

**PROOF OF CONCEPT: GTFS DATA AS A
BASIS FOR OPTIMIZATION OF
OREGON'S REGIONAL AND
STATEWIDE TRANSIT NETWORKS**

Final Report

SPR 752



**PROOF OF CONCEPT: GTFS DATA AS A BASIS FOR
OPTIMIZATION OF OREGON'S REGIONAL AND
STATEWIDE TRANSIT NETWORKS**

Final Report

SPR 752

by

J. David Porter, David S. Kim, Saeed Ghanbartehrani
School of Mechanical, Industrial and Manufacturing Engineering
Oregon State University

for

Oregon Department of Transportation
Research Section
555 13th Street, Suite 1
Salem OR 97301

and

Federal Highway Administration
400 Seventh Street, SW
Washington, DC 20590-0003

May 2014

1. Report No. FHWA-OR-RD-14-12		2. Government Accession No.		3. Recipient's Catalog No.	
4. Title and Subtitle Proof Of Concept: GTFS Data As A Basis For Optimization Of Oregon's Regional And Statewide Transit Networks				5. Report Date May 2014	
				6. Performing Organization Code	
7. Author(s) J. David Porter, David S. Kim, Saeed Ghanbarteherani, School of Mechanical, Industrial and Manufacturing Engineering, Oregon State University				8. Performing Organization Report No. SPR 752	
9. Performing Organization Name and Address Oregon Department of Transportation Research Section 500 13 th Street NE, Suite 1 Salem, OR 97301				10. Work Unit No. (TRAIS)	
				11. Contract or Grant No.	
12. Sponsoring Agency Name and Address Oregon Department of Transportation Research Section 500 13 th Street NE, Suite 1 Salem, OR 97301-5192 and Federal Highway Administration 400 Seventh Street, SW Washington, DC 20590-0003				13. Type of Report and Period Covered Final Report	
				14. Sponsoring Agency Code	
15. Supplementary Notes					
16. Abstract Assessing the current "state of health" of individual transit networks is a fundamental part of studies aimed at planning changes and/or upgrades to the transportation network serving a region. To be able to effect changes that benefit both the individual transit networks as well as the larger transportation system, organizations need to develop meaningful strategies guided by specific performance metrics. A fundamental requirement for the development of these performance metrics is the availability of accurate data regarding transit networks. Prior to 2005, transit data was not readily available. This situation complicated the assessment of single transit networks, let alone performing a state-wide or region-wide study. The advent of the General Transit Feed Specification (GTFS) changed this constrained landscape and motivated transit operators to release their schedules and route information to third party developers. In this report, the development work conducted to create an open source software tool to help the Oregon Department of Transportation's Public Transit Division gain a better understanding and more efficient utilization of existing state-wide transit networks is described. The final product, referred to as the Transit Network Analysis software tool, incorporates GTFS data and census data as its main inputs and can be used to visualize, analyze and report on the Oregon transit network.					
17. Key Words CENSUS DATA, GTFS, OPENTRIPPLANNER, OPEN SOURCE SOFTWARE, TRANSIT NETWORK.			18. Distribution Statement Copies available from NTIS, and online at http://www.oregon.gov/ODOT/TD/TP_RES/		
19. Security Classification (of this report) Unclassified		20. Security Classification (of this page) Unclassified		21. No. of Pages 72	22. Price

SI* (MODERN METRIC) CONVERSION FACTORS

APPROXIMATE CONVERSIONS TO SI UNITS					APPROXIMATE CONVERSIONS FROM SI UNITS				
Symbol	When You Know	Multiply By	To Find	Symbol	Symbol	When You Know	Multiply By	To Find	Symbol
<u>LENGTH</u>					<u>LENGTH</u>				
in	inches	25.4	millimeters	mm	mm	millimeters	0.039	inches	in
ft	feet	0.305	meters	m	m	meters	3.28	feet	ft
yd	yards	0.914	meters	m	m	meters	1.09	yards	yd
mi	miles	1.61	kilometers	km	km	kilometers	0.621	miles	mi
<u>AREA</u>					<u>AREA</u>				
in ²	square inches	645.2	millimeters squared	mm ²	mm ²	millimeters squared	0.0016	square inches	in ²
ft ²	square feet	0.093	meters squared	m ²	m ²	meters squared	10.764	square feet	ft ²
yd ²	square yards	0.836	meters squared	m ²	m ²	meters squared	1.196	square yards	yd ²
ac	acres	0.405	hectares	ha	ha	hectares	2.47	acres	ac
mi ²	square miles	2.59	kilometers squared	km ²	km ²	kilometers squared	0.386	square miles	mi ²
<u>VOLUME</u>					<u>VOLUME</u>				
fl oz	fluid ounces	29.57	milliliters	ml	ml	milliliters	0.034	fluid ounces	fl oz
gal	gallons	3.785	liters	L	L	liters	0.264	gallons	gal
ft ³	cubic feet	0.028	meters cubed	m ³	m ³	meters cubed	35.315	cubic feet	ft ³
yd ³	cubic yards	0.765	meters cubed	m ³	m ³	meters cubed	1.308	cubic yards	yd ³
NOTE: Volumes greater than 1000 L shall be shown in m ³ .									
<u>MASS</u>					<u>MASS</u>				
oz	ounces	28.35	grams	g	g	grams	0.035	ounces	oz
lb	pounds	0.454	kilograms	kg	kg	kilograms	2.205	pounds	lb
T	short tons (2000 lb)	0.907	megagrams	Mg	Mg	megagrams	1.102	short tons (2000 lb)	T
<u>TEMPERATURE (exact)</u>					<u>TEMPERATURE (exact)</u>				
°F	Fahrenheit	(F-32)/1.8	Celsius	°C	°C	Celsius	1.8C+32	Fahrenheit	°F

*SI is the symbol for the International System of Measurement

ACKNOWLEDGEMENTS

This research was funded by the Oregon Department of Transportation (ODOT) Research Section. We would like to thank Matthew Barnes from ODOT's Rail & Public Transit Division for providing the vision for this project. We would also like to thank Bernie Jones and Lyn Cornell (ODOT Research) for their assistance and support of this research.

DISCLAIMER

This document is disseminated under the sponsorship of the Oregon Department of Transportation and the United States Department of Transportation in the interest of information exchange. The State of Oregon and the United States Government assume no liability of its contents or use thereof.

The contents of this report reflect the view of the authors who are solely responsible for the facts and accuracy of the material presented. The contents do not necessarily reflect the official views of the Oregon Department of Transportation or the United States Department of Transportation.

The State of Oregon and the United States Government do not endorse products of manufacturers. Trademarks or manufacturers' names appear herein only because they are considered essential to the object of this document.

This report does not constitute a standard, specification, or regulation.

TABLE OF CONTENTS

1.0	INTRODUCTION.....	9
1.1	PROJECT OBJECTIVE	9
1.2	REPORT ORGANIZATION	10
2.0	LITERATURE REVIEW ON TRANSIT NETWORK ANALYSIS TOOLS	11
2.1	CONCLUSIONS FROM THE LITERATURE REVIEW.....	12
3.0	DEVELOPMENT OF THE TNA SOFTWARE TOOL	15
3.1	THE OPEN TRIP PLANNER ARCHITECTURE.....	15
3.1.1	<i>Additional Software Development Tools.....</i>	<i>16</i>
3.2	SERVER-SIDE DEVELOPMENT TASKS	17
3.2.1	<i>Hierarchical List of Transit Agencies.....</i>	<i>17</i>
3.2.2	<i>Stops by Agency ID.....</i>	<i>18</i>
3.2.3	<i>Stops by Route.....</i>	<i>18</i>
3.2.4	<i>Route Shape by Route Variant</i>	<i>18</i>
3.2.5	<i>Transit Agency Summary Report</i>	<i>19</i>
3.2.6	<i>Transit Agency Extended Report</i>	<i>19</i>
3.2.7	<i>Routes Report.....</i>	<i>19</i>
3.2.8	<i>Stops Report.....</i>	<i>19</i>
3.2.9	<i>Population within X Distance of Stops</i>	<i>19</i>
3.2.10	<i>Route Miles</i>	<i>19</i>
3.2.11	<i>Service Miles per Day.....</i>	<i>20</i>
3.2.12	<i>Additional Server-Side Development Tasks</i>	<i>21</i>
3.3	CLIENT-SIDE DEVELOPMENT TASKS	21
3.3.1	<i>The Tree Structure of the Transit Agency Pane</i>	<i>22</i>
3.3.2	<i>Resizable Transit Agency Pane.....</i>	<i>22</i>
3.3.3	<i>Map Tiles</i>	<i>22</i>
3.3.4	<i>Displaying Stops and Route Shapes.....</i>	<i>23</i>
3.3.5	<i>Clustering Stops.....</i>	<i>23</i>
3.3.6	<i>Reports.....</i>	<i>23</i>
3.3.7	<i>Sorting Reports</i>	<i>24</i>
3.3.8	<i>Displaying County Shapes over the Map.....</i>	<i>24</i>
3.3.9	<i>Report Export Functionality</i>	<i>24</i>
3.3.10	<i>Report Print Functionality.....</i>	<i>24</i>
3.3.11	<i>Linking Reports.....</i>	<i>24</i>
3.3.12	<i>Styling Reports</i>	<i>25</i>
3.4	OTHER DEVELOPMENT	25
3.4.1	<i>Acquiring Up-To-Date Oregon Map Data.....</i>	<i>25</i>
3.4.2	<i>Building the Graph File.....</i>	<i>25</i>
3.4.3	<i>Incorporating Census Data</i>	<i>26</i>
4.0	TRANSIT ANALYSIS NETWORK TOOL QUICK USER GUIDE	29
4.1	ACCESSING THE TNA SOFTWARE TOOL	29
4.2	INTERFACING WITH THE TNA SOFTWARE TOOL.....	29
4.3	THE TRANSIT AGENCY PANE	30
4.3.1	<i>Pane Width.....</i>	<i>30</i>
4.3.2	<i>Pane Tree Structure.....</i>	<i>31</i>

