

PondiônsTracker: A framework based on GTFS-RT to identify delays and estimate arrivals dynamically in Public Transportation Network

Pedro Pongelupe Lopes

Programa de Pós-Graduação em Informática

December 04 2023



PUC Minas

Contents

1 Introduction

2 Theoretical Reference

3 Related Work

4 PondiônsTracker

5 Results

6 Conclusion

Introduction

Motivation

- Public Transportation Network
- Huge centers
- GTFS and GTFS-RT specifications



PUC Minas

Motivation

GTFS-RT Matching Identifiers Issue

To work with GTFS-RT, it is **required** to track vehicles in real-time. But, in some cases, it is not easy to match the identifier between a real-time record and the GTFS static data. In Rome in 2016, this issue was reported by Raghothama et al. (2016). We still face this issue in Belo Horizonte in 2023.



Objectives

Main Objective

- Proposing and validating PondiônsTracker

Specific Objectives

- Collecting data from the real-time API and combining with the GTFS
- Understanding if Belo Horizonte's delays are spatial and temporal dependent by analyzing delays among bus stops
- Comparing the arrival times defined at the GTFS with the arrival times generated by *PondiônsTracker*.



Theoretical Reference

Main Ideas

- Smart Cities
- Urban Computing
- Human Mobility
- Graphs and Complex Network



PUC Minas

Complex Networks and Graphs

Definition

Bacciu et al., 2020 state that "a graph has a compositional nature, being a compound of atomic information pieces and a relational nature, as the links defining its structure denote relationships between the linked entities".

- $g = (V_g, E_g, X_g, A_g)$

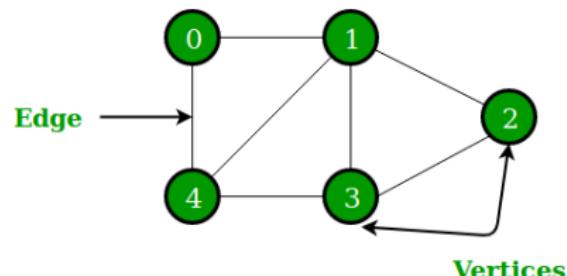
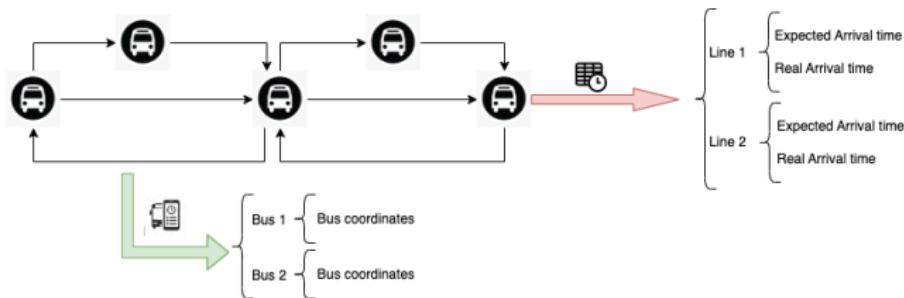


Figura: Example of an undirected graph



Public Transportation Network as a Complex Network

$$G^l = (V_g, E_g, X_g, A_g)$$



G^l : Graph G at a given time /

V_g : Bus stops

E_g : Routes connecting two bus stops

X_g : Additional information about bus stops (V_g)

A_g : Additional information about routes connecting two bus stops (E_g)



PUC Minas

Related Work

Main Ideas

- GTFS-RT matching identifiers issue
- Public Transportation Network as a complex network
- Delay analysis
- Algorithm outline after monitoring vehicles in real-time



PUC Minas

PondiônsTracker

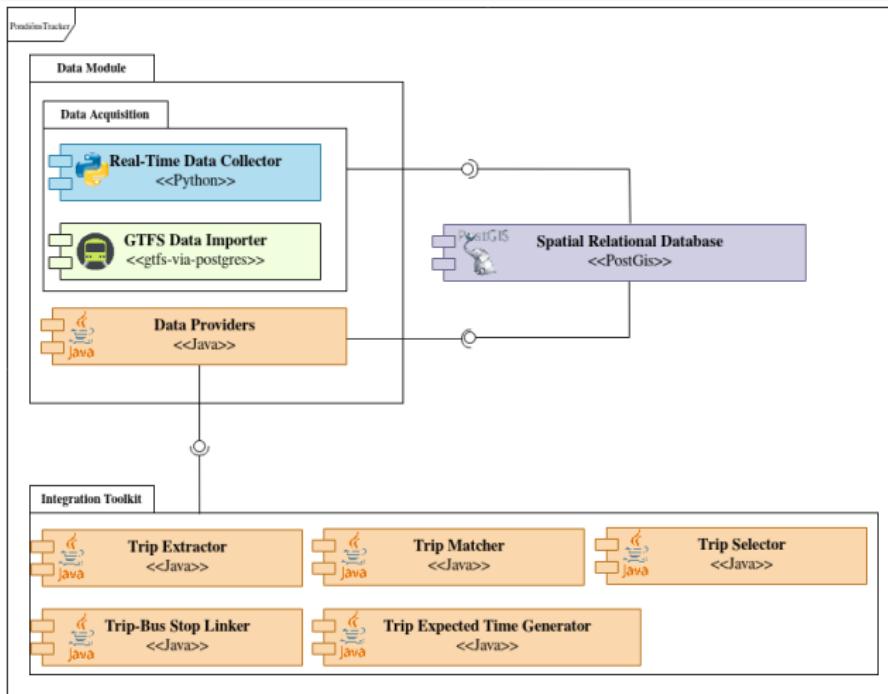
Overview

PondiônsTracker^a is a framework to enrich GTFS data with real-time data. The name *PondiônsTracker* is a small gag from the sonority of the expression '*bus stop*' when pronounced in Portuguese with the accent from Minas Gerais.

