GTFS-RIDE: TRANSIT RIDERSHIP DATA STANDARD ECOSYSTEM DEVELOPMENT

Final Report

PROJECT GTFS-Ride

GTFS-RIDE: TRANSIT RIDERSHIP DATA STANDARD ECOSYSTEM DEVELOPMENT

Final Report

PROJECT GTFS-Ride

by

J. David Porter, PhD, Phillip Carleton, Sylvan Hoover, Ben Fields
School of Mechanical, Industrial and Manufacturing Engineering
Oregon State University
Corvallis, Oregon 97331

for

January 2021

1. Report No. FHWA-OR-STIC-21-10	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle GTFS-ride: Transit Ridership Data Standard Ecosystem Development		5. Report Date January 2021
		6. Performing Organization Code
7. Author(s) J. David Porter, PhD, https://orcid.org/0000-0001-5937-1808 Phillip Carleton, https://orcid.org/0000-0002-0724-9832 Sylvan Hoover, https://orcid.org/0000-0003-1099-5112 Ben Fields		8. Performing Organization Report No.
9. Performing Organization Name and Address Oregon Department of Transportation Research Section 555 13th Street NE, Suite 1 Salem, OR 97301		10. Work Unit No. (TRAIS)
		11. Contract or Grant No.
12. Sponsoring Agency Name and Address Oregon Dept. of Transportation Research Section 555 13th Street NE, Suite 1 Salem, OR 97301		13. Type of Report and Period Covered Final Report
		14. Sponsoring Agency Code
15. Supplementary Notes		

16. Abstract: This report documents the results of a research project whose objective was to build on the results of project SPR 803 to identify the value proposition for adoption of the GTFS-ride data standard and to continue to develop an ecosystem of supporting software tools. By completing this project, the current state of practice, level of technical capabilities, and challenges related to ridership data faced by Oregon transit agencies are now better understood. Although it is recognized that the GTFS-ride data standard has the potential to solve many ridership data challenges, Oregon transit agencies are hesitant to implement the GTFS-ride data standard without integration into the software tools they currently use. With only slightly over a year in existence, the GTFS-ride data standard is still in its infancy, and additional education and dissemination efforts are needed in the future to achieve its widespread use. Despite these challenges, this project accomplished significant progress toward the goal of furthering the use of the GTFS-ride data standard. With improved supporting software tools and website, salient data conversion pilot case studies, interested stakeholder group creation, and extensive discussion and presentation, the project's dissemination efforts significantly increased the awareness and interest of the GTFS-ride data standard within public transit industry. With additional resources and strategic partnerships, there are great opportunities for furthering the adoption and expansion of the GTFS-ride data standard and for the realization of its benefits.

17. Key Words: Ridership; data standard		18. Distribution Statement	
19. Security Classification (of this report) Unclassified	20. Security Classification (of this page) Unclassified	21. No. of Pages 114	22. Price

Technical Report Form DOT F 1700.7 (8-72)   Reproduction of completed page authorized   Printed on recycled paper

SI* (MODERN METRIC) CONVERSION FACTORS

APPROXIMATE CONVERSIONS TO SI UNITS

Symbol	When You Know	Multiply By	To Find	Symbol
LENGTH				
in	inches	25.4	millimeters	mm
ft	feet	0.305	meters	m
yd	yards	0.914	meters	m
mi	miles	1.61	kilometers	km
AREA				
in^2	square inches	645.2	millimeters squared	mm^2
ft^2	square feet	0.093	meters squared	m^2
yd^2	square yards	0.836	meters squared	m^2
ac	acres	0.405	hectares	ha
mi^2	square miles	2.59	kilometers squared	km^2
VOLUME				
fl oz	fluid ounces	29.57	milliliters	ml
gal	gallons	3.785	liters	L
ft^3	cubic feet	0.028	meters cubed	m^3
yd^3	cubic yards	0.765	meters cubed	m^3
~~NOTE: Volumes greater than 1000 L shall be shown in m^3.				
MASS				
oz	ounces	28.35	grams	g
lb	pounds	0.454	kilograms	kg
T	short tons (2000 lb)	0.907	megagrams	Mg
TEMPERATURE (exact)				
°F	Fahrenheit	(F-32)/1.8	Celsius	°C

APPROXIMATE CONVERSIONS FROM SI UNITS

Symbol	When You Know	Multiply By	To Find	Symbol
LENGTH				
mm	millimeters	0.039	inches	in
m	meters	3.28	feet	ft
m	meters	1.09	yards	yd
km	kilometers	0.621	miles	mi
AREA				
mm^2	millimeters squared	0.0016	square inches	in^2
m^2	meters squared	10.764	square feet	ft^2
m^2	meters squared	1.196	square yards	yd^2
ha	hectares	2.47	acres	ac
km^2	kilometers squared	0.386	square miles	mi^2
VOLUME				
ml	milliliters	0.034	fluid ounces	fl oz
L	liters	0.264	gallons	gal
m^3	meters cubed	35.315	cubic feet	ft^3
m^3	meters cubed	1.308	cubic yards	yd^3
MASS				
g	grams	0.035	ounces	oz
kg	kilograms	2.205	pounds	lb
Mg	megagrams	1.102	short tons (2000 lb)	T
TEMPERATURE (exact)				
°C	Celsius	1.8C+32	Fahrenheit	°F

*SI is the symbol for the International System of Measurement

ACKNOWLEDGEMENTS

DISCLAIMER

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

EXECUTIVE SUMMARY

This report documents the results of the research project "*GTFS-ride: Transit Ridership Data Standard Ecosystem Development*", whose main objective was to identify the value proposition for adoption of the fixed-route transit ridership data standard known as GTFS-ride and to continue to develop an ecosystem of supporting software tools. The overall scope of the research project, which was executed between January 2018 and March 2019, was divided into the following six main tasks:

1. Identify and engage 4-6 transit providers in the use of GTFS-ride and accompanying open source software tools.

2. Engage other potential partners to disseminate use of GTFS-ride standard.

3. Pilot use of GTFS-ride with identified transit providers.

4. Develop and improve open source software tools.

5. Prepare draft final report and draft research note.

6. Final report and final research note.

As a result of the knowledge gained through initial conversations with Oregon transit agencies, ODOT and the OSU research team agreed to expand the scope of task #2 to engage more deeply with stakeholders to disseminate information and promote the adoption of the GTFS-ride data standard. To this end, the scope of tasks #1, #3, and #4 was revised and expanded to be more at the service of the dissemination and promotion activities at the core of task #2.

Reviewers of this report will be able to view the full development process, follow the justification for project decisions, evaluate the project outputs, and understand the challenges that have been overcome and those that still exist in the adoption of GTFS-ride as a fixed-route ridership data standard.

ENGAGEMENT WITH TRANSIT AGENCIES TO PILOT GTFS-RIDE STANDARD

Six Oregon transit agencies with varying capabilities, levels of service, and geographic locations were selected initially to learn more about their opportunities, challenges, levels of enthusiasm, and willingness to adopt the GTFS-ride data standard and its supporting software tools. A consistent finding in these discussions was that all these transit agencies currently use a myriad of different methods and procedures to process ridership data. In hopes of finding useful commonalities and better opportunities to develop generalizable solutions, four additional transit agencies were contacted.

All transit agencies contacted were also asked to provide a representative sample of their ridership data. The review of these ridership datasets provided a more thorough understanding of the varied ridership data environments that currently exist at transit agencies across Oregon, and to prompt ideas for potential enhancement to the GTFS-ride data standard and its supporting software tools. A small number of the ridership datasets collected were used to pilot conversion processes to create GTFS-ride compliant feeds.

The discussions with Oregon transit agencies and the review of several ridership datasets revealed several common themes:

- Due to the current state of practice with regards to ridership data (i.e., dominated by the use of custom- and/or internally-developed procedures for handling ridership data), Oregon transit agencies recognize the potential value of the GTFS-ride data standard.

- The majority of Oregon transit agencies actively use the Remix platform for analysis of their services, and mentioned that an integration of Remix with the GTFS-ride data standard would be helpful in facilitating the adoption of the latter.

- Oregon transit agencies would be much more inclined to implement the GTFS-ride data standard if it could be seamlessly integrated with the software tools they currently use to collect and process ridership data.

Piloting the Creation of GTFS-ride Datasets

The raw ridership datasets of four Oregon transit agencies were used to pilot the creation of GTFS-ride compliant feeds. The transit agencies that participated in this pilot were Lane Transit District (LTD), Rogue Valley Transportation District (RVTD), Josephine Community Transit (JCT), and Union County Public Transit (UCPT). Additional conversations, either by phone or in-person, were conducted with each of these transit agencies to discuss their specific ridership data practices before processing their raw ridership datasets to create GTFS-ride compliant feeds.

The lack of a consistent structure and the use of IDs generated internally by the transit agency (or generated automatically by their APC/ITS software) to identify stops and trips were the main challenges found in the datasets provided by LTD, RVTD, JCT, and UCPT. Due to these challenges, custom procedures had to be developed for each individual transit agency to convert their raw ridership dataset into compliant GTFS-ride feeds.

The scenarios encountered with the ridership datasets provided by LTD, RVTD, JCT, and UCPT helped to identify clear actions that would facilitate the adoption of the GTFS-ride standard. First, the standardization of best practices associated with the collection and representation of raw ridership data will aid in the conversion to GTFS-ride compliant feeds. Second, and perhaps more importantly, a great opportunity exists for transit data software and hardware vendors to offer GTFS-ride support as a native feature of their products.

ENGAGEMENT WITH OTHER PARTNERS TO DISSEMINATE GTFS-RIDE STANDARD

This project placed a significant emphasis on activities that disseminated the value of the GTFS-ride data standard and that developed a path for its widespread adoption. For example, a total of eight invited presentations were given at academic and practitioner conferences in 2018, and at least two more will be given in 2019. In addition, an archival journal article has been accepted for publication in the Transportation Research Record.

Another significant effort was the engagement with a broad group of industry stakeholders to both advice on the project development and to aid in the advocacy and promotion of the GTFS-ride data standard to the broader transit community. Contacts at potential member agencies were invited to join what would become known as the GTFS-ride Consortium Group in May 2018. A total of three 90-minute web-based consortium meetings were conducted (i.e., June and October of 2018, and a wrap-up meeting in February 2019). The output and related documents from all three consortium group meetings (e.g., meeting recordings, chat transcripts, presentation slides, etc.) can be accessed at https://www.gtfs-ride.org/consortium/meetings.html.

Finally, additional management activities were undertaken to carry over the discussions held during the consortium group webinar meetings to other various digital platforms. These activities involved managing the official GitHub repository of the GTFS-ride data standard and an updated version of a GTFS-ride Google Group discussion forum.

IMPROVEMENTS TO OPEN SOURCE SOFTWARE TOOLS

One of the main objectives of this project was to improve and further develop the open source software tools housed in WebHub (http://webhub.gtfs-ride.org/) that support the GTFS-ride data standard. As mentioned before, the knowledge gained at the onset of the project through the engagement and discussions with transit agencies and software vendors revealed that the software development efforts should be refocused towards a more demonstrative role in the support of the dissemination and promotion efforts of the GTFS-ride data standard. Therefore, the improvements made to WebHub were intended to help fulfill the new focus.

First, a completely redesigned database schema was created to align more closely with the needs of the GTFS-ride data standard. Also, several elements of WebHub's website were redesigned, including the aesthetic and layout of several web pages; the addition of user account security features to validate GTFS-ride feed uploads to the database; the form and function of the user interfaces (UIs) across the website; and the depiction of and interaction with report and visualization outputs. Finally, the software Tableau Public was implemented as the main tool to create reports and visualizations in a much more streamlined and flexible fashion.

Another effort in this project related to software development was the creation of an automated conversion tool for JCT using the Excel Visual Basic for Applications (VBA) programming language. JCT's conversion tool was tested and used to create a GTFS-ride compliant feed that was later uploaded to WebHub's redesigned database. The development, functionality, and usage of JCT's conversion tool, along with the associated analyses and data visualizations created with Tableau Public were demonstrated to both staff at JCT and too many additional stakeholders as a part of this project's dissemination efforts.

CONCLUSIONS, CHALLENGES, AND OPPORTUNITIES

The current state of practice, level of technical capabilities, and challenges related to ridership data faced by Oregon transit agencies are now better understood. Based on the feedback captured in discussions with transit industry stakeholders, it is recognized that the GTFS-ride data standard has the potential to solve many ridership data challenges. However, Oregon transit agencies are hesitant to implement the GTFS-ride data standard without integration into the software tools they currently use. With only slightly over a year in existence, the GTFS-ride data standard is still in its infancy, and additional education and dissemination efforts are needed in the future to achieve its widespread use.

Despite these challenges, this project accomplished significant progress toward the goal of furthering the use of the GTFS-ride data standard. With improved supporting software tools and website, salient data conversion pilot case studies, interested stakeholder group creation, and extensive discussion and presentation, the project's dissemination efforts significantly increased the awareness and interest of the GTFS-ride data standard within public transit industry. With additional resources and strategic partnerships, there are great opportunities for furthering the adoption and expansion of the GTFS-ride data standard and for the realization of its benefits.

1.0   INTRODUCTION

Personnel from the Rail and Public Transit Division (RPTD) of the Oregon Department of Transportation (ODOT) and a research team from Oregon State University (OSU) collaborated on the State Planning & Research (SPR) project titled "*SPR 803: Statewide Data Standards to Support Current and Future Strategic Public Transit Investment*" from August 2016 to December 2017. The main objective of project SPR 803 was the development of a standard to facilitate the collection, storage, sharing, reporting, and analysis of fixed-route ridership data for all Oregon public transit agencies. The main product of project SPR 803 was a fixed-route transit ridership data standard known as GTFS-ride, whose first version was released on September 6, 2017. As its name implies, the GTFS-ride data standard extends the general transit feed specification (GTFS) data standard by incorporating five additional files which transit agencies can use to reflect their ridership information. These five files are *board_alight.txt*, *trip_capacity.txt*, *rider_trip.txt*, *ridership.txt*, and *ride_feed_info.txt* (Porter, Carleton, Hoover, and Fields, 2018).

1.1   REPORT OBJECTIVE AND AUDIENCE

This report documents the results of the research project "*GTFS-ride: Transit Ridership Data Standard Ecosystem Development*", whose objective was to build on the results of project SPR 803 to identify the value proposition for adoption of the GTFS-ride data standard and to continue to develop an ecosystem of supporting software tools.

Table 1.1 shows that the overall scope of the research project was divided into six main tasks, which were completed between January 2018 and March 2019. Task #5 and task #6 focused on the preparation of draft and final versions of a final report and a research note. Different chapters and sections of this report document the work performed and the deliverables produced for each of these tasks. The entries shown in the last column of Table 1.1 are intended to help the reader make this connection.

Table 1.1: Main Task of Research Project

Task #	Description	Section(s)
1	Identify and engage 4-6 transit providers in the use of GTFS-ride and accompanying open source software tools.	3.0, 3.1, 3.2
2	Engage other potential partners to disseminate use of GTFS-ride standard.	4.0, 4.1, 4.2, 4.3., 4.3.1, 4.3.2
3	Pilot use of GTFS-ride with identified transit providers.	3.3, 3.3.1, 3.3.2, 3.3.3, 3.3.4
4	Develop and improve open source software tools.	3.4, 5.0, 5.1, 5.2, 5.3, 5.4
5	Prepare draft final report and draft research note.	
6	Final report and final research note.	

Within the first two months after the start of the research project, ODOT and the OSU research team realized that the time and resources estimated for each task had to be distributed differently than originally proposed in the work plan to better align with the current state of readiness, capabilities, understanding, and enthusiasm of Oregon transit agencies for the adoption of GTSF-ride. More specifically, it was discovered that many transit agencies in Oregon would need significantly more technological assistance in adopting the GTFS-ride data standard than this research project would be able to provide. Therefore, ODOT and the OSU research team decided to expand the scope of task #2 to engage more deeply with stakeholders to disseminate information and promote the adoption of the GTFS-ride data standard. To this end, the scope of tasks #1, #3, and #4 was revised and expanded to be more at the service of the dissemination and promotion activities described in task #2.

The main audience for this report is the project sponsors at ODOT and the Federal Highway Administration (FHWA). Other intended audiences include industry stakeholders (e.g., state and federal transportation agencies, transit agencies, regional planners, modelers, software and hardware vendors, etc.) and academic researchers. Reviewers of this report will be able to view the full development process, follow the justification for project decisions, evaluate the project outputs, and understand the challenges that have been overcome and those that still exist in the adoption of GTFS-ride as a fixed-route ridership data standard.

1.2   REPORT ORGANIZATION

This report follows the logical and chronological steps undertaken by the OSU research team to complete the work outlined in the work plan. If any adjustments were made to a task, these are documented within each section. Where needed, the reader is referred to online resources to obtain more detailed specifics of some of the individual project elements.

The remainder of this report is organized as follows. Chapter 2 documents the knowledge and experiences gathered through the engagement with a wide spectrum of Oregon transit agencies to

pilot the GTFS-ride data standard. Chapter 3 details the activities conducted to disseminate the GTFS-ride data standard at several conference, as well as with a consortium composed of transit agencies, transportation consultants, and software vendors. Chapter 4 documents the process and results of improving the web-based software tools in support of the dissemination and adoption of the GTFS-ride data standard. Finally, Chapter 5.0 presents the conclusions of the research project and outlines the opportunities and recommendations for future work.

2.0 ENGAGEMENT WITH TRANSIT AGENCIES TO PILOT GTFS-RIDE STANDARD

The first task in this research project involved engaging with a representative set of Oregon transit agencies. The main goal of this engagement was to learn more about the opportunities, challenges, levels of enthusiasm, and willingness of these transit agencies to consider the adoption of the GTFS-ride data standard and its supporting software tools. The findings that resulted from the discussions with these transit agencies were both enlightening and unanticipated. In particular, it became clear that most of the transit agencies contacted initially would face significant challenges in adopting the GTFS-ride data standard given their current level of readiness and technical capabilities. Therefore, the overall scope and the main focus of the research project had to be altered.

The rest of this chapter is organized as follow. Section 2.1 describes outreach activities conducted with Oregon transit agencies. Section 2.2 summarizes the main findings gleaned from the engagement with transit agencies. Section 2.3 outlines the work performed to pilot the use of the GTFS-ride data standard using raw ridership data from a select group of transit agencies. Finally, Section 2.4 discusses the development of a custom GTFS-ride conversion tool created for a small transit agency in Oregon.

2.1 TRANSIT AGENCY SELECTION AND OUTREACH

The Oregon transit agencies that were engaged at the onset of the research project were selected based on consultations with ODOT and by referencing questionnaire responses collected in project SPR 803, which belonged to transit agencies that expressed future willingness to engage with the OSU research team. An initial set of six transit agencies with varying capabilities, levels of service, and geographic locations representing the breadth of scenarios seen in Oregon was selected from these two sources. In discussing potential challenges of adopting the GTFS-ride data standard with this initial small set of transit agencies, it was found that they currently use a myriad of different methods and procedures to process their ridership data. Therefore, the outreach and data collection efforts were expanded in hopes of finding useful commonalities and better opportunities to develop generalizable solutions.

In addition to conducting a combination of face-to-face, email, phone, and web discussions about the GTFS-ride project, all transit agencies contacted were asked to provide a representative sample of their ridership data to the OSU research team. While most transit agencies were able to provide ridership data at the lowest level collected and reported, some were not able to provide data due to various reasons. In addition, the OSU research team had access to a small number of ridership datasets obtained either while completing project SPR 803 or through intermediary sources without holding direct discussions with a transit agency. Many of these datasets, while not necessarily candidates for piloting the GTFS-ride data standard, helped to provide a more thorough understanding of the varied ridership data environments that currently exist at transit

agencies across Oregon, and to prompt ideas for potential enhancement to the GTFS-ride data standard and its supporting software tools.

Table 2.1 shows the Oregon transit agencies that were targeted for direct engagement, those transit agencies that were actually contacted, and the source of the ridership data that was analyzed by the OSU research team.

Table 2.1: Oregon Transit Agency Engagement and Data Collection Activities

Agency/Service	Direct Engagement	Ridership Data
Amtrak	No	Yes (through ODOT)
Salem Area Mass Transit District (Cherriots)	Yes	No
Columbia Gorge Express (CGE)	Yes	Yes
Corvallis Transit System (CTS)	Yes	No
Josephine Community Transit (JCT)	Yes	Yes
Lane Transit District (LTD)	Yes	Yes
Metro Transit	Yes (through prior phase)	Yes (through prior phase)
The POINT	Yes	Yes
Rogue Valley Transportation District (RVTD)	Yes	Yes
Tillamook County Transportation District (TCTD)	Yes	Yes
TriMet	Yes	Yes (small sample through NITC PORTAL)
Union County Public Transit (UCPT)	Yes	Yes

2.2 MAIN FINDINGS FROM DISCUSSIONS WITH TRANSIT AGENCIES

The discussions with Oregon transit agencies and the review of the ridership datasets began to reveal some common themes. While it was known through the information collected with the questionnaire administered in project SPR 803 that ridership data suffers from a lack of standardization in data formats and is collected, processed, stored, and analyzed in a variety of ways, the scale of the problem was not fully grasped until the detailed discussions with transit agencies took place.

As previously mentioned, each transit agency contacted follows their own internal procedures for handling ridership data with its own specific formatting and fields, and with their own unique challenges and issues, often compounded by a lack of sufficient resources (e.g., funding,

available staff time, staff education, proper authorization, technological capability, etc.). Some transit agencies have additional challenges due to the structure of the transit services they operate. For example, the POINT intercity bus service managed by ODOT is operated through several contracted carriers and ticketed through other agencies (i.e., Amtrak and Greyhound). This structure adds extra complexity to ridership data since it comes through such varied sources. Due to the current state of practice with regards to ridership data, a common theme heard from transit agencies was the recognition of potential value in GTFS-ride as a data standard. Even smaller agencies like JCT and UCPT expressed interest in having greater analysis capability of their ridership data.

It was also learned that the majority of Oregon transit agencies actively use the Remix platform (www.remix.com) for analysis of their services, and mentioned that an integration of Remix with the GTFS-ride data standard would be helpful in facilitating the adoption of the latter. Further discussions with transit agencies also revealed that they would be much more inclined to implement the GTFS-ride data standard if it could be seamlessly integrated with the software tools they currently use to collect and process ridership data. However, this integration must include the data collection and creation side through APC, AFC, CAD/AVL hardware and software, as well as the data consumption side through analysis software to be effective.

The discussions with Oregon transit agencies about the hardware and software they utilize to collect and process ridership data led to direct engagements with some vendors of these solutions. The OSU research team had discussions with moovel (www.moovel-group.com) and TripSpark (www.tripspark.com) which helped to reinforce the potential value of the GTFS-ride data standard and provided additional insights on project challenges and opportunities.

2.3   PILOTING THE CREATION OF GTFS-RIDE DATASETS

The raw ridership data provided by a few Oregon transit agencies were selected to pilot the creation of GTFS-ride compliant feeds. The Oregon transit agencies selected, shown in Table 2.2, operate fleets and serve populations of varied sizes. Lane Transit District (LTD) serves the urban area of Eugene-Springfield and surrounding communities of Lane County. Rogue Valley Transportation District (RVTD) operates in the smaller urban communities of Jackson County (e.g., Medford, Ashland, etc.) in southwestern Oregon, while Josephine Community Transit (JCT) operates mostly in the town of Grants Pass in the more rural, adjacent Josephine County. Union County Public Transit (UCPT), a part of Northeast Oregon Public Transit, operates one fixed route in the small town of La Grande in eastern Oregon.

Table 2.2: Oregon Transit Agencies Used to Pilot the GTFS-ride Data Standard

Transit Agency	# of Buses	Population	Area (sq. miles)
Lane Transit District (LTD)	85	302,000	482
Rogue Valley Transportation District (RVTD)	26	132,000	50
Josephine Community Transit (JCT)	11	48,000	80
Union County Public Transit (UCPT)	2	13,000	5

There were additional conversations, either by phone or in-person, with each transit agency listed in Table 2.2 to discuss their specific ridership data practices before processing their raw ridership data to create GTFS-ride compliant feeds. It is important to note that the OSU research team was not provided with raw APC output data even though some of the transit agencies involved do use APCs to collect ridership. Instead, transit agencies provided pre-processed data extracts from proprietary APC data systems.