4.4	SELECTING TRANSIT AGENCIES FOR VISUALIZATION	32
4.4.1	Stops Clusters	32
4.4.2	Zooming In on Stops	33
4.4.3	Displaying Routes	35
4.5	REPORTS	39
4.5.1	Transit Agency Summary Report	40
4.5.2	Transit Agency Extended Report	42
4.5.3	Routes Report.....	43
4.5.4	Stops Report.....	44
5.0	CONCLUSIONS AND FUTURE WORK	47
6.0	REFERENCES.....	49

APPENDIX A: LIST OF REVIEWED COMMERCIALY AVAILABLE TNA SOFTWARE TOOLS

APPENDIX B: GLOSSARY OF TERMS

LIST OF TABLES

Table 2.1: Selection Criteria for Open Source Framework	13
Table 3.1: References and Software Libraries	17
Table 3.2: Salient Features of Population Distribution Methods	26

LIST OF FIGURES

Figure 3.1: Main components of the OTP architecture.....	15
Figure 3.2: TNA software tool transit agency pane tree structure	18
Figure 3.3: Relationships among routes, trips, calendar data and schedules in the GTFS data	20
Figure 4.1: The TNA software tool graphical user interface	30
Figure 4.2: Modifying the size of the transit agency pane.....	31
Figure 4.3: Transit agency pane tree structure.....	31
Figure 4.4: Selecting transit agencies for visualization	32
Figure 4.5: Examples of stops clusters for two different transit agencies.....	33
Figure 4.6: Second level decomposition of stops clusters for the City of Albany/Linn Benton Loop Transit Agency	33
Figure 4.7: Service area envelope of a 10-stops cluster.....	34
Figure 4.8: Transit agency name tool tip and stop location balloon	34
Figure 4.9: Route visualization example for the Corvallis Transit System	36
Figure 4.10: Route selection visualization options with a list box	37
Figure 4.11: Selecting all the route shapes from the Corvallis Transit Agency for visualization.....	38
Figure 4.12: Selecting all the route shapes and stops clusters from the Corvallis Transit Agency for visualization...38	
Figure 4.13: Accessing other reports from the "Transit Agency Summary" report.....	39
Figure 4.14: Buttons to Allow User to Export, Print or Close a Report	39
Figure 4.15: "Transit Agency Summary" report.....	41
Figure 4.16: "Transit Agency Extended" report.....	42
Figure 4.17: Routes report for the Columbia County Rider transit agency	44
Figure 4.18: "Stops" report for the Northeast Oregon Public Transit agency.....	45

1.0 INTRODUCTION

Assessing the current "state of health" of individual transit networks is a fundamental part of studies aimed at planning changes and/or upgrades to the transportation network serving a region. To be able to effect changes that benefit both the individual transit networks as well as the larger transportation system, organizations need to develop meaningful strategies guided by specific performance metrics. A fundamental requirement for the development of these performance metrics is the availability of accurate data regarding transit networks.

Prior to 2005, software engineers had to "data scrape" an agency's web site or were required to submit *Freedom of Information Act* requests to obtain transit data (*Roth 2010*). This situation complicated the assessment of single transit networks, let alone performing a state-wide or region-wide study. The advent of the *General Transit Feed Specification* (GTFS) changed this constrained landscape and motivated transit operators to release their schedules and route information to third party developers. The GTFS is a common format for public transportation schedules and associated geographic information. Using the GTFS specification, a public or private transit agency can describe such characteristics of their transit network as service calendar, stop times, stop locations, trips (a specific stop pattern) and routes (collection of trips), to name a few. The resulting GTFS feed can be used to acquire public transportation information about an agency in space and time. Many transit agencies in the U.S. (and across the world) have already created and adopted the GTFS data standard to make information about their network available to users (*googletransitdatafeed undated*). In the state of Oregon, approximately 85% of fixed route transit providers have GTFS data for their services (*ODOT GTFS undated*).

1.1 PROJECT OBJECTIVE

Based on the Oregon Department of Transportation's (ODOT) Public Transit Division (PTD) long experience in dealing with public transit issues, the majority of investments made in transit service by state, local, and federal government agencies are:

1. Based on local service providers' individual priorities, and
2. Made with limited ability to consider the impact that these investments may have on the larger transit network.

The main objective of this project was to develop an open source software tool that will help the ODOT's PTD gain a better understanding of the structure (i.e., number of stops and routes) and geographical coverage of the larger Oregon state transit network and how it correlates to populations within the state. The final product, referred to as the "transit network analysis" (TNA) software tool, incorporates publicly available GTFS data and census data as its main inputs and can be used to visualize, analyze and report on the Oregon transit network.

1.2 REPORT ORGANIZATION

The remainder of this report is organized as follows. Chapter 2 presents the results of the literature conducted on available transit network analysis tools. Chapter 3 details the process followed and software packages/utilities used in the development of the TNA software tool. Chapter 4 is structured as a quick user guide to introduce users to the features of the TNA software tool. Finally, chapter 5 presents the conclusions and recommendations for future work.

2.0 LITERATURE REVIEW ON TRANSIT NETWORK ANALYSIS TOOLS

A "transit network" (or "transportation network") refers to a network of roads, streets, or any structure that allows vehicular movement (*Xie and Levinson 2011*). A review conducted as part of this study concluded that a large number of software packages are currently available in the marketplace that perform some form of transit network analysis. These packages vary largely in terms of the software platforms they are built upon as well as the analysis capabilities they offer.

To compare these available software packages, a set of criteria was needed. However, universally accepted criteria to compare such software packages does not exist. Therefore, a set of criteria was defined to assess whether currently available software packages offer similar features to those to be included in the TNA software tool developed. The assessment criteria for a software package are:

- Be capable of collecting and storing GTFS data.
- Be capable of keeping GTFS data up-to-date.
- Allow the visualization of GTFS data on a map. Desirable features of this capability may include: displaying stops, routes, transit agency information and other information provided on a map; allowing the user to select what feed/feeds to display; and providing a web-based interface.
- Be capable of generating different kinds of reports based on queries from GTFS data.
- Be developed using open source tools.

The criteria for open source development was a strategic decision made by the ODOT's PTD to enhance the accessibility of the benefits provided by the TNA software tool. Developing the tool in an open source environment will also enhance the functionality of the tool by making its code available to organizations and developers with similar interests.

A total of 43 software packages in different areas related to the analysis of transit networks were identified and analyzed. The complete list of these software packages is included in Appendix A in a tabular format. The set of criteria used to classify these software packages is included in this table.

2.1 CONCLUSIONS FROM THE LITERATURE REVIEW

After completing the review of all 43 software packages, the short list of options shown in Table 2.1 were considered as potential development frameworks for the TNA software tool. For each of these options, the advantages, disadvantages and the required development effort were documented.

The conclusion reached from this analysis is that only OpenTripPlanner (OTP) met most of the criteria discussed before and provided the best balance between already available features and required development work. OTP is a multi-modal trip planner launched in 2009 as an open source community driven trip planner that comes with GTFS data storage and visualization capabilities. Currently Portland's TriMet uses OTP as its official trip planner (*OpenTripPlanner undated*).

Table 2.1: Selection Criteria for Open Source Framework

Framework Alternative	Advantages	Disadvantages	Required Development by OSU
<p>Open Trip Planner (OTP) OTP is an open source multi-modal trip planner, which runs on Linux, Windows, or potentially any platform with a Java virtual machine.</p>	<ul style="list-style-type: none"> - Trip planner (i.e., routing engine) can be used to enable future functionality of the tool. - Supports GTFS and GTFS real-time natively. - Provides a web interface and an application programming interface (API). - Everything on the server side is developed in Java, except for the web interface which requires JavaScript. 	<ul style="list-style-type: none"> - Uses an XML graph file as a database (instead of a relational database). 	<ul style="list-style-type: none"> - Adding customized data fields into the graph file. - Customizing web interface to accommodate layer selection, querying and reporting features. - Adding querying features to the engine. - Adding reporting features to the engine.
<p>Quantum GIS (QGIS) QGIS is a cross platform open source GIS package (i.e., it can run on Windows, Mac, Linux, Android) licensed under GPL.</p>	<ul style="list-style-type: none"> - Has a desktop component to edit base maps and custom layers. - Has a standard-compliant web map service (WMS) 1.3 server that can be easily configured to share QGIS desktop project files. - - Has a web client based on open layers and GeoExt. (JavaScript). 	<ul style="list-style-type: none"> - Currently, QGIS is not compatible with GTFS data. - Map server is just a WMS (does not provide routing or querying features). 	<ul style="list-style-type: none"> - Adding a PostGIS database and querying component for the storage of other data. - Customizing web interface to accommodate querying and reporting features. - Developing a querying engine. - Adding reporting features to the engine.
<p>Building from scratch The best combination of open source tools seem to be GeoDjango, PostGIS and OpenLayers.</p>	<ul style="list-style-type: none"> - Designing everything from scratch based on our needs. 	<ul style="list-style-type: none"> - Some features that are already available in OTP and QGIS may be difficult to implement. 	<ul style="list-style-type: none"> - Everything.

3.0 DEVELOPMENT OF THE TNA SOFTWARE TOOL

In this chapter, the development work conducted to construct the "transit network analysis" (TNA) software tool is described. The majority of the development tasks involved making modifications and/or upgrading the client-server architecture of OpenTripPlanner (OTP).

This chapter is organized as follows. Section 3.1 describes the main features of the OTP client-server architecture. Section 3.2 describes the development tasks completed on the *server* side of OTP, whereas section 3.3 describes the development tasks completed on the *client* side. Section 3.4 describes additional development tasks that improve the functionality of the TNA software tool, but did not involve working with the server or the client side of the OTP architecture.

3.1 THE OPEN TRIP PLANNER ARCHITECTURE

OTP is a multi-modal trip planner launched in 2009 as an open source, community driven trip planner that allows for GTFS data storage and provides visualization capabilities. The OTP version 0.9.2 was used in the development of the TNA software tool. Figure 3.1 depicts the main components of OTP and its basic architecture.