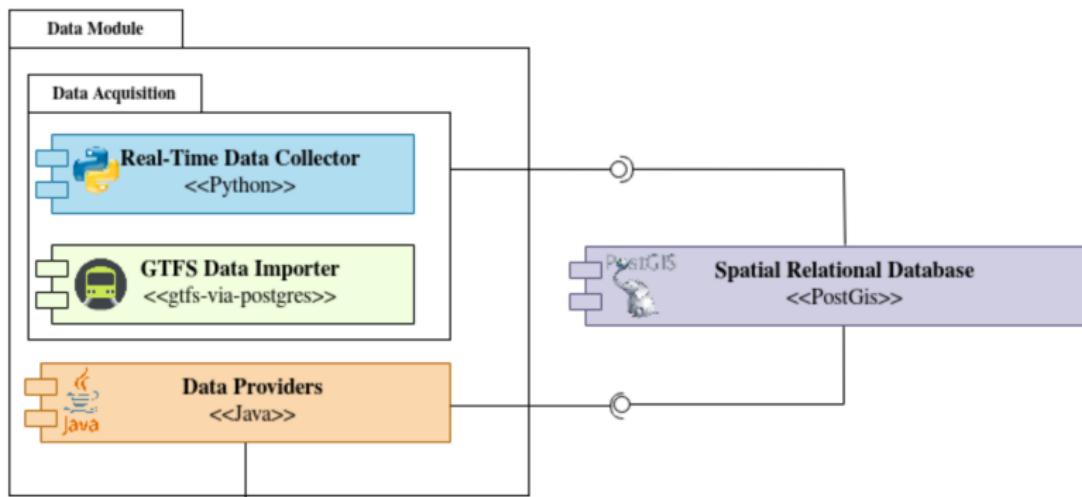
^aAvailable at <https://github.com/Pongelupe/PondionsTracker/>



PondiônsTracker's Architecture



Data Module Overview



Data Acquisition - GTFS Data Importer

gtfs-via-postgres

Import [GTFS Static/Schedule](#) datasets into a [PostgreSQL database](#), to allow for efficient querying and analysis.

npm v4.8.2 binary build failing license Prosperity/Apache node >=16.17 support me donate chat with me on Twitter

- handles [daylight saving time correctly](#) but retains reasonable lookup performance
- supports [frequencies.txt](#)
- ⚡ joins [stop_times.txt](#) / [frequencies.txt](#) , [calendar.txt](#) / [calendar_dates.txt](#) , [trips.txt](#) , [route.txt](#) & [stops.txt](#) into [views](#) for straightforward data analysis (see below)
- 🚀 is carefully optimised to let PostgreSQL's query planner do its magic, yielding quick lookups even with large datasets (see [performance section](#))
- validates and imports [translations.txt](#)
- ⚡ exposes (almost) all data via GraphQL using [PostGraphile](#)

Figura: *gtfs-via-postgres*'s README



PUC Minas

Data Acquisition - Real-Time Data Collector

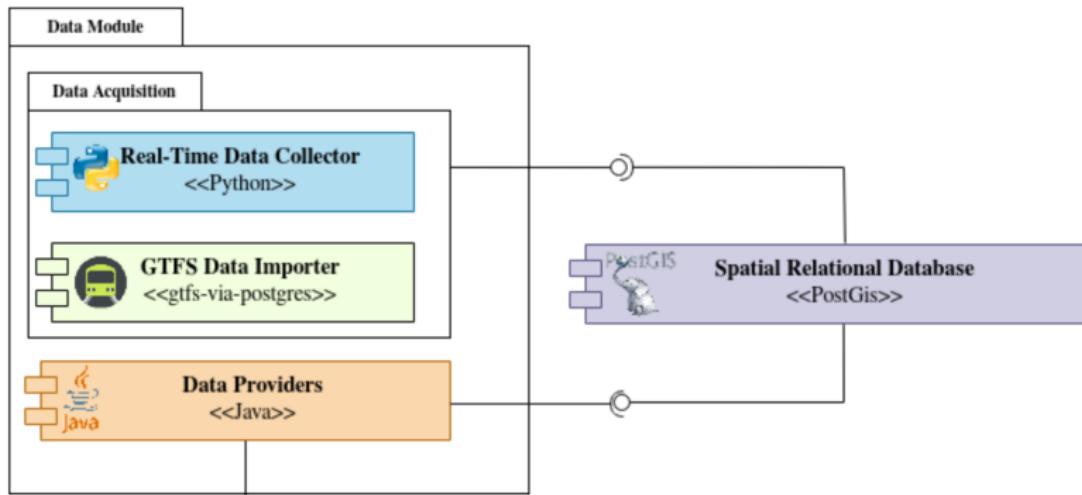
Considerations

This component incorporates the data from real-time traffic API provided by some external source.



PUC Minas

Data Providers



Data Providers

```
1 <dependency>
2   <groupId>br.pondionstracker</groupId>
3   <artifactId>data-module</artifactId>
4   <version>1.0.0</version>
5 </dependency>
```