The following sections describe the steps followed to process raw ridership data to create GTFS-ride compliant feeds.

2.3.1   Lane Transit District

LTD provided two comma-separated values (CSV) files containing custom datasets of raw, stop-level ridership data collected via APC technology and processed through the transit scheduling software HASTUS (www.giro.ca/en). Examples of these two datasets are depicted in Figure 2.1 and Figure 2.2. A data dictionary that explained the meanings of the data fields was also provided.

Both raw ridership datasets provided by LTD were very close to containing the required elements to create a compliant GTFS-ride feed. However, the absence of a `stop_sequence` prevented the development of a simple automated conversion process. It was determined that through data processing steps and reasonable assumptions, it would be possible to infer the missing `stop_sequence` data needed to make the link from the raw ridership data to the associated base GTFS feed.

The procedure to find the matching `stop_sequence` required for a GTFS-ride compliant feed for LTD was coded in the Scala programming language and processed with the Apache Spark analytics engine housed at Databricks (databricks.com). Databricks is a cloud-based, distributed computing platform that is easily scalable to handle large amounts of data and helps in reducing the required processing time. The Databricks distributed computing platform was chosen (as opposed to single machine computing) due to the number of steps required to process each record in LTD's raw ridership datasets, and in anticipation of the need to process much larger volumes of raw ridership data for LTD.

After trimming LTD's raw ridership datasets to match the corresponding GTFS active date range, no trips containing a stop visited more than twice were found in LTD's GTFS feed. Therefore, any trips in the raw ridership dataset containing stops visited more than twice were considered as errors and filtered out, which resulted in a loss of only 0.3% of the records.

In the ridership dataset provided initially by LTD (see Figure 2.1), it was also discovered that each stop visit entry contained only an associated data collection time. To solve the issue of the missing `stop_sequence` element, a second ridership dataset was requested from LTD (see Figure 2.2) that also included an ordinal sequence value for each stop visit entry. Two different methods that used the values of these two data fields (i.e., data collection time and ordinal sequence) were implemented and compared for assigning a GTFS `stop_sequence` to each stop visit entry. Each method sorted one of these two data fields and searched the GTFS feed to make a confident match inference to the needed `stop_sequence`. At the end, the sorting method based

on the ordinal sequence data field performed better than the sorting method based on the data collection time with 99.8% of data entries being able to be attributed to a value in GTFS.

After the two processing steps mentioned above (i.e., removing multi-stop visit errors and record sorting and matching based on the ordinal sequence value) were applied to the raw ridership dataset provided by LTD depicted in Figure 2.2, a full 99.3% of stop visit entries were successfully matched to GTFS stop_sequence values. However, the sorting method based on the data collection time was required for the raw ridership dataset provided by LTD depicted in Figure 2.1 since it lacked the ordinal sequence value field. It is important to note that the valid GTFS-ride feed that was created with the first raw ridership dataset provided by LTD was used for analysis throughout the remainder of the project.

The processing steps needed to convert LTD's raw ridership data into a GTFS-ride compliant feed are depicted in Figure 2.3.

	A	B	C	D	E	F	G	H	I
1	calendar_date	route	dir	stop	trip_sn	msg_time	board	alight	departure_load
2	2/10/2017	41	I	1028	1756042	18:35	4	1	28
3	2/10/2017	41	I	1029	1756042	18:37	0	1	27
4	2/10/2017	41	I	1032	1756042	18:39	0	1	26
5	2/10/2017	41	I	1035	1756042	18:41	2	1	27
6	2/10/2017	41	I	1036	1756042	18:42	0	5	22
7	2/10/2017	41	I	1540	1756042	18:44	0	1	21
8	2/10/2017	41	I	escenter	1756042	18:46	0	17	4
9	2/10/2017	11	O	2505	1755197	14:52	10	0	10
10	2/10/2017	11	O	1489	1755197	15:01	2	0	12
11	2/10/2017	11	O	42	1755197	15:06	0	1	11
12	2/10/2017	11	O	37	1755197	15:08	1	2	10
13	2/10/2017	11	O	1776	1755197	15:15	0	2	8
14	2/10/2017	11	O	1	1755197	15:19	2	3	7
15	2/10/2017	11	O	2	1755197	15:20	3	0	10
16	2/10/2017	11	O	6	1755197	15:23	1	0	11
17	2/10/2017	11	O	12	1755197	15:25	0	2	9
18	2/10/2017	11	O	14	1755197	15:26	0	4	5
19	2/10/2017	11	O	15	1755197	15:27	0	1	4
20	2/10/2017	11	O	1328	1755197	15:28	0	1	3
21	2/10/2017	11	O	2505	1755204	16:02	7	0	7
22	2/10/2017	11	O	249	1755204	16:13	0	1	6
23	2/10/2017	11	O	50	1755204	16:14	0	2	4

LTD Data 9-18-16-3-18-17

Figure 2.1: First raw ridership dataset provided by LTD

block	msg_time	route	dir	pattern_id	pattern	stop	stop_name	pc_latitude	pc_longitude	gn_latitude	gn_longitude	the_bus	bus_class	artic	service_type_text	service_type_general	trip_id	trip_sn	trip_end	block_stop_order	board	alight	departure_load
1020	9:00	95	O	22441	13	2107	Eugene Station, Bay G	440483681	-1230940719	440483031	-1230940871	1101	1100	n	Sunday	sun	605911	1812598	9:41	2	2	1	1
1020	9:15	95	O	22441	13	967	E/S of Hwy 99 N of Roosevelt	440642664	-1231371782	440642566	-1231372131	1101	1100	n	Sunday	sun	605911	1812598	9:41	8	1	0	2
1020	9:16	95	O	22441	13	965	E/S of Hwy 99 N of Side	440670437	-1231397829	440669902	-1231398053	1101	1100	n	Sunday	sun	605911	1812598	9:41	9	1	0	3
1020	9:18	95	O	22441	13	961	E/S of Hwy 99 N of Fairfield	440752141	-1231474297	440751551	-1231474175	1101	1100	n	Sunday	sun	605911	1812598	9:41	11	0	1	2
1020	9:27	95	O	22441	13	1951	N/S of Recovery Way E of Drea	441698352	-1232013793	441648341	-1231951208	1101	1100	n	Sunday	sun	605911	1812598	9:41	19	0	1	1
1020	9:39	95	O	22441	13	9008	N/S of 8th E of Holly	442209743	-1232035343	442209652	-1232035184	1101	1100	n	Sunday	sun	605911	1812598	9:41	25	0	2	0
1020	9:39	95	I	22385	12	9008	N/S of 8th E of Holly	442209743	-1232035343	442209652	-1232035184	1101	1100	n	Sunday	sun	605912	1812600	10:25	26	6	0	6
1020	9:45	95	I	22385	12	9850	N/S of W. 10th W of Ivy (JC)	442226518	-1232052113	442226134	-1232051637	1101	1100	n	Sunday	sun	605912	1812600	10:25	27	0	1	5
1020	9:48	95	I	22385	12	9853	W/S of Maple N of W. 1st	442152138	-1232088127	442152028	-1232088118	1101	1100	n	Sunday	sun	605912	1812600	10:25	30	1	0	6
1020	10:10	95	I	22385	12	959	W/S of Hwy 99 of Lakewood	440798717	-1231514414	440799138	-1231514491	1101	1100	n	Sunday	sun	605912	1812600	10:25	38	0	3	3
1023	17:33	95	O	22441	13	2107	Eugene Station, Bay G	440483241	-1230941030	440483031	-1230940871	6217	6200	n	Sunday	sun	605973	1812599	18:43	357	1	0	1
1023	18:15	95	O	22441	13	967	E/S of Hwy 99 N of Roosevelt	440642474	-1231372054	440642566	-1231372131	6217	6200	n	Sunday	sun	605973	1812599	18:43	363	1	0	2
1023	18:39	95	O	22441	13	9847	S/S of E. 1st E of Railroad tracks	442149501	-1232035496	442149376	-1232027933	6217	6200	n	Sunday	sun	605973	1812599	18:43	375	0	1	1
1023	18:43	95	O	22441	13	9008	N/S of 8th E of Holly	442209736	-1232035263	442209652	-1232035184	6217	6200	n	Sunday	sun	605973	1812599	18:43	380	0	2	0
1023	18:43	95	I	22385	12	9008	N/S of 8th E of Holly	442209736	-1232035263	442209652	-1232035184	6217	6200	n	Sunday	sun	605974	1812601	19:25	381	1	0	1
1023	18:45	95	I	22385	12	9850	N/S of W. 10th W of Ivy (JC)	442226371	-1232051683	442226134	-1232051637	6217	6200	n	Sunday	sun	605974	1812601	19:25	382	1	0	2
1023	18:49	95	I	22385	12	9853	W/S of Maple N of W. 1st	442152160	-1232087940	442152028	-1232088118	6217	6200	n	Sunday	sun	605974	1812601	19:25	385	1	0	3
1023	19:22	95	I	22385	12	1186	S/S of 7th W of Washington	440520700	-1231006119	440521018	-1231006045	6217	6200	n	Sunday	sun	605974	1812601	19:25	401	0	1	2
1023	19:25	95	I	22385	12 escenter		Eugene Sta central arrival zone	440478635	-1230936773	440482110	-1230935701	6217	6200	n	Sunday	sun	605974	1812601	19:25	403	0	3	0
13	6:00	95	O	22441	13	2107	Eugene Station, Bay G	440483459	-1230940862	440483031	-1230940871	6219	6200	n	Weekday	wkd	606327	1810240	6:40	2	7	0	7
13	6:04	95	O	22441	13	323	N/S of 6th W of Willamette	440532831	-1230931405	440532484	-1230932861	6219	6200	n	Weekday	wkd	606327	1810240	6:40	5	1	0	8
13	6:10	95	O	22441	13	967	E/S of Hwy 99 N of Roosevelt	440642859	-1231372051	440642566	-1231372131	6219	6200	n	Weekday	wkd	606327	1810240	6:40	8	1	0	9
13	6:17	95	O	22441	13	9841	E/S of Hwy 99 N of Irving	441036756	-1231723930	441037780	-1231724397	6219	6200	n	Weekday	wkd	606327	1810240	6:40	16	0	2	7
13	6:18	95	O	22441	13	9233	E/S of Hwy 99 N of Airport Rd	441140291	-1231821062	441139710	-1231821333	6219	6200	n	Weekday	wkd	606327	1810240	6:40	17	0	2	5
13	6:26	95	O	22441	13	1951	N/S of Recovery Way E of Drea	441648373	-1231951372	441648341	-1231951208	6219	6200	n	Weekday	wkd	606327	1810240	6:40	19	0	1	4
13	6:33	95	O	22441	13	9843	S/S of E 1st E of second drivew	442149400	-1231992920	442149091	-1231993558	6219	6200	n	Weekday	wkd	606327	1810240	6:40	21	0	2	2
13	6:34	95	O	22441	13	9848	E/S of Birch N of E. 3rd	442171279	-1231966812	442170520	-1231966753	6219	6200	n	Weekday	wkd	606327	1810240	6:40	22	3	0	5
13	6:36	95	O	22441	13	9849	N/S of E 6th W Bi-mart DW	442193531	-1232007839	442193352	-1232007847	6219	6200	n	Weekday	wkd	606327	1810240	6:40	24	3	0	8
13	6:37	95	O	22441	13	9008	N/S of 8th E of Holly	442209829	-1232034967	442209652	-1232035184	6219	6200	n	Weekday	wkd	606327	1810240	6:40	25	0	1	7
13	6:37	95	I	22385	12	9008	N/S of 8th E of Holly	442209829	-1232034967	442209652	-1232035184	6219	6200	n	Weekday	wkd	606328	1810237	7:25	26	7	0	14
13	6:42	95	I	22385	12	9850	N/S of W. 10th W of Ivy (JC)	442226480	-1232051363	442226134	-1232051637	6219	6200	n	Weekday	wkd	606328	1810237	7:25	27	0	1	13
13	6:43	95	I	22385	12	9851	N/S of W 10th East of Maple	442225526	-1232087531	442225078	-1232087305	6219	6200	n	Weekday	wkd	606328	1810237	7:25	28	1	0	14

GTFS Ride LTD

Figure 2.2: Second raw ridership dataset provided by LTD
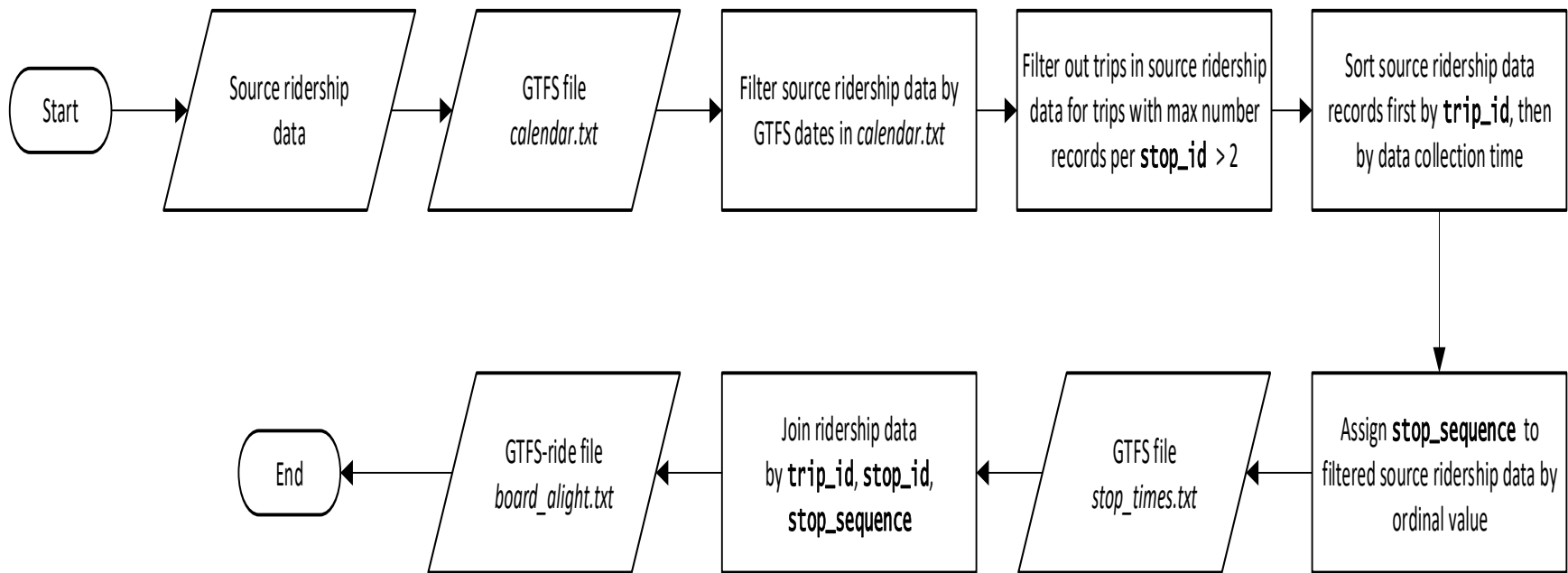
Figure 2.3: Steps to convert LTD's raw ridership data into a GTFS-ride compliant feed

2.3.2   Rogue Valley Transportation District

Similar to LTD, RVTD also has buses equipped with APCs and provided custom extracts (in CSV format) from their fixed-route transit software developed by TripSpark (www.tripspark.com). An example of RVTD's raw ridership data is depicted in Figure 2.4. Also similar to LTD, the ridership data provided by RVTD contained almost all the required fields necessary to represent stop-visit level data in the GTFS-ride file *board_alight.txt*. However, values for the GTFS fields' `trip_id` and `stop_id` were missing, thus preventing the use of the same methods developed to process LTD's ridership data into GTFS-ride compliant feeds.

The processing steps needed to convert RVTD's raw ridership data into a GTFS-ride compliant feed are depicted in Figure 2.5. As a first step, a simple join query based on the field `stop_code` (present in both the raw ridership data and the GTFS feed) was used to match the data to the respective GTFS `stop_id`. However, there was no field in the raw ridership data that could serve similarly as a corollary for the field `trip_id`. To address this issue, two methods to infer the correct `trip_id` from information contained in the raw ridership data were developed. The first method used the stop times and the set of stops all under one `trip_id` (this ID bared no relation to the GTFS `trip_id`) of the raw ridership data to look for corresponding matches in RVTD's GTFS feed. A second method parsed out information from the text contained in the field `trip_name` (second column in Figure 2.4) of the raw ridership dataset, which contained concatenated text strings of the route number, direction, and trip start time. The parsed data was then used to search for matches in RVTD's GTFS feed. As these methods could be resource intensive when applied to large ridership datasets, the distributed computing platform Databricks was used again to perform the main processing tasks. Each of the two methods was able to match approximately 96% of the trips in the raw ridership dataset provided by RVTD to a corresponding GTFS `trip_id` before being trimmed to fit within the active date range of a single GTFS feed.

Although the methods used to process the raw ridership datasets of RVTD and LTD were automated and executed quickly (due to the small temporal scales of the datasets and the use of distributed processing), there was a relatively extensive effort required to build and test these custom solutions. It would appear that simple changes to the current practices employed by the transit agencies to collect and store ridership data could prevent similar challenges from occurring. For example, the data conversion procedures developed and used by the OSU research team would not have been needed if a policy were implemented requiring the use of GTFS IDs (e.g., `stop_id` and `trip_id`) to identify the corresponding transit system elements (i.e., a stop and a trip) instead of IDs generated internally by the transit agency based on tribal knowledge or generated automatically by their APC/ITS software.

Nevertheless, the scenarios encountered at LTD and RVTD show the great opportunity for transit data software and hardware vendors to offer GTFS-ride support as a native feature of their products.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1	trip_id	trip_name	stop_code	stope_sequence	record_use	schedule_relationship	boardings	alightings	current_load	load_type	rack_down	bike_boardings	bike_alightings	ramp_used	ramp_boardings	ramp_alightings	service_arrival_date	service_arrival_time	source
2	24331	24 - IB - 16:38	11250	18	0	0	0	0	6	0	NULL	NULL	NULL	NULL	NULL	NULL	20170101	14:41:00	1
3	24900	24 - OB - 14:00	1	0	0	0	1	2	19	0	NULL	NULL	NULL	NULL	NULL	NULL	20170104	14:08:00	1
4	24900	24 - OB - 14:00	1	0	0	0	9	0	9	0	NULL	NULL	NULL	NULL	NULL	NULL	20170106	13:59:00	1
5	24900	24 - OB - 14:00	1	0	0	0	6	0	7	0	NULL	NULL	NULL	NULL	NULL	NULL	20170109	14:14:00	1
6	24900	24 - OB - 14:00	1	0	0	0	4	6	16	0	NULL	NULL	NULL	NULL	NULL	NULL	20170112	14:01:00	1
7	24900	24 - OB - 14:00	1	0	0	0	3	0	0	0	NULL	NULL	NULL	NULL	NULL	NULL	20170113	13:58:00	1
8	24900	24 - OB - 14:00	1	0	0	0	0	0	0	0	NULL	NULL	NULL	NULL	NULL	NULL	20170116	13:58:00	1
9	24900	24 - OB - 14:00	1	0	0	0	6	8	5	0	NULL	NULL	NULL	NULL	NULL	NULL	20170119	14:00:00	1
10	24900	24 - OB - 14:00	1	0	0	0	0	0	0	0	NULL	NULL	NULL	NULL	NULL	NULL	20170123	13:56:00	1
11	24900	24 - OB - 14:00	1	0	0	0	12	7	56	0	NULL	NULL	NULL	NULL	NULL	NULL	20170125	14:01:00	1
12	24900	24 - OB - 14:00	1	0	0	0	0	0	30	0	NULL	NULL	NULL	NULL	NULL	NULL	20170126	13:56:00	1
13	24900	24 - OB - 14:00	1	0	0	0	0	0	10	0	NULL	NULL	NULL	NULL	NULL	NULL	20170201	14:00:00	1
14	24900	24 - OB - 14:00	1	0	0	0	3	5	0	0	NULL	NULL	NULL	NULL	NULL	NULL	20170207	14:01:00	1
15	24900	24 - OB - 14:00	1	0	0	0	5	4	7	0	NULL	NULL	NULL	NULL	NULL	NULL	20170210	14:02:00	1
16	24900	24 - OB - 14:00	1	0	0	0	3	6	3	0	NULL	NULL	NULL	NULL	NULL	NULL	20170213	14:01:00	1
17	24900	24 - OB - 14:00	1	0	0	0	0	0	7	0	NULL	NULL	NULL	NULL	NULL	NULL	20170110	13:59:00	1
18	24900	24 - OB - 14:00	1	0	0	0	2	6	3	0	NULL	NULL	NULL	NULL	NULL	NULL	20170110	14:02:00	1
19	24900	24 - OB - 14:00	1	0	0	0	0	0	8	0	NULL	NULL	NULL	NULL	NULL	NULL	20170118	14:01:00	1
20	24900	24 - OB - 14:00	1	0	0	0	0	0	7	0	NULL	NULL	NULL	NULL	NULL	NULL	20170119	13:57:00	1
21	24900	24 - OB - 14:00	1	0	0	0	1	3	0	0	NULL	NULL	NULL	NULL	NULL	NULL	20170131	14:00:00	1
22	24900	24 - OB - 14:00	1	0	0	0	0	0	25	0	NULL	NULL	NULL	NULL	NULL	NULL	20170202	13:58:00	1
23	24900	24 - OB - 14:00	1	0	0	0	5	6	4	0	NULL	NULL	NULL	NULL	NULL	NULL	20170203	14:00:00	1
24	24900	24 - OB - 14:00	1	0	0	0	0	0	3	0	NULL	NULL	NULL	NULL	NULL	NULL	20170208	13:59:00	1
25	24900	24 - OB - 14:00	1	0	0	0	4	2	5	0	NULL	NULL	NULL	NULL	NULL	NULL	20170208	14:02:00	1
26	24900	24 - OB - 14:00	1	0	0	0	2	7	0	0	NULL	NULL	NULL	NULL	NULL	NULL	20170214	14:02:00	1
27	24900	24 - OB - 14:00	1	0	0	0	0	0	2	0	NULL	NULL	NULL	NULL	NULL	NULL	20170217	14:08:00	1
28	24900	24 - OB - 14:00	1	0	0	0	0	6	0	0	NULL	NULL	NULL	NULL	NULL	NULL	20170217	14:11:00	1
29	24900	24 - OB - 14:00	1	0	0	0	4	0	3	0	NULL	NULL	NULL	NULL	NULL	NULL	20170220	13:56:00	1
30	24900	24 - OB - 14:00	1	0	0	0	3	0	36	0	NULL	NULL	NULL	NULL	NULL	NULL	20170222	14:01:00	1
31	24900	24 - OB - 14:00	1	0	0	0	2	0	4	0	NULL	NULL	NULL	NULL	NULL	NULL	20170224	14:02:00	1

boardings_alightings_rvtd_2017

Figure 2.4: RVTD raw ridership data

13
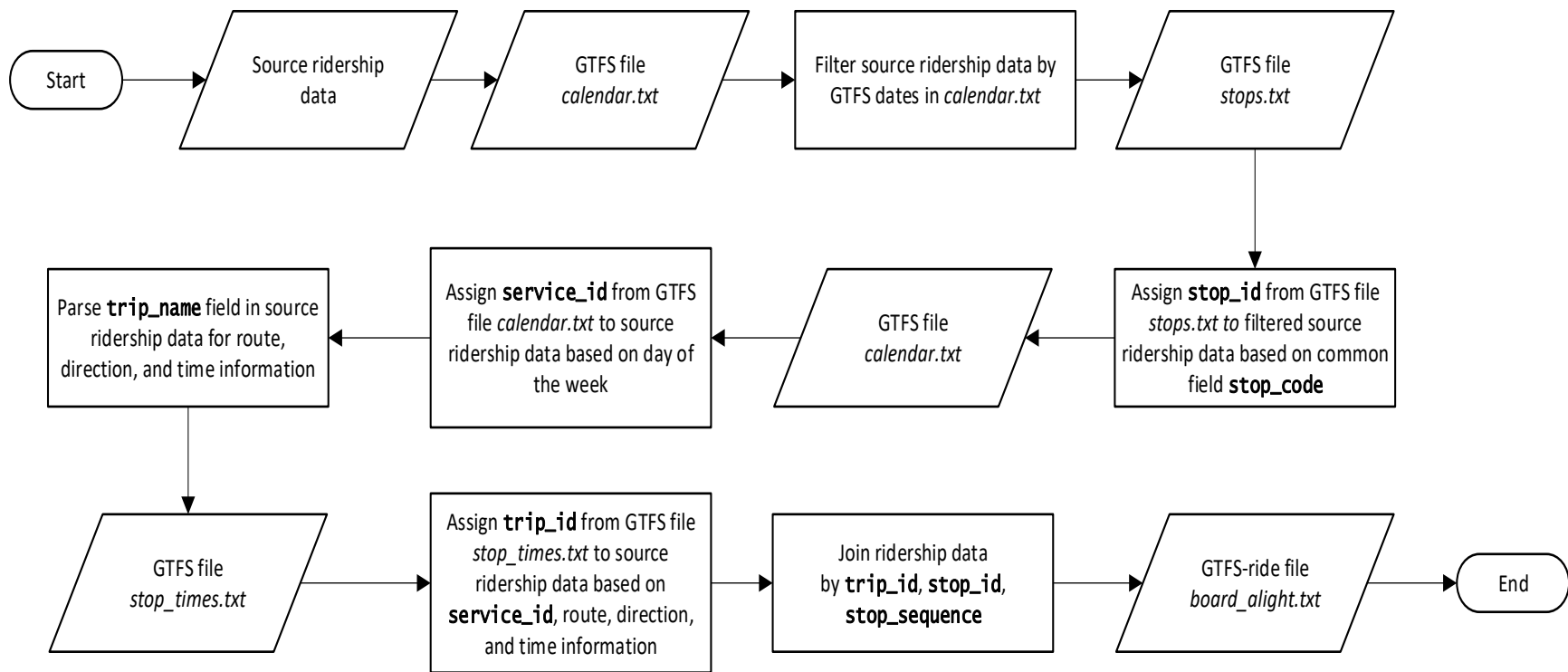
Figure 2.5: Steps to convert RVTD's raw ridership data into a GTFS-ride compliant feed

2.3.3 Josephine Community Transit

As seen in Table 2.2, JCT operates a much smaller transit system (when compared to LTD and RVTD) with only five regular fixed routes and one intercity commuter route. Unlike LTD and RVTD, JCT collects ridership data manually. Vehicle operators use manual tally counters and driver logs to record boardings by rider type, which are then entered in tables in MS Excel spreadsheets. This manual collection process requires ridership to be recorded at a higher level of aggregation for both the five regular routes (daily route-level boardings) and for the four-stop commuter route (daily stop-level boardings). Occasionally, JCT also performs a one-day manual count of stop-level boardings and alighting by visually reviewing video footage collected by security cameras installed on their buses. Figure 2.6 depicts samples of the raw ridership data provided by JCT that were collected for the five regular routes, the four-stop commuter route, and from the review of bus security camera video footage.