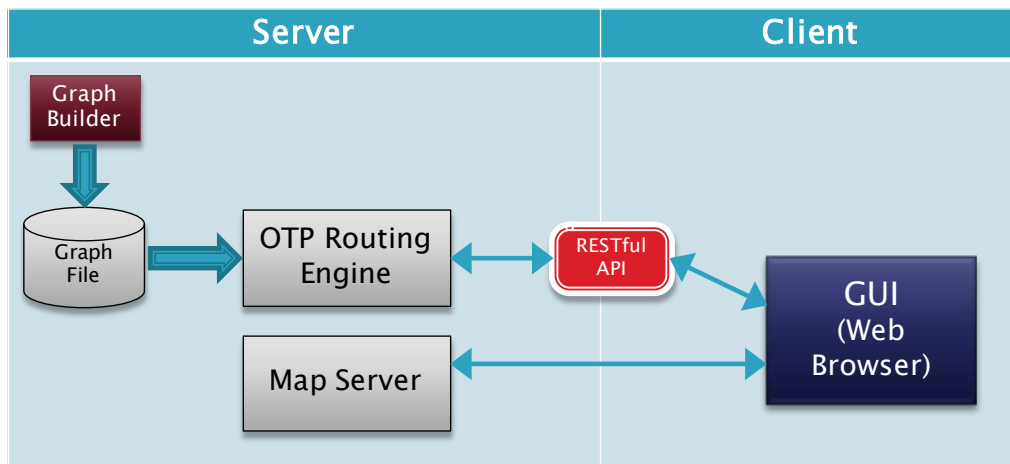


Figure 3.1: Main components of the OTP architecture

OTP is based on a client-server architecture. The resource intensive computations are performed by the server side and make it possible to run OTP on any computer with any type of operating system and hardware configuration. The server application of OTP can run on Windows, Mac, and Linux platforms and is developed in Java. The only requirement on the client side is a web browser (e.g., Mozilla Firefox or Google Chrome) which acts as the Graphical User Interface (GUI).

A fundamental component of the OTP architecture is the "Graph File," which is an *extensible markup language* (XML) based text file for storing road network data as well as GTFS data. The "Graph File" needs to be created once the server is set up using map and GTFS data for the region of interest (a "Graph File" builder module is available for this purpose). The "OTP Routing Engine" is responsible for processing requests from the GUI, extracting the requested data from the "Graph File," creating the response and sending it back to the GUI. The "RESTful API" is the main interface between the GUI and the "OTP Routing Engine." A map server can be used to provide map tiles for the area that is being displayed on the GUI. OTP provides documentation, ready-to-go builds and sample data files for developers (*GitHub undated b*). It also has a wiki (*GitHub undated a*) and a developers' mailing list (*Google undated*) to aid developers in finding solutions to programming issues.

3.1.1 Additional Software Development Tools

In this section, the software tools and software libraries used to add functionality to the TNA software tool are introduced. OTP comes with set up tutorials (*GitHub undated c*) for both Eclipse and Netbeans. Eclipse was chosen as the development environment due to familiarity and past experience with this environment.

Eclipse is a free and open source integrated development environment (IDE) which can be used to develop applications in Java, as well as other programming languages (i.e., using plug-ins) such as C, C++, JavaScript, Python, and CSS, to name a few (*Eclipse undated*). In order to properly import and work on all OTP modules in Eclipse, plug-ins like EGit, m2eclipse, and WTP were necessary. Detailed information on configuring the Eclipse IDE is provided in "Getting Started with OTP in Eclipse" tutorial (*GitHub undated d*) available on OTP's wiki page.

The OTP components webapp, graph builder and other server side components have been developed in Java. The client side application is written in hypertext markup language (HTML) format (*W3C[®] undated a*). Cascading Style Sheets (CSS) (*W3C[®] undated b*) is used for styling the web pages and JavaScript (*Flanagan 2011*) is the main scripting language used in OTP's GUI.

Other than the tools mentioned above, some additional JavaScript plug-ins and libraries were also used to add functionality to the client side of the TNA software tool. Table 3.1 provides a brief description of these libraries and points to specific references where more information can be found.

Table 3.1: References and Software Libraries

Name	Description	Reference
jQuery	jQuery is a feature rich JavaScript library that is compatible with most web browsers.	<i>(jQuery undated a)</i>
jQuery UI	jQuery UI is a plug-in for jQuery that provides some effects, widgets and interactions for jQuery.	<i>(jQuery undated b)</i>
jstree	An open source free JavaScript plug-in that provides a tree menu structure.	<i>(Bozhanov undated)</i>
Leaflet	Leaflet is a versatile and widely used JavaScript library, which is open source and free. It is used for building interactive maps.	<i>(Agafonkin undated)</i>
leaflet markercluster	Leaflet markercluster is a JavaScript library that provides clustering capability for leaflet markers. Markercluster can display up to 50,000 markers on map efficiently.	<i>(GitHub undated e)</i>
Sorttable	A JavaScript library used for implementing a table sorting feature.	<i>(Langridge 2003)</i>
Leaflet.encoded	An open source leaflet library used to decode the encoded polyline into an array of L.LatLng objects that can be displayed on a map.	<i>(GitHub undated f)</i>
jQuery UI Multiselect	An open source jQuery UI widget that allows creating multi-selectable drop down list of checkboxes.	<i>(Hynds 2010)</i>

3.2 SERVER-SIDE DEVELOPMENT TASKS

One of the main benefits of a client-server architecture is that any resource intensive operations needed by a software solution can be performed on the server side to achieve the best possible user experience on the client side. The TNA software tool takes maximum advantage of this benefit.

The majority of the development tasks presented in this section involved the design, implementation and testing of queries that provided data to be displayed either directly on the graphical user interface (GUI) or were included in one of the reports generated by the TNA software tool. These queries were designed to minimize further data processing on the client side of the OTP architecture.

3.2.1 Hierarchical List of Transit Agencies

This query is used to provide the required data for populating the transit agency pane on the TNA software tool GUI. The transit agency pane uses a tree structure that lists transit agencies at its top level, followed by routes and then route variants, as depicted in Figure 3.2.

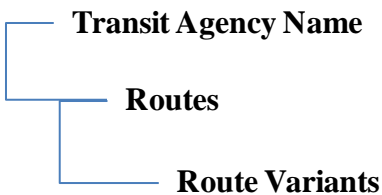


Figure 3.2: TNA software tool transit agency pane tree structure

Since the jstree library is used to populate the transit agency pane on the GUI, a JavaScript Object Notation (JSON) object formatted according to the jstree input specifications is an ideal form of providing data for the menu. JSON is a lightweight data interchange format which is easy to read and write for both humans and computers (*ECMA 2013*). More details on how the menu was implemented is available at section 3.3.1.

3.2.2 Stops by Agency ID

This query provides the name and coordinates of the stops for a given transit agency identified by its agency ID. Graph builder (i.e., OTP's software module that creates the graph file) assigns an agency ID to every stop, which is the agency ID of the transit agency that serves the stop.

It is important to note that there are a few Oregon GTFS feeds that contain data for multiple transit agencies. In these cases, the graph builder assigns all the stops contained in the multi-agency GTFS feed to a default agency ID that is specified in the graph builder configuration file. Therefore, filtering stops by the agency ID assigned by the graph builder may not work well since the stops served by the agencies that are not specified as the default agency in the graph builder configuration file cannot be retrieved.

In order to query the stops by agency ID, all the routes for the given transit agency are queried first and then the stops for every route are queried and copied to a list. Since a single stop can be served by multiple routes, every stop is added to the list only if it is not already in it. This querying schema also solves the problem faced with multi-agency GTFS feeds by querying stops by looking at the route data instead of stops themselves.

3.2.3 Stops by Route

This query provides a list of stops for a given route and transit agency and is used when a route is clicked on the transit agency pane tree structure to retrieve stop coordinates and stop names and display them on the map.

3.2.4 Route Shape by Route Variant

This query provides the shape for a route variant, given the transit agency ID, route ID and route variant name. This query is used when a route variant is selected on the tree menu to be displayed on the map and returns the coordinates of the route shape in the form of an *encoded polyline*. The specifications and algorithm of an *encoded polyline* are explained in google maps

API (Google undated b). The encoded polyline is then converted into L.LatLNg objects using Leaflet.encoded so that it can be displayed on a map.

3.2.5 Transit Agency Summary Report

This query generates the "Transit Agency Summary" report in JSON format. This query has no input parameters, and once it is called, it generates a list that includes the *agency ID*, *agency name*, *phone number*, count of *total routes*, count of *total stops*, *population served*, and *fare URL*, for all the transit agencies in Oregon for which a GTFS feed is available.

3.2.6 Transit Agency Extended Report

This query generates the "Transit Agency Extended" report in JSON format. This query generates a list that includes, for a given transit agency, the fields *agency ID*, *agency name*, *route miles*, *route stops*, *stops per route mile*, *population served by route*, *service miles*, *service stops*, and *population served by service*. This report can be generated for a single day of the week, a selection of multiple days of the week, or for the entire week.

3.2.7 Routes Report

This query generates the "Routes" report. This query generates a list that includes, for a single route of a given transit agency, *route ID*, *route name*, *route long name*, *route type*, *route length*, *total stops* count for every route, *unduplicated population*, *service stops*, *population served by service*, and *route description*. This report can be generated for a single day of the week, a selection of multiple days of the week, or for the entire week.

3.2.8 Stops Report

This query generates the "Stops" report. This query generates a list that includes, for a single stop of a single route of a given transit agency, *stop ID*, *stop name*, the *route(s) that the stop belongs to*, and *population served*.

3.2.9 Population within X Distance of Stops

This query computes the population around a point (given its coordinates) within a certain distance *X* (in miles) from either a stop, a route, or a transit agency. More information on how this query works is provided in section 3.4.3.

3.2.10 Route Miles

This query computes route miles for a given agency ID. Route miles is the summation of all route lengths for a given agency. Since length of a route can vary based on the service schedule, longest route length is considered in this computation.

3.2.11 Service Miles per Day

This query computes service miles for a transit agency for a given day of the week. In order to compute the service miles, the parameters *number of trips* and *length of every trip* are queried for the given day of the week and then they are summed up over all routes.