1

Figura: *DataModule's* maven dependency



PUC Minas

Integration Module

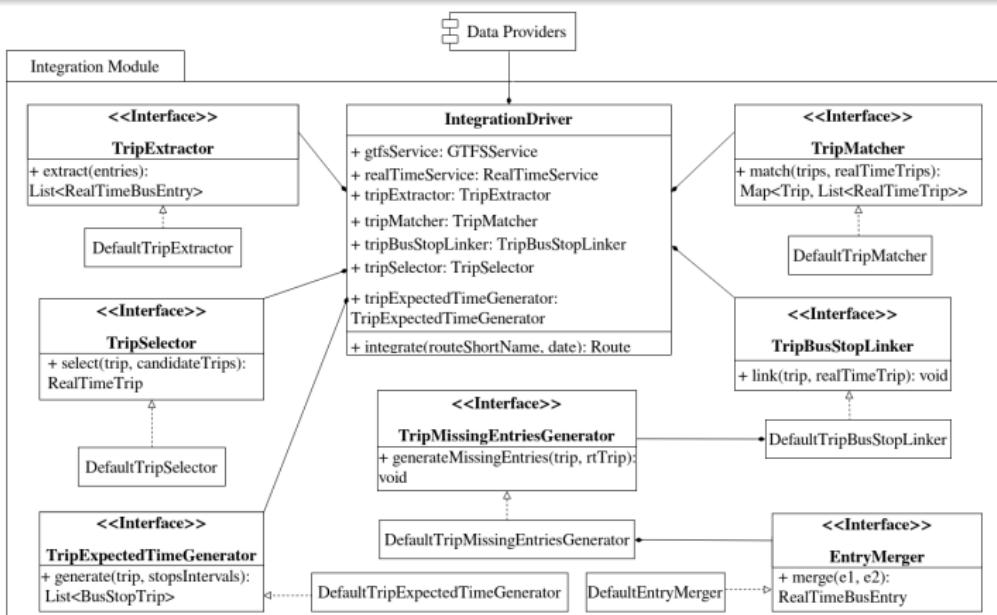


Figura: Integration Module Class Diagram

Integration Driver

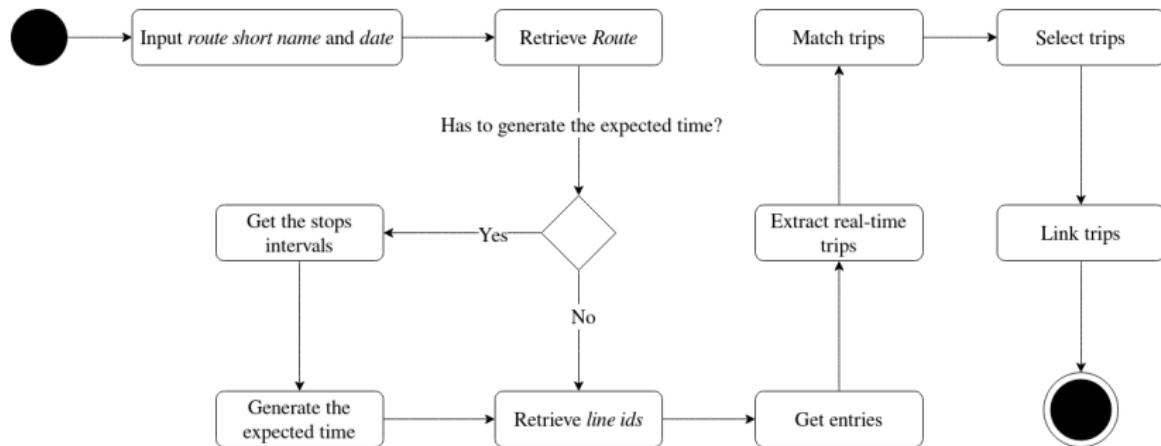


Figura: Integration Driver Activity Diagram



Integration Driver - 1st Step

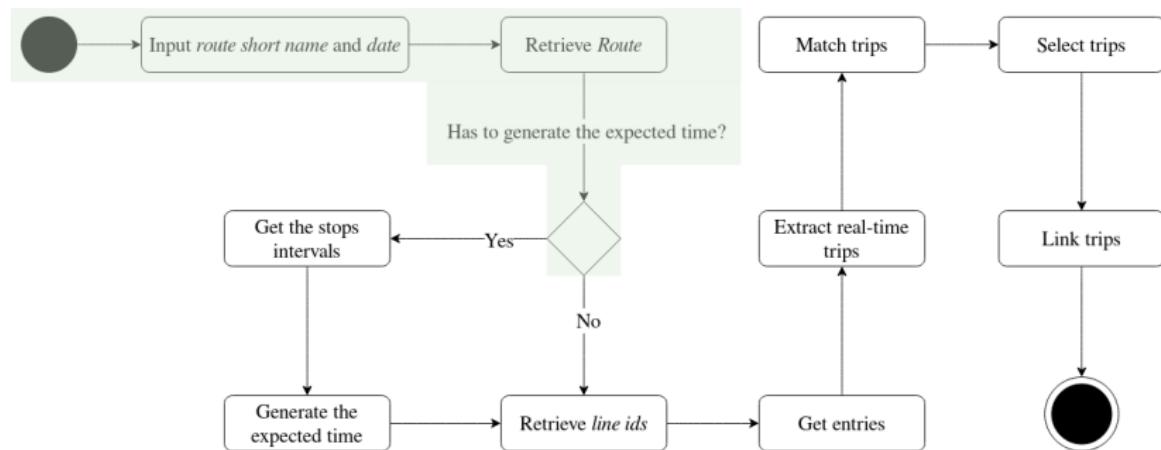


Figura: Integration Driver Activity Diagram



Has to generate the expected time?

<code>arrival_time</code>	Time	Conditionally required	Arrival time at a specific stop for a specific trip on a route. If there are not separate times for arrival and departure at a stop, enter the same value for <code>arrival_time</code> and <code>departure_time</code> . For times occurring after midnight on the service day, enter the time as a value greater than 24:00:00 in HH:MM:SS local time for the day on which the trip schedule begins.
			Scheduled stops where the vehicle strictly adheres to the specified arrival and departure times are timepoints. If this stop is not a timepoint, it is recommended to provide an estimated or interpolated time. If this is not available, <code>arrival_time</code> can be left empty. Further, indicate that interpolated times are provided with <code>timepoint=0</code> . If interpolated times are indicated with <code>timepoint=0</code> , then time points must be indicated with <code>timepoint=1</code> . Provide arrival times for all stops that are time points. An arrival time must be specified for the first and the last stop in a trip.