The level of aggregation of the raw ridership data provided by JCT required the use of the GTFS-ride file *ridership.txt*, whose flexibility allowed for both the route-level and the stop-level raw ridership data to be represented in a single file. The samples of ridership data depicted in Figure 2.6(a) and Figure 2.6(b) show that the information needed for the required fields of the GTFS-ride file *ridership.txt* are present, but they are not recorded in the specified format. For example, the table depicted in Figure 2.6(a) shows that the ridership data on a particular route is denoted by a table header (i.e., "Rt 10"). However, the GTFS-ride file *ridership.txt* requires route-level ridership to be indicated by a `route_id` contained in the transit agency's GTFS feed. While the `route_short_name` for route "Rt 10" is "10", the string "Rt 10" does not occur anywhere in JCT's GTFS feed. Several other data format issues and challenges with GTFS identifiers existed in the raw ridership data provided by JCT.

(a) Sample raw ridership data for JCT's five regular routes

	A	B	C	D	E	F	G	H	I
2			JCT PUBLIC BUS PASSENGER REPORT						
3	Rt 10								
4	DATE	RCC	REG	SR	PASS	FREE	TRSF	TOTAL	CASH
5	7/1/2017								
6	7/2/2017								
7	7/3/2017	0	23	20	31	8	38	120	$123.30
8	7/4/2017								
9	7/5/2017	2	42	24	46	7	32	153	$50.01
10	7/6/2017	3	55	29	61	9	24	181	$75.47
11	7/7/2017	2	60	19	35	4	40	160	$51.64
12	7/8/2017							0	

(b) Sample raw ridership data for JCT's four-stop commuter route

	V	W	X	Y	Z	AA	AB
3	RVCL - RT100						
4	DATE	GP	RR	GH	RVTD	TOTAL	
5	7/1/2017						
6	7/2/2017						
7	7/3/2017	19	9	3	27	58	
8	7/4/2017					0	
9	7/5/2017	27	6	4	35	72	
10	7/6/2017	21	6	3	24	54	

(c) Sample raw ridership data for JCT's bus security camera video

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	0	Route 10/10a			1/30/2015			Attribute					
2	0.3												
3	0.2	10a											
4	0.3	Distance											
5	0.3		Stop List:		Visable Locator					on	off	Total	%
6	0.5	0	Ann Basker Auditorium					Shelter		198	151	349	12.64%
7	0.5	0.3	6th and G St		Dutch Bros.					44	40	84	3.04%
8	0.2	0.5	6th and K St							22	33	55	1.99%
9	0.3	0.8	6th St		Riverside Complex					50	23	73	2.64%
10	0.3	1.1	West Park St		RV Fitness Center					9	14	23	0.83%
11	0.3	1.6	Ringuette		Senior Meal Site / before driveway			Shelter		54	54	108	3.91%
12	0.4	2.1	Fairgrounds/YMCA							3	19	22	0.80%
13	0.4	2.3	Redwood Ave & Allen Creek							5	25	30	1.09%
14	0.1	2.6	Redwood Ave		Spring Meadow		10'west of fire hydrant			0	8	8	0.29%

Figure 2.6: Examples of JCT's raw ridership data

The MS Excel spreadsheets provided by JCT that contained counts of ridership data also lacked standardized table formats, which prevented the development of an automated data conversion process for creating a GTFS-ride compliant feed. Therefore, the GTFS-ride files were created manually though a process of reconciling the information in the customized MS Excel tables with the identifiers in their corresponding GTFS feeds. The raw data for the route-level ridership included designations that could be verified manually and matched to the field `route_short_name` in the GTFS file *routes.txt*. The stop-level ridership data from the commuter route required more steps to manually verify all the identification numbers that link a stop all the way back to the transit agency in the GTFS feed. This was a critical step to connect the stop designations in the raw ridership data for JCT's commuter route to stops in the corresponding GTFS feed. The required `stop_id` information was verified manually, and the raw ridership data was converted though a sequential process of linking related IDs through the GTFS files *agency.txt, routes.txt, trips.txt, stop_times.txt,* and finally *stops.txt*. The main steps of this manual conversion process are depicted in Figure 2.7. It is important to distinguish this aggregated, stop-level data from the disaggregated, stop-level data processed in the LTD and RVTD scenarios. The raw ridership data for JCT's commuter route did not contain enough information to distinguish boardings at a stop on a specific running of the commuter route (i.e., a trip). Therefore, all boardings at a stop had to be aggregated over all trips run over the course of the service day.

The conversion process for the small amount of stop-level data collected through the review of video footage (i.e., Figure 2.6c) proved very time-intensive and of uncertain quality. The internal, agency-specific descriptions of some of the stops (e.g., "at the end of the walkway", "just before the end of construction sign", etc.) made it very difficult to (1) distinguish among several stops that were named very similarly, or (2) find any matches in the GTFS file *stops.txt*. Although attempts to address this challenge by comparing the raw ridership data sorted by distance traveled (i.e., column B in Figure 2.6c) with the GTFS file *stop_times.txt* sorted by `stop_sequence` did help make a few additional stop matches, it was decided that the minimal value added by the process was overshadowed by the effort required.
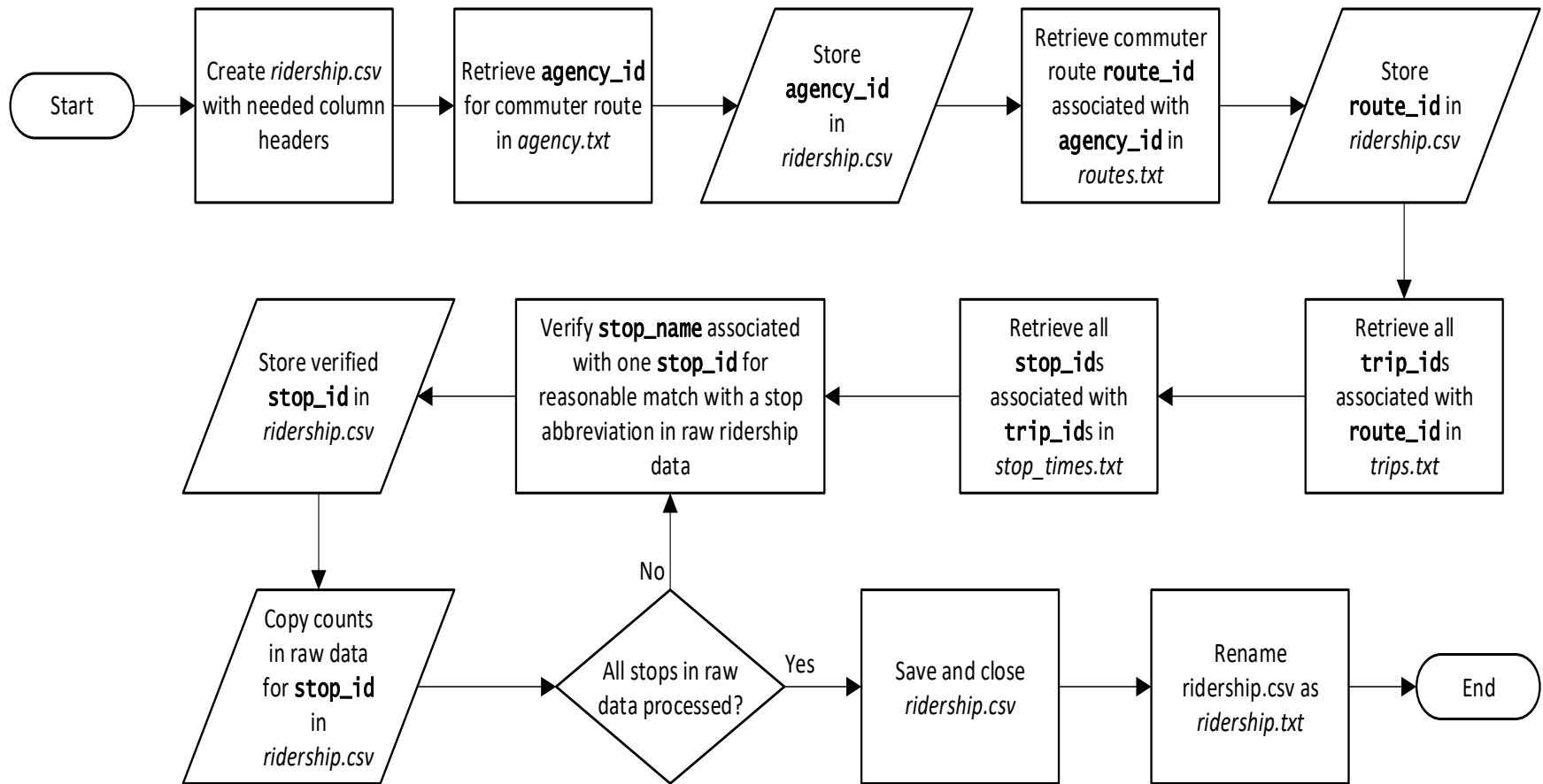
Figure 2.7: Manual conversion process for JCT's commuter route

18

2.3.4　Union County Public Transit

UCPTs fixed-route system services a single-loop route with a single vehicle on standard service days. Like JCT, ridership data is collected manually by drivers with tickers and entered into their transit software the next morning. UCPT provided a simple, standard ridership report exported from StrataGen ADEPT (www.ddswireless.com/adept) in MS Excel format spanning about ten months of service. The process required to create a GTFS-ride compliant feed from this raw ridership data was similar (in concept) to the methods used in the JCT case study. However, it was a much quicker and less involved process due to the simplicity of the scenario.

Through a discussion with UCPT, it was learned that the fixed-route service is occasionally altered or expanded for holidays and special events (e.g., county fair, St. Patty's day Pub Crawl), and that the daily ridership counts are delineated as occurring either in the AM or PM. Figure 2.8 shows that in the ridership dataset provided by UCPT, these distinctions are indicated by a unique, internal RouteID. After reviewing the UCPT GTFS feed, it was found that these distinctions are not reflected with unique GTFS IDs. Therefore, the UCPT GTFS-ride records were created by simply summing the ridership over these service distinctions for each day and recorded with the verified `route_id` from the GTFS file *routes.txt*. While the simplicity of this process lends itself to automation, it was determined to be more efficient to process the data manually in this instance.

Figure 2.8: UCPT raw ridership data

The JCT and UCPT scenarios exemplify several potential issues in the adoption of the GTFS-ride data standard. One issue is that standardizing best practices around collecting and representing raw ridership data will aid in the conversion to the GTFS-ride data standard until, hopefully, the conversion process is not needed due to software/hardware vendor adoption. Another self-evident issue is that the manual data collection of ridership data limits the amount of granularity that can be achieved, thus potentially limiting its value. A less evident challenge is the potential loss of some detail in the conversion process. The details of rider types at JCT and

19

the distinction of AM, PM, or special services at UCPT were lost in the conversion of ridership data to the GTFS-ride format.

2.4   AUTOMATED GTFS-RIDE CONVERSION TOOL FOR JCT

Through the course of creating GTFS-ride compliant feeds from the raw ridership data provided by transit agencies, it became apparent that there was an opportunity to develop a simple and automated ridership data conversion tool. Due to the potential expediency of development and to add the greatest value, this effort focused on smaller transit agencies that use MS Excel as the primary tool for storing and using ridership data.

The transit agency JCT was selected as the prime candidate based on the characteristics of the raw ridership data they collect. Further discussions with JCT staff were scheduled to seek approval and to acquire further insights into their ridership data and analysis requirements. Through these discussions, it was also learned that JCT staff had recently revised and improved the consistency of formatting of the MS Excel spreadsheets they use to store and organize their ridership data. In support of the development of the automated conversion tool, JCT staff made a commitment to maintain the consistency across future data files for the foreseeable future. It was the challenge posed by this previous lack of consistency which had precluded the use of automated methods for the initial conversion of JCT raw ridership data to a GTFS-ride compliant feed. Finally, a newer ridership dataset was requested and obtained to test the conversion tool during development.

The conversion tool for JCT was developed with the Excel Visual Basic for Applications (VBA) programming language. The development of the conversion tool took advantage of the specific structure of JCT's template data and the assumption of consistency in that much functionality was "hard coded" into the program. Thus, the Excel VBA code will not function properly if this structure is altered, and it is not transferable to another transit agency storing ridership data in MS Excel. Figure 2.9 depicts the workflow and logic employed in the Excel VBA program. Figure 2.10 depicts the instructions that are provided to a user in the conversion tool via the MS Excel user interface. The complete Excel VBA code listing is included in Appendix A.

JCT's conversion tool was tested and used to create a GTFS-ride compliant feed. The GTFS-ride compliant feed created was later uploaded to WebHub's redesigned database. Also, new visualizations of the data were created with the software Tableau Public. The development, functionality, and usage of JCT's conversion tool, along with the associated analyses, were then demonstrated to both staff at JCT and too many additional stakeholders as a part of this project's dissemination efforts.

It is important to note that JCT's conversion tool is a custom solution. As such, it applies only to JCT's specific ridership data representation. Nevertheless, JCT's conversion tool served a role in both aiding JCT in the creation of GTFS-ride compliant feeds, and in demonstrating potential future opportunities to other transit agencies and software vendors.
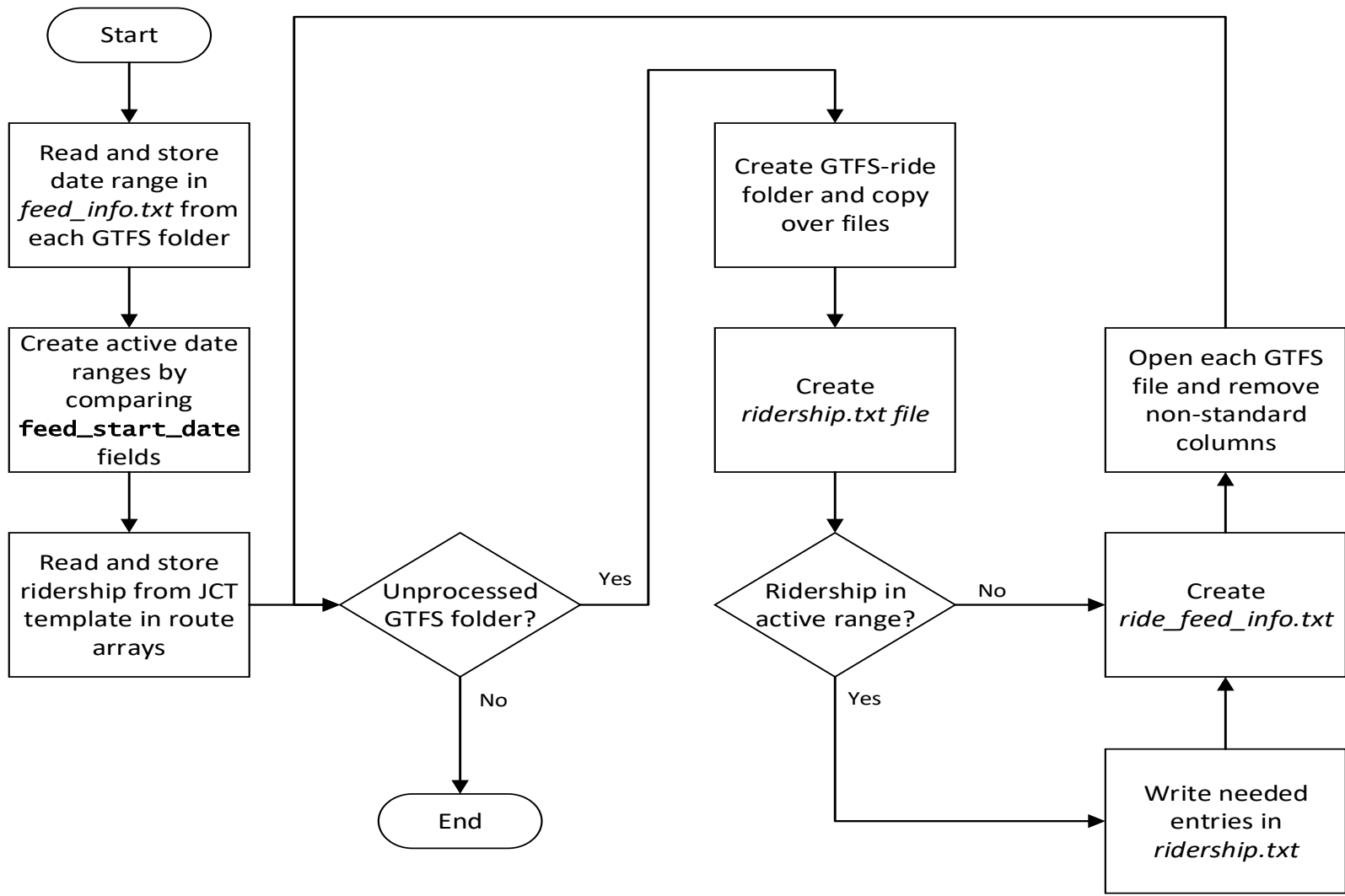
Figure 2.9: Workflow of JCT's GTFS-ride conversion tool

JCT GTFS-ride Feed Creation Tool

Process Steps

1. Create folder to hold input files - name and location unimportant

2. Download or move the required GTFS feeds (i.e., those necessary to cover the ridership data date range) into the folder - recommended to name by published date

3. Unzip GTFS feed folders (if zipped) into folder

4. Place the template files (*ride_feed_info.csv* and *ridership.csv*) into the folder

5. Place the raw JCT ridership data file into the folder

6. Place this file (*JCT_GTFSride_DataConverter.xlsm*) into the folder

7. Specify the final date of ridership data contained within the raw data file in the following cell (*format mm/dd/yyy*): 9/19/2018

8. Press the following button to begin data conversion:
Note: Procedure may make Excel unresponsive and screen blank while running. Procedure will take more time with more GTFS feeds (up to 2 minutes). Please wait. Control will be returned when process is complete.

Convert Ridership

9. GTFS-ride feed folders will be in the folder with "-ride" appended to original GTFS feed folder names. They may now be zipped for distribution or used as desired.

Figure 2.10: User interface of JCT's GTFS-ride conversion tool

3.0 ENGAGEMENT WITH OTHER PARTNERS TO DISSEMINATE GTFS-RIDE STANDARD

The value and potential for wide adoption of the GTFS-ride data standard were established through the completion of project SPR 803 (Porter et al., 2018). In the research project "*GTFS-ride: Transit Ridership Data Standard Ecosystem Development*", a significant emphasis was placed on activities that disseminated this value and that developed a path for widespread adoption of the GTFS-ride data standard.

This chapter documents the results of conducting these dissemination activities, which included presentations at academic and practitioner conferences, forming a consortium group of stakeholders, hosting consortium group webinars, and leading and managing online discussions.

3.1 INDUSTRY CONFERENCE ATTENDANCE AND PRESENTATION

Table 3.1 highlights the extensive efforts to disseminate the value and potential of the GTFS-ride data standard that were made through presentations at academic and practitioner conferences. The recognition of the potential value of the GTFS-ride data standard is evident by the fact that each conference presentation was selected competitively from a submitted abstract or funded by an inviting organization. Contacts built through prior engagements with industry stakeholders (particularly with Trillium Solutions, Inc.) assisted in facilitating several of the listed conference presentations. These contacts continue to be productive in providing new dissemination opportunities and have also led to informal presentations and discussions with the state DOTs of North Carolina and California.

The composition of the audience at the various conferences varied in terms of their general familiarity with the GTFS standard (on which the GTFS-ride data standard is based) and the transit issues around ridership data. For example, the audience at the 2018 Institute of Industrial and Systems Engineering (IISE) conference was predominately academic researchers in the field of Industrial Engineering, while the attendees at the State Public Transportation Partnerships (SPTP) conference were predominately state DOT officials. Even so, the response from conference presentation attendees was very positive and receptive to the ideas proposed and the potential of the GTFS-ride data standard. Many of the presentations prompted more in-depth, one-on-one conversations. Also, some conference presentations led to further opportunities. For example, the SPTP conference in Minneapolis led to an invitation to give a presentation at the California Association for Coordinated Transportation (CALACT) conference in Napa, which itself may lead to additional dissemination opportunities. There is also future potential for the 2019 Transportation Research Board (TRB) and 2019 IISE conference presentations to be expanded into archival journal publications.

Table 3.1: Conference Presentations Made to Disseminate the GTFS-ride Data Standard

Conference	Date and Location	Presentation Title	Invited/Selected
2018 Institute of Industrial and Systems Engineering (IISE) Annual Conference and Expo	May 2018 Orlando, FL	GTFS-ride: The First Standard for Transit Ridership Data	Competitive Selection
2018 State Public Transportation Partnerships (SPTP) Conference	August 2018 Minneapolis, MN	GTFS-ride: Standards-Based Rider Data for Better Decisions	Invited (travel funded by CALACT)
23rd National Conference on Rural Public and Intercity Bus Transportation (RPIBT)	September 2018 Breckenridge, CO	GTFS-Ride: The First Standard for Transit Ridership Data	Competitive Selection
2018 Oregon Public Transportation Conference (OPTC)	October 2018 Bend, OR	The GTFS-ride Data Standard with APC and AFC Data	Invited (travel funded by ODOT)
2018 Oregon Public Transportation Conference (OPTC)	October 2018 Bend, OR	The GTFS-ride Data Standard: Using GTFS Datasets for Ridership	Invited (travel funded by ODOT)
California Association for Coordinated Transportation (CALACT) 2018 Autumn Technology & Shared Mobility Conference	November 2018 Napa, CA	The GTFS-ride Data Standard: Using GTFS Datasets for Ridership	Invited (travel funded by CALACT)
2019 Transportation Research Board (TRB) Annual Meeting	January 2019 Washington DC	GTFS-ride: A Unifying Standard for Fixed-Route Ridership Data	Competitive Selection
2019 Institute of Industrial and Systems Engineering (IISE) Annual Conference and Expo	May 2019 Orlando, FL	Implementing the First Standard for Transit Ridership Data: Lessons Learned	Competitive Selection

3.2   CONSORTIUM GROUP FORMATION AND ENGAGEMENT

One of the main efforts of this research project was the engagement with a broad group of industry stakeholders to elicit their active participation in the dissemination, improvement, and adoption of the GTFS-ride data standard. Therefore, it was determined early on that an industry stakeholder group was desired to both advice on the project development and to aid in the advocacy and promotion of the GTFS-ride data standard to the broader transit community. When envisioning this group, there was also a strong desire to bring public transit agencies and their supplying vendors together for discussions aimed at solving problems and strengthening the business case for adopting the GTFS-ride data standard.

