RITA Documentation

N:\Planning - New File Structure\GIS\VRT\_PythonScripts\Performance\OnTimePerformance\RITA Documentation.docx

This script reads should read in a GTFS Feed, a RouteMatch On Time Performance Detail Report, and the shapefiles behind [ACHD’s RITA Map](http://achd.maps.arcgis.com/apps/webappviewer/index.html?id=2de7a5bc4fe842fab9020b4647343ebf), and identify route segments where road construction has adversely affected on time performance. **TODO: this script is still pretty rough, but I wanted to leave it in because it has a lot of potential. Pairing it with APC data would also get the number of people affected. The split\_line\_by\_nearest\_points function also has a lot of other potential applications.**

Requisites

* [Python 3](https://www.python.org/)
* [Geopandas](http://geopandas.org)
* [Pandas](https://pandas.pydata.org/pandas-docs/stable/)
* [TKInter](https://docs.python.org/3/library/tkinter.html)
* [Shapely](https://shapely.readthedocs.io/en/stable/manual.html)
* [GTFSTK](https://mrcagney.github.io/gtfstk_docs/)
* [Matplotlib](https://matplotlib.org/)
* [Seaborn](https://seaborn.pydata.org/)
* [Jupyter](https://jupyter.org/documentation)/IPython

Usage

1. Export an On Time Performance Detail Report from RouteMatch as a Flat Formatted Report (aka a CSV). Set the Late to “Depart” more than “5” minutes, and the Early to “Depart” more than “1” minutes. Group By “Route.” If you’re trying to use more than a couple weeks at a time, you’ll probably have to filter by route and repeat a few times to create manageable chunks of data.
2. Run the script. It will join up route segments and RITA streets with lane restrictions or closures. It then returns a list of affected routes and their on time performance. **TODO: build GUI, clean up output, a lot of other stuff.**