The diagram depicted in Figure 3.3 illustrates how the GTFS relates route and trip data to daily schedules and calendar information. Every route for a given transit agency is identified by a unique route ID. For a given route ID, there has to be at least one trip ID that matches the information related to arrival time, departure time, and stop sequence in the *stop_times.txt* file. Therefore, trips identified by unique trip IDs, are different instances of the same route being served on different hours in the file *stop_times.txt*. The attribute "service_id" in the file *calendar.txt* is the key to match trip IDs with different days of the week. For every service pattern represented by a set of days (e.g., Monday, Wednesday, Thursday, and Friday) that some service is active on, there needs to be a unique "service_id" that can be matched to one or more trip IDs.

Therefore, in order to compute service miles per day for a given transit agency, all the routes along with their corresponding trip IDs and service IDs need to be queried. Once this information is obtained, service IDs need to be compared with calendar data for the given day to find which service IDs are in effect for the day of interest. Finally, the trip lengths for the effective trips on the given day are computed and summed up to find service miles per day.

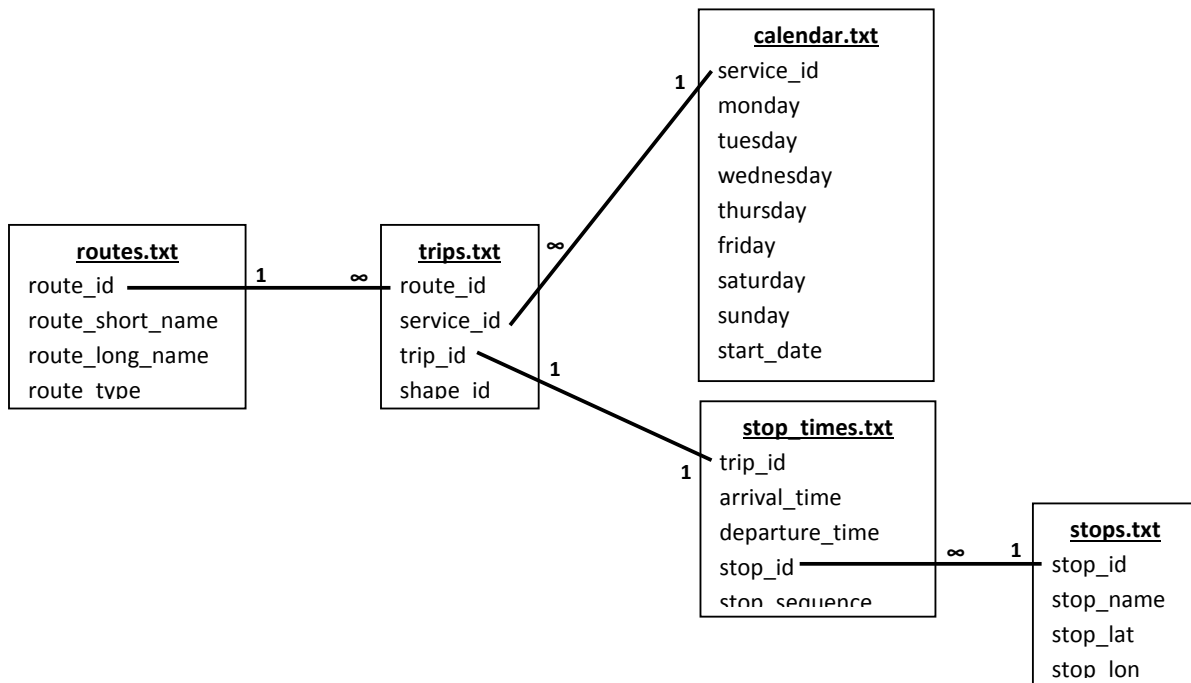


Figure 3.3: Relationships among routes, trips, calendar data and schedules in the GTFS data

3.2.12 Additional Server-Side Development Tasks

3.2.12.1 *Configuring OTP to Work with Two Graph Files*

As it is explained in section 3.4.3, the census data retrieved from the U.S. census bureau web site was converted into a GTFS format. In order to avoid slowing down the querying performance of the OTP server, the census data was loaded into a separate graph file. Thus, the TNA software tool operates with two separate graph files; one containing transit data and the other containing census data. Since OTP is capable of working with multiple graph files simultaneously, a few configuration changes had to be made to allow the second graph file to work properly. The instructions on how to configure OTP to work with multiple graph files are available on the OTP wiki (*GitHub undated g*).

3.2.12.2 *Updating graphbuilder Transit Index Builder*

In order to be able to query calendar data as well as the number of trips per day for a given route variant, some changes had to be made to the transit index builder in *opentripplanner-graphbuilder* module to allow the required information into the OTP graph file. These changes are suggested by user “tuure” and are available on OTP page on GitHub (*GitHub undated h*). After making the changes, a new graph file was built so that the required information (i.e., calendar data, trip frequencies and service ids) were available for querying.

3.3 CLIENT-SIDE DEVELOPMENT TASKS

The OTP Analyst Extension user interface was used as the basis for developing the GUI of the TNA software tool. The OTP Analyst Extension was chosen mainly because it does not come with any of the unnecessary functions that are available in the OTP standard web interface like trip planning, bike rental, and so on. Therefore, it is easier to modify.

The main page layout design of the TNA software tool was based on the page layout of the original OTP Analyst Extension. The additional development tasks conducted to fit the requirements of the TNA software tool to this base GUI are described in the following sections.

3.3.1 The Tree Structure of the Transit Agency Pane

A transit agency pane that followed a tree structure was implemented in order to allow a user to view and select one or multiple transit agencies for visualization. The transit agency pane tree structure has three levels:

- **Level 1.** Lists all transit agencies in the state of Oregon in alphabetical order. Clicking an item at this level unchecks all of its child nodes (if checked) and displays the stops that belong to the selected transit agency on the map.
- **Level 2.** Lists all routes that belong to the selected transit agency (parent node). Selecting any nodes at this level unchecks its parent node (if checked) and displays the stops that are served by the selected route.
- **Level 3.** Lists all route variants for the route (expanded parent node). Checking a node at this level, displays the route shape on map.

The open source JavaScript plug-in called *jstree* was used to implement the tree structure used in the transit agency pane. The plug-in *jstree* is customized so that a user of the TNA software tool can select any of the nodes in the tree by checking any of the available boxes. Checking a box in the tree structure triggers an event based on the type of node selected. In order to populate the tree structure menu, a JSON object is created on the server side application (see section 3.2.1). Therefore, no extra manipulation is required for data conversion, which ensures a swift user experience on the client side of the TNA software tool. The tree structure menu is loaded once the client side main page is called and it is used as long as user keeps the page open.

3.3.2 Resizable Transit Agency Pane

The strings names of some of the objects in the transit agency pane (e.g., transit agency name) may be very long. Thus, the transit agency pane can be resized to avoid forcing the user to scroll left and right. Making the tree structure resizable also allows the ability to enlarge the map visualization area, if needed. The ability to resize the transit agency pane was implemented using the “resizable” method in the jQuery UI.

3.3.3 Map Tiles

The OTP Analyst Extension originally comes with map tiles from mapquest (*Mapquest undated*) and mapbox. Mapbox does not offer free map tiles anymore and the mapquest tiles URL has been changed, so the client was not able to display map tiles anymore. Both aerial photo and street map layers where replaced with the latest free map tiles from mapquest.

A new set of tiles known as “Toner” by stamen design (*Stamen Design 2011*) was also added to the map. The toner map tiles provide improved visibility of a transit agency characteristics (i.e., stops and routes) for visualization and printing purposes.

3.3.4 Displaying Stops and Route Shapes

As described in section 3.2.2 and section 3.2.4, queries were developed on the server to provide stops and route shapes to the client side of the TNA software tool. For all of the visualization capabilities of the TNA software tool, a JavaScript library named “Leaflet” is used.

In the case of displaying stops, the list of stop names and coordinates are provided. The route shapes are received in encoded polyline form from the server and therefore need to be decoded into coordinates in order to be displayed on the map using Leaflet. Leaflet encoded is a plug-in for leaflet that adds encoding and decoding encoded polyline capabilities to Leaflet.

3.3.5 Clustering Stops

The TNA software tool displays stops as *clusters* that change in size as the level of zooming is changed. This dynamic approach to displaying stop clusters proved to be the most efficient way to display thousands of stops on a map. The software component used to accomplish this was Leaflet markercluster.

3.3.6 Reports

A reporting interface was added to the TNA software tool so that specific characteristics of the transit agencies could be visualized either on the GUI, sent to a printer, or downloaded as a comma-separated value (.csv) file for further analysis. The “Reports” button shown on the upper right corner of Figure 4.1 opens a new page that contains the "Transit Agency Summary Report." This report provides general information on all transit agencies in the state of Oregon as well as access to the "Routes" report, the "Stops" Report, and the "Transit Agency Extended" report. The queries that generate these reports are described in section 3.2.5 through section 3.2.8.

As originally designed, the "Transit Agency Summary" report was supposed to provide complete information on all Oregon transit agencies. However, the original report took a long time to load due to the size of the GTFS feeds. Therefore, the information fields *route miles*, *service miles*, *population served*, *service stops*, and *population served by service* were moved from the "Transit Agency Summary" report into the "Transit Agency Extended" report. Moving the aforementioned information fields to the "Transit Agency Extended" report reduced the "Transit Agency Summary" report's loading time significantly.

All the reports available through the TNA software tool share the same design for consistency. Buttons for printing, closing and exporting reports are located in the same position and order on the GUI. These buttons also trigger the same JavaScript methods that perform the corresponding reporting function.

The jQuery UI Multiselect widget is used on the Transit Agency Extended report and the Routes report to allow for the selection of multiple weekdays, so service stops, population served by service, service miles, and population served by service can be computed for the selected days of the week.

3.3.7 Sorting Reports

The majority of the reports generated by the TNA software tool can be sorted based on a single category (e.g., *transit agency ID* or *total routes*). The "Transit Agency Extended" report cannot be sorted because it always returns information pertaining to a single transit agency.