Figura: `arrival_time` definition from `stop_times.txt`



PUC Minas

Integration Driver - 2nd Step*

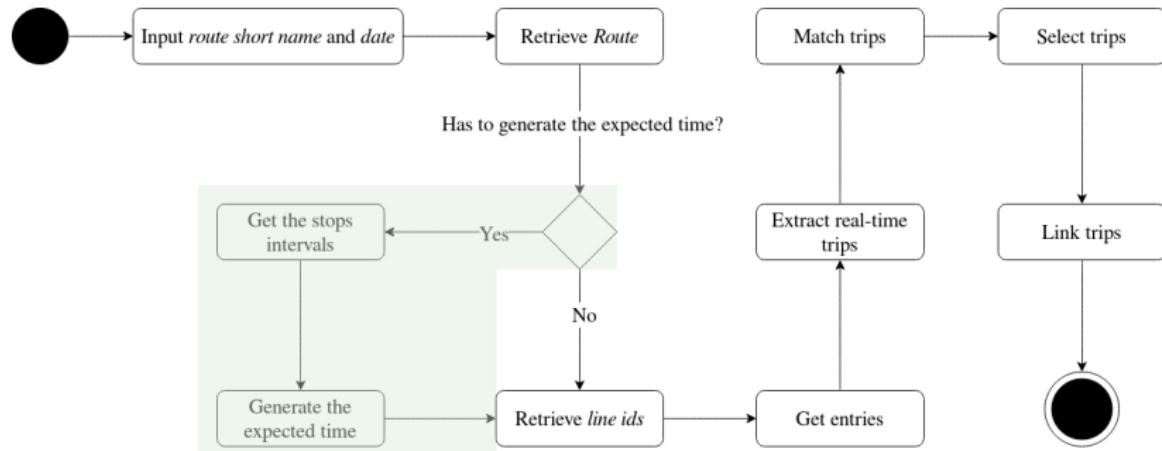


Figura: Integration Driver Activity Diagram



Integration Driver - 2nd Step*

Overview

- ① Get the stop intervals
- ② Generate the expected time using *TripExpectedTimeGenerator*

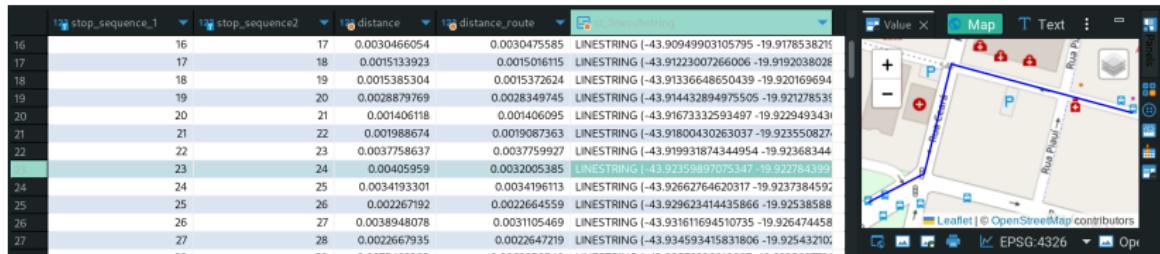


Figura: Example of bus stop couples and their distance



PUC Minas

TripExpectedTimeGenerator

Default Implementation key ideas

- Stop times are incremental in the trip and are incremented at each stop.
- *arrival_time* → *departure_time*
- The bus *should* travel the trip in average speed
- Minor deviations to the original *departure_time*



