

Python/PandasでGTFS - WebでうごくGTFS-GO

GTFSxOSGeo研究会



MIERUNE

IGUCHI Kanahiro



GIS/OSS Enthusiast

井口 奏大

IGUCHI Kanahiro

CTO of MIERUNE Inc.

MapLibre User Group Japan(MUG-JP)



<https://github.com/Kanahiro>



<https://qiita.com/Kanahiro>

01 | gtfs-parser

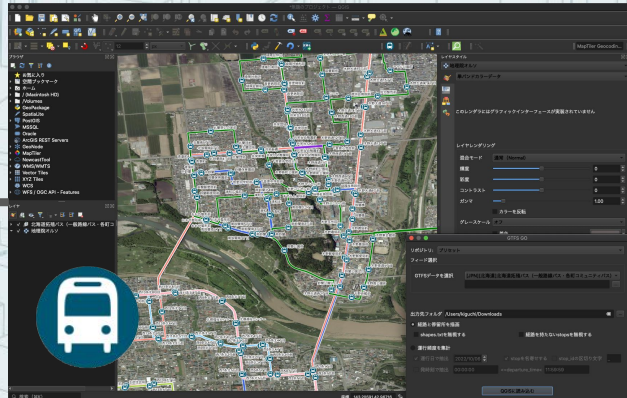
02 | gtfs-parserの技術的背景

03 | WebでうごくGTFS-GO

01

gtfs-parser

- GTFSをパースする処理は当初、GTFS-GO内で実装していました
- GTFS-GOの外でも同様にパース・集計処理をしたかったので、2023年に、独立したライブラリとして切り出しました(PyPIでリリース済み)。
 - これがgtfs-parserです



GTFS-GO

Installation

```
pip install gtfs-parser
```

PyPIからインストール可能

コマンドラインでGTFS
データをGeoJSONに変
換可能

CLI

```
usage: gtfs-parser [-h] [--parse_ignoresshapes] [--parse_ignoreroute]
                  [--aggregate_yyyymmdd AGGREGATE_YYYYMMDD]
                  [--aggregate_nounifystops]
                  [--aggregate_delimiter AGGREGATE_DELIMITER]
                  [--aggregate_begintime AGGREGATE_BEGINTIME]
                  [--aggregate_endtime AGGREGATE_ENDTIME]
                  mode src dst
```

positional arguments:

```
mode
src
dst
```

optional arguments:

```
-h, --help            show this help message and exit
--parse_ignoresshapes
--parse_ignoreroute
--aggregate_yyyymmdd AGGREGATE_YYYYMMDD
--aggregate_nounifystops
--aggregate_delimiter AGGREGATE_DELIMITER
--aggregate_begintime AGGREGATE_BEGINTIME
--aggregate_endtime AGGREGATE_ENDTIME
```

Example

```
gtfs-parser parse gtfs.zip output
gtfs-parser parse gtfs_dir output --parse_ignoresshapes
gtfs-parser aggregate gtfs.zip output
gtfs-parser aggregate gtfs_dir output --aggregate_nounifystops
```

API

```
import gtfs_parser

# construct GTFS object
gtfs = gtfs_parser.GTFSFactory(zip_path)

# parse as GeoJSON
stops = gtfs_parser.parse.read_stops(gtfs)
routes = gtfs_parser.parse.read_routes(gtfs)

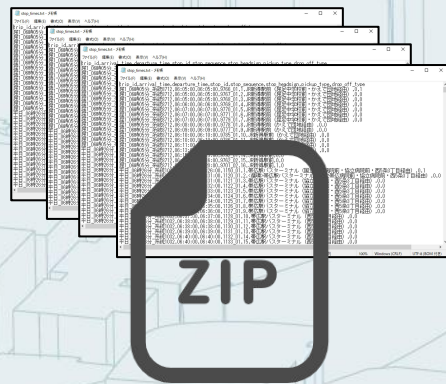
# aggregate frequency
aggregator = gtfs_parser.aggregate.Aggregator(gtfs, yyyyymmdd=yyyyymmdd)
interpolated_stops = aggregator.read_interpolated_stops()
route_freq = aggregator.read_route_frequency()
```

Python-APIも提供されています

つまり任意のアプリケーションに組み込むことができます！

02

gtfs-parserの技術的背景



入力はGTFSのファイル群が格納されたZIP



テーブルごとのDataFrame

stops
stop_times
routes
trips...

DataFrame

stops

routes

trips

routes

stops

stop_times

merge,
merge,
merge...