The JavaScript library *sortable* was used to implement the sorting feature. Sorting a report by a specific information field is accomplished by simply clicking the corresponding table heading.

3.3.8 Displaying County Shapes over the Map

The TNA software tool has the ability to display the shapes of all Oregon counties as a map layer. The Oregon county shapes were obtained from (*TIGER undated*) already converted from *.shp* format (i.e., the native format provided by the U.S. census bureau) into GeoJson format (i.e., format that Leaflet supports). Each county shape includes county name and surface area (in square miles) which are displayed in a L.control Leaflet object that shows up when the mouse is hovered over a county shape on the map.

The county shape layer is styled to be transparent to allow other layers beneath it to remain visible. There are also functions that slightly highlight the county shape and update the data displayed in the box over the map when the mouse is hovered over a county shape. If a county shape is clicked on with the mouse, the map zooms in on the corresponding county.

3.3.9 Report Export Functionality

The report export button on the TNA software tool report interface creates a *comma separated value* (CSV) instance of the report table contents on the client application. Currently, a JavaScript method named *window.open* is used for downloading the data, which does not allow to specify the file name. Therefore, the web browser automatically generates a random string that is used as the file name when the file is downloaded.

3.3.10 Report Print Functionality

The report print button on the TNA software tool report interface uses the *window.print* JavaScript function. The functionality provided by the *window.print* JavaScript function is identical to that experienced when using the internal printing function available on all web browsers. One disadvantage of the *window.print* JavaScript function is that it does not provide a print preview feature. However, most web browsers such as Mozilla Firefox and Google Chrome provide this feature.

3.3.11 Linking Reports

As mentioned earlier, there are four reports available in the current version of the TNA software tool. The "Transit Agency Summary" report is the main report that provides access to the other reports by clicking on either the *agency ID*, *routes* or *stop* numbers.

To implement this feature, a hyperlink tag was added to the corresponding fields in the "Transit Agency Summary" report table. Since these links are not real hyperlinks (i.e., they do not refer to a URL), a JavaScript function was implemented to capture the click events. Two custom attributes *Type* and *ID* were necessary to identify the type of report that needed to be loaded. Therefore, each time a hyperlink is clicked, the JavaScript function decides which type of report to load by looking into the *Type* attribute and then pass the appropriate parameters to the query that generates the report by looking into the *ID* attribute.

3.3.12 Styling Reports

All reports have a tabular design that makes them easy to read. The report tables are designed to extend up to 99% of the page width. Therefore, if they are minimized or displayed on a lower resolution screen (e.g., video projector or computers with smaller displays), the table width will be self-adjusted to prevent horizontal scrolling. To make table rows easier to follow, odd and even rows are colored differently and the entire row is highlighted when the mouse is hovered over the table row. These effects and all other formatting for the reports were implemented using *Cascading Style Sheets (CSS)*.

3.4 OTHER DEVELOPMENT

3.4.1 Acquiring Up-To-Date Oregon Map Data

As explained in section 3.1, OTP relies on an XML-based text file known as the "Graph File" for storing road network data as well as GTFS data. One of the essential components to build a graph file is map data. OpenStreetMap (OSM) is a very common format for distribution of map data (*Wiki undated*).

OSM is a project that aims at providing free editable world map to the public (*OSM undated*). OSM makes their data available to the world in osm format on a daily basis. However, extracting a region's data (e.g., state of Oregon) from a 25-Gyobyte base file is time consuming and requires some tools. An alternative is using the download page of CloudMade (*CloudMade undated*), which contains extracts of maps available for download. Oregon State map data were downloaded from this source to build the graph file.

3.4.2 Building the Graph File

As the development efforts progressed, the OTP graph files needed to be updated since the GTFS feeds for the Oregon transit agencies were updated quite frequently. The first step in building a graph file is data preparation. Acquiring the latest map data was explained in section 3.4.1. Apart from the map data, transit data in the form of GTFS feeds were also required. Before the introduction of the GTFS feed archive tool, collecting all GTFS feeds from the Oregon's GTFS data page (*ODOT GTFS undated*) was a time intensive task. GTFS data came from different sources that had to be collected and organized.

The next step in building the graph file was configuring the graph builder configuration file. The XML configuration file of the graph builder needs references to all data files that are being incorporated into the graph file, as well as a working directory (i.e., where the graph builder can

store temporary files as it processes the data) and output folder (i.e., where the graph builder writes the graph file). More information on configuring the graph builder and possible configurations can be found on OTP's wiki (*GitHub undated i*).

3.4.3 Incorporating Census Data

Census data can be represented in various formats depending on the type of analysis it is used for. The most essential element in census-based reports is finding the number of people served within a given distance of a point or set of points. Census data is collected every 10 years by the U.S. Census Bureau and made available to the public through its web site (*US Census Bureau undated*). The latest available dataset dates back to the year 2010.

Census data plays a key role in the reporting capabilities of the TNA software tool since one of its main requirements was that it reported an accurate population count within an *X*-mile radius of a single stop, a route and a transit agency. Several population distribution methods were explored to provide this capability including *chloroplethic simple weighted average*, *chloroplethic weighted average with decay*, *dasyymmetric mapping*, *discretized grid network with decay*, and *chloroplethic centroid*. Table 3.2 summarizes the salient features of these methods.

Table 3.2: Salient Features of Population Distribution Methods

Method	Description	Pros	Cons
Chloroplethic simple weighted average (<i>Laporte et. al 2002</i>)	Assumes uniform distribution of population over the land area.	Simple and easy to implement.	Not accurate for rural/ non-uniformly populated areas.
Chloroplethic weighted average with decay (<i>Laporte et. al 2002</i>)	Assumes a decay factor in addition to the above method to account for the population decrement as distance from the stop increases.	More accurate than the first method.	Decay function type and parameters have to be determined based on location.
Dasyymmetric mapping (<i>Holt et. al 2004</i>)	Assumes uniform distribution of population over the residential portion of the land area.	More accurate for areas with large non-residential blocks.	Requires land use (residential/non-residential) data.
Discretized grid network with decay (<i>Bruno et. al 2002</i>)	Maps the population to points on a discrete grid based on walking distance from transit stops.	Incorporates decay and walking distance.	Walking distance is not identical for urban and rural areas.
Chloroplethic centroid (<i>Kamatu et. al 2007</i>)	Maps the population to the census block centroids.	Easy to calculate; does not require extra data.	Accuracy varies by different census block sizes.

The population distribution method *chloroplethic centroid* was chosen to incorporate census data into the TNA software tool. In this method, the total population living in a census block (which is the highest resolution of the census data available) is allocated to the centroid of each census

block. In order to map the population of a census block to its centroid, topologically integrated geographic encoding and referencing (TIGER) shape files were also retrieved from the U.S. Census Bureau web site (*TIGER undated*). The census block data and the TIGER shape files were then imported into ArcGIS and latitude/longitude coordinates for every census block centroid were computed and matched with the total population count.

As mentioned before, OTP relies on the "Graph File" as its main data repository. In order to incorporate the census data for each census block centroid into the "Graph File" so that it could be queried to generate reports, a census GTFS feed was created. In the census GTFS feed, each centroid was represented as a regular transit stop. The field "stop_name" was then used to associate the corresponding population count to each block centroid. The census GTFS feed was validated like any other regular GTFS feed with the Google FeedValidator tool (*Google undated c*) before being imported into the "Graph File."

4.0 TRANSIT ANALYSIS NETWORK TOOL QUICK USER GUIDE

The information included in this chapter is intended to quickly familiarize a new user with the features of the "transit network analysis" (TNA) software tool.

4.1 ACCESSING THE TNA SOFTWARE TOOL

The TNA software tool has been developed to work optimally with the Mozilla Firefox and Google Chrome browsers. The TNA software tool can be accessed by typing (or by copying and pasting) the following URL in a browser's address bar:

<http://mime-sim01.engr.oregonstate.edu:8088/TNA-software-tool/>

4.2 INTERFACING WITH THE TNA SOFTWARE TOOL

Figure 4.1 depicts the main graphical user interface (GUI) of the TNA software tool. The GUI includes three main components:

1. A pane that lists all the Oregon transit agencies with an available general transit feed specification (GTFS) feed.
2. A map interface to visualize the routes and stops that compose each transit agency's service area as well as the spatial relationships among these service areas. The map has zooming capabilities to enable different levels of visualization regarding one or multiple transit agencies at a time.
3. A <Reports> button. Clicking on this button opens the "Transit Agency Summary" report from which other reports can be accessed.