PUC Minas

Integration Driver - 3rd Step

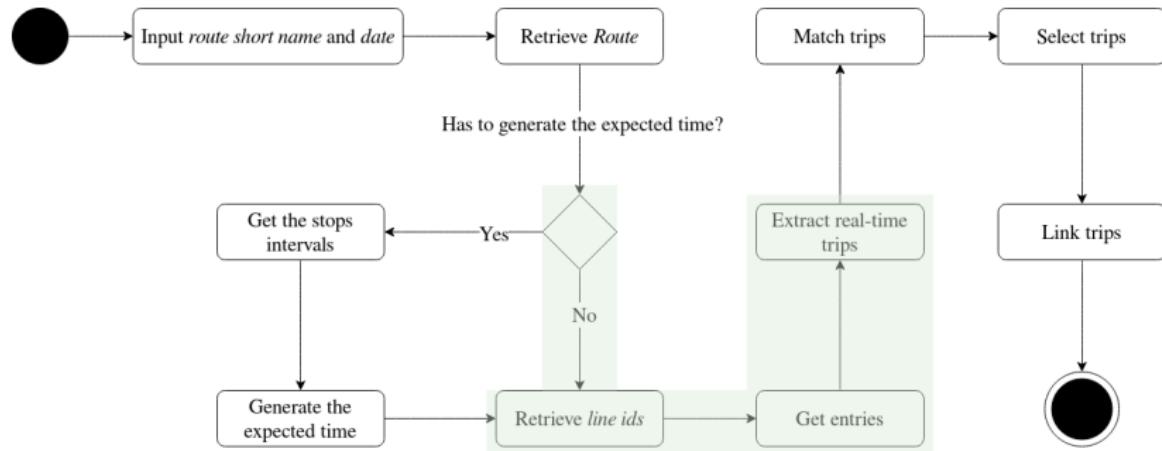


Figura: Integration Driver Activity Diagram



Integration Driver - 4th Step

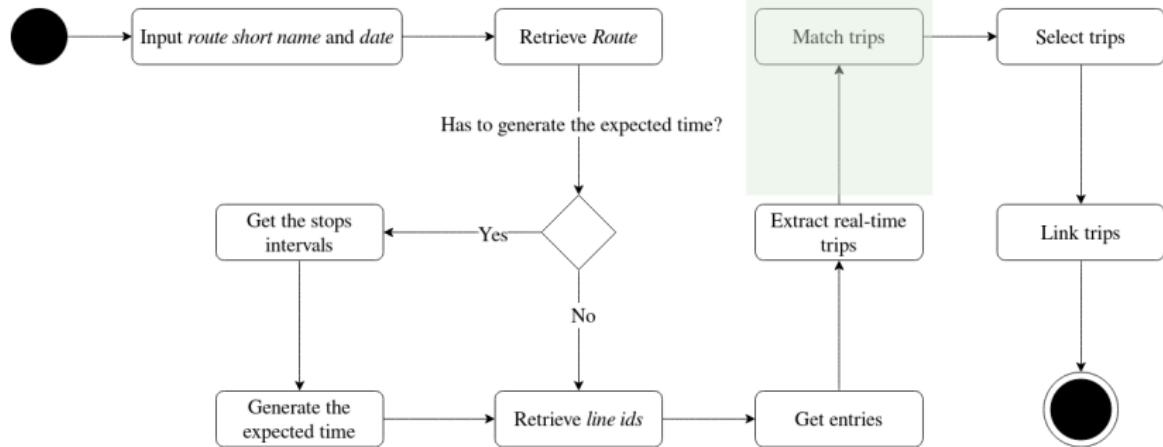


Figura: Integration Driver Activity Diagram



TripMatcher

Default Implementation key ideas

- Linking static data to real-time data
- Compare schedules using padding
- For a **scheduled trip** there are many **candidate trips**



PUC Minas

Integration Driver - 5th Step

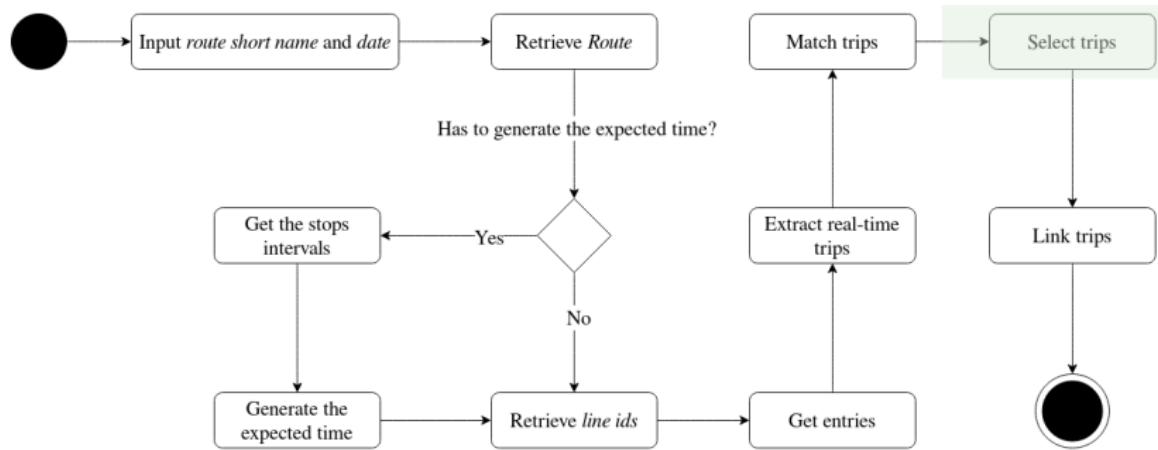


Figura: Integration Driver Activity Diagram



PUC Minas

Integration Driver - 6th Step

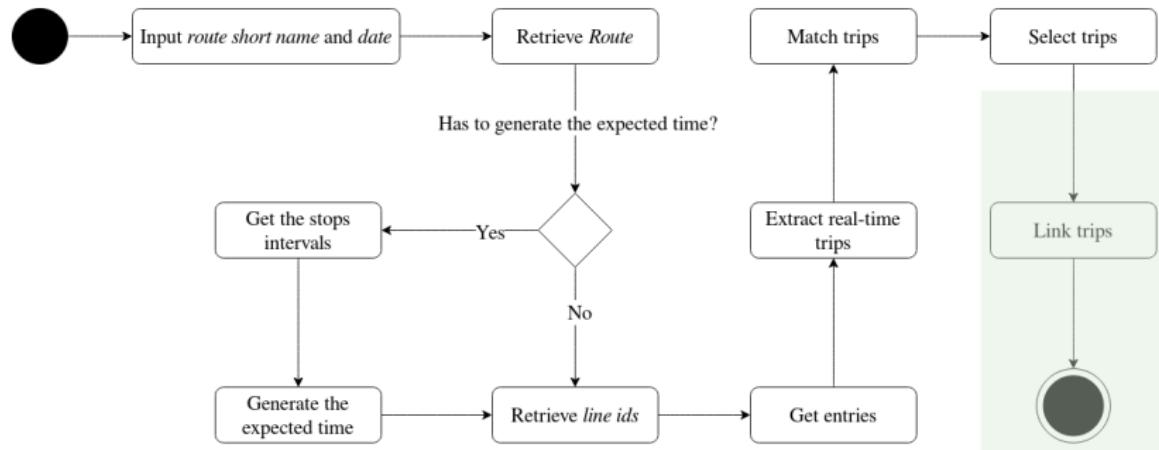


Figura: Integration Driver Activity Diagram



PUC Minas

TripBusStopLinker

Default Implementation Premises

- *RealTimeTrip* have been **around** every bus stop from their route
- What is the concept of an entry to be **around** a bus stop?
- Distance threshold d_t

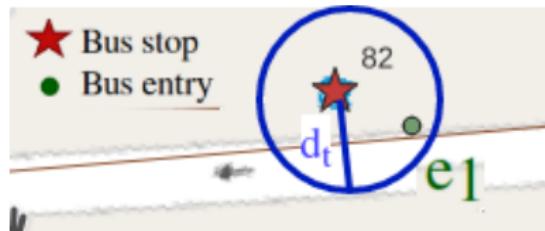


Figura: Entries and d_t representations

TripBusStopLinker

Default Implementation key ideas

Iterate over *Trip's busStopSequence* and for each bus stop s_x it is searched for all the **valid** entries within d_t^* , then scanning the *RealTimeTrip's entries* set.