停留所データ(stops.geojson)

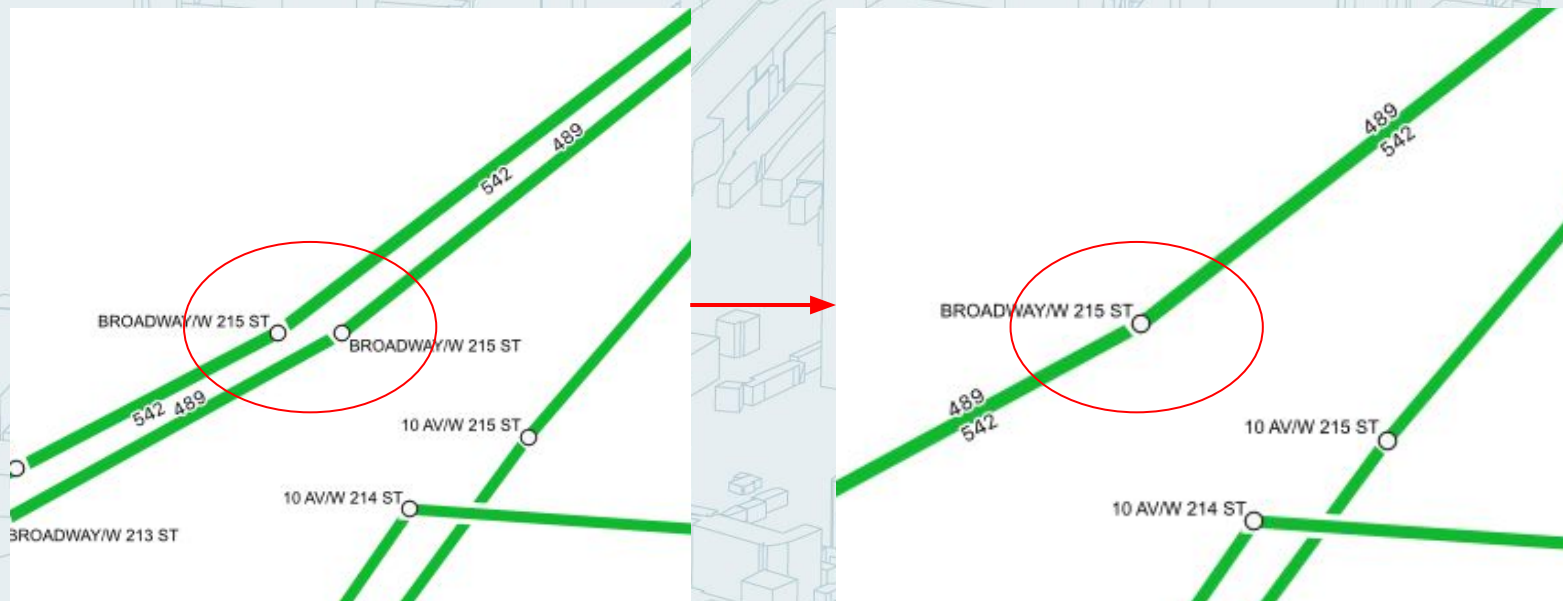
経路データ(routes.geojson)

● 各テーブルの役割・テーブル間の関係の把握がたいへん

- IDをキーに複数のテーブルをJOINすれば良いのはわかるが、どのテーブルを持ってくるべきなのかが初めはわからなかった

● GTFSでは多くの値がオプションである

- 列どころかオプションなテーブルすらあり、規格としては緩め
 - たとえば路線ごとのルート的位置情報を表すshapes.txtはオプション。
- データの存在有無に応じた分岐が必要。



- GTFSでは、たとえば道路を挟んで明らかにペアで存在するような停留所も、別々に定義します。
- 運行頻度を集計する際、こういった「類似」する停留所をグルーピングできるとうれしい。

「類似」する停留所の定義

- 同一の親停留所をもつ停留所群
 - GTFSの停留所は、仕様上「親停留所(parent_station)」を持つことができます。
 - この場合は親停留所の位置に集約します。
- stop_idに同一の接頭辞をもつ停留所群
 - stop_idにhoge_1/hoge_2, fuga-A/fuga-BのようなIDを与えることが多いため。
 - この場合は、群の地理的重心に集約します。
- stop_nameが一致していてかつ地理的に近傍にある停留所群
 - たとえば道路を挟んで同じ名称となっている停留所が多いため。
 - この場合は、群の地理的重心に集約します。