Contacts at potential member agencies were invited to join what would become known as the GTFS-ride Consortium Group in May 2018. Table 3.2 lists the original members of the GTFS-ride Consortium Group. The invitees were asked to commit to participating in three, 90-minute meetings in 2018 and early 2019, to review project documents, and to provide their expertise (see Appendix B). Specifically, they were asked to provide insights on the following questions:

- Does the GTFS-ride data standard need to be adjusted so that it will better meet your future needs and provide value to other GTFS-ride users? How?

- What set of software functionality should be developed to provide value to GTFS-ride users?

- What can we collectively do to move GTFS-ride from a data format proposal to a data standard?

The 90-minute meetings were conducted as online webinars with an initial, kick-off meeting in June 2018, an update meeting in October 2018, and a wrap-up meeting in February 2019. The consortium meetings were coordinated and facilitated by the communications consulting firm COGITO Partners, and physically hosted at ODOT facilities in Salem, OR. The teams from ODOT, OSU, and COGITO attended the webinar meetings at the Salem location, and the consortium members attended the meetings virtually from their respective locations.

Table 3.2: GTFS-ride Project Consortium Group Members

Consortium Member	Organization Type
Conveyal	Consultant and Software Vendor
Dilax	Hardware Vendor
IBI Group	Urban Design Firm
Interline Technologies	Consultant and Software Vendor
Josephine Community Transit	Public Transit Agency
Korbato	Consultant and Software Vendor
Lane Transit District	Public Transit Agency
Massachusetts Bay Transportation Authority	Public Transit Agency
Metro Transit	Public Transit Agency
MIT Transit Lab	Academic Research Center
moovel Group	Software Vendor
The POINT	Public Transit Service (through ODOT)
Remix	Software Vendor
Rogue Valley Transportation District	Public Transit Agency
Swiftly	Software Vendor
Tillamook County Transportation District	Public Transit Agency
TransitCenter	Transit Advocacy Foundation
Transport Foundry	Consultant and Software Vendor
TransSight	Consultant
Trillium Solutions	Consultant
TriMet	Public Transit Agency
TripSpark Technologies	Hardware and Software Vendor
University of Chicago Urban Labs	Academic Research Center
Urban Transportation Associates (UTA)	Hardware and Software Vendor

In each of the webinar meetings, the OSU/ODOT teams conducted presentations on project elements coupled with plans for activities going forward, and the attending consortium members were invited to provide feedback and prompted to answer specific questions through audio connection or webinar chat box. In the second consortium webinar meeting, the consortium members listed in Table 3.3 were invited to participate by conducting presentations on the GTFS-ride data standard and ecosystem development from the perspective of their organization. Members from the Zephyr Foundation participated in the third consortium meeting and shared their vision for a working group whose main objective will be to further the growth and dissemination of the GTFS-ride standard under their organization.

Table 3.3: Second Consortium Meeting Member Presentations

Organization/Project	Presenter	Presentation Title
moovel	Mindy Montgomery	The Importance of Data Standards in a Multi-Modal World
Remix	Charlie Bailey	Remix Introduction and Possibilities with GTFS-ride
TIDES project	John Levin	Transit ITS Data Exchange Specification TIDES
Trillium	Aaron Antrim	Lessons from GTFS: How do we make a de facto standard?

The participation level at the first consortium webinar was relatively high as indicated by the number of attendees and the responses submitted by participants through the chat box interface. The feedback from consortium members about ongoing issues and concerns proved to be a valuable source of material to steer ongoing conversations. Following the meeting, the transcript of this feedback was coded and categorized. The general topics that emerged as outputs of this categorization process served as the foundation for a series of posts made to a Google Group discussion forum (see Section 0). The consortium members that participated in the first meeting generally praised the prior work on the GTFS-ride data standard and acknowledged the potential benefits of its wide adoption. While some consortium members did provide suggestions on potential solutions to outstanding issues and additional reference material, there were no concrete actions taken by consortium members that can be directly attributed as an outcome of the meeting.

The second consortium meeting had lower attendance than the first, but those that did attend were very involved in discussion and feedback. As shown in Table 3.3, four consortium members actively participated through both presenting and responding to questions during the webinar. Based on evidence obtained through follow up conversations and attendance to conferences, the second consortium meeting was successful in increasing the active engagement of the transit industry with the GTFS-ride data standard.

The third and final consortium group meeting was marginally better attended than the second consortium meeting, with a noticeably higher proportion of transit vendors and consultants than transit agencies. The third consortium meeting focused predominately on updating members on final project activities, discussing the feedback received at key transportation conferences about the GTFS-ride standard, and on the plan to continue the growth and dissemination of GTFS-ride after the current project phase. To address the latter point, Clint Daniels, a funding board member of the Zephyr Foundation (https://zephyrtransport.org/) participated in the meeting via audio. Mr. Daniels discussed the origin of the Zephyr Foundation and explained how Zephyr anticipates supporting a working group to work on the GTFS-ride standard. A key message from this presentation was a request for the consortium members to remain active in furthering the cause of the GTFS-ride data standard, particularly by joining the GTFS-ride Zephyr working group. Several consortium members responded to the request with pledges to participate in this project group.

The output and related documents from each of the consortium group meetings (e.g., meeting recordings, chat transcripts, presentation slides, etc.) can be accessed at https://www.gtfs-ride.org/consortium/meetings.html. While there is anecdotal evidence that the first consortium group meeting helped in creating a broader awareness and understanding in the public transit industry about the benefits and challenges in the dissemination and possible adoption of the GTFS-ride data standard, there were no consortium member actions attributable to the second group meeting. It is hoped that interactive discussions about implementing and improving the GTFS-ride data standard are continuing on an individual basis among members of the consortium group.

3.3   GTFS-RIDE REPOSITORY AND DISCUSSION FORUM ACTIVITY

During the time period between consortium meetings, several additional management activities were undertaken to carry over the discussions held during the consortium group webinar meetings to other various digital platforms. The role of facilitating the official management of the GTFS-ride data standard established in project SPR 803 was also carried over as an active task of this project phase. Section 3.3.1 discusses the formal and informal activities conducted to manage the GTFS-ride data standard on its official GitHub repository, whereas Section 0 discusses the project tasks conducted to manage an updated version of a Google Group discussion forum.

3.3.1   Management of GTFS-ride GitHub Repository

The official GTFS-ride management tasks undertaken in this project associated with the GitHub repository included:

- Responding to initiated pull requests,

- Interacting with users on opened issues, and

- Conducting quarterly reviews of updates to GTFS and approved changes to the GTFS-ride data standard.

All of the above tasks were documented as part of the history of activities stored in the GTFS-ride GitHub repository accessible via https://github.com/ODOT-PTS/GTFS-ride. While these tasks constituted a small portion of the engagement and dissemination efforts, they served an important role in maintaining the evolution of the GTFS-ride data standard and in establishing continuity from the prior project phase.

There were no newly initiated pull requests asking for a change to the GTFS-ride data standard, and only one pull request remained open over the course of this project. The open pull request (https://github.com/ODOT-PTS/GTFS-ride/pull/11) was initiated towards the end of project SPR 803, but it did not follow the official protocol for a valid change to the GTFS-ride data standard as stipulated by the change process described at https://github.com/ODOT-PTS/GTFS-ride/blob/master/CHANGES.md. The required announcement on the Google Group page was never made, and the pull request was not withdrawn by its author, despite policy reiteration and

active discussion in the pull request conversation thread. Therefore, the pull request continued in an open and unresolved status over the course of this project.

There was more GitHub activity in managing GitHub issues (https://github.com/ODOT-PTS/GTFS-ride/issues) that were opened over the course of this project. In total, twelve new issues were initiated. Six of these issues have been closed (five with no specific action needed), and six still remain open. Of the six issues that are still open, five have been assigned to a GitHub user to address with continued discussion or a pull request solution. One GTFS-ride GitHub issue remains open and unassigned due to its complex nature. This issue is related to base GTFS publishing, versioning, and the practice of using historical GTFS feeds to associate various data to the transit services provided (as GTFS-ride does). These problems have been recognized as much larger issues within the industry and, as yet, have no concrete or implemented solutions. This issue will likely remain open until a consensus solution among many industry stakeholders is reached.

The quarterly review management task was relatively uneventful over the course of this project. The only change to the GTFS standard that took place did not impact the functionality of the GTFS-ride data standard. Therefore, no updates to the GTFS-ride data standard were needed as part of the quarterly review process. A previously approved change to include the optional field load_count in the GTFS-ride file *board_alight.txt* was enacted with the January 2018 review. No other pull request or changes were enacted during the four quarterly reviews of 2018.

3.3.2   Management of GTFS-ride Google Group Forum

Early in this project, it was recognized that there was potential for confusion in the official processes that must be followed to:

1. Conduct general discussions about the GTFS-ride data standard, and

2. Seek a change in the GTFS-ride data standard.

The general process for interaction with the GTFS-ride data standard is depicted in Figure 3.1. The confusion arose from the name choice for the Google Group and from the dual-purpose nature of this discussion forum. The Google Group, previously named "GTFS-ride Changes" through project SPR 803, was intended to serve as the official announcement platform to the transit community for a suggested change to the GTFS-ride data standard and also as a general discussion forum. The creation of the GTFS-ride consortium group furthered the need for an online forum to continue the discussions started in the webinar meetings. Rather than creating a second Google Group to serve as the general discussion forum apart from the "changes" group, the chosen solution was to rename the Google Group to be simply "GTFS-ride", understanding that the full community and not just followers of a "changes" forum would want to be informed of proposed changes to the standard. The renamed GTFS-ride Google Group can be accessed via https://groups.google.com/forum/#!forum/gtfs-ride.

It was anticipated that the newly renamed Google Group would be a center of discussion and a catalyst to spark action and beneficial partnerships among consortium members following webinar meetings. Several consortium members did begin their own post on the Google Group,

but to further initiate and facilitate interactions, new topic threads were created periodically following the first consortium meeting. The subject of each new thread was developed as a result of coding and categorizing the chat transcript of the first consortium meeting, whereas the prompting questions in each initial thread post were themes brought up as concerns or opportunities by consortium members. In total, eight different discussion threads were initiated on the following topics:

- Commercial Interests in GTFS-ride

- Data Quality in GTFS-ride

- GTFS-ride Adoption Challenges

- Logistics and Technical Challenges

- Analysis and Planning Opportunities

- Privacy

- Interoperability and Standardization

- Sharing and Reporting

The overall response to the posts to the Google Group was far lower than expected. Several postings received no response, and the others received only minimal response (one or two outside comments). Although Google does offer those with a management roll the ability to see the number of web views each posting receives, most members of the GTFS-ride Google Group opted to receive the postings as a direct email. This means that the numbers showing the count of web views is likely not a meaningful way of gauging the visibility of each post to group members. It may also be the case that many members were not aware of the capability to reply to a posting by directly replying to the email notification of the post. These factors may have contributed to the appearance of low involvement.

Even with the low apparent involvement, several consortium members urged the continuation of these topical GTFS-ride group posts during the second webinar meeting, in hopes that involvement would increase in the future. The postings to the Google Group and the responses received to date can be seen in Appendix C.
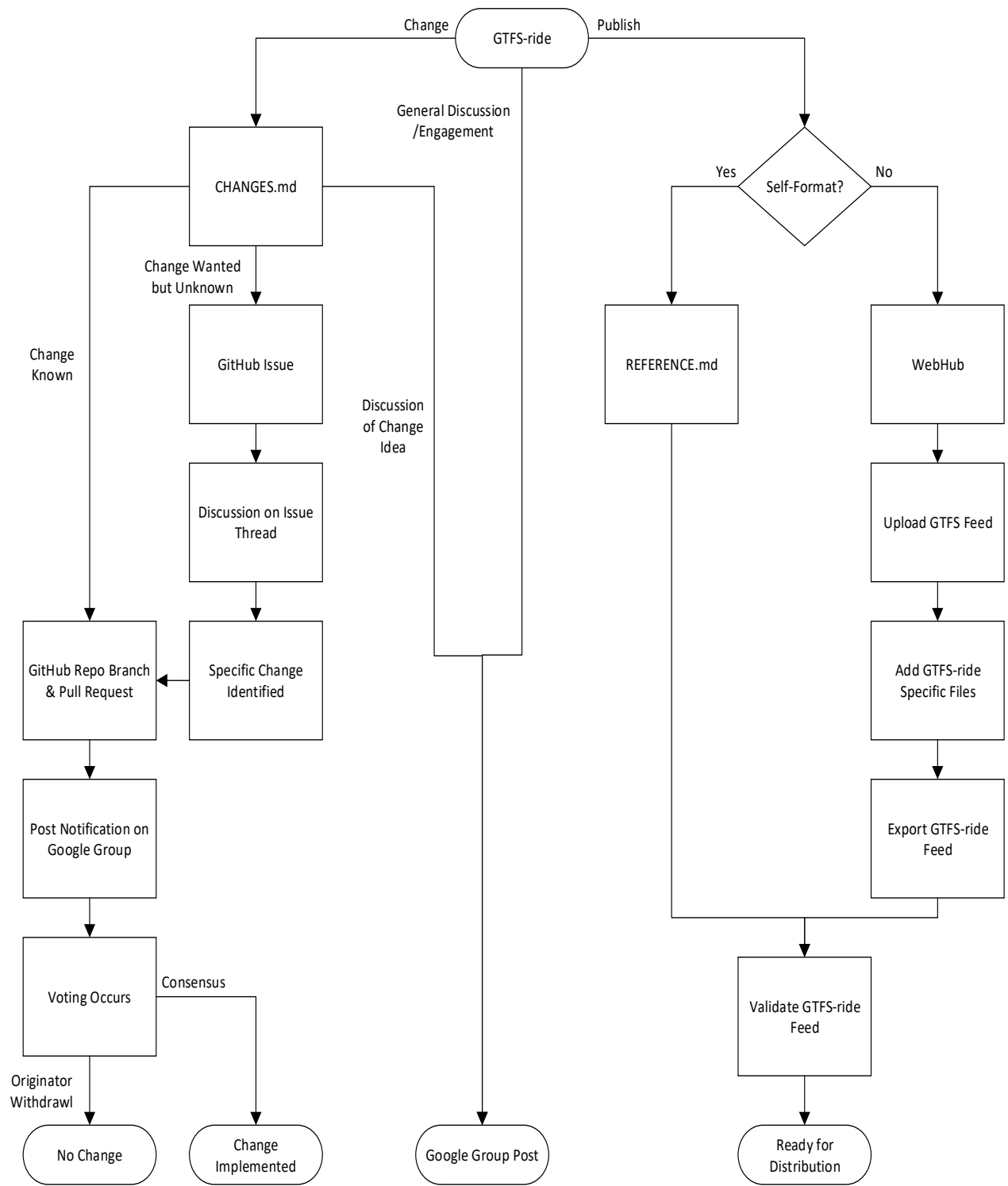
Figure 3.1: User interaction with GTFS-ride data standard

4.0   IMPROVEMENTS TO OPEN SOURCE SOFTWARE TOOLS

The website WebHub (http://webhub.gtfs-ride.org/), originally built in project SPR 803, houses the major components of the open source software tools developed to support the GTFS-ride data standard. WebHub is a centralized suite of web-based tools that allows a user to interact with their ridership data by uploading (or creating) a valid GTFS-ride feed to a cloud-base relational database. The uploaded GTFS-ride compliant data can then be used to create reports and visualizations.

The main objective of task #4 in this project was to improve and further develop the open source software tools housed in WebHub. However, the knowledge gained in the early stages of this project through the engagement and discussions with transit agencies and software vendors revealed that the software development efforts should be refocused towards a more demonstrative role in the support of the dissemination and promotion efforts of the GTFS-ride data standard.

This chapter documents the additional development and improvements made to WebHub. Section 4.1 describes the redesign of the cloud database. Section 4.2 describes the full redesign of the WebHub interface and website, whereas Section 0 documents the improvements made to the querying and visualization capabilities of WebHub. Finally, Section 0 outlines the change made to the GTFS-ride validator tool.

4.1   DATABASE REDESIGN AND IMPROVEMENT

The version of WebHub developed in project SPR 803 created a solid foundation to support GTFS-ride. However, there were several unresolved functionality issues that could be traced back to the relational database used to store the GTFS-ride feed uploads, including:

- Slow processing and uploading

- Error prone uploading

- Incorrect handling of multi-agency GTFS feeds

- Large, complex, and superfluous libraries and tables

- Inability to hold more than one feeds from a single transit agency simultaneously

- Poor support documentation

- Time consuming development, testing, and debugging

The development processes, database schema, programming structures, and programming environments employed to construct the original version of WebHub were chosen to speed up the developments process and to leverage the expertise gained in a previous project sponsored by ODOT (Porter, et al., 2017). While these choices did enable a relatively rapid development of the original version of WebHub, they also unintentionally introduced several issues. Therefore, the decision was made to move completely away from the Java-based development, Maven project structure, and OneBusAway database schema used in project SPR 803.

The use of PostgreSQL, administered through pgAdmin (https://www.pgadmin.org/), was maintained in this phase of the project. However, a completely redesigned database schema (depicted in Figure 4.1) was created to align more closely with the needs of the GTFS-ride data standard. The database schema used in project SPR 803, which is based on the OneBusAway project (https://onebusaway.org/), was not very conducive to the historical nature of the GTFS-ride data standard because OneBusAway was designed to accommodate predominately real-time transit information.

Figure 4.1 shows that a custom table titled "feeds" was added to the redesigned database schema to enable sequential feed enumeration as a method of versioning, and to allow storage of multiple historical GTFS-ride feeds from the same transit agency. The table "feeds" also facilitates the querying of data across multiple contemporaneous versions of a transit agency's base GTFS feed. It is important to note that the data field `population` in table "feeds" allows for the possibility of creating user-defined transit service population metrics not included in GTFS or GTFS-ride data for per capita calculations and visualizations.

The communications and controls of WebHub's relational database were also redesigned using the JavaScript-based, run-time environment Node.js (https://nodejs.org/en/) as opposed to the Java-based, project environment Apache Maven (https://maven.apache.org/). The MS Visual Studio Code (https://code.visualstudio.com/) integrated development environment (IDE) was used in this phase of the project instead of the Eclipse IDE (https://www.eclipse.org/ide/) used in project SPR 803. The new programming environment allowed for quicker development and debugging because there is more documentation and support for JavaScript-based methods due to their widespread use in modern web development. The new programming environment also facilitated the resolution of many of the previously unresolved GTFS-ride feed upload issues (e.g., unexplained errors and slow processing).
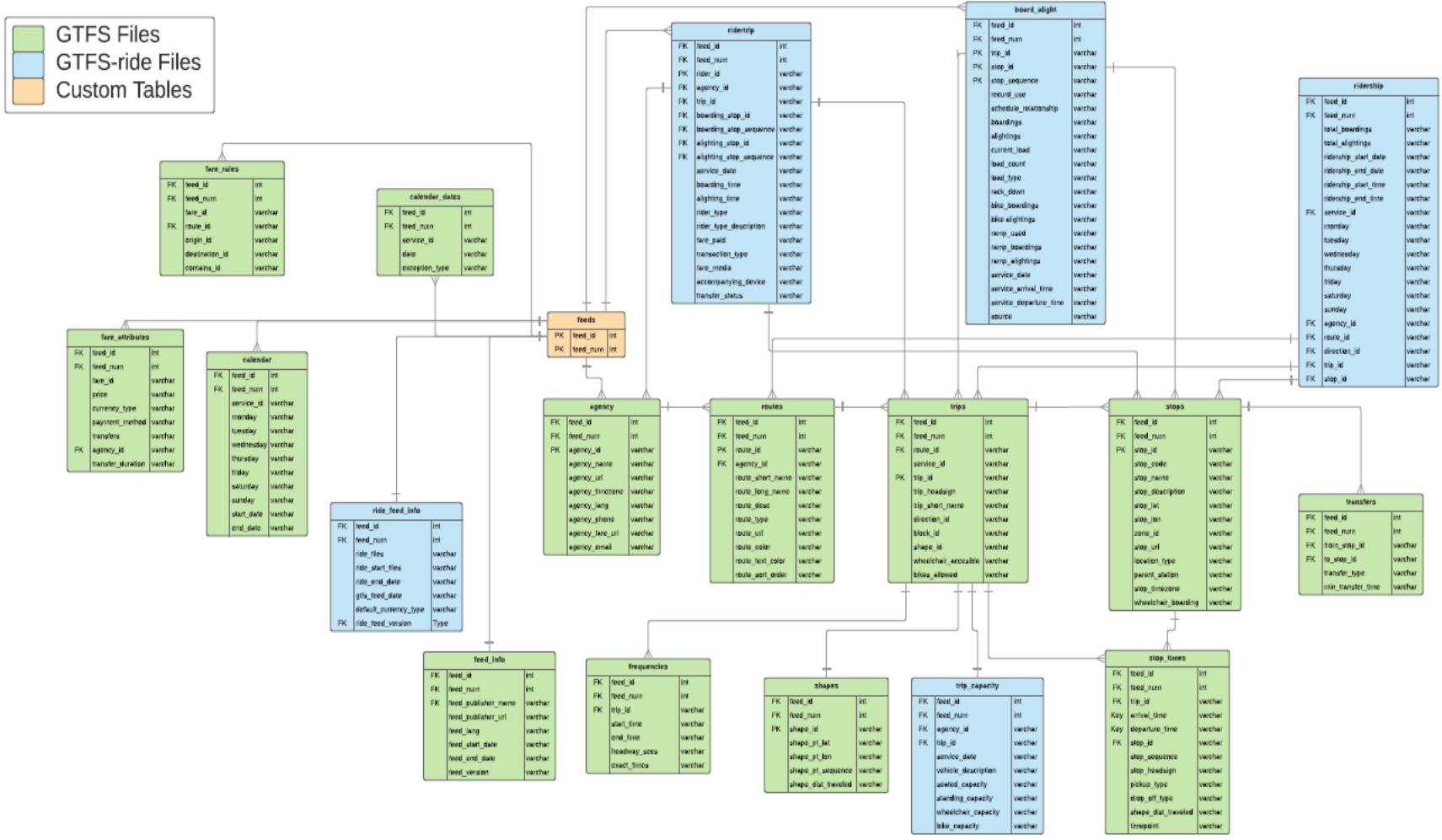
Figure 4.1: Redesigned GTFS-ride relational database schema

4.2 WEBHUB AND GTFS-RIDE WEBSITE REDESIGN

Several elements of WebHub's website were also redesigned to support the promotion and dissemination efforts of the GTFS-ride data standard, including:

- The aesthetic and layout of several web pages,

- The addition of user account security features to validate GTFS-ride feed uploads to the database,

- The form and function of the user interfaces (UIs) across the website, and

- The depiction of and interaction with report and visualization outputs.