Relationship between Entries and Bus Stops

An entry e to a bus stop s_x :

- ① An entry e can be associated with only one s_x
- ② A s_x can be related to more than one entry e
- ③ A s_x can be related to **zero entries**



Empty Set of Entries

Why do we have an empty set?

This does not imply that the bus has not been around a stop on the trip, it might be only a GPS positioning error. For example, if a bus is at a certain speed, it passes by a bus stop without stopping because there is no boarding or landing at that given stop.

Figura: A bus stop with no entries related



TripMissingEntriesGenerator

Default Implementation Overview

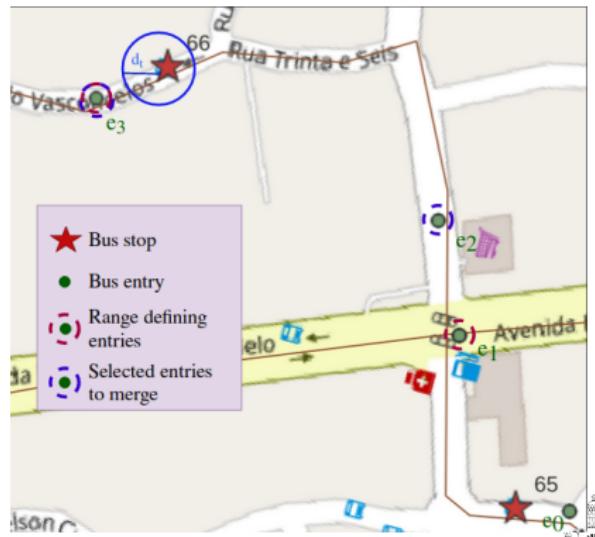
- Generate an artificial entry by merging a couple of entry
- The *EntryMerger* is the component in charge of merging two entries
 - **Linear Interpolation**



TripMissingEntriesGenerator

Which entries are going to be merged?

- Defining an interval of candidates entries
- *Lower Bound Entry* and *Upper Bound Entry*
- The closest entries to the bus stop from each side of the interval



Integration Driver

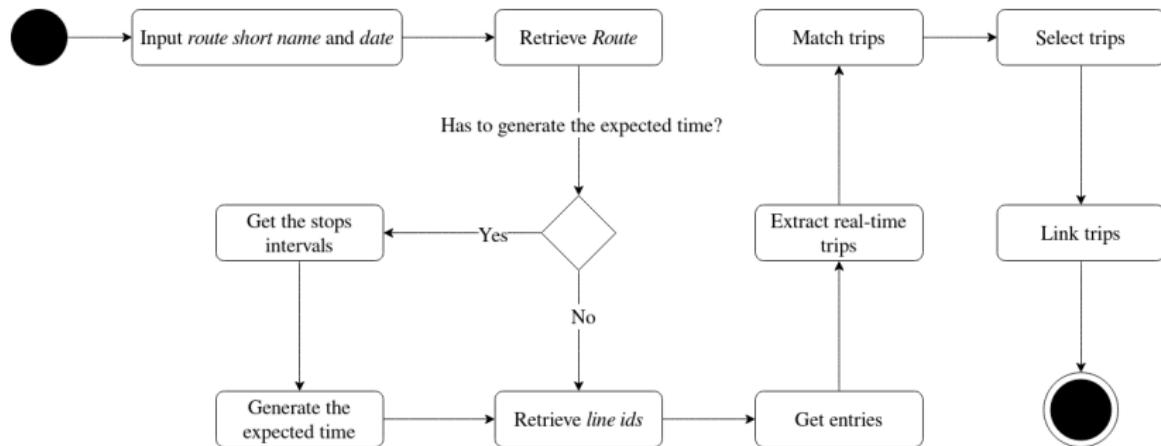


Figura: Integration Driver Activity Diagram



Integration Module

```
1 <dependency>
2   <groupId>br.pondionstracker</groupId>
3   <artifactId>integration-module</artifactId>
4   <version>1.0.0</version>
5 </dependency>
```

1

Figura: *IntegrationModule*'s maven dependency



PUC Minas

PondiônsTracker-BH

Overview

PondiônsTracker-BH^a is a *PondiônsTracker*'s specialization created to deal with Belo Horizonte's Public Transportation Network particularities. So, we have implemented our own *Real-Time Data collector*, and we have overwritten the method *getIdsLineByRouteId* from the *RealTimeService*.

- *BHTrans* → *GTFS*
- *Transfacil* → *Traffic API*

^aAvailable at <https://github.com/Pongelupe/PondionsTracker-BH>



Belo Horizonte's RealTimeService

BHRealTimeService

BHRealTimeService extends *DefaultRealTimeService* and overrides *getIdsLineByRouteld* method. There is a **one-to-many** relationship between the GTFS and the real-time data.