stop_nameが一致していてかつ地理的に近傍にある停留所群

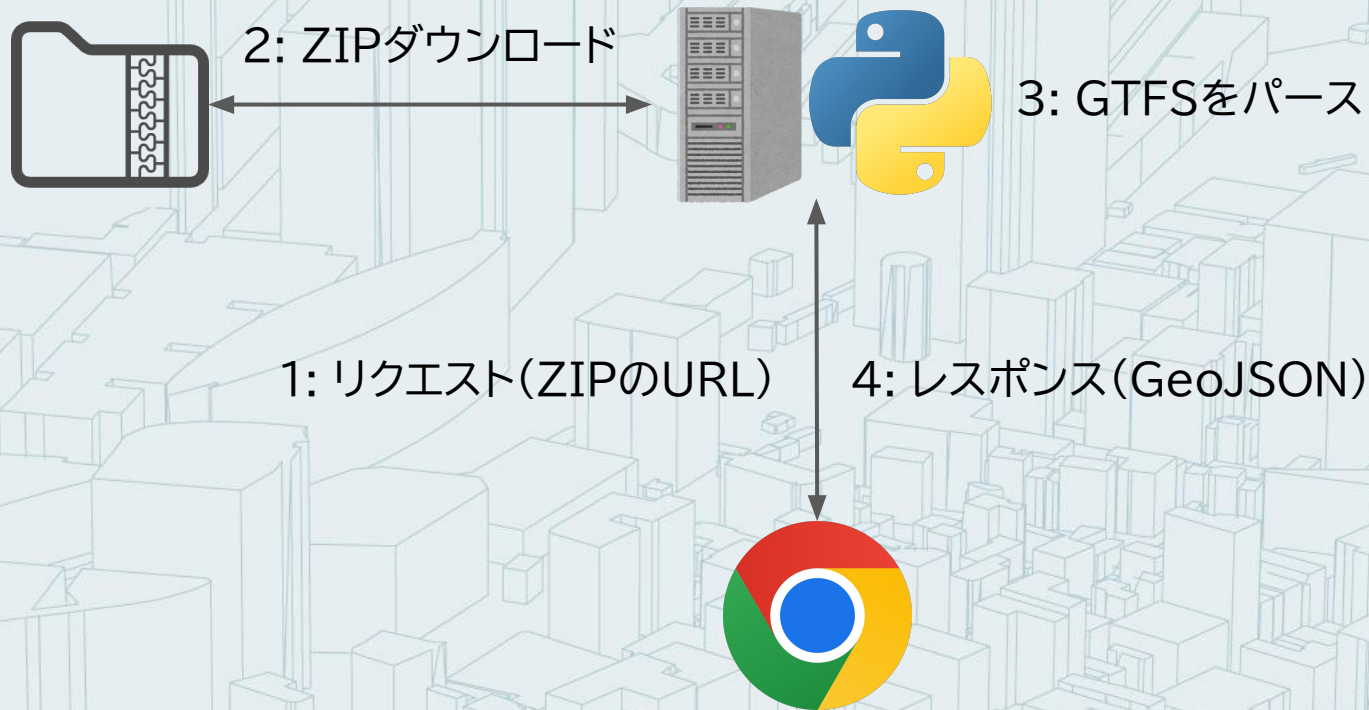


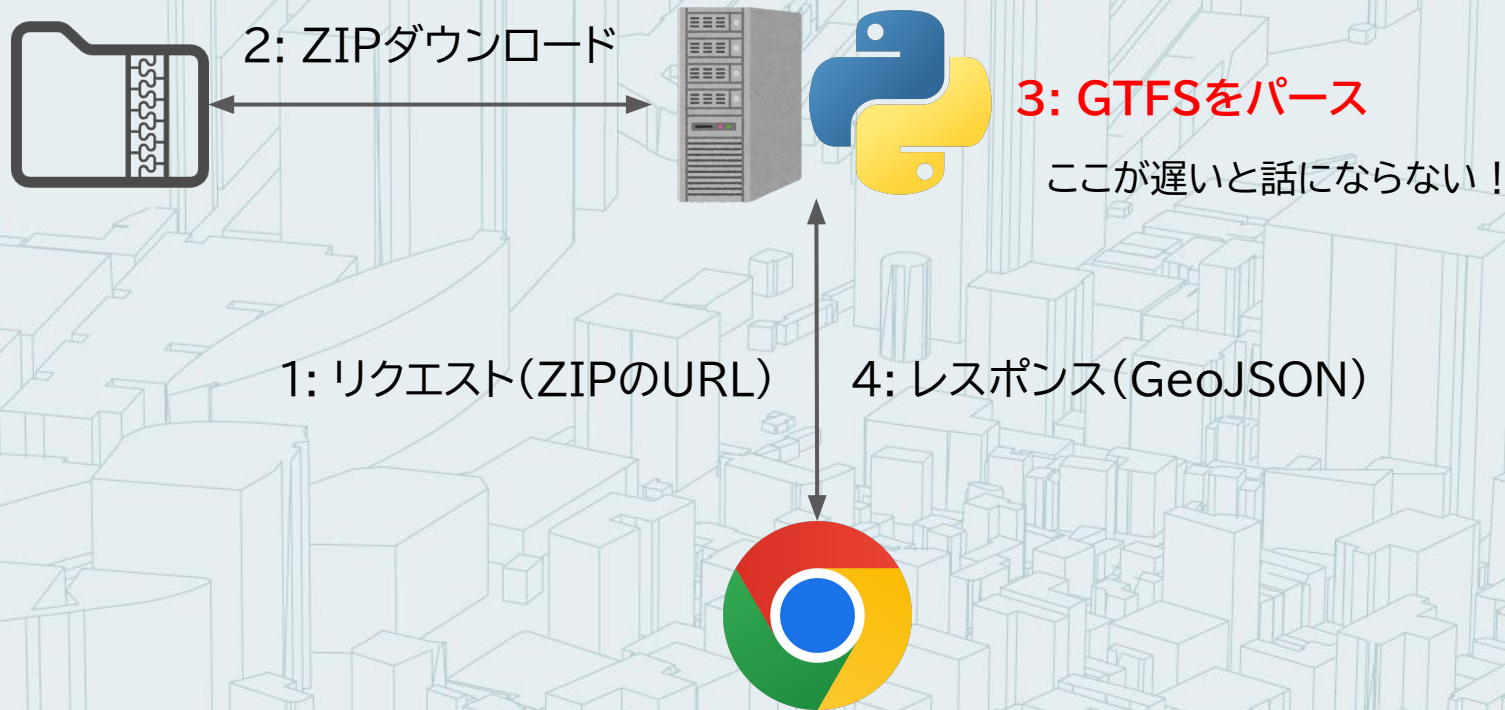
一定距離内の同名停留所をグルーピング

重心に集約する

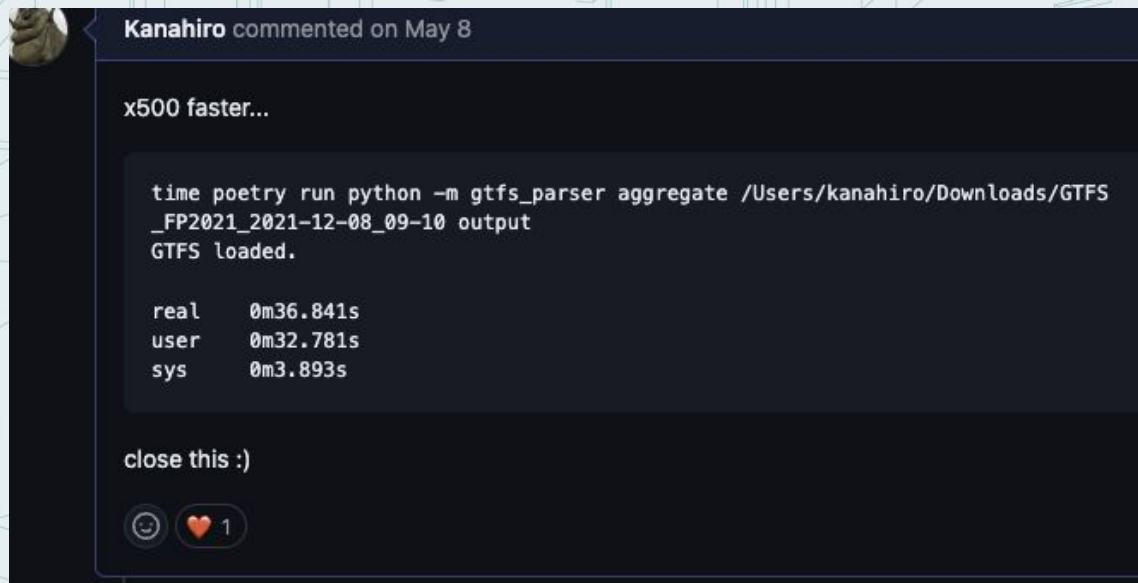
03

WebでうごくGTFS-GO





GTFS-GOがv4でパフォーマンスが向上したと紹介しましたが、これはコアモジュールであるgtfs-parserの性能が向上したということでした(x500)



Kanahiro commented on May 8

x500 faster...