Since it was determined early on in this research project that the suite of software support tools was to serve primarily as a demonstrative tool to aid in the promotion of the GTFS-ride data standard, there was a need to update WebHub's website to present a more professional appearance and to improve website navigation. The various pages of the website were coded with cascading style sheets (CSS) and hypertext markup language (HTML) with updated logos and palettes to match the aesthetic established on the main GTFS-ride website (www.GTFS-ride.org) created in project SPR 803. Figure 4.2 depicts the appearance of WebHub's redesigned website. For comparison purposes, Figure 4.3 depicts the appearance of WebHub's website created in project SPR 803.
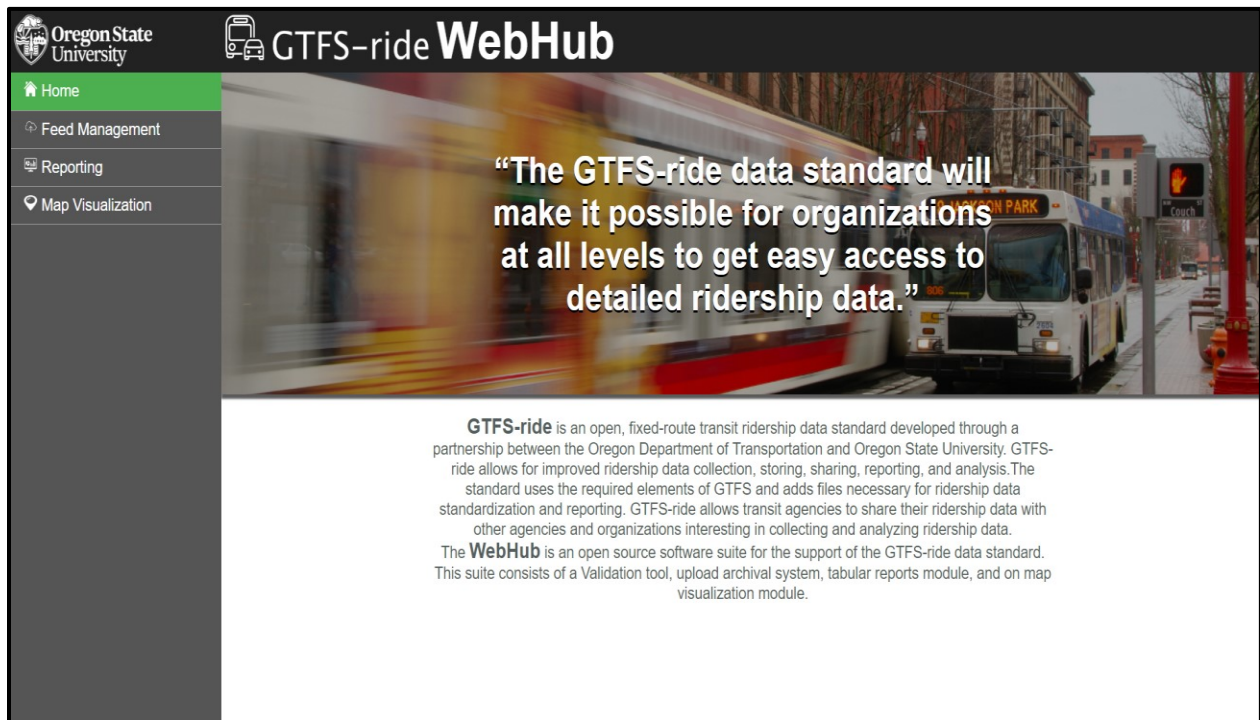
Figure 4.2: Redesigned GTFS-ride WebHub homepage
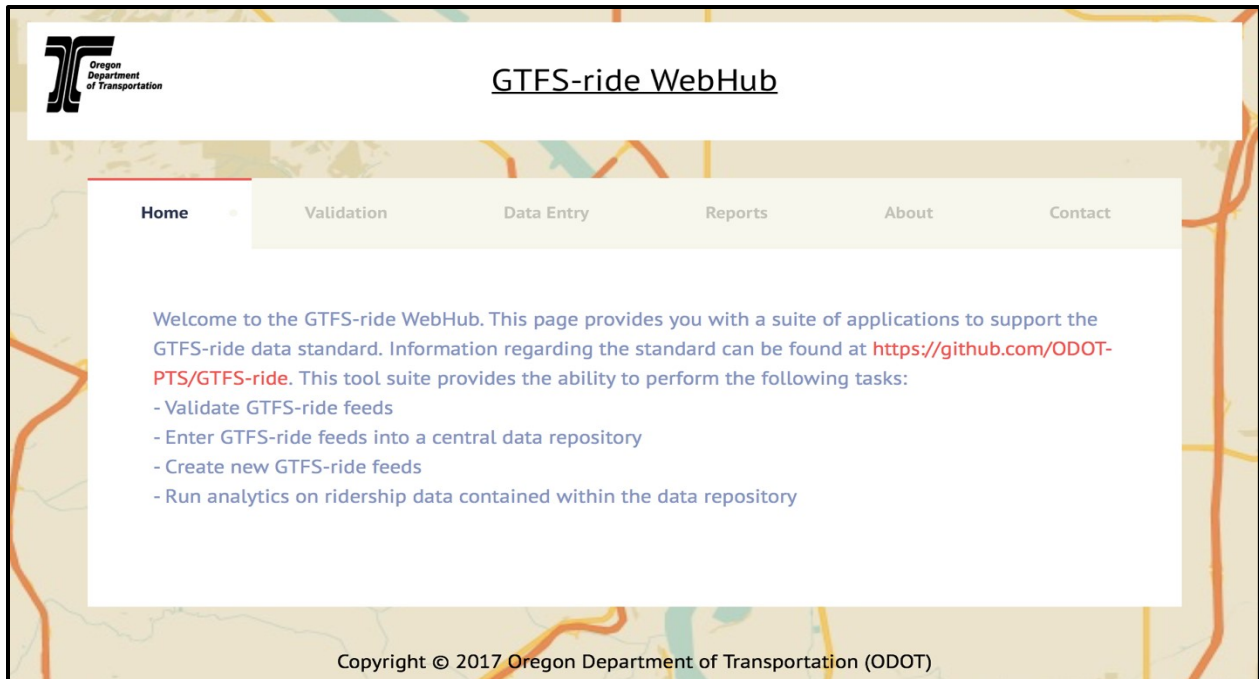
Figure 4.3: GTFS-ride WebHub homepage created in project SPR 803

Another feature added to the redesigned version of WebHub was a simple login control interface to prevent unauthorized uploads to the GTFS-ride relational database. The new login feature, depicted in Figure 4.4, restricts access to the feed management module (i.e., upload tools and feed history) to only those website users with pre-established authorization and manually created accounts and credentials.
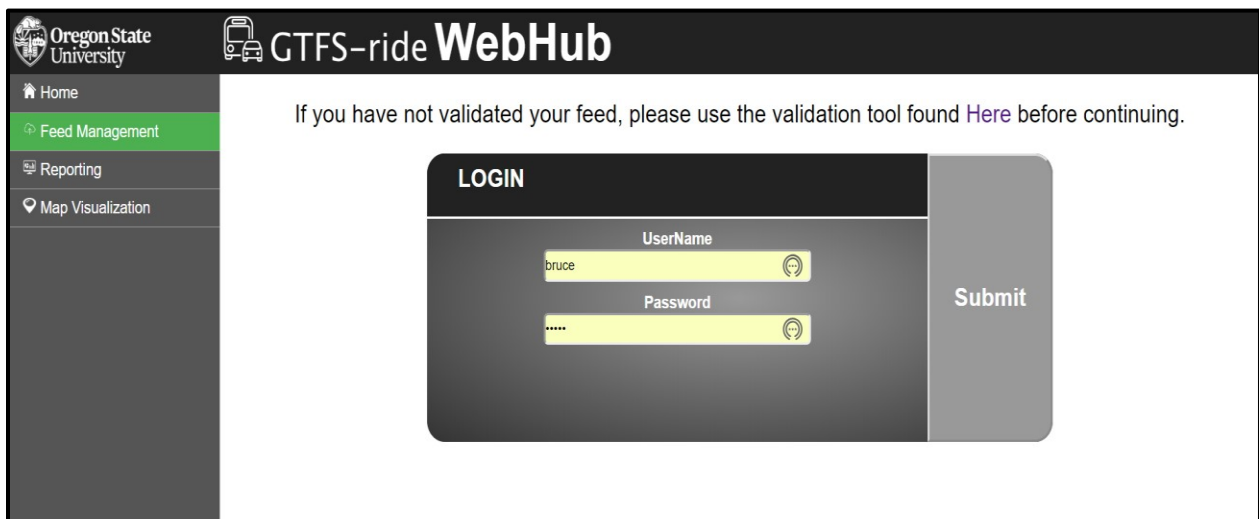
Figure 4.4: New GTFS-ride WebHub user account feature for feed uploads

The UI of the page navigation, layout, user interaction, and controls of WebHub's website were also redesigned and improved. The front-end coding and controls created with the JavaScript

programming language were maintained from the original version of WebHub. However, the React JavaScript libraries (https://reactjs.org/) were used in the development process instead of the project structure available with Apache Maven (https://maven.apache.org/). This transition was simplified by the fact that the back-end components and controls in the redesigned version of WebHub were now also coded in JavaScript, as described in Section 4.1. The React JavaScript UI libraries enabled for the development of a cleaner, more intuitive, and conspicuous WebHub UI and controls, as seen in Figure 4.5. For comparison purposes, Figure 4.6 depicts the WebHub's UI developed in project SPR 803.
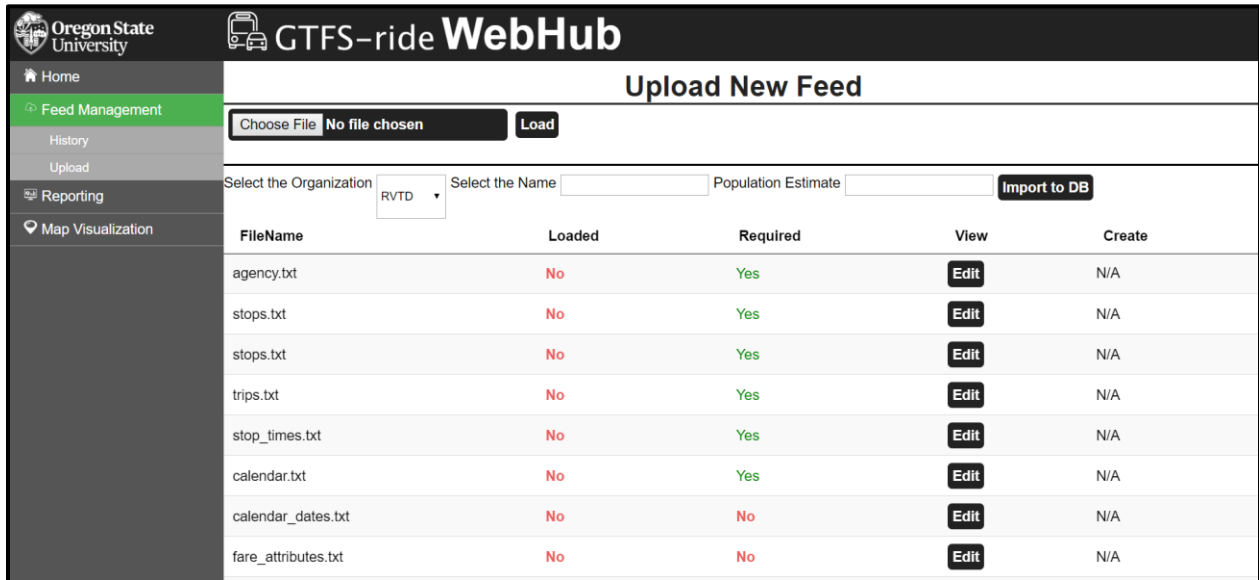
Figure 4.5: Redesigned GTFS-ride WebHub feed upload module
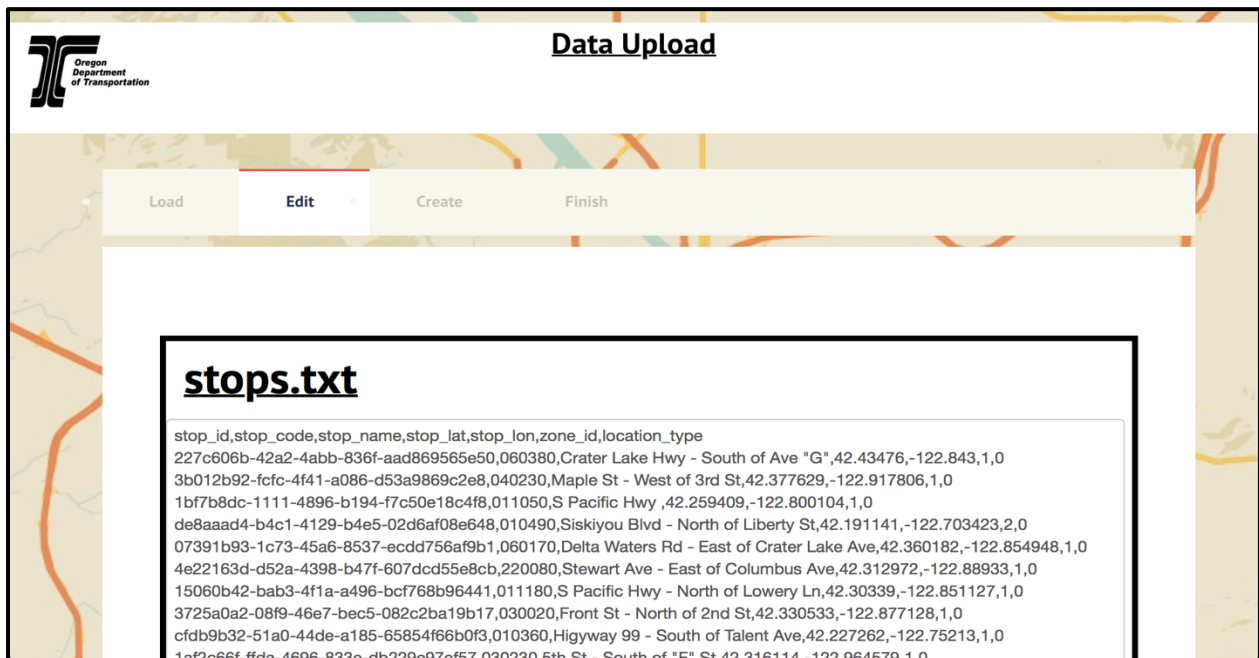
Figure 4.6: GTFS-ride WebHub feed upload module created in SPR 803 project

38

4.3   IMPROVED REPORTING AND VISUALIZATION

A major component of the efforts to improve the functionality of WebHub was the redesign of its reporting and visualization capabilities. The previous method of generating web-based reports from WebHub relied on the creation of custom SQL queries which produced a limited number of pre-specified, static reports. These reports were displayed to the website using the JavaScript library PDFMake (http://pdfmake.org/#/). The data visualizations available through the original version of WebHub consisted of simple map overlays of aggregated ridership data using the JavaScript library Leaflet (https://leafletjs.com/). The coding needed to construct the data reports and visualizations was restrictively time consuming and offered little useful output as a result.

The software Tableau Public (https://public.tableau.com/s/), which included the ability to host and share created reports and visualizations in a much more streamlined and flexible fashion, was implemented in WebHub in this phase of the project. An example of the redesigned WebHub's UI combined with visualizations through Tableau Public is depicted in Figure 4.7. Figure 4.8 depicts WebHub's UI created in project SPR 803 for comparison purposes.
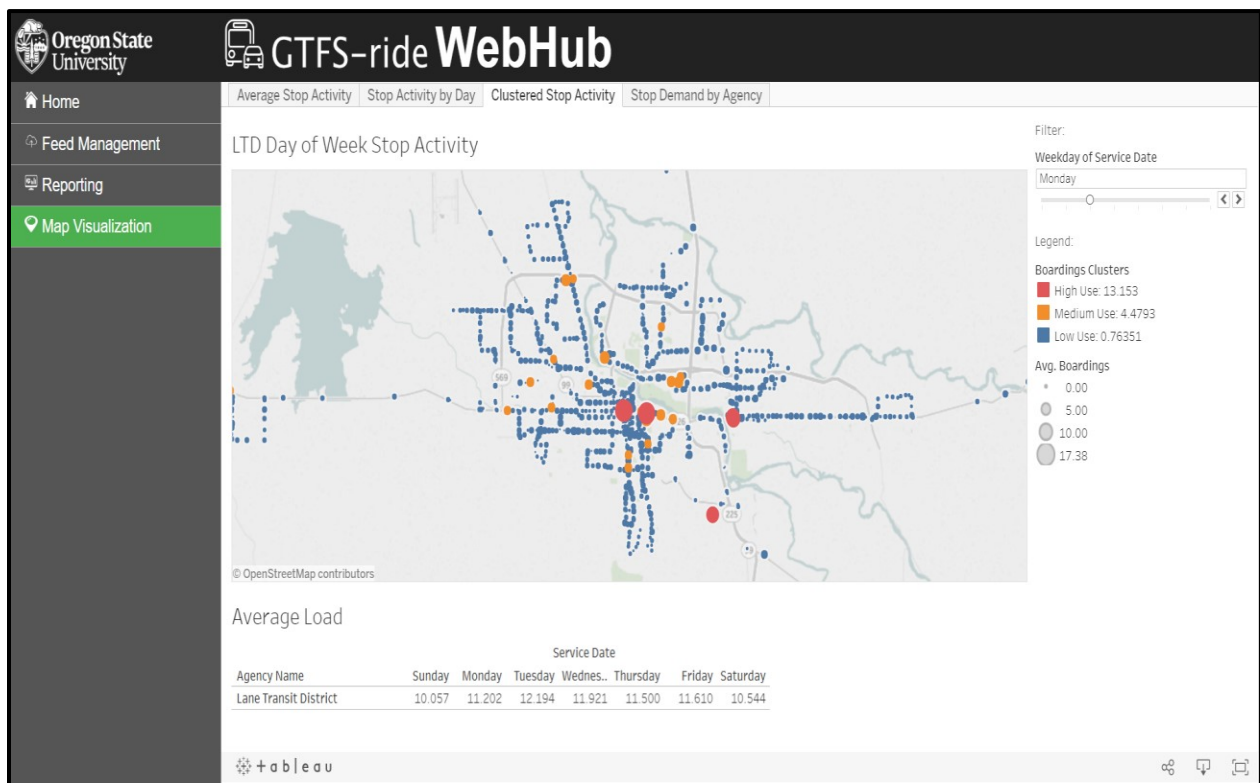
Figure 4.7: Redesigned GTFS-ride WebHub reports and visualization module
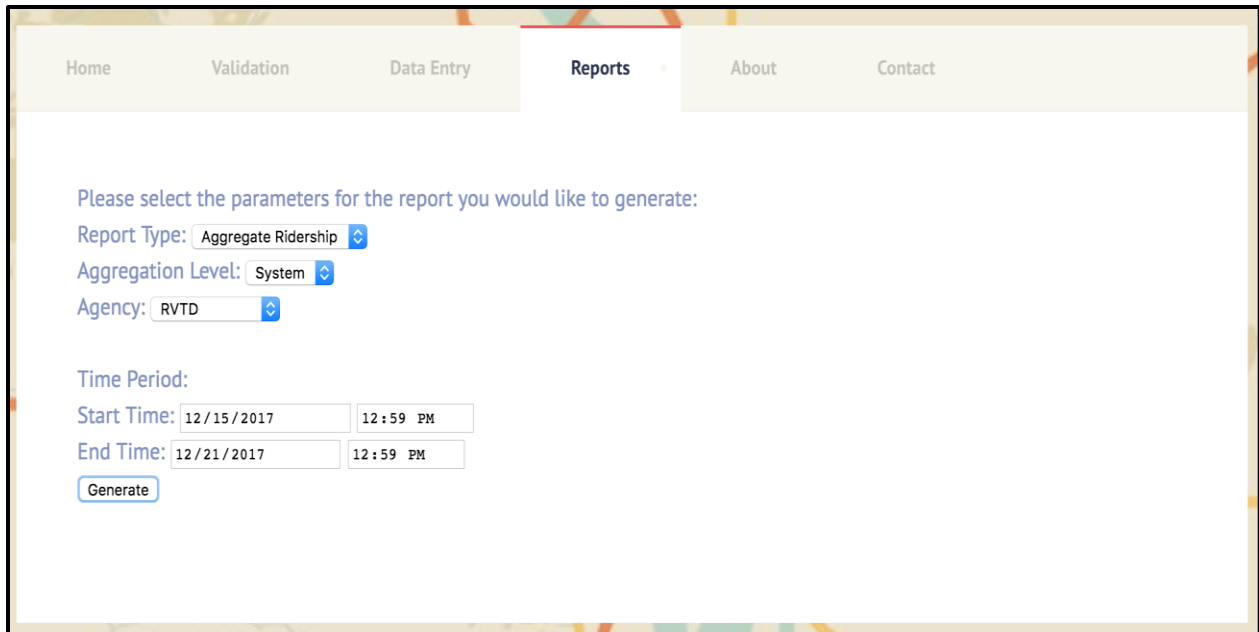
Figure 4.8: GTFS-ride WebHub reports and visualization module created in project SPR 803

The Tableau Public software seamlessly created the required table joins through a live connection to the PostgreSQL database, as depicted in Figure 4.9. Defining the data table connection structure using the desired identifiers (e.g., route_id, stop_id, etc.) allowed for the creation of a multitude of reports and visualizations relatively quickly and easily within Tableau Public because all the required database queries are generated automatically in the background when adding desired data elements to visualizations and reports. The various visualizations were then combined into a dashboard of synchronized and interactive reports, as depicted in Figure 4.7. The dashboards were then paired with a required database extract of ridership data and shared to a Tableau Public account. These Tableau Public hosted dashboards were then embedded within the React UI framework on WebHub's website.
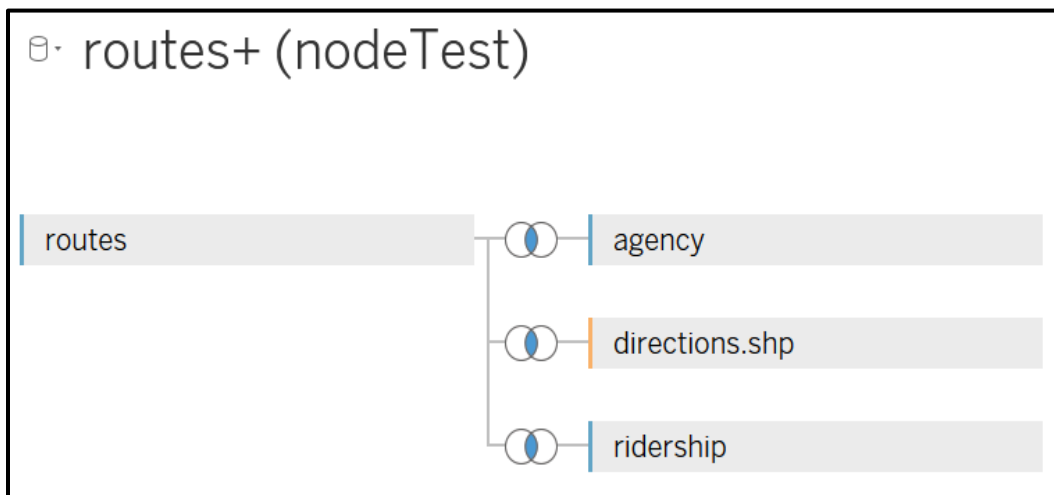
Figure 4.9: Database connection and table joining in Tableau Public

The vast increase in available reports and visualizations in this phase of the project created the issue of having an abundance of choices without a clear sense of the reports and visualizations that would be desired across transit agencies of varied sizes and levels of sophistication. Therefore, the reports and visualizations included as part of the redesigned WebHub's website were chosen to represent a range of possibilities that would aid in marketing and promoting the use of the GTFS-ride data standard. Reports were also selected based on the constraints imposed by the Tableau Public software, which limits the type of database connection (i.e., static extract vs. live) and the number of records contained in a data extract. The example dashboard depicted in Figure 4.10 was produced to be more tailored to the needs of JCT and was used as a part of the demonstration of the automated data conversion tool discussed in Section 2.4. The on-map route displayed in Figure 4.10 was created by incorporating a shape file extract of JCT's GTFS feed through the Remix software (https://www.remix.com/).
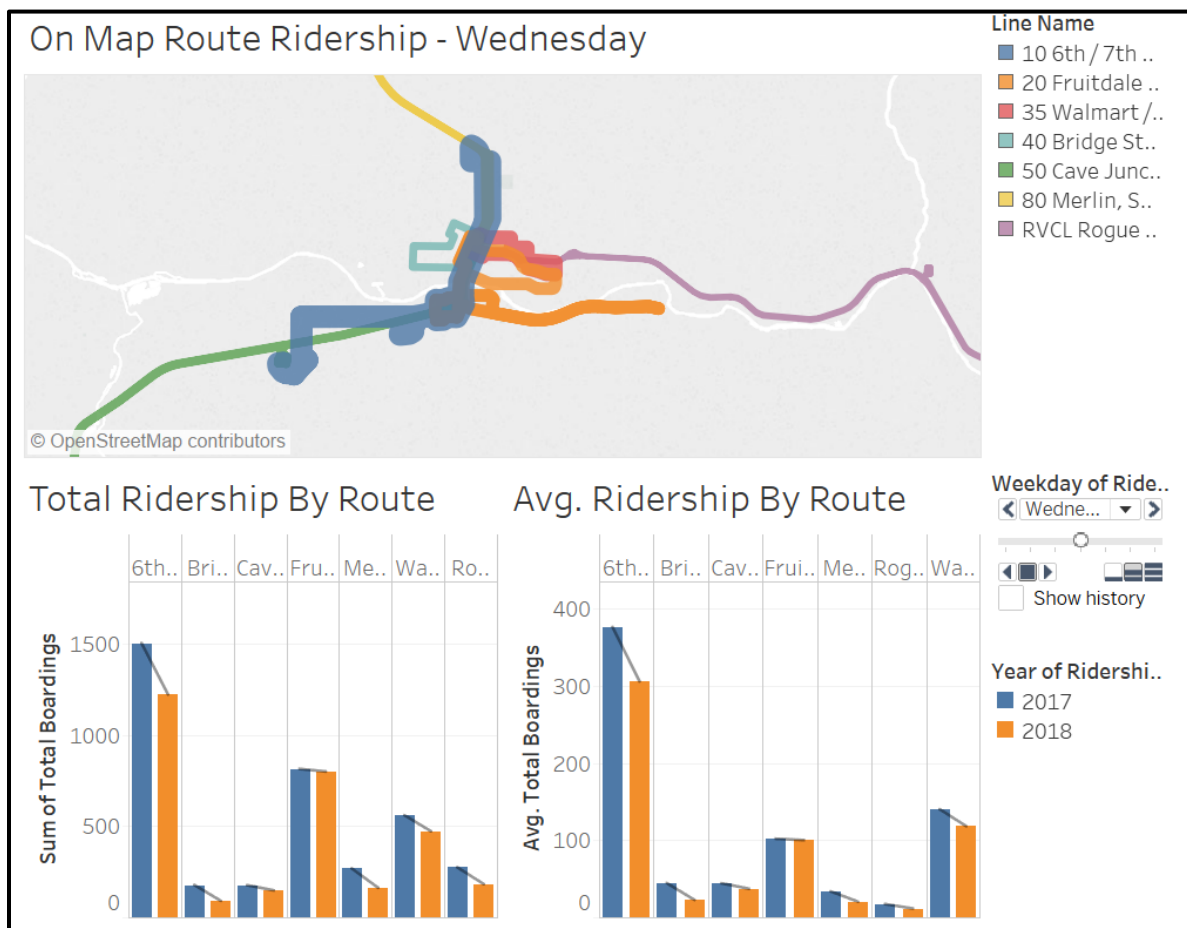
Figure 4.10: Example JCT Tableau ridership dashboard

4.4   GTFS-RIDE VALIDATION TOOL UPDATE

This section describes partially resolved and unresolved issues associated with the GTFS-ride validation tool developed in project SPR 803 (https://github.com/ODOT-PTS/transitfeed-ride). The main reason why these issues have not been addressed completely is because they would

require changes to the structure of files of the GTFS standard or to Google's GTFS feed validator tool and, therefore, are outside of the control of the OSU research team.

