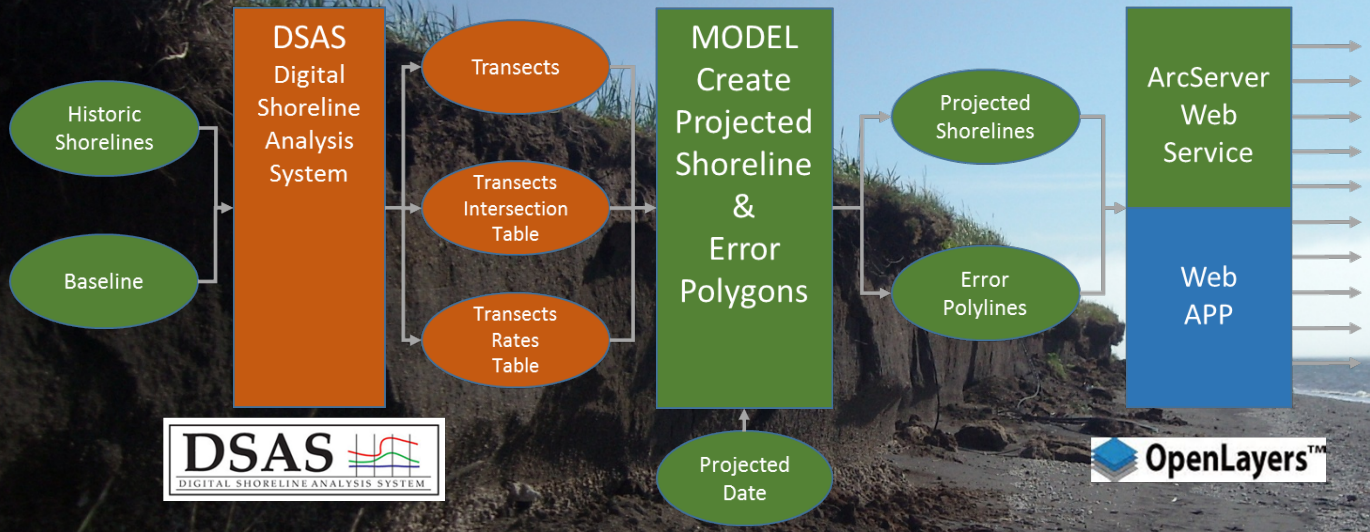
Example: Lidar Datasets of Alaska

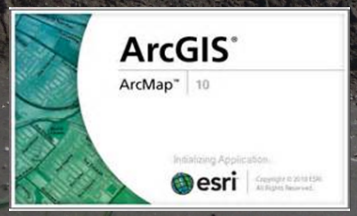


Future Work

- Develop more efficient tool in Python
- Develop tools and metrics to semiautomate the creation of baselines for use in DSAS
- Publish a geoprocessing service of this tool

Creating and Serving Alaska Shoreline Change Predictions





Mike Hendricks
Geospatial Analyst
Alaska Division of Geological & Geophysical Surveys
3354 College Rd, Fairbanks AK 99709
mike.hendricks@alaska.gov