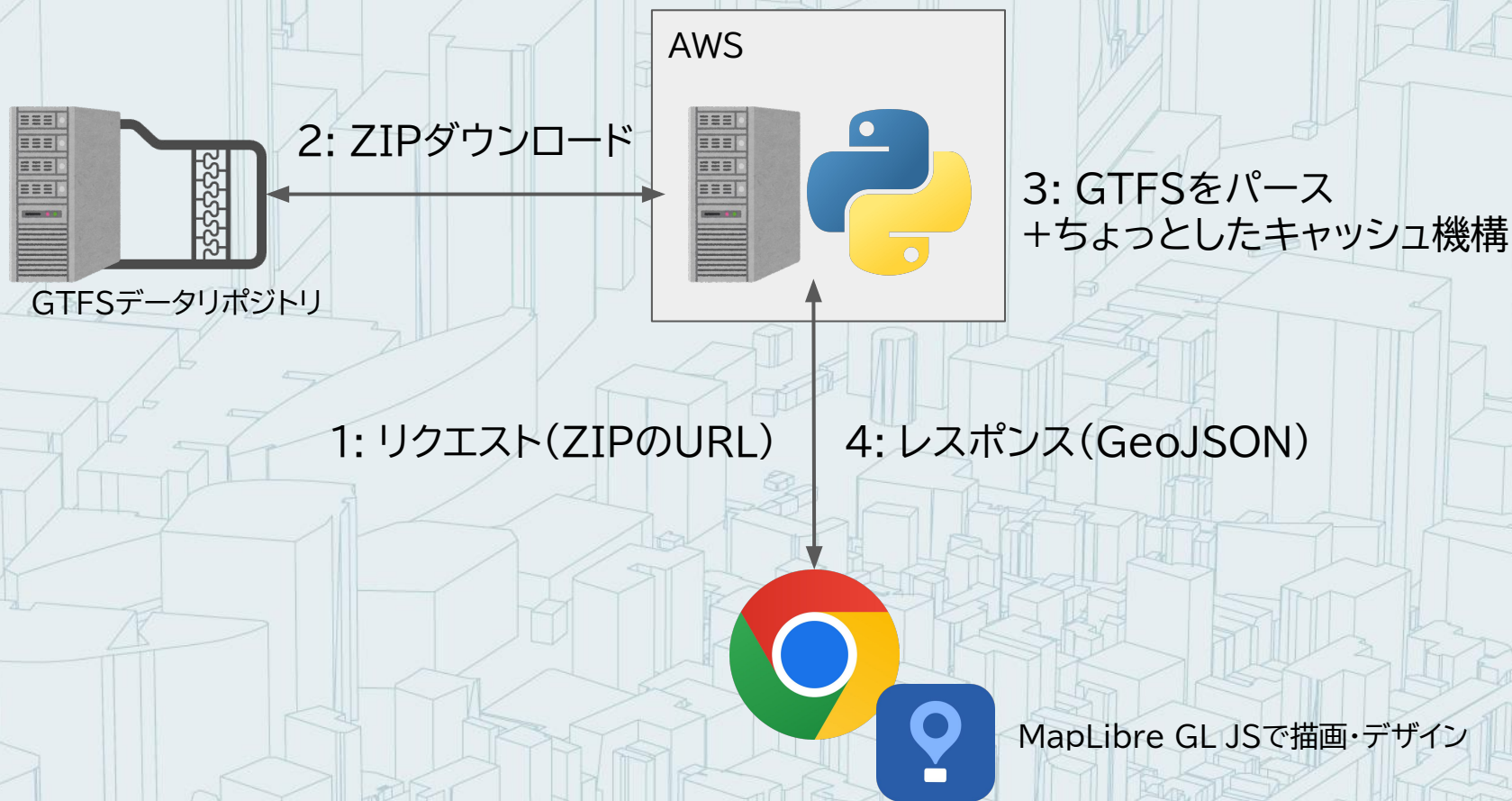
```
time poetry run python -m gtfs_parser aggregate /Users/kanahiro/Downloads/GTFS_FP2021_2021-12-08_09-10 output
GTFS loaded.

real    0m36.841s
user    0m32.781s
sys     0m3.893s
```

close this :)

👍 1

gtfs-parserがかなり速くなったので
リクエスト時にGTFSをパースしてGeoJSONを返すような
サーバーの実装が現実的になりました



何らかの形での公開を予定しています
coming soon...

 MIERUNE