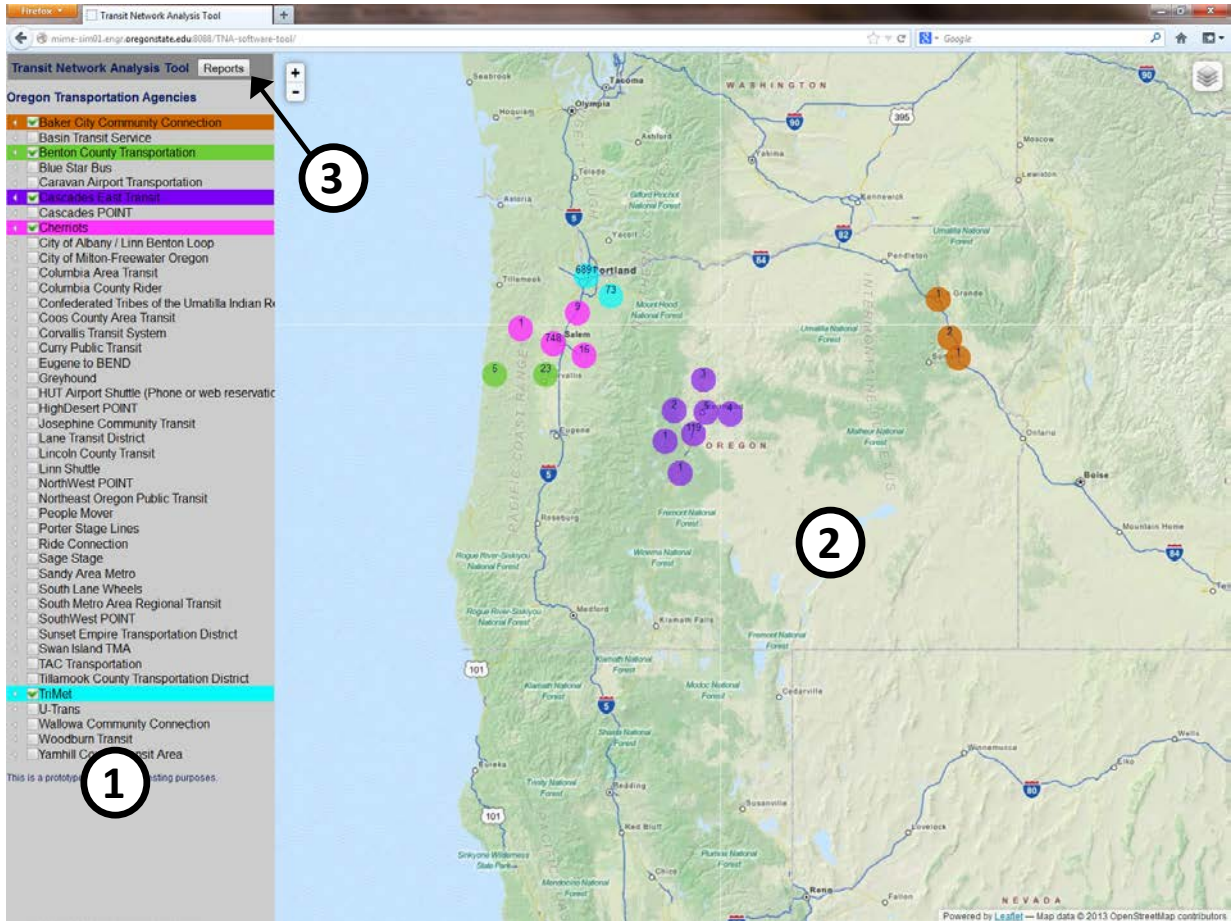


Figure 4.1: The TNA software tool graphical user interface

4.3 THE TRANSIT AGENCY PANE

The following sections describe the main features of the transit agency pane of the TNA software tool.

4.3.1 Pane Width

The width of the transit agency pane is set to a default width when the GUI of the TNA software tool is first loaded, as depicted in Figure 4.2a. The default width can be modified by placing the cursor on the edge between the transit agency pane and the map area and, while holding the left-mouse button, dragging the pane to the desired width. Figure 4.2b depicts a wider transit agency pane. This feature is convenient when the name of the transit agency (or its routes) is too long to fit in the default width of the transit agency pane. Conversely, the width of the transit agency pane can be reduced if a larger map view is desired.

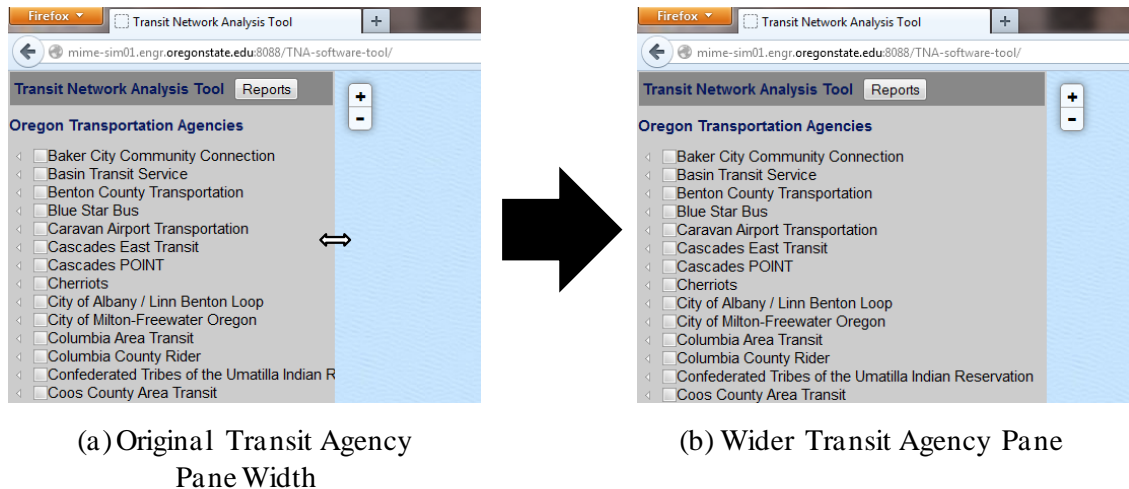


Figure 4.2: Modifying the size of the transit agency pane

4.3.2 Pane Tree Structure

The transit agency pane uses a tree structure to organize the information regarding a transit agency (i.e., routes and route variants), as depicted in Figure 4.3a. A transit agency represents the root of the tree. By clicking on the white arrow icon (◀), the routes operated by the transit agency are revealed. Figure 4.3b shows an example for the South Metro Regional Transit agency, which operates a total of six routes. The last level of the tree structure is shown for the 95th Ave route, which includes two different route variants. If available through the GTFS feed, a route name or a route variant name may display characters enclosed in parentheses at the end of their names representing the "route short name" (e.g., "5" or "101_14572").

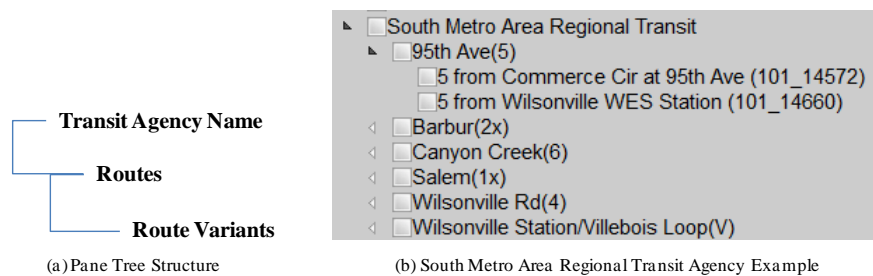


Figure 4.3: Transit agency pane tree structure

The different levels of the transit agency tree structure are collapsed by clicking on their corresponding black arrow icon (▶). *A convenient way to return the TNA software tool to its default form is to simply reload it by pressing the reload symbol at the end of the browser's address bar.*

4.4 SELECTING TRANSIT AGENCIES FOR VISUALIZATION

To select a single transit agency (or multiple transit agencies) for visualization in the map area simply select them by clicking on their corresponding check box, as depicted in Figure 4.4. Transit agencies will be assigned one of six colors (i.e., green, pink, teal, red, purple and brown) as they are selected. The same color assigned to an agency on the transit agency pane will be used to identify its stop clusters on the map.

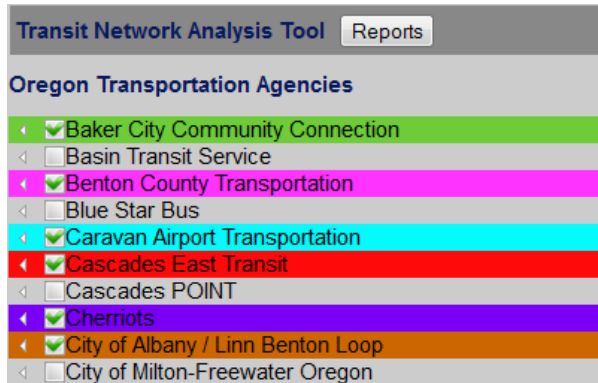


Figure 4.4: Selecting transit agencies for visualization

4.4.1 Stops Clusters

The location and size (i.e., total number of stops) of a transit agency network is represented in the map area in the form of *stops clusters*. The symbol for a stops cluster is a circle of the same color assigned to the transit agency in the transit agency pane. The number shown inside a stops cluster (or the summation of the numbers, if several stops cluster are displayed for a single agency) represents the total number of stops operated by the transit agency.

For example, Figure 4.5a depicts the stops cluster representation for the City of Albany/Linn Benton Loop transit agency when this transit agency is first selected on the transit agency pane. As indicated by the number inside the single stops cluster, this transit agency operates a total of 98 stops. Figure 4.5b depicts the stops cluster representation for the Salem-based transit agency Cherriots when it is first selected on the transit agency pane. In this case, four stops clusters are displayed amounting to a total of 774 stops.

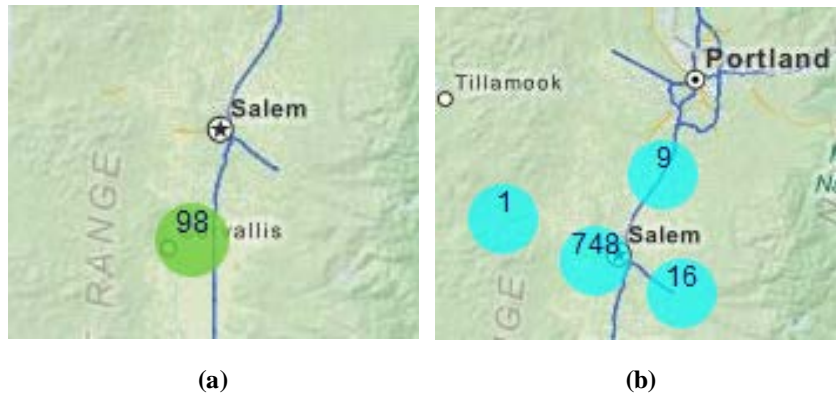


Figure 4.5: Examples of stops clusters for two different transit agencies

4.4.2 Zooming In on Stops

The best way to zoom in on an area serviced by a specific transit agency is by clicking on one of its stops clusters with the mouse pointer. This action will begin to decompose the higher level stops cluster into lower level stops cluster(s). Figure 4.6 illustrates the resulting effect of clicking on the 98-stops cluster of the City of Albany/Linn Benton Loop transit agency. Now, the 98-stops cluster has been decomposed into 27 different stops clusters comprised of anywhere from one to as many as 10 stops. Clicking on stops clusters that contain more than one stop automatically zooms in further on that area of the transit agency's service area.