4.4.1   Issues Related to the GTFS standard

The GTFS-ride feed validation tool required an update to resolve an issue with the upload process of GTFS-ride feeds to the cloud database. The still unresolved issue with the upload process of GTFS-ride feeds has to do with experimental data fields published in GTFS feeds that are not an official part of the GTFS standard. The process of uploading a GTFS-ride feed to the cloud database does not execute properly when one of these unsanctioned, experimental data fields are encountered.

In this phase of the project, the GTFS-ride validation tool was updated to flag these unrecognized data columns as errors (i.e., as an indication to a user that an attempt to upload the feed may not be successful) only as a temporary fix until a more permanent solution can be implemented. It should be noted that the issue with the GTFS-ride feed upload process was addressed proactively in the GTFS-ride conversion tool (see Section 2.4) developed for JCT by removing the problem data fields as part of the automated conversion process.

4.4.2   Issues Related to Google's GTFS Feed Validation Tool

Recent changes to the GTFS data standard have not been incorporated by Google into their GTFS feed validation tool. For instance, a new field was added to the GTFS file *routes.txt*. However, the GTFS feed validator tool has not been revised to recognize this new field. As a consequence, any transit agency that (correctly) includes the new field in their GTFS feed will receive an error message when they run their GTFS feed through Google's GTFS feed validation tool.

Since the GTFS-ride feed validation tool is based on Google's GTFS feed validation tool, these errors also appear when validating GTFS-ride feeds. The OSU research team will continue to monitor the situation and will update the GTFS-ride feed validation tool as needed once Google makes the necessary updates to their GTFS feed validation tool.

5.0 CONCLUSIONS, CHALLENGES, AND OPPORTUNITIES

The main objective of the research project titled "*GTFS-ride: Transit Ridership Data Standard Ecosystem Development*" was to identify the value proposition for adoption of the GTFS-ride data standard and to develop an ecosystem of supporting software tools. The overall scope was divided into six main tasks, which were completed between January 2018 and March 2019. After the completion of the project tasks, these are the key accomplishments of the project:

- Broadened the awareness and the status of the GTFS-ride data standard within the transit community through the engagement in numerous dissemination and promotion activities.

- Developed and demonstrated custom methods for creating GTFS-ride compliant feeds from existing ridership datasets provided by Oregon transit agencies of various sizes.

- Developed and tested an automated data conversion tool for Josephine Community Transit (JCT) as a proof-of-concept and demonstration tool for other transit agencies and transit technology vendors.

- Formed a consortium of active industry stakeholders who possess the capability to further enhance the dissemination and adoption of the GTFS-ride data standard.

- Enhanced the aesthetics, capabilities, and performance of a suite of software support tools that will serve in a demonstrative role to support the adoption of the GTFS-ride data standard.

- Conducted additional management activities to carry over the discussions held during the consortium group webinar meetings to other various digital platforms (i.e., GitHub and GTFS-ride Google Discussion Group).

The following sections present the most salient conclusions, challenges, and opportunities identified in this research project.

5.1 CONCLUSIONS

The current state of practice, level of technical capabilities, and challenges related to ridership data faced by Oregon transit agencies are now better understood. The large majority of these transit agencies operate custom- and/or internally-developed systems for collecting, storing, and analyzing ridership and related data. Those that use software tools in support of these activities are typically forced to adapt their data processing practices to the software tools instead of the other way around. By "recreating the wheel" over and over, lots of time and money are wasted and a full utilization of ridership data is never achieved. It could be argued that Oregon's ridership data landscape is very representative of other states in the US.

Based on the feedback captured from the many discussions with Oregon transit agencies, hardware/software vendors, and other members of the public transit community, it is evident that the GTFS-ride data standard has the potential to solve many (if not all) of the aforementioned issues. A similar message was received from general audiences and through personal conversations with key individuals at the different academic and practitioner conferences where the GTFS-ride data standard was introduced. However, the main obstacle for its widespread adoption appears to be the lack of resources (e.g., technical experience, training, staff time, funds, upper management support, etc.) at transit agencies.

One of the main findings of this project is that Oregon transit agencies would be much more inclined to implement the GTFS-ride data standard if it could be seamlessly integrated with the software tools they currently use to collect and process ridership data. Hardware and software vendors that currently work with some of these transit agencies are now fully aware of this opportunity. Unfortunately, no one seems willing to take the first step that could result in a domino effect to change the current ridership data practices in Oregon. A possible solution for this stalling could be the infusion of state or federal funding that could motivate transit agencies toward the use of the GTFS-ride data standard and also incentivize vendors to seamlessly integrate the standard into their products.

With only slightly over a year in existence, the GTFS-ride data standard is still in its infancy. Without a doubt, additional education and dissemination efforts are needed in the future to achieve its widespread use.

5.2   CHALLENGES

There were several challenges that were identified in this project. Developing a concise value proposition for the GTFS-ride data standard proved to be a key challenge (especially for smaller, much more resource-constrained transit agencies) due to the lack of a "killer" application which resembled the success experienced by the integration of the GTFS data standard into Google's trip planner.

Another significant technical challenge involved making the correct association of ridership data to the corresponding GTFS feed. Multiple, frequently published versions of a transit agency's GTFS feed make the assumed associations ambiguous (and possibly erroneous) and have the potential to produce large and superfluous datasets due to large amounts of duplicated information. This issue is acknowledged in the open GitHub issues and, more generally, in the transit data community as existing in many areas, not just within the GTFS-ride data standard.

5.3   OPPORTUNITIES

There are several future opportunities that could further the evolution, dissemination, and adoption of the GTFS-ride data standard.

A subscription to Tableau Online would enable further enhancements to WebHub, including a live database link and the execution of more complex database queries. These enhancements would permit transit agencies to upload GTFS-ride compliant feeds to WebHub and experience real-time updates of their reports and data visualizations.

The GTFS-ride data standard could be expanded to accommodate ridership data generated by modes other than fixed-route (e.g., demand response, paratransit, commercial lines, vanpool, etc.) and from Mobility-as-a-Service (MaaS) type services such as car-share, bike-share, and ride-share, to name a few. Also, integration with other similar data standards (e.g., GTFS+, GTFS-Flex, GTFS-Vehicles, GTFS-Realtime, retro-GTFS, GBFS, etc.) could be explored.

Finally, additional funding opportunities should be pursued. This funding would enable new piloting and promotion opportunities that could bring together transit agencies, vendor organizations, and other entities that benefit from the knowledge gained from ridership data (e.g., Federal Transit Administration) to create successful use cases of the adoption of the GTFS-ride data standard.

6.0   REFERENCES

Porter, J.D., Kim, D.S., Mohseni, A., Barahimi, P., Venkatesa, S., & Prasad, V. (2021). *Phase IV: Development of Transit Network Tools, Standards, and Processes (not published)*. Salem, Oregon. Oregon Department of Transportation.

Porter, J. D., Carleton, P., Hoover, S., & Fields, B. (2018). *Statewide Data Standards to Support Current and Future Strategic Public Transit Investment* (Publication No. FHWA-OR-RD-18-13). Salem, OR: Oregon Department of Transportation. Retrieved from https://www.oregon.gov/ODOT/Programs/ResearchDocuments/SPR_803_Final Strategic Public Transit Investment.pdf .

APPENDIX A: EXCEL VBA CODE FOR JCT CONVERSION TOOL

```vbnet
'Macro to Convert ridership in JCT template "PASSENGER COUNT" into the corresponding
GTFS-ride feeds
Sub ConvertData()

'Declaring variables
Dim objFSO As Object
Dim objFolder As Object
Dim objFile As Object
Dim objSubFolders As Object
Dim objGTFSFolders As Object
Dim objSF As Object
Dim objNewFolder As Object
Dim openWB As Workbook
Dim openSheet As Worksheet
Dim activeDRstr() As String
Dim activeDRdate() As Date
Dim rawFileName As String
Dim lastModified As Date
Dim yearDay As Integer
Dim rt10(366, 2), rt20(366, 2), rt35(366, 2), rt40(366, 2), rt50(366, 2), rt80(366, 2), rvcl(366, 5)
Dim agencycol() As String: agencycol = Split("agency_id agency_name agency_url
agency_timezone agency_lang agency_phone agency_fare_url agency_email")
Dim stopscol() As String: stopscol() = Split("stop_id stop_code stop_name stop_desc stop_lat
stop_lon zone_id stop_url location_type parent_station stop_timezone wheelchair_boarding")
Dim routescol() As String: routescol() = Split("route_id agency_id route_short_name
route_long_name route_desc route_type route_url route_color route_text_color
route_sort_order")
Dim tripscol() As String: tripscol() = Split("route_id service_id trip_id trip_headsign
trip_short_name direction_id block_id shape_id wheelchair_accessible bikes_allowed")
Dim stop_timescol() As String: stop_timescol() = Split("trip_id arrival_time departure_time
stop_id stop_sequence stop_headsign pickup_type drop_off_type shape_dist_traveled
timepoint")
Dim calendarcol() As String: calendarcol() = Split("service_id monday tuesday wednesday
thursday friday saturday sunday start_date end_date")
Dim calendar_datescol() As String: calendar_datescol() = Split("service_id date
exception_type")
Dim fare_attributescol() As String: fare_attributescol() = Split("fare_id price currency_type
payment_method transfers agency_id transfer_duration")
Dim fare_rulescol() As String: fare_rulescol() = Split("fare_id route_id origin_id destination_id
contains_id")
Dim shapescol() As String: shapescol() = Split("shape_id shape_pt_lat shape_pt_lon
shape_pt_sequence shape_dist_traveled")
Dim frequenciescol() As String: frequenciescol() = Split("trip_id start_time end_time
headway_secs exact_times")
Dim transferscol() As String: transferscol() = Split("from_stop_id to_stop_id transfer_type
min_transfer_time")
```

```
Dim feed_infocol() As String: feed_infocol() = Split("feed_publisher_name feed_publisher_url
feed_lang feed_start_date feed_end_date feed_version")

'For cleaner execution
With Application
    .DisplayAlerts = False
    .ScreenUpdating = False
    .EnableEvents = False
End With

Set objFSO = CreateObject("Scripting.FileSystemObject")
Set objFolder = objFSO.GetFolder(ActiveWorkbook.Path) 'Uses directory housing the Macro
file
Set objSubFolders = objFolder.SubFolders 'Assums that GTFS downloads are unzipped and the
only subfolders in the working folder
Set objGTFSFolders = objSubFolders

'Store path and filename for JCT ridership file
For Each objFile In objFolder.Files

    If Left(objFile.Name, 15) = "PASSENGER COUNT" Then 'Assumes filenames for JCT
ridership will only be updated by changing the year portion of the filename

        rawFileName = objFile.Path
        'lastModified = objFile.DateLastModified 'This property will be changed each time the file
is opened, even if not saved. Recommend not opening the JCT template file past the date on
which the last ridership information was entered to avoid creating possibly erroneous "0" entries
in ridership.txt

    End If

Next objFile

ReDim activeDRstr(objSubFolders.Count, 3) 'Size array depending on the number of GTFS feed
folders
ReDim activeDRdate(objSubFolders.Count, 3) 'Size array depending on the number of GTFS
feed folders
lastModified = DateAdd("d", 1, Range("M12").Value) 'Initialize variable for final date of
ridership in raw data file


i = 1 'counter to index activeDateRange array

'Read in the active date ranges from each GTFS feed
For Each objSF In objSubFolders
```

```
        For Each objFile In objSF.Files

            If objFile.Name = "feed_info.txt" Then


                Call objFSO.CopyFile(objFile.Path, objSF.Path + "\feed_info_copy.csv", True) 'Create a
.csv copy of feed_info.txt to manipulate and extract date information
                Set openWB = Application.Workbooks.Open(objSF.Path + "\feed_info_copy.csv")

                'stores date range info as a string - uses match function to prevent future column addtions
from interfering - assumes that feed_info will only have 1 entry
                activeDRstr(i, 1) = openWB.Application.Range("A1").Offset(1,
Application.WorksheetFunction.Match("feed_start_date", openWB.ActiveSheet.Rows(1), 0) -
1).Value 'feed_start_date which will be compared to ridership record date
                activeDRstr(i, 2) = openWB.Application.Range("A1").Offset(1,
Application.WorksheetFunction.Match("feed_end_date", openWB.ActiveSheet.Rows(1), 0) -
1).Value 'feed_end_date which will be double checked but likely not a limiting contraint in
practice
                activeDRstr(i, 3) = objSF.Path 'to store path of associated files with active dates when
buidling the ride files later

                'converts to date data type needed for comparisons
                activeDRdate(i, 1) = DateSerial(Left(activeDRstr(i, 1), 4), Mid(activeDRstr(i, 1), 5, 2),
Right(activeDRstr(i, 1), 2))
                activeDRdate(i, 2) = DateSerial(Left(activeDRstr(i, 2), 4), Mid(activeDRstr(i, 2), 5, 2),
Right(activeDRstr(i, 2), 2))
                openWB.Close

                i = i + 1

                objFSO.DeleteFile (objSF.Path + "\feed_info_copy.csv") 'delete created .csv file to keep
orginial GTFS folder unchanged for copying files over to GTFS-ride folder later

            End If

        Next objFile

    Next objSF

    'Set actual upper bound of date range based on subsequently published GTFS start date
    'This assumes that each pushlished GTFS feed active date range superceeds any overlap with a
    previously published GTFS feed
    'This interpretation is not explicitly defined in the GTFS-ride spec
    For i = UBound(activeDRdate) To LBound(activeDRdate) Step -1

        If i = UBound(activeDRdate) Then
```

```vba
        activeDRdate(i, 3) = lastModified

    Else

        activeDRdate(i, 3) = activeDRdate(i + 1, 1)

    End If

Next i


'Read in ridership data from JCT template file

'Create a workbook object from JCT template file
Set openWB = Application.Workbooks.Open(rawFileName)

yearDay = 1

For i = 1 To 12 'Cycling through the month worksheets

    openWB.Sheets(i + 1).Activate


    For j = 1 To 31 'Reading each days date and ridership

        If IsEmpty(Range("A4").Offset(j, 0)) = False And Range("A4").Offset(j, 0) <= lastModified
Then 'Only read in data upto the day the file was last saved

            'Ranges assume the sheets and tables and fields are in fixed positions
            rt10(yearDay, 1) = Range("A4").Offset(j, 0).Value
            rt10(yearDay, 2) = Range("H4").Offset(j, 0).Value + Range("R4").Offset(j, 0).Value

            rt20(yearDay, 1) = Range("A41").Offset(j, 0).Value
            rt20(yearDay, 2) = Range("H41").Offset(j, 0).Value

            rt35(yearDay, 1) = Range("A78").Offset(j, 0).Value
            rt35(yearDay, 2) = Range("H78").Offset(j, 0).Value

            rt40(yearDay, 1) = Range("K78").Offset(j, 0).Value
            rt40(yearDay, 2) = Range("R78").Offset(j, 0).Value

            rt50(yearDay, 1) = Range("A115").Offset(j, 0).Value
            rt50(yearDay, 2) = Range("H115").Offset(j, 0).Value
```

```
            rt80(yearDay, 1) = Range("K115").Offset(j, 0).Value
            rt80(yearDay, 2) = Range("R115").Offset(j, 0).Value

            rvcl(yearDay, 1) = Range("K152").Offset(j, 0).Value
            rvcl(yearDay, 2) = Range("L152").Offset(j, 0).Value
            rvcl(yearDay, 3) = Range("M152").Offset(j, 0).Value
            rvcl(yearDay, 4) = Range("N152").Offset(j, 0).Value
            rvcl(yearDay, 5) = Range("O152").Offset(j, 0).Value

            yearDay = yearDay + 1

        End If

    Next j

Next i

openWB.Close

'Replace Empty values with 0s for rvcl
        For i = 1 To 366
          For j = 2 To 5
            If IsEmpty(rvcl(i, 1)) = False And IsEmpty(rvcl(i, j)) = True Then
                rvcl(i, j) = 0
            End If
          Next j
        Next i

'Create new folder for GTFS-ride feeds, copy GTFS files, create ridership.txt and
ride_feed_info.txt, and clean up extra columns in GTFS files
For Each objSF In objSubFolders

   'Create GTFS-ride folders
   Set objNewFolder = objFSO.CreateFolder(objFolder.Path + "\" + objSF.Name + "-ride")

   For Each objFile In objSF.Files

      'Copy over GTFS files to ride folder
      Call objFSO.CopyFile(objFile.Path, objNewFolder.Path + "\" + objFile.Name, True)

   Next objFile

   'Create ridership.txt and ride_feed_info.txt
   Call objFSO.CopyFile(objFolder.Path + "\" + "ridership_template.csv", objNewFolder.Path +
"\" + "ridership.csv", True)
```

Call objFSO.CopyFile(objFolder.Path + "\" + "ride_feed_info_template.csv", objNewFolder.Path + "\" + "ride_feed_info.csv", True)
    Set openWB = Application.Workbooks.Open(objNewFolder.Path + "\ridership.csv")
    rowcounter = 0 'initializing the row counter
    earliest = lastModified 'initializing the tracker for ealiest date of ridership to the latest date the JCT template file was opened
    latest = activeDRdate(1, 1) 'initializing the tracker for the latest date of ridership to feed_start_date date in the first read subfolder (assumes the subfolders are read in chronologically, leading to the earlierst possible date - this may require GTFS folder naming conventions to ensure ordered listing)

    '***Write needed fields in ridership.csv*****************************
    For i = 1 To 366 'Day of year

        If IsEmpty(rt10(i, 1)) = False Then 'Only for dates upto last modified date of JCT template file by checking if data in rt10 array. This can mistakenly add in 0 ridership if the file has been saved on a date after the day on which the last ridership was recorded

            For j = 1 To UBound(activeDRstr, 1) 'Loop over number of GTFS folders to store index j of correct date range

                If activeDRstr(j, 3) = objSF.Path Then 'Making sure to match of GTFS-ride folder with approrpiate date range

                    If rt10(i, 1) >= activeDRdate(j, 1) And rt10(i, 1) < activeDRdate(j, 3) And rt10(i, 1) <= activeDRdate(j, 2) Then 'Making sure date of ridership is within active range

                        If rt10(i, 1) < earliest Then earliest = rt10(i, 1) 'Storing date of earliest ridership
                        If rt10(i, 1) >= latest Then latest = rt10(i, 1) 'Storing date of earliest ridership

                        '**********Write new row for ridership on rt10 for day i*************************
                        Range("A2").Offset(rowcounter, 0).Value = rt10(i, 2) 'Write to total_boardings field
                        Range("B2").Offset(rowcounter, 0).Value = rt10(i, 2) 'Write to total_alightings field
                        Range("C2").Offset(rowcounter, 0).Value = Format(rt10(i, 1), "yyyymmdd") 'Write to ridership_start_date field
                        Range("D2").Offset(rowcounter, 0).Value = Format(rt10(i, 1), "yyyymmdd") 'Write to ridership_end_date field
                        Range("O2").Offset(rowcounter, 0).Value = Str(94) 'Write to agency_id field, 94 = Josephine Community Transit
                        Range("P2").Offset(rowcounter, 0).Value = Str(439) 'Write to route_id field, 439 = Route 10: 6th/7th St - RCC

rowcounter = rowcounter + 1 'advance counter to write next ridership entry in next row

'**
******

End If

If rt20(i, 1) >= activeDRdate(j, 1) And rt20(i, 1) < activeDRdate(j, 3) And rt20(i, 1) <= activeDRdate(j, 2) Then 'Making sure date of ridership is within active range

If rt20(i, 1) < earliest Then earliest = rt20(i, 1) 'Storing date of earliest ridership
If rt20(i, 1) >= latest Then latest = rt20(i, 1) 'Storing date of earliest ridership

'**********Write new row for ridership on rt20 for day
¡***********************
Range("A2").Offset(rowcounter, 0).Value = rt20(i, 2) 'Write to total_boardings field
Range("B2").Offset(rowcounter, 0).Value = rt20(i, 2) 'Write to total_alightings field
Range("C2").Offset(rowcounter, 0).Value = Format(rt20(i, 1), "yyyymmdd") 'Write to ridership_start_date field
Range("D2").Offset(rowcounter, 0).Value = Format(rt20(i, 1), "yyyymmdd") 'Write to ridership_end_date field
Range("O2").Offset(rowcounter, 0).Value = Str(94) 'Write to agency_id field, 94 = Josephine Community Transit
Range("P2").Offset(rowcounter, 0).Value = Str(440) 'Write to route_id field, 440 = Route 20: Fruitdale Ave

rowcounter = rowcounter + 1 'advance counter to write next ridership entry in next row

'**
******

End If

If rt35(i, 1) >= activeDRdate(j, 1) And rt35(i, 1) < activeDRdate(j, 3) And rt35(i, 1) <= activeDRdate(j, 2) Then 'Making sure date of ridership is within active range

If rt35(i, 1) < earliest Then earliest = rt35(i, 1) 'Storing date of earliest ridership
If rt35(i, 1) >= latest Then latest = rt35(i, 1) 'Storing date of earliest ridership