PUC Minas

Workload Overview

Workload

- Data collected for 11 days straight in August 2023
- 30 Gigabytes

Date	Day-of-Week	Entries
29-07-23	Saturday	22,319,765
30-07-23	Sunday	22,635,117
31-07-23	Monday	22,583,380
01-08-23	Tuesday	22,432,739
02-08-23	Wednesday	21,970,073
03-08-23	Thursday	22,050,579
04-08-23	Friday	22,402,865
05-08-23	Saturday	22,642,955
06-08-23	Sunday	22,786,254
07-08-23	Monday	22,109,606
08-08-23	Tuesday	22,405,222
Total	-	246,338,555



Schedule Analysis

Schedule-Filled Percentage

Schedule-Filled Percentage = Matched Trips / Scheduled Trips

- **Total:** $156,628 / 205,884 = 76.08\%$
- **Weekdays:** $118,559 / 159,418 = 74.37\%$
- **Saturdays:** $22,796 / 28,200 = 80.84\%$
- **Sundays:** $15,273 / 18,266 = 83.61\%$



PUC Minas

Schedule Analysis - Schedule Deviations

Schedule Deviations

Regarding the real-time API, there are collected trips which were not defined at Belo Horizonte's GTFS.

82 - Estação São Gabriel / Savassi Via Hospitais

On Sundays, the GTFS does not schedule any trip for route 82, but the API provided entries regarding this route twice during the period observed.



PUC Minas

Delay Analysis

Delay Notation

- **Delay:** ≥ 1 minute after
- **Ahead-of-Schedule:** ≥ 1 minute before
- **On time:** ≤ 59 seconds after OR ≤ 59 seconds before

	Weekday	Saturday	Sunday
Total trips matched	118,559	22,796	15,273
Trips entirely out of schedule	60,244	10,899	7,148
Trips with departure or arrival on time	39,403	8,731	5,988
Trips with departure and arrival on time	324	95	56
Trips entirely on time	1	2	1

Figura: Delays detailed in whole Public Transportation Network scale



PUC Minas

Delay Analysis

331 - Estação Barreiro/Conjunto Antonio Teixeira Dias Via Upa

Has 32 bus stops, representing a length of almost 9 kilometers.

- ① Jul. 29 15:30:00 - 15:56:27 → Jul. 29 15:30:03 - 15:57:03
- ② Jul. 30 08:20:00 - 08:46:27 → Jul. 30 08:20:31 - 08:46:15
- ③ Aug. 04 05:40:00 - 06:06:27 → Aug. 04 05:40:30 - 06:06:00
- ④ Aug. 05 17:10:00 - 17:36:27 → Aug. 05 17:10:45 - 17:36:49



Delay Analysis

Distribution of each status over the network

- **Delay:** 89.8%
- **Ahead-of-Schedule:** 6.9%
- **On time:** 3.3%

Attention!

The predominance of *DELAYED* in the Public Transportation Network **does not imply** that the network is not working nor completely stopped!



PUC Minas

Delay Analysis

Delays distribution over the network

- Bus Stop
- Trips

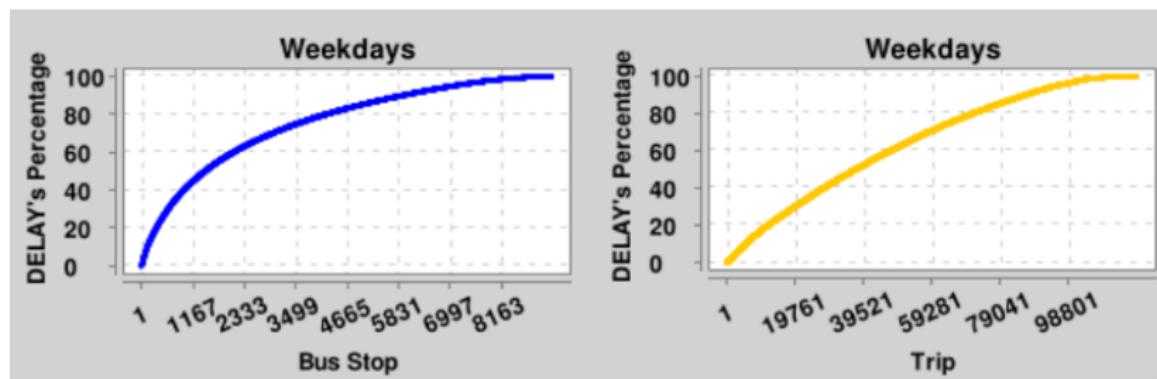


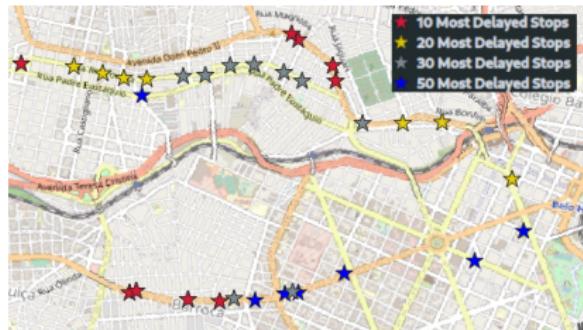
Figura: DELAYs Distribution: Bus Stop and Trip

Delay Analysis

Figura: 300 Most Delayed Stops



Figura: Fragment of the 50 Most Delayed Stops



PUC Minas

Delay Analysis

Three Most Delayed Stops for Weekdays

- ① #14793268 - *Avenida Dom Pedro II 1520* with 7,309 delays
- ② #14791617 - *Avenida Amazonas 7309* with 7,009 delays
- ③ #14790997 - *Avenida Dom Pedro II 1980* with 6,692 delays

Constants

- ① *Global Ahead Average:* 13.42 minutes
- ② *Global Delay Average:* 20.49 minutes