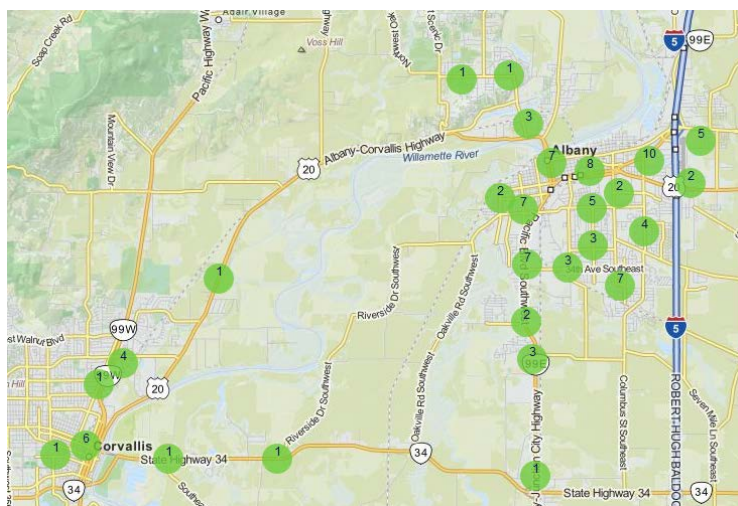


Figure 4.6: Second level decomposition of stops clusters for the City of Albany/Linn Benton Loop Transit Agency

Stops clusters have additional features that enhance the functionality of the TNA software tool. For example, if the mouse pointer is hovered over a stops cluster whose number of stops is larger than two, a blue convex envelope will appear. This blue convex envelope is available at all levels of zooming for those stops clusters that contain more than two stops. The blue convex envelope

represents the smallest convex polygon that includes all the stops serviced by the transit agency. However, it is important to note that the level of zooming may have to be adjusted to completely visualize the blue convex envelope for a stops cluster. Figure 4.7 depicts an example of this feature of the TNA software tool for a 10-stops cluster that belongs to the City of Albany/Linn Benton Loop transit agency.



Figure 4.7: Service area envelope of a 10-stops cluster

Once a stops cluster is at its lowest level of aggregation (i.e., one stop only), a tool tip is available that displays the name of the transit agency that the stop belongs to. The tool tip will appear when the mouse is hovered over the one-stop cluster. Also, by clicking on the one-stop cluster, the location of the stop (and any additional information contained in the GTFS feed about this stop) will be displayed on the map inside a balloon shape. The balloon shape can be closed by clicking on the "X" mark located on its upper right corner. Figure 4.8 shows an exploded view of the 5-stops cluster of the City of Albany/Linn Benton Loop transit agency. The one-stop cluster located along Knox Butte Rd East illustrates the tool tip and the stop information balloon.

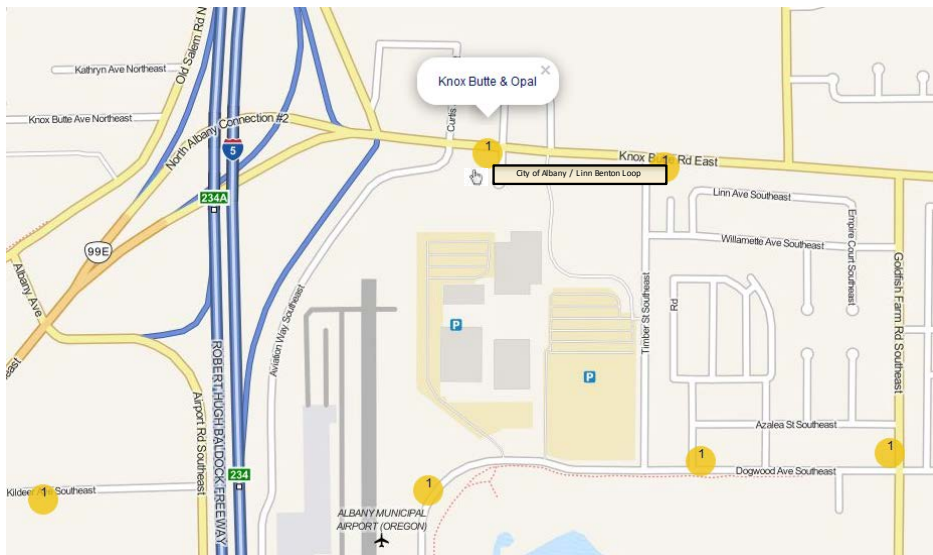


Figure 4.8: Transit agency name tool tip and stop location balloon

4.4.3 Displaying Routes

The TNA software tool provides different options for the visualization of the routes formed by the stops operated by a transit agency. The following sections detail the steps needed to enable each of these visualization options.

4.4.3.1 Displaying a Single Route

The shape of a single route formed by its stops can be visualized in the TNA software tool by following these steps:

1. Reveal the routes operated by the transit agency by clicking on the white arrow next to its name.
2. Identify the route to be displayed on the map for visualization. Reveal its variants by clicking on the white arrow next to its name.
3. Click the check box that identifies the route name. This action will display the location of the stops cluster that corresponds to this route.
4. Click on the stops cluster. This action will zoom in and will display the location of the stops on the route.
5. Click the check box that identifies the route variant's name. This action will draw the shape of the route by joining the stops with a line. To change the color that identifies the route shape, continue to click on the check box until the desired color is used.

Figure 4.9 depicts an example of a portion of a route located in northwest Corvallis. The route is represented by the green line joining the red one-stop clusters on the map. It is important to note, however, that this feature is only available for the routes of those agencies that include a *shapes.txt* file in their GTFS feed¹.

¹ The *shapes.txt* file is **optional** in the GTFS specification.

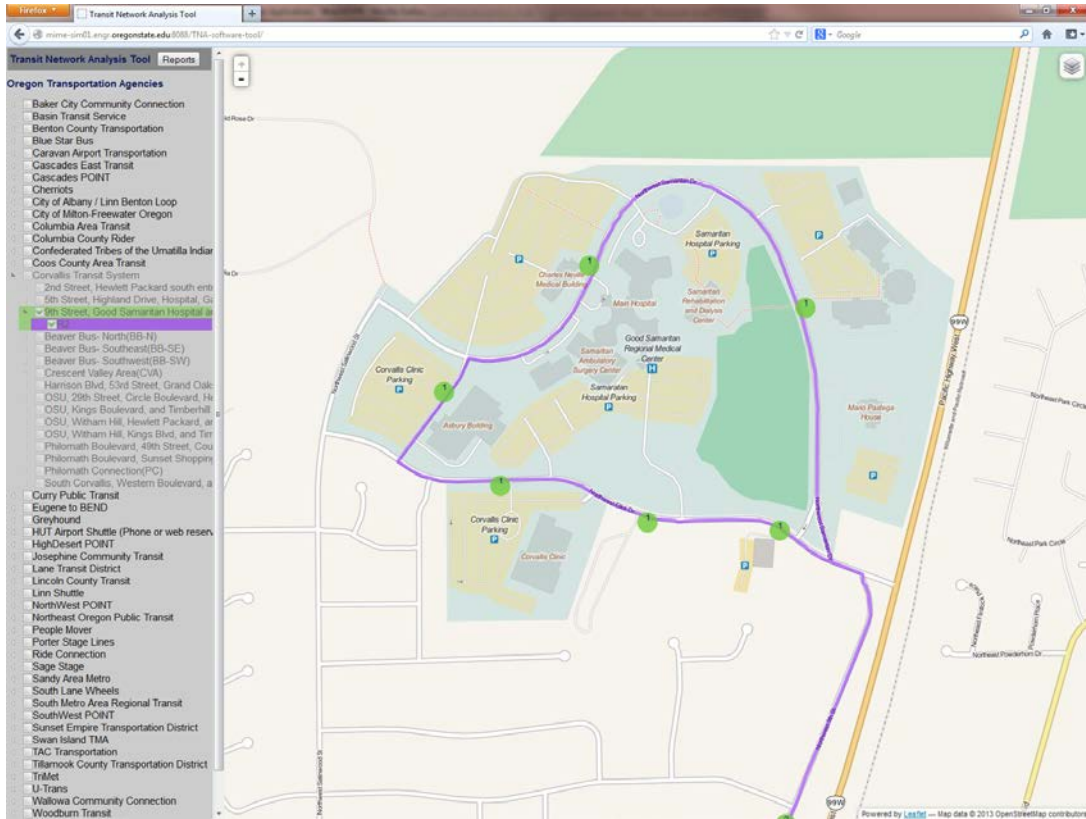


Figure 4.9: Route visualization example for the Corvallis Transit System

4.4.3.2 Displaying Multiple Routes

There are a couple of ways to display multiple routes on the GUI of the TNA software tool. If the user is interested on visualizing specific routes (and route variants) operated by a transit agency, the steps described in section 4.4.3.1 should be followed. The difference is that instead of selecting a single route, multiple routes and routes variants must be selected.

Alternatively, the user can visualize all the route shapes for an agency (with or without displaying the stops clusters) by following these steps:

1. Position the mouse cursor on the transit agency of interest. This action should change the background color of the transit agency's name from gray to light blue, as depicted in Figure 4.10.

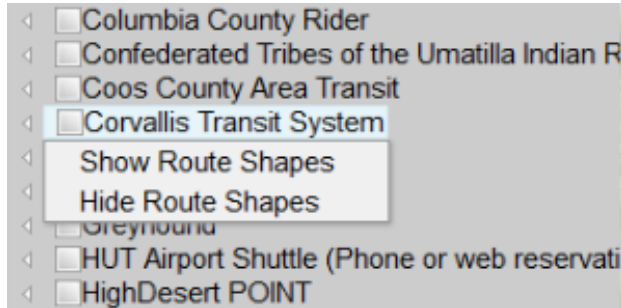


Figure 4.10: Route selection visualization options with a list box

2. Right-mouse click on the transit agency name to reveal a list box with the options “Show Route Shapes” and “Hide Route Shapes.” This list box is also depicted in Figure 4.10.
3. Selecting the option “Show Route Shapes” from the list (without selecting the check box next to the transit agency name) displays all the routes operated by the transit agency on the map, as depicted in Figure 4.11. If the check box next to the transit agency's name is selected, the stops clusters are displayed as well, as depicted in Figure 4.12.