'**********Write new row for ridership on rt35 for day
¡***********************

A-7

```vba
                Range("A2").Offset(rowcounter, 0).Value = rt35(i, 2) 'Write to total_boardings
field
                Range("B2").Offset(rowcounter, 0).Value = rt35(i, 2) 'Write to total_alightings
field
                Range("C2").Offset(rowcounter, 0).Value = Format(rt35(i, 1), "yyyymmdd")
'Write to ridership_start_date field
                Range("D2").Offset(rowcounter, 0).Value = Format(rt35(i, 1), "yyyymmdd")
'Write to ridership_end_date field
                Range("O2").Offset(rowcounter, 0).Value = Str(94) 'Write to agency_id field, 94
= Josephine Community Transit
                Range("P2").Offset(rowcounter, 0).Value = Str(633) 'Write to route_id field, 633
= Route 35: Walmart / GPHS

                rowcounter = rowcounter + 1 'advance counter to write next ridership entry in
next row

'**************************************************************************
******

        End If

            If rt40(i, 1) >= activeDRdate(j, 1) And rt40(i, 1) < activeDRdate(j, 3) And rt40(i, 1)
<= activeDRdate(j, 2) Then 'Making sure date of ridership is within active range

                If rt40(i, 1) < earliest Then earliest = rt40(i, 1) 'Storing date of earliest ridership
                If rt40(i, 1) >= latest Then latest = rt40(i, 1) 'Storing date of earliest ridership

                '**********Write new row for ridership on rt40 for day
¡************************
                Range("A2").Offset(rowcounter, 0).Value = rt40(i, 2) 'Write to total_boardings
field
                Range("B2").Offset(rowcounter, 0).Value = rt40(i, 2) 'Write to total_alightings
field
                Range("C2").Offset(rowcounter, 0).Value = Format(rt40(i, 1), "yyyymmdd")
'Write to ridership_start_date field
                Range("D2").Offset(rowcounter, 0).Value = Format(rt40(i, 1), "yyyymmdd")
'Write to ridership_end_date field
                Range("O2").Offset(rowcounter, 0).Value = Str(94) 'Write to agency_id field, 94
= Josephine Community Transit
                Range("P2").Offset(rowcounter, 0).Value = Str(441) 'Write to route_id field, 633
= Route 40: Bridge Street

                rowcounter = rowcounter + 1 'advance counter to write next ridership entry in
next row
```

'**
******

        End If

        If rt50(i, 1) >= activeDRdate(j, 1) And rt50(i, 1) < activeDRdate(j, 3) And rt50(i, 1) <= activeDRdate(j, 2) Then 'Making sure date of ridership is within active range

        If rt50(i, 1) < earliest Then earliest = rt50(i, 1) 'Storing date of earliest ridership
        If rt50(i, 1) >= latest Then latest = rt50(i, 1) 'Storing date of earliest ridership

        '**********Write new row for ridership on rt50 for day
¡************************
        Range("A2").Offset(rowcounter, 0).Value = rt50(i, 2) 'Write to total_boardings field
        Range("B2").Offset(rowcounter, 0).Value = rt50(i, 2) 'Write to total_alightings field
        Range("C2").Offset(rowcounter, 0).Value = Format(rt50(i, 1), "yyyymmdd") 'Write to ridership_start_date field
        Range("D2").Offset(rowcounter, 0).Value = Format(rt50(i, 1), "yyyymmdd") 'Write to ridership_end_date field
        Range("O2").Offset(rowcounter, 0).Value = Str(94) 'Write to agency_id field, 94 = Josephine Community Transit
        Range("P2").Offset(rowcounter, 0).Value = Str(442) 'Write to route_id field, 633 = Route 50: Cave Junction

        rowcounter = rowcounter + 1 'advance counter to write next ridership entry in next row

'**
******

        End If

        If rt80(i, 1) >= activeDRdate(j, 1) And rt80(i, 1) < activeDRdate(j, 3) And rt80(i, 1) <= activeDRdate(j, 2) Then 'Making sure date of ridership is within active range

        If rt80(i, 1) < earliest Then earliest = rt80(i, 1) 'Storing date of earliest ridership
        If rt80(i, 1) >= latest Then latest = rt80(i, 1) 'Storing date of earliest ridership

        '**********Write new row for ridership on rt80 for day
¡************************
        Range("A2").Offset(rowcounter, 0).Value = rt80(i, 2) 'Write to total_boardings field

```vba
                Range("B2").Offset(rowcounter, 0).Value = rt80(i, 2) 'Write to total_alightings
field
                Range("C2").Offset(rowcounter, 0).Value = Format(rt80(i, 1), "yyyymmdd")
'Write to ridership_start_date field
                Range("D2").Offset(rowcounter, 0).Value = Format(rt80(i, 1), "yyyymmdd")
'Write to ridership_end_date field
                Range("O2").Offset(rowcounter, 0).Value = Str(94) 'Write to agency_id field, 94
= Josephine Community Transit
                Range("P2").Offset(rowcounter, 0).Value = Str(443) 'Write to route_id field, 633
= Route 80: Merlin, Sunny Valley and Wolf Creek

                rowcounter = rowcounter + 1 'advance counter to write next ridership entry in
next row

'************************************************************************
******

        End If

        If rvcl(i, 1) >= activeDRdate(j, 1) And rvcl(i, 1) < activeDRdate(j, 3) And rvcl(i, 1)
<= activeDRdate(j, 2) Then 'Making sure date of ridership is within active range

                If rvcl(i, 1) < earliest Then earliest = rvcl(i, 1) 'Storing date of earliest ridership
                If rvcl(i, 1) >= latest Then latest = rvcl(i, 1) 'Storing date of earliest ridership

                '**********Write new row for ridership on rvcl at Grants Pass stop for day
i*************************
                Range("A2").Offset(rowcounter, 0).Value = rvcl(i, 2) 'Write to total_boardings
field for GP stop (note this stop_id recently changed, reflected below)
                Range("C2").Offset(rowcounter, 0).Value = Format(rvcl(i, 1), "yyyymmdd")
'Write to ridership_start_date field
                Range("D2").Offset(rowcounter, 0).Value = Format(rvcl(i, 1), "yyyymmdd")
'Write to ridership_end_date field
                Range("O2").Offset(rowcounter, 0).Value = Str(225) 'Write to agency_id field,
225 = Rogue Valley Commuter Line
                Range("P2").Offset(rowcounter, 0).Value = Str(1380) 'Write to route_id field,
1380 = Rogue Valley Commuter Line
                Range("S2").Offset(rowcounter, 0).Value = Str(13837) 'Write to stop_id field,
13837 = Anne Basker Auditorium 500 NW 6th St (note this stop_id is hard coded in and will not
update if GTFS is updated)

                rowcounter = rowcounter + 1 'advance counter to write next ridership entry in
next row

'************************************************************************
******
```

```vba
'**********Write new row for ridership on rvcl at Rogue River stop for day
i**************************
            Range("A2").Offset(rowcounter, 0).Value = rvcl(i, 3) 'Write to total_boardings
field for RR stop
            Range("C2").Offset(rowcounter, 0).Value = Format(rvcl(i, 1), "yyyymmdd")
'Write to ridership_start_date field
            Range("D2").Offset(rowcounter, 0).Value = Format(rvcl(i, 1), "yyyymmdd")
'Write to ridership_end_date field
            Range("O2").Offset(rowcounter, 0).Value = Str(225) 'Write to agency_id field,
225 = Rogue Valley Commuter Line
            Range("P2").Offset(rowcounter, 0).Value = Str(1380) 'Write to route_id field,
1380 = Rogue Valley Commuter Line
            Range("S2").Offset(rowcounter, 0).Value = Str(770062) 'Write to stop_id field,
770062 = Rogue River (note this stop_id is hard coded in and will not update if GTFS is
updated)

            rowcounter = rowcounter + 1 'advance counter to write next ridership entry in
next row

'*******************************************************************************
******

'**********Write new row for ridership on rvcl at Gold Hill stop for day
i**************************
            Range("A2").Offset(rowcounter, 0).Value = rvcl(i, 4) 'Write to total_boardings
field for GH stop
            Range("C2").Offset(rowcounter, 0).Value = Format(rvcl(i, 1), "yyyymmdd")
'Write to ridership_start_date field
            Range("D2").Offset(rowcounter, 0).Value = Format(rvcl(i, 1), "yyyymmdd")
'Write to ridership_end_date field
            Range("O2").Offset(rowcounter, 0).Value = Str(225) 'Write to agency_id field,
225 = Rogue Valley Commuter Line
            Range("P2").Offset(rowcounter, 0).Value = Str(1380) 'Write to route_id field,
1380 = Rogue Valley Commuter Line
            Range("S2").Offset(rowcounter, 0).Value = Str(770063) 'Write to stop_id field,
770063 = Gold Hill - Medford (note this stop_id is hard coded in and will not update if GTFS is
updated)

            rowcounter = rowcounter + 1 'advance counter to write next ridership entry in
next row

'*******************************************************************************
******
```

```vba
            '**********Write new row for ridership on rvcl at Medford stop for day
i**************************
                Range("A2").Offset(rowcounter, 0).Value = rvcl(i, 5) 'Write to total_boardings
field for RVTD stop
                Range("C2").Offset(rowcounter, 0).Value = Format(rvcl(i, 1), "yyyymmdd")
'Write to ridership_start_date field
                Range("D2").Offset(rowcounter, 0).Value = Format(rvcl(i, 1), "yyyymmdd")
'Write to ridership_end_date field
                Range("O2").Offset(rowcounter, 0).Value = Str(225) 'Write to agency_id field,
225 = Rogue Valley Commuter Line
                Range("P2").Offset(rowcounter, 0).Value = Str(1380) 'Write to route_id field,
1380 = Rogue Valley Commuter Line
                Range("S2").Offset(rowcounter, 0).Value = Str(770064) 'Write to stop_id field,
770064 = Rogue Valley Commute Line (note this stop_id is hard coded in and will not update if
GTFS is updated)

                rowcounter = rowcounter + 1 'advance counter to write next ridership entry in
next row

'******************************************************************************
******

            End If

        End If

      Next j

    End If

  Next i

  rowcounter = 0 'Reset row counter

  openWB.Close (True) 'Close ridership.csv
  Call objFSO.CopyFile(objNewFolder.Path + "\" + "ridership.csv", objNewFolder.Path + "\" +
"ridership.txt", True) 'Convert ridership.csv to ridership.txt
  Call objFSO.DeleteFile(objNewFolder.Path + "\" + "ridership.csv") 'Delete ridership.csv

'******************************************************************************
*********************************************

  '***********************Writing needed fields in
ride_feed_info.txt****************************************************
  Set openWB = Application.Workbooks.Open(objNewFolder.Path + "\" + "ride_feed_info.csv")
```

```
    Range("A2").Value = Str(2) 'To indicate only ridership.txt is provided file containing ridership
data in ride_files field
    Range("B2").Value = Format(earliest, "yyyymmdd") 'Writing ride_start_date field
    Range("c2").Value = Format(latest, "yyyymmdd") 'Writing ride_end_date field

    openWB.Close (True) 'Close ride_feed_info.csv
    Call objFSO.CopyFile(objNewFolder.Path + "\" + "ride_feed_info.csv", objNewFolder.Path +
"\" + "ride_feed_info.txt", True) 'Convert ridership.csv to ridership.txt
    Call objFSO.DeleteFile(objNewFolder.Path + "\" + "ride_feed_info.csv") 'Delete
ride_feed_info.csv

'**************************************************************************
*******************************************

'****************************Removing non-standard GTFS
columns**************************************************

'Checks only standard GTFS files as defined on 09/28/2018. Ignores additional files
For Each objFile In objNewFolder.Files

    '********************************Checking agency.txt
columns************************************
    If objFile.Name = "agency.txt" Then

        Call objFSO.CopyFile(objFile.Path, objNewFolder.Path + "\agency.csv", True)
        Set openWB = Application.Workbooks.Open(objNewFolder.Path + "\agency.csv")

        '***********Check the number of columns in file***********************
        columncount = 0
        emptycolumn = False
        numDeleted = 0

        While emptycolumn = False
          If IsEmpty(Range("A1").Offset(0, columncount)) = False Then
            columncount = columncount + 1
          Else
            emptycolumn = True
          End If
        Wend

'**************************************************************************

        '***********Checking each column for match to GTFS allowed column******
        For i = 1 To columncount

          colMatch = False
```

```vba
        For j = 0 To UBound(agencycol, 1)

            If Range("A1").Offset(0, i - 1 - numDeleted).Value = agencycol(j) Then colMatch = _
True

        Next j

        If colMatch = False Then

          If numDeleted = 0 Then

              Columns(i).EntireColumn.Delete 'Deleting the first unmatched column
              numDeleted = numDeleted + 1 'counting the number of columns deleted since the
act of deleted a column shifts the column index of all subsequent columns

          Else

              Columns(i - numDeleted).EntireColumn.Delete 'Deleting any remaining
unmatched columns
              numDeleted = numDeleted + 1 'counting the number of columns deleted since the
act of deleted a column shifts the column index of all subsequent columns

          End If

        End If

    Next i

'************************************************************************

    openWB.Close (True)
    Call objFSO.CopyFile(objNewFolder.Path + "\" + "agency.csv", objNewFolder.Path + "\" _
+ "agency.txt", True)
    Call objFSO.DeleteFile(objNewFolder.Path + "\" + "agency.csv")

  End If

'*******************************Checking stops.txt
columns*********************************
  If objFile.Name = "stops.txt" Then

    Call objFSO.CopyFile(objFile.Path, objNewFolder.Path + "\stops.csv", True)
    Set openWB = Application.Workbooks.Open(objNewFolder.Path + "\stops.csv")

    '***********Check the number of columns in file***********************
```

```vba
        columncount = 0
        emptycolumn = False
        numDeleted = 0

        While emptycolumn = False
           If IsEmpty(Range("A1").Offset(0, columncount)) = False Then
              columncount = columncount + 1
           Else
              emptycolumn = True
           End If
        Wend

'*************************************************************************

        '*************Checking each column for match to GTFS allowed column*******
        For i = 1 To columncount

           colMatch = False

           For j = 0 To UBound(stopscol, 1)

             If Range("A1").Offset(0, i - 1 - numDeleted).Value = stopscol(j) Then colMatch =
True

           Next j

           If colMatch = False Then

              If numDeleted = 0 Then

                 Columns(i).EntireColumn.Delete 'Deleting the first unmatched column
                 numDeleted = numDeleted + 1 'counting the number of columns deleted since the
act of deleted a column shifts the column index of all subsequent columns

              Else

                 Columns(i - numDeleted).EntireColumn.Delete 'Deleting any remaining
unmatched columns
                 numDeleted = numDeleted + 1 'counting the number of columns deleted since the
act of deleted a column shifts the column index of all subsequent columns

              End If

           End If

        Next i
```

```
'**********************************************************************

         openWB.Close (True)
         Call objFSO.CopyFile(objNewFolder.Path + "\" + "stops.csv", objNewFolder.Path + "\" +
"stops.txt", True)
         Call objFSO.DeleteFile(objNewFolder.Path + "\" + "stops.csv")

    End If

    '*******************************Checking routes.txt
columns***********************************
    If objFile.Name = "routes.txt" Then

         Call objFSO.CopyFile(objFile.Path, objNewFolder.Path + "\routes.csv", True)
         Set openWB = Application.Workbooks.Open(objNewFolder.Path + "\routes.csv")

         '***********Check the number of columns in file************************
         columncount = 0
         emptycolumn = False
         numDeleted = 0

         While emptycolumn = False
           If IsEmpty(Range("A1").Offset(0, columncount)) = False Then
              columncount = columncount + 1
           Else
              emptycolumn = True
           End If
         Wend

'**********************************************************************

         '***********Checking each column for match to GTFS allowed column******
         For i = 1 To columncount

           colMatch = False

           For j = 0 To UBound(routescol, 1)

             If Range("A1").Offset(0, i - 1 - numDeleted).Value = routescol(j) Then colMatch =
True

           Next j

           If colMatch = False Then
```

```
        If numDeleted = 0 Then

                Columns(i).EntireColumn.Delete 'Deleting the first unmatched column
                numDeleted = numDeleted + 1 'counting the number of columns deleted since the
act of deleted a column shifts the column index of all subsequent columns

            Else

                Columns(i - numDeleted).EntireColumn.Delete 'Deleting any remaining
unmatched columns
                numDeleted = numDeleted + 1 'counting the number of columns deleted since the
act of deleted a column shifts the column index of all subsequent columns

            End If

        End If

    Next i

'******************************************************************************

    openWB.Close (True)
    Call objFSO.CopyFile(objNewFolder.Path + "\" + "routes.csv", objNewFolder.Path + "\"
+ "routes.txt", True)
    Call objFSO.DeleteFile(objNewFolder.Path + "\" + "routes.csv")

  End If

    '*********************************Checking trips.txt
columns*************************************
    If objFile.Name = "trips.txt" Then

        Call objFSO.CopyFile(objFile.Path, objNewFolder.Path + "\trips.csv", True)
        Set openWB = Application.Workbooks.Open(objNewFolder.Path + "\trips.csv")

        '***********Check the number of columns in file***********************
        columncount = 0
        emptycolumn = False
        numDeleted = 0

        While emptycolumn = False
          If IsEmpty(Range("A1").Offset(0, columncount)) = False Then
            columncount = columncount + 1
          Else
            emptycolumn = True
          End If
```

```
        Wend

'*****************************************************************************

        '************Checking each column for match to GTFS allowed column******
        For i = 1 To columncount

            colMatch = False

            For j = 0 To UBound(tripscol, 1)

                If Range("A1").Offset(0, i - 1 - numDeleted).Value = tripscol(j) Then colMatch =
True

            Next j

            If colMatch = False Then

                If numDeleted = 0 Then

                    Columns(i).EntireColumn.Delete 'Deleting the first unmatched column
                    numDeleted = numDeleted + 1 'counting the number of columns deleted since the
act of deleted a column shifts the column index of all subsequent columns

                Else

                    Columns(i - numDeleted).EntireColumn.Delete 'Deleting any remaining
unmatched columns
                    numDeleted = numDeleted + 1 'counting the number of columns deleted since the
act of deleted a column shifts the column index of all subsequent columns

                End If

            End If

        Next i

'*****************************************************************************

        openWB.Close (True)
        Call objFSO.CopyFile(objNewFolder.Path + "\" + "trips.csv", objNewFolder.Path + "\" +
"trips.txt", True)
        Call objFSO.DeleteFile(objNewFolder.Path + "\" + "trips.csv")

    End If
```

```vba
'**********************************Checking stop_times.txt
columns*********************************
    If objFile.Name = "stop_times.txt" Then

        Call objFSO.CopyFile(objFile.Path, objNewFolder.Path + "\stop_times.csv", True)
        Set openWB = Application.Workbooks.Open(objNewFolder.Path + "\stop_times.csv")

        '***********Check the number of columns in file**********************
        columncount = 0
        emptycolumn = False
        numDeleted = 0

        While emptycolumn = False
          If IsEmpty(Range("A1").Offset(0, columncount)) = False Then
             columncount = columncount + 1
          Else
             emptycolumn = True
          End If
        Wend


'*********************************************************************************

        '***********Checking each column for match to GTFS allowed column*******
        For i = 1 To columncount

           colMatch = False

           For j = 0 To UBound(stop_timescol, 1)

             If Range("A1").Offset(0, i - 1 - numDeleted).Value = stop_timescol(j) Then colMatch = True

           Next j

           If colMatch = False Then

              If numDeleted = 0 Then

                 Columns(i).EntireColumn.Delete 'Deleting the first unmatched column
                 numDeleted = numDeleted + 1 'counting the number of columns deleted since the
act of deleted a column shifts the column index of all subsequent columns

              Else

                 Columns(i - numDeleted).EntireColumn.Delete 'Deleting any remaining
unmatched columns
```

```
                    numDeleted = numDeleted + 1 'counting the number of columns deleted since the
act of deleted a column shifts the column index of all subsequent columns

                End If

            End If

        Next i

'*************************************************************************

        openWB.Close (True)
        Call objFSO.CopyFile(objNewFolder.Path + "\" + "stop_times.csv", objNewFolder.Path
+ "\" + "stop_times.txt", True)
        Call objFSO.DeleteFile(objNewFolder.Path + "\" + "stop_times.csv")

    End If

    '*********************************Checking calendar.txt
columns***********************************
    If objFile.Name = "calendar.txt" Then

        Call objFSO.CopyFile(objFile.Path, objNewFolder.Path + "\calendar.csv", True)
        Set openWB = Application.Workbooks.Open(objNewFolder.Path + "\calendar.csv")

        '************Check the number of columns in file************************
        columncount = 0
        emptycolumn = False
        numDeleted = 0

        While emptycolumn = False
            If IsEmpty(Range("A1").Offset(0, columncount)) = False Then
                columncount = columncount + 1
            Else
                emptycolumn = True
            End If
        Wend

'*************************************************************************

        '************Checking each column for match to GTFS allowed column******
        For i = 1 To columncount

            colMatch = False

            For j = 0 To UBound(calendarcol, 1)
```

```
            If Range("A1").Offset(0, i - 1 - numDeleted).Value = calendarcol(j) Then colMatch =
True

        Next j

        If colMatch = False Then

           If numDeleted = 0 Then

               Columns(i).EntireColumn.Delete 'Deleting the first unmatched column
               numDeleted = numDeleted + 1 'counting the number of columns deleted since the
act of deleted a column shifts the column index of all subsequent columns

           Else

               Columns(i - numDeleted).EntireColumn.Delete 'Deleting any remaining
unmatched columns
               numDeleted = numDeleted + 1 'counting the number of columns deleted since the
act of deleted a column shifts the column index of all subsequent columns

           End If

        End If

     Next i

'**************************************************************************


     openWB.Close (True)
     Call objFSO.CopyFile(objNewFolder.Path + "\" + "calendar.csv", objNewFolder.Path +
"\" + "calendar.txt", True)
     Call objFSO.DeleteFile(objNewFolder.Path + "\" + "calendar.csv")

   End If

   '*******************************Checking calendar_dates.txt
columns***********************************
   If objFile.Name = "calendar_dates.txt" Then

     Call objFSO.CopyFile(objFile.Path, objNewFolder.Path + "\calendar_dates.csv", True)
     Set openWB = Application.Workbooks.Open(objNewFolder.Path +
"\calendar_dates.csv")

     '***********Check the number of columns in file***********************
     columncount = 0
```

```vba
      emptycolumn = False
      numDeleted = 0

      While emptycolumn = False
        If IsEmpty(Range("A1").Offset(0, columncount)) = False Then
          columncount = columncount + 1
        Else
          emptycolumn = True
        End If
      Wend


'**********************************************************************

      '************Checking each column for match to GTFS allowed column******
      For i = 1 To columncount

        colMatch = False

        For j = 0 To UBound(calendar_datescol, 1)

          If Range("A1").Offset(0, i - 1 - numDeleted).Value = calendar_datescol(j) Then
colMatch = True

        Next j

        If colMatch = False Then

          If numDeleted = 0 Then

            Columns(i).EntireColumn.Delete 'Deleting the first unmatched column
            numDeleted = numDeleted + 1 'counting the number of columns deleted since the
act of deleted a column shifts the column index of all subsequent columns

          Else

            Columns(i - numDeleted).EntireColumn.Delete 'Deleting any remaining
unmatched columns
            numDeleted = numDeleted + 1 'counting the number of columns deleted since the
act of deleted a column shifts the column index of all subsequent columns

          End If

        End If

      Next i
```

```
'****************************************************************************

        openWB.Close (True)
        Call objFSO.CopyFile(objNewFolder.Path + "\" + "calendar_dates.csv",
objNewFolder.Path + "\" + "calendar_dates.txt", True)
        Call objFSO.DeleteFile(objNewFolder.Path + "\" + "calendar_dates.csv")

    End If

    '*******************************Checking fare_attributes.txt
columns***********************************
    If objFile.Name = "fare_attributes.txt" Then

        Call objFSO.CopyFile(objFile.Path, objNewFolder.Path + "\fare_attributes.csv", True)
        Set openWB = Application.Workbooks.Open(objNewFolder.Path + "\fare_attributes.csv")

        '************Check the number of columns in file***********************
        columncount = 0
        emptycolumn = False
        numDeleted = 0

        While emptycolumn = False
          If IsEmpty(Range("A1").Offset(0, columncount)) = False Then
             columncount = columncount + 1
          Else
             emptycolumn = True
          End If
        Wend

'****************************************************************************

        '************Checking each column for match to GTFS allowed column******
        For i = 1 To columncount

          colMatch = False

          For j = 0 To UBound(fare_attributescol, 1)

            If Range("A1").Offset(0, i - 1 - numDeleted).Value = fare_attributescol(j) Then
colMatch = True

            Next j

            If colMatch = False Then
```

```vba
            If numDeleted = 0 Then

                    Columns(i).EntireColumn.Delete 'Deleting the first unmatched column
                    numDeleted = numDeleted + 1 'counting the number of columns deleted since the
act of deleted a column shifts the column index of all subsequent columns

                Else

                    Columns(i - numDeleted).EntireColumn.Delete 'Deleting any remaining
unmatched columns
                    numDeleted = numDeleted + 1 'counting the number of columns deleted since the
act of deleted a column shifts the column index of all subsequent columns

                End If

            End If

        Next i

'**************************************************************************

        openWB.Close (True)
        Call objFSO.CopyFile(objNewFolder.Path + "\" + "fare_attributes.csv",
objNewFolder.Path + "\" + "fare_attributes.txt", True)
        Call objFSO.DeleteFile(objNewFolder.Path + "\" + "fare_attributes.csv")

    End If

    '*******************************Checking fare_rules.txt
columns************************************
    If objFile.Name = "fare_rules.txt" Then

        Call objFSO.CopyFile(objFile.Path, objNewFolder.Path + "\fare_rules.csv", True)
        Set openWB = Application.Workbooks.Open(objNewFolder.Path + "\fare_rules.csv")

        '***********Check the number of columns in file***********************
        columncount = 0
        emptycolumn = False
        numDeleted = 0

        While emptycolumn = False
          If IsEmpty(Range("A1").Offset(0, columncount)) = False Then
            columncount = columncount + 1
          Else
            emptycolumn = True
          End If
```

```
        Wend

'*************************************************************

        '************Checking each column for match to GTFS allowed column******
        For i = 1 To columncount

            colMatch = False

            For j = 0 To UBound(fare_rulescol, 1)

                If Range("A1").Offset(0, i - 1 - numDeleted).Value = fare_rulescol(j) Then colMatch = True

            Next j

            If colMatch = False Then

                If numDeleted = 0 Then

                    Columns(i).EntireColumn.Delete 'Deleting the first unmatched column
                    numDeleted = numDeleted + 1 'counting the number of columns deleted since the
act of deleted a column shifts the column index of all subsequent columns

                Else

                    Columns(i - numDeleted).EntireColumn.Delete 'Deleting any remaining
unmatched columns
                    numDeleted = numDeleted + 1 'counting the number of columns deleted since the
act of deleted a column shifts the column index of all subsequent columns

                End If

            End If

        Next i

'*************************************************************

        openWB.Close (True)
        Call objFSO.CopyFile(objNewFolder.Path + "\" + "fare_rules.csv", objNewFolder.Path +
"\" + "fare_rules.txt", True)
        Call objFSO.DeleteFile(objNewFolder.Path + "\" + "fare_rules.csv")

    End If
```

```
'*******************************Checking shapes.txt
columns****************************************
     If objFile.Name = "shapes.txt" Then

         Call objFSO.CopyFile(objFile.Path, objNewFolder.Path + "\shapes.csv", True)
         Set openWB = Application.Workbooks.Open(objNewFolder.Path + "\shapes.csv")

         '************Check the number of columns in file**********************
         columncount = 0
         emptycolumn = False
         numDeleted = 0

         While emptycolumn = False
             If IsEmpty(Range("A1").Offset(0, columncount)) = False Then
                 columncount = columncount + 1
             Else
                 emptycolumn = True
             End If
         Wend


'***********************************************************************

         '************Checking each column for match to GTFS allowed column*******
         For i = 1 To columncount

             colMatch = False

             For j = 0 To UBound(shapescol, 1)

                 If Range("A1").Offset(0, i - 1 - numDeleted).Value = shapescol(j) Then colMatch =
True

             Next j

             If colMatch = False Then

                 If numDeleted = 0 Then

                     Columns(i).EntireColumn.Delete 'Deleting the first unmatched column
                     numDeleted = numDeleted + 1 'counting the number of columns deleted since the
act of deleted a column shifts the column index of all subsequent columns

                 Else

                     Columns(i - numDeleted).EntireColumn.Delete 'Deleting any remaining
unmatched columns
```