Delay Analysis

Stops #14793268 and
#14790997

- 462 meters
- 2,590 trips



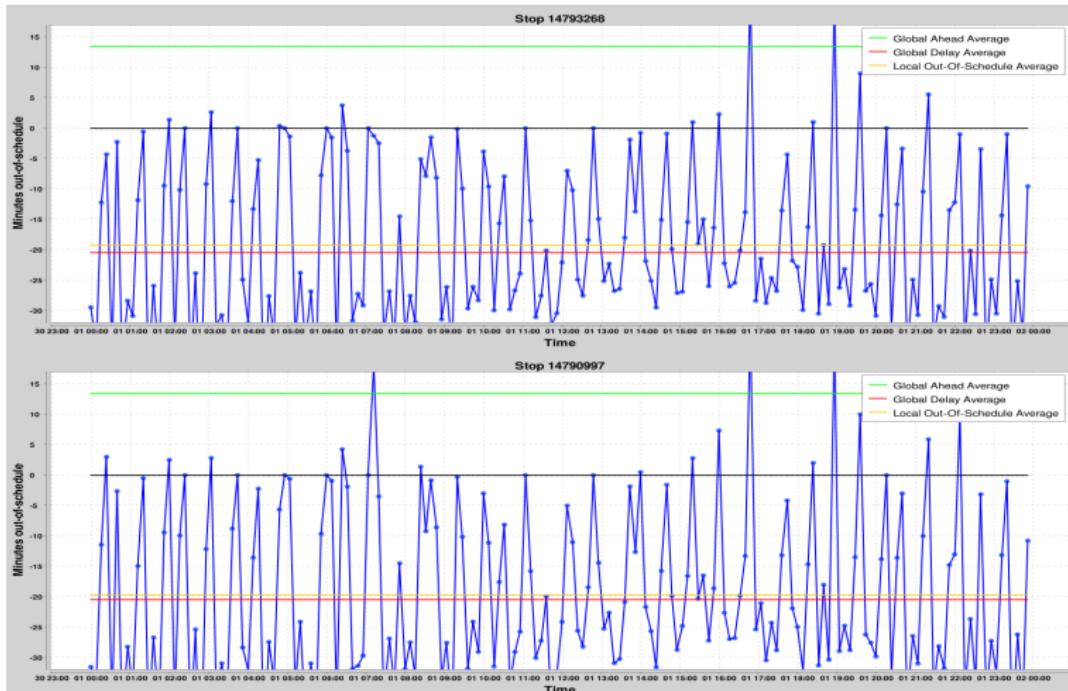
Local Out-Of-Schedule Average

- #14793268: 19.29 minutes
- #14790997: 19.68 minutes



PUC Minas

Delay Analysis



Comparison Between Generated and Real Data

Overview

The previous analysis was only possible because Belo Horizonte's GTFS defines the expected time for all bus stops on every trip. The *Trip Expected Time Generator* generates the expected times when missing, so, we executed this component with Belo Horizonte's data and compared the expected times generated with those defined at the GTFS.



Comparison Between Generated and Real Data

Trip entirely out of schedule

- GTFS: 78,291
- Generated: 75,073
- **Diff:** 3,218 (**4.29%**)

Trip entirely on time

- GTFS: 4
- Generated: 0
- **Diff:** 4 (**100%**)

Trips with departure **or** arrival on time

- GTFS: 54,122
- Generated: 54,271
- **Diff:** 149 (**0.27%**)

Trips with departure **and** arrival on time

- GTFS: 475
- Generated: 596
- **Diff:** 121 (**25.47%**)



Comparison Between Generated and Real Data

		GTFS	Generated
Weekday	<i>ON_TIME</i>	3.3%	3.2%
	<i>AHEAD_OF_SCHEDULE</i>	6.9%	17.8%
	<i>DELAYED</i>	89.8%	79.0%
Saturday	<i>ON_TIME</i>	3.9%	3.5%
	<i>AHEAD_OF_SCHEDULE</i>	6.5%	18.4%
	<i>DELAYED</i>	89.6%	78.1%
Sunday	<i>ON_TIME</i>	4.4%	3.9%
	<i>AHEAD_OF_SCHEDULE</i>	5.4%	18.5%
	<i>DELAYED</i>	90.2%	77.6%



Comparison Between Generated and Real Data

Global Averages

- *Global Ahead Average*
 - GTFS: 13.42 minutes
 - Generated: 38.57 minutes
 - **Diff:** 25.15 minutes
- *Global Delay Average*
 - GTFS: 20.49 minutes
 - Generated: 24.75 minutes
 - **Diff:** 4.26 minutes

Local Out-Of-Schedule Average

- #14793268
 - GTFS: 19.29 minutes
 - Generated: 15.54 minutes
 - **Diff:** 3.75 minutes
- #14790997
 - GTFS: 19.68 minutes
 - Generated: 14.68 minutes
 - **Diff:** 5 minutes



Limitations

Limitations

The *Real-Time Data Collector* is the most fragile component due to the third-party real-time traffic API interface.

- Size and quality of the data
- Scheduled routes with no entries reported
 - ① 720 - Circular Saúde MG20 missed 175 trips
 - ② 912 - Conjunto Taquaril/Praça Che Guevara missed 210 trips



Conclusion

Concluding Remarks

- Delays in Belo Horizonte follow a *log-normal* distribution
- Analysis using data generated with the *Trip Expected Time Generator*
- *PondiônsTracker* as a viable option when GTFS-RT is unavailable

Future Work

- Further explore Belo Horizonte Public Transportation Network using deep learning for graphs approaches
- Reproduce Belo Horizonte's results with other cities



UFSC Minas

Davide Bacciu, Federico Errica, Alessio Micheli, and Marco Podda.
2020. A gentle introduction to deep learning for graphs. *Neural Networks* 129 (sep 2020), 203–221.
<https://doi.org/10.1016/j.neunet.2020.06.006>

Jayanth Raghothama, Vinutha Magal Shreenath, and Sebastiaan Meijer. 2016. Analytics on Public Transport Delays with Spatial Big Data. In *Proceedings of the 5th ACM SIGSPATIAL International Workshop on Analytics for Big Geospatial Data* (Burlingame, California) (*BigSpatial '16*). Association for Computing Machinery, New York, NY, USA, 28–33.
<https://doi.org/10.1145/3006386.3006387>



Conclusion

Thanks!!