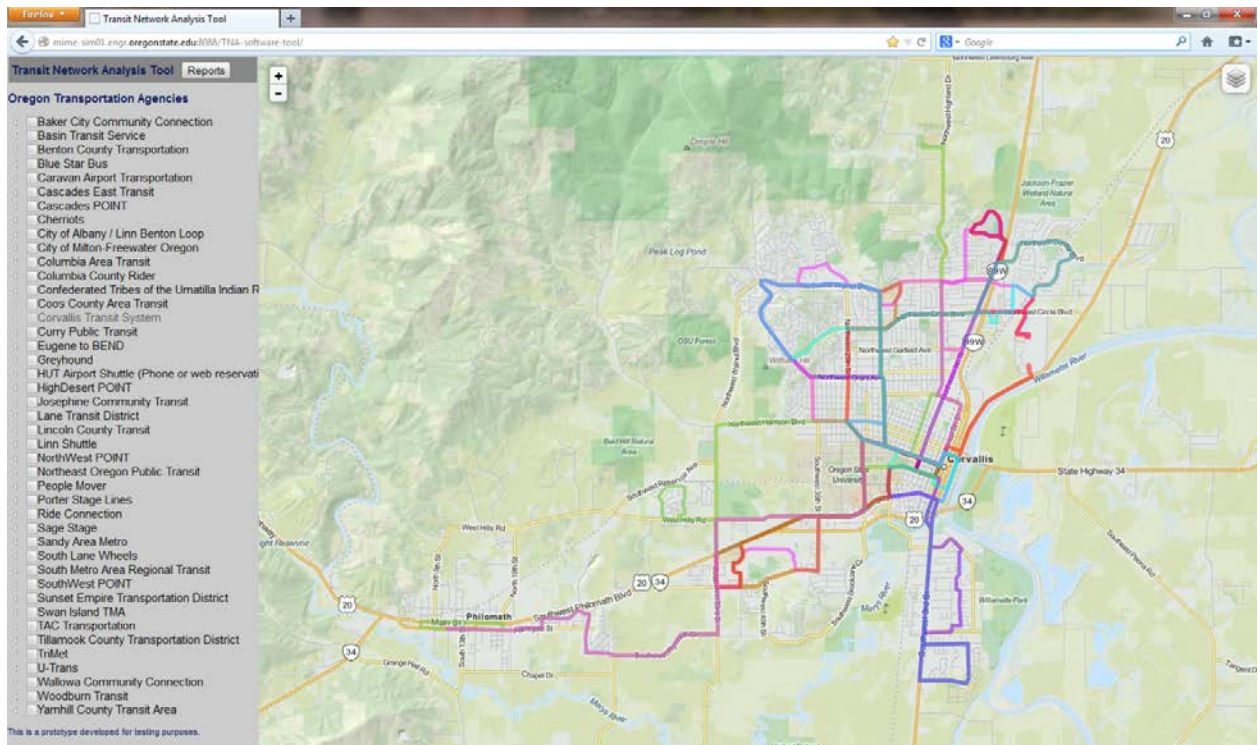


Figure 4.11: Selecting all the route shapes from the Corvallis Transit Agency for visualization

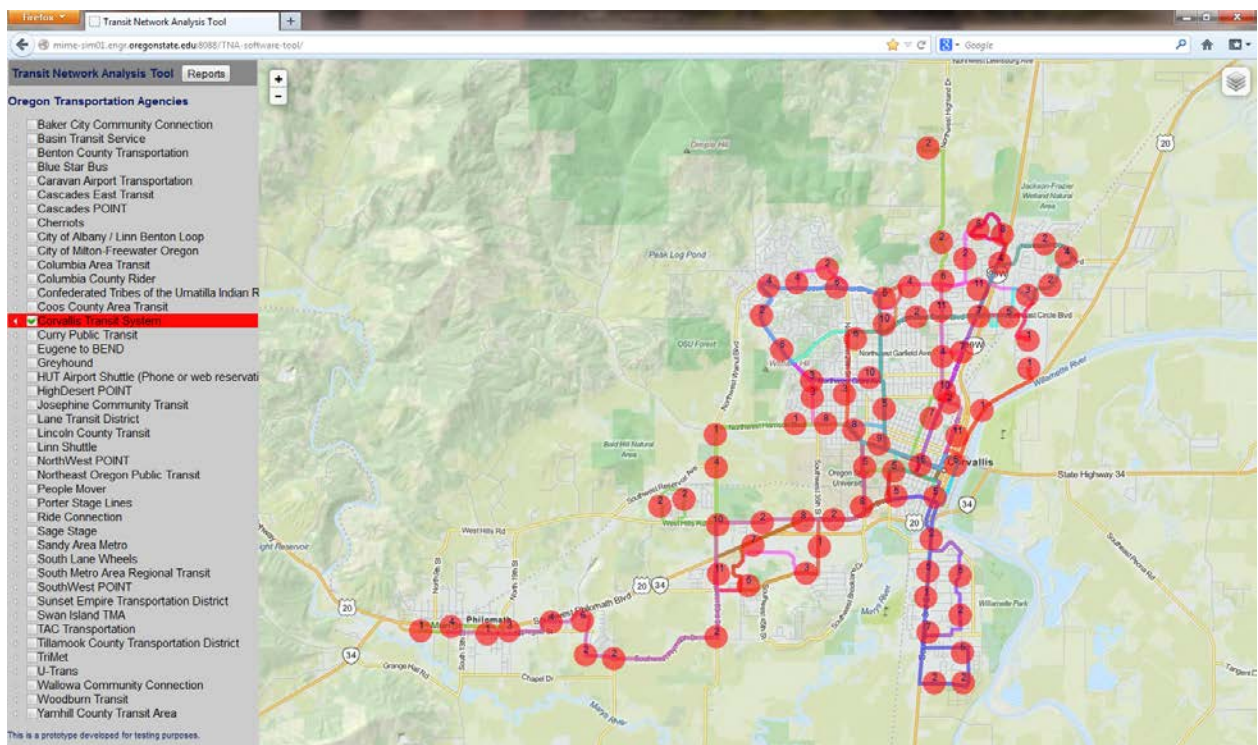


Figure 4.12: Selecting all the route shapes and stops clusters from the Corvallis Transit Agency for visualization

4. It should be noted that if the check box next to the transit agency's name is **not** selected, the color of the font displaying the name of the transit agency changes to a light gray to indicate that some lower component of the tree structure (i.e., routes or route variants) are selected.

4.5 REPORTS

Several reports are available through the TNA software tool by clicking on the **<Reports>** button located on the upper right corner of the transit agency pane. The first report that is presented to the user when the **<Reports>** button is pressed is the "Transit Agency Summary" report. Three additional reports are accessible from the "Transit Agency Summary" report:

1. "Transit Agency Extended" report.
2. "Routes" report.
3. "Stops" report.

The three reports mentioned above can be accessed by clicking on hyperlinked values that correspond to a specific transit agency. For example, Figure 4.13 shows the hyperlinked values for the transit agency "Baker City Community Connection" as displayed in the "Transit Agency Summary" report. To access the "Transit Agency Extended" report for the "Baker City Community Connection" transit agency, the hyperlinked number that represents the "Agency ID" (i.e., 159) must be clicked. Similarly, to access the "Routes" report or the "Stops" report, the hyperlinked numbers displayed in the columns labeled "Total Routes" and "Total Stops" (i.e., 1 and 4, respectively) must be clicked. The hyperlink depicted in Figure 4.13 for the column labeled "Agency Name" opens the transit agency's official web site, whereas the hyperlink for the column labeled "Fare URL" displays a web page with the transit agency's fare information (if available).

Agency ID	Agency Name ▾	Phone #	Total Routes	Total Stops	Population Served	Fare URL
159	Baker City Community Connection	1-800-823-6591	1	4	44	Fares

Figure 4.13: Accessing other reports from the "Transit Agency Summary" report

All the reports available in the TNA software tool can be viewed in a separate browser window. Reports can also be printed as a hard copy or exported as a *comma-separated values* (CSV) file for further processing in other applications such as Microsoft® Excel. To print or to export any of the reports, the user needs to press the buttons **<Export Report>** or **<Print Report>** located on the upper right corner of the browser's window. A third button labeled **<Close Report>** will close the browser window where the report is displayed. The **<Export Report>**, **<Print Report>**, and **<Close Report>** buttons are illustrated in Figure 4.14.

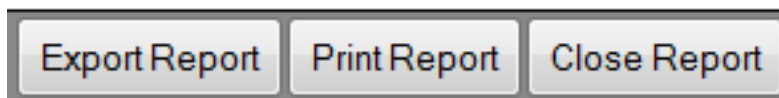


Figure 4.14: Buttons to Allow User to Export, Print or Close a Report

4.5.1 Transit Agency Summary Report

When the <Reports> button is first pressed, the "Transit Agency Summary" report is generated. By default, this report is sorted by "Agency Name" (notice the white arrow pointing down next to this heading). The "Transit Agency Summary" report is composed of the following data elements (see Figure 4.15):

- **Agency ID.** This is the identification number reported in the transit agency's GTFS feed.
- **Agency Name.** This is the agency name reported in the transit agency's GTFS feed.
- **Phone #.** Main phone number to contact the transit agency.
- **Total Routes.** Total number of routes operated by the transit agency.
- **Total Stops.** Total number of stops operated by the transit agency.
- **Population Served.** Summation of the unduplicated population count within an X radius (i.e., stop distance) of each stop operated by the transit agency. The default value for X is 0.1 miles. However, the value of the radius can be changed by the user on the text box shown on the upper right corner of the report and then pressing the <Submit> button.
- **Fare URL.** If available, this field points to the fare information published by the transit agency on its web site.

The "Transit Agency Summary" report can be sorted based on any of the above fields by clicking on the corresponding column heading.

Transit Network Analysis Tool Reports

Export Report Print Report Close Report

Transit Agency Summary Report Population Search