numDeleted = numDeleted + 1 'counting the number of columns deleted since the act of deleted a column shifts the column index of all subsequent columns

      End If

     End If

    Next i

'***

   openWB.Close (True)
   Call objFSO.CopyFile(objNewFolder.Path + "\" + "shapes.csv", objNewFolder.Path + "\" + "shapes.txt", True)
   Call objFSO.DeleteFile(objNewFolder.Path + "\" + "shapes.csv")

  End If

  '*********************************Checking frequencies.txt columns***********************************
  If objFile.Name = "frequencies.txt" Then

   Call objFSO.CopyFile(objFile.Path, objNewFolder.Path + "\frequencies.csv", True)
   Set openWB = Application.Workbooks.Open(objNewFolder.Path + "\frequencies.csv")

   '************Check the number of columns in file************************
   columncount = 0
   emptycolumn = False
   numDeleted = 0

   While emptycolumn = False
    If IsEmpty(Range("A1").Offset(0, columncount)) = False Then
     columncount = columncount + 1
    Else
     emptycolumn = True
    End If
   Wend

'***

   '************Checking each column for match to GTFS allowed column******
   For i = 1 To columncount

    colMatch = False

    For j = 0 To UBound(frequenciescol, 1)

```
        If Range("A1").Offset(0, i - 1 - numDeleted).Value = frequenciescol(j) Then
colMatch = True

        Next j

        If colMatch = False Then

          If numDeleted = 0 Then

            Columns(i).EntireColumn.Delete 'Deleting the first unmatched column
            numDeleted = numDeleted + 1 'counting the number of columns deleted since the
act of deleted a column shifts the column index of all subsequent columns

          Else

            Columns(i - numDeleted).EntireColumn.Delete 'Deleting any remaining
unmatched columns
            numDeleted = numDeleted + 1 'counting the number of columns deleted since the
act of deleted a column shifts the column index of all subsequent columns

          End If

        End If

      Next i

'**************************************************************************

      openWB.Close (True)
      Call objFSO.CopyFile(objNewFolder.Path + "\" + "frequencies.csv", objNewFolder.Path
+ "\" + "frequencies.txt", True)
      Call objFSO.DeleteFile(objNewFolder.Path + "\" + "frequencies.csv")

    End If

'*********************************Checking transfers.txt
columns************************************
    If objFile.Name = "transfers.txt" Then

      Call objFSO.CopyFile(objFile.Path, objNewFolder.Path + "\transfers.csv", True)
      Set openWB = Application.Workbooks.Open(objNewFolder.Path + "\transfers.csv")

      '***********Check the number of columns in file***********************
      columncount = 0
      emptycolumn = False
```

```
        numDeleted = 0

        While emptycolumn = False
          If IsEmpty(Range("A1").Offset(0, columncount)) = False Then
            columncount = columncount + 1
          Else
            emptycolumn = True
          End If
        Wend

'***********************************************************************

        '************Checking each column for match to GTFS allowed column******
        For i = 1 To columncount

          colMatch = False

          For j = 0 To UBound(transferscol, 1)

            If Range("A1").Offset(0, i - 1 - numDeleted).Value = transferscol(j) Then colMatch =
True

          Next j

          If colMatch = False Then

            If numDeleted = 0 Then

              Columns(i).EntireColumn.Delete 'Deleting the first unmatched column
              numDeleted = numDeleted + 1 'counting the number of columns deleted since the
act of deleted a column shifts the column index of all subsequent columns

            Else

              Columns(i - numDeleted).EntireColumn.Delete 'Deleting any remaining
unmatched columns
              numDeleted = numDeleted + 1 'counting the number of columns deleted since the
act of deleted a column shifts the column index of all subsequent columns

            End If

          End If

        Next i

'***********************************************************************
```

```
        openWB.Close (True)
        Call objFSO.CopyFile(objNewFolder.Path + "\" + "transfers.csv", objNewFolder.Path +
"\" + "transfers.txt", True)
        Call objFSO.DeleteFile(objNewFolder.Path + "\" + "transfers.csv")

    End If

    '*******************************Checking feed_info.txt
columns*************************************
    If objFile.Name = "feed_info.txt" Then

        Call objFSO.CopyFile(objFile.Path, objNewFolder.Path + "\feed_info.csv", True)
        Set openWB = Application.Workbooks.Open(objNewFolder.Path + "\feed_info.csv")

        '***********Check the number of columns in file***********************
        columncount = 0
        emptycolumn = False
        numDeleted = 0

        While emptycolumn = False
            If IsEmpty(Range("A1").Offset(0, columncount)) = False Then
                columncount = columncount + 1
            Else
                emptycolumn = True
            End If
        Wend

    '**********************************************************************************

        '***********Checking each column for match to GTFS allowed column******
        For i = 1 To columncount

            colMatch = False

            For j = 0 To UBound(feed_infocol, 1)

                If Range("A1").Offset(0, i - 1 - numDeleted).Value = feed_infocol(j) Then colMatch
= True

            Next j

            If colMatch = False Then

                If numDeleted = 0 Then
```

```
            Columns(i).EntireColumn.Delete 'Deleting the first unmatched column
                numDeleted = numDeleted + 1 'counting the number of columns deleted since the
act of deleted a column shifts the column index of all subsequent columns

            Else

                Columns(i - numDeleted).EntireColumn.Delete 'Deleting any remaining
unmatched columns
                numDeleted = numDeleted + 1 'counting the number of columns deleted since the
act of deleted a column shifts the column index of all subsequent columns

            End If

        End If

    Next i

'*************************************************************************

    openWB.Close (True)
    Call objFSO.CopyFile(objNewFolder.Path + "\" + "feed_info.csv", objNewFolder.Path +
"\" + "feed_info.txt", True)
    Call objFSO.DeleteFile(objNewFolder.Path + "\" + "feed_info.csv")

    End If

  Next objFile

'*************************************************************************
*********************************************

Next objSF


With Application
    .DisplayAlerts = False
    .ScreenUpdating = False
    .EnableEvents = False
End With


End Sub
```

APPENDIX B: CONSORTIUM GROUP TALKING POINTS

- **What is GTFS-ride?**

  GTFS-ride is a common format for fixed-route transit ridership data. GTFS-ride will allow transit agencies at many different levels of maturity and technological capability to reflect their ridership in a standardized way to facilitate information sharing and the use of common software tools.

- **Who are the GTFS-ride Consortium Group members?**

  A select national group of transit, research, state and federal agency representatives, transportation consultants, transit hardware/software vendors, and industry stakeholders.

- **What are the vision and goals of the Consortium Group?**

  The vision the Consortium Group is a future where industry stakeholders are motivated and active in adopting and using the GTFS-ride data standard. The Consortium Group goals are then to:

  o Help bring to fruition the recognized potential benefits of the GTFS-ride data standard.

  o Help steer and advise the development of the data standard and associated supporting software tools.

  o Help communicate the value and potential of adopting the GTFS-ride standard both within and outside of their respective organizations.

- What are the expectations of the Consortium Group members?

  o Participation in three, 90-minute meetings in 2018 and early 2019.

  o Review project documents and establish an understanding of the data standard.

  o Provide expert opinion, advice, and feedback to the project team when appropriate.

  o Facilitate and/or participate in ongoing discussions of related issues among group members and in the broader transit community.

APPENDIX C: TOPICAL GOOGLE GROUP POSTINGS

Commercial Interests in GTFS-ride

Greetings consortium members and transit stakeholders,

This thread focuses on the opportunities and challenges that vendors may face in incorporating/supporting the GTFS-ride standard in their commercial products and services. Please provide any feedback, questions, or comments that you feel are relevant to this topic. Also, feel free to pose questions to other members of this forum who might have unique insight and expertise on your issue. More specifically, please comment if you have information or opinions related to the following:

- To transit agencies, what would you like vendors to hear about the support for GTFS-ride in their commercial products that would be desired at your agency?

- Have you discussed GTFS-ride within your organization or with your customers?

- Will your customers or your organization benefit directly from the interoperability afforded through the GTFS-ride data standard?

- What products or services do you foresee incorporating GTFS-ride in the future and in what capacity?

- Do you foresee your organization being able to assist your customers in producing ridership data in the GTFS-ride standard or in discovering value from their ridership data already in the GTFS-ride standard (i.e., consuming GTFS-ride feeds)?

- Do you believe supporting GTFS-ride in software created by your organization would add to the value proposition and utility for your customers?

The GTFS-ride project team highly values your feedback and feels these discussion threads will help engage the stakeholder community and further the development of the GTFS-ride data standard.

Responses:

Data Quality in GTFS-ride

Greetings consortium members and transit stakeholders,

This thread focuses on the issues surrounding data quality and its implications for the GTFS-ride standard. More specifically, please comment if you have information or opinions related to the following questions:

- Do you (or your organization) have concerns about the quality of your collected (raw) ridership data?

- If concerns do exist, what would be needed to help your organization feel comfortable with publishing ridership data in the GTFS-ride standard? For example, a categorical field could be added to the standard to describe the level of data quality (e.g., "uncorrected raw data", "complete data", "incomplete data", "complete data w/inferences", etc.)

- Will having ridership in GTFS-ride help your organization facilitate quality assurance and/or quality control procedures?

- What types of additional metadata regarding ridership data quality would you feel is needed for GTFS-ride? (The current file *ride_feed_info.txt* and the **field's** record_use and **schedule**_relationship in *board_alight.txt* could be considered metadata currently in the GTFS-ride standard).

The GTFS-ride project team highly values your feedback and feels these discussion threads will help engage the stakeholder community and further the development of the GTFS-ride data standard. Therefore, please feel free to provide any feedback or comments that you think are relevant to this topic. If you have a specific concern or recommendation for the standard, we invite you to start an issue or pull request in the GitHub repository.

Responses:

Andrew Martin

- Our GTFS ride feed had a very small percentage of boardings data that was dropped in the conversion process. I have small concerns about releasing data that likely won't match our NTD report (if you released a whole year at a time). These concerns are very minor.

- I don't think it's really a large deal, but something to be aware of. Data consistency is important when releasing information to the public.

- I'm not sure how they would at this point.

- I'm not sure what else would be needed at this point.

GTFS-ride Adoption Challenges

Greetings consortium members and transit stakeholders,

We are looking to learn more about the challenges and practical considerations for adopting the GTFS-ride data standard. What challenges and practical considerations are most important to focus work upon? Also, please comment if you have information or insights related the following questions:

- Do you foresee significant challenges in connecting your ridership data to GTFS elements (e.g., trip_id, stop_id, route_id, etc.)?

- What "wow" factors would add value to your organization and help incentivize the adoption of GTFS-ride?

- Would estimates of NTD metrics (e.g., passenger miles traveled/capita, vehicle revenue miles/capita, etc.) be a useful functionality of GTFS-ride support software and help incentivize its adoption?

- What help or assistance would your organization require to adopt the continued use of GTFS-ride?

- Are there any partnerships with other organizations that you envision could be beneficial in helping to facilitate the implementation of GTFS-ride?

- How do you see your organization using GTFS-ride?

- Does your organization have any current plans for the adoption of the GTFS-ride data standard?

The GTFS-ride project team highly values your feedback and feels these discussion threads will help engage the stakeholder community and further the development of the GTFS-ride data standard.

Responses:

GTFS-ride

Any of our new members to the group have feedback for the GTFS-ride project team?

Angelic Peters

I foresee a significant challenge for rural agencies, particularly anywhere not using automatic passenger counters, to meet the needs for this project with the data that is currently exportable.

Matthew M Barnes

For GTFS-ride to be a practical option for smaller transit agencies (no dedicated technical staff), there is a need for software tools (GTFS-ride editor + simple analysis functionality) that compare favorably to manual spreadsheet ridership tracking and analysis practices.

Analysis and Planning Opportunities

Greetings consortium members and transit stakeholders,

Please take this opportunity to engage in the development and adoption of the GTFS-ride data standard. This thread focuses on the issues surrounding Analysis and Planning opportunities with the GTFS-ride standard. More specifically, please comment if you have information or opinions related to the following questions:

- In what types of analyses do you think GTFS-ride feeds would prove beneficial?
- What in the standard is particularly helpful for the analyses your organization performs or would like to perform?
- What additions to the standard would help your analysis and planning goals?
- What functionality would you like to see in GTFS-ride supporting analysis and planning software (i.e., software consuming GTFS-ride feeds)?
- Would having data in the GTFS-ride standard help your organization to share analysis tools and/or results?

The GTFS-ride project team highly values your feedback and feels these discussion threads will help engage the stakeholder community and further the development of the GTFS-ride data standard. Therefore, please feel free to provide any feedback or comments that you think are relevant to this topic. If you have a specific concern or recommendation for the standard, we invite you to start an issue or pull request in the GitHub repository.

Responses:

Logistics and Technical Challenges

Greetings consortium members and transit stakeholders,

It was great to see some of you and discuss GTFS-ride at the recent transit conferences in Bend and Napa. In efforts to keep the conversation going, please take this opportunity to engage in the development and adoption of the GTFS-ride data standard. This thread focuses on the issues surrounding logistics and technical challenges with the GTFS-ride standard. More specifically, please comment if you have information or opinions related to the following issues:

- There is a concern about GTFS-ride creating large files, especially in the case of the following:
  - Large volumes of stop-level data in board_alight.txt.
  - When many base GTFS feeds are needed to cover a time range of ridership data due to frequent republishing of GTFS.
- There has been interest expressed in adding fields for the option to provide summarized data (e.g., averages, std. dev., samples size, etc.) or processed data (e.g., estimated, extrapolated, cleaned, truncated, etc.) as opposed to only total counts.
  - What technical challenges would you foresee in implementing such fields?
  - What metadata fields would be desired to put any reported ridership into proper context?
- There is a recurring technical issue in making connections between ridership data (as it currently exists at agencies) and the required GTFS IDs (e.g., route_id, stop_id, trip_id, stop_sequence).
  - How much of a challenge would this be for your agency and staff?
  - What are your ideas for solutions to the issue, generalizable or unique?
  - What GTFS best practices could be adopted to make this transition easier?

Also, please see the outstanding issues on the GTFS-ride GitHub repository here: https://github.com/ODOT-PTS/GTFS-ride/issues. Many of these issues are also open technical challenges. If you have another concern or recommendation for the standard, we invite you to start a new issue or pull request in the GitHub repository.

The GTFS-ride project team highly values your feedback and feels these discussion threads will help engage the stakeholder community and further the development of the GTFS-ride data standard.

Responses:

Privacy Issues

Hello again transit community,

As we mentioned in the second consortium meeting in October, we are continuing our posts discussing the potential opportunities and challenges facing the GTFS-ride data standard. This post will focus on the privacy and public availability concerns arising from the use of GTFS-ride. Please comment if you have information or opinions related to the following:

- There are concerns that the unique **rider_id** in *rider_trip.txt* and stop-level travel patterns from low-density service in *board_alight.txt* could expose identities of specific riders. (See GitHub [Issue](#))
  - Would this be an active concern?
  - Are there any known or potential solutions?
  - Would adding language to the standard clarifying that any such data using GTFS-ride not necessarily be required to be public be a potential solution?

- It has also been expressed that agencies may not feel comfortable releasing some ridership data publicly due to quality concerns, conflicts with other data (e.g., NTD), and the potential for misinterpretation.
  - How much of a concern would this be for your organization?
  - Would more opportunity for context (e.g., metadata, summary statistics, quality measures, etc.) built into the standard alleviate these concerns?
  - What potential opportunities could be lost by keeping ridership data private?

Feel free to provide any feedback or comments that you think are relevant to this topic. If you have a specific concern or recommendation for the standard, we invite you to start an [issue](#) or [pull request](#) in the GitHub repository.

Also, consortium members, please offer your thoughts on these other past discussion posts:

[Logistics and Technical Challenges](#)
[Analysis and Planning Opportunities](#)
[GTFS-ride Adoption Challenges](#)
[Data Quality in GTFS-ride](#)
[Commercial Interests in GTFS-ride](#)

Responses:

Sharing and Reporting in GTFS-ride

Good Morning GTFS-ride Google Group,
This is the final discussion topic post stemming from the Consortium Group webinar meetings. This discussion focuses on the topic of sharing and reporting GTFS-ride datasets. Please comment if you have information or opinions related to the following:

- As previously mentioned, there have been concerns expressed about sharing GTFS-ride data.
  - Do you have concerns about sharing GTFS-ride data? In whole, or in part?
  - If you have concerns, what parts of a GTFS-ride feed would be helpful to be made public, and which may be best kept private?

- Would adopting GTFS-ride help you to share internally developed tools more broadly?

- What are your organization's thoughts and attitudes towards open source tools and open data sharing?

- What are your thoughts about how GTFS-ride data could potentially integrate with and/or ease NDT reporting?
  - What are your ideas on how to advance the value of GTFS-ride for NTD reporting?

- What challenges may arise from potentially large file sizes of GTFS-ride datasets?

Feel free to provide any feedback or comments that you think are relevant to this topic. If you have a specific concern or recommendation for the standard, we invite you to start an issue or pull request in the GitHub repository.
In addition, there is still a chance for consortium members to offer their thoughts on the following past discussion posts:
Logistics and Technical Challenges
Analysis and Planning Opportunities
GTFS-ride Adoption Challenges
Data Quality in GTFS-ride
Commercial Interests in GTFS-ride
Privacy Issues in GTFS-ride

We greatly appreciate the involvement and participation of different transit communities in helping to improve and promote the GTFS-ride data standard.

Responses:

Andrew Martin
- As previously mentioned, there have been concerns expressed about sharing GTFS-ride data.
  - Do you have concerns about sharing GTFS-ride data? In whole, or in part?
  - If you have concerns, what parts of a GTFS-ride feed would be helpful to be made public, and which may be best kept private?

I don't think that LTD has serious concerns about sharing most of our data. Part of that is based on what we are currently collecting. We will soon be implementing an electronic fare collection system. Once this is in place, we may have more concerns. We have already identified that we do not want to share our customer ridership information, which is one of the files in the GTFS-ride standard. Obviously the ability to link rider trips to service is important, but it's not something we would want to publish externally at a disaggregate level. Overall, most of our records are subject to records requests and we are willing to share the information. In my work specifically, we are trying to put more of our information online before we get requests to cut down on headaches for ourselves and our public when they have to go through a request process to get at the information that they are ultimately entitled to.

- Would adopting GTFS-ride help you to share internally developed tools more broadly?

I think if GTFS-ride tools were developed, some of our stakeholders would certainly want us to share the information in this format. One of the challenges we face currently is the format that we can share ridership information in. Every request ends up being a semi-custom data dump to the person who requested it. Having a widely accepted standard would allow us to be more consistent about what is published and to share that more broadly once we have created the files.

- What are your organization's thoughts and attitudes towards open source tools and open data sharing?

As I mentioned above, at least some individuals at LTD are starting to talk about publishing our information more broadly in an effort to avoid the burden of having more requests for data. We are required to provide almost anything we have to the public, so publishing it rather than making people do a records request seems to make sense to me, particularly for information like ridership numbers.

- What are your thoughts about how GTFS-ride data could potentially integrate with and/or ease NTD reporting?

I'm not sure how it could help us. LTD reports many different modes and not all of our data gets put into the same database depending on which service it was collected from. We may be unique in that aspect, but I can't say. We already have a fairly extensive back-end set up of views and tables to help us report NTD. Any problems we have would not be related to the format that we are collecting and then reading the data from.

  o What are your ideas on how to advance the value of GTFS-ride for NTD reporting?

- What challenges may arise from potentially large file sizes of GTFS-ride datasets?

I think generating and processing the data sets may end up being a challenge. LTD collects very disaggregate ridership data and will soon have more data from our electronic fare system to join to our APC and AVL data. I worry that once it becomes public knowledge that we can provide very

specific types of information for basically any date range, people will expect us to regularly publish all of that. That will end up being a very large amount of data for us to process and store for a relatively small amount of people to look through. Some level of aggregation will be necessary for any kind of regular publication of our data to control the size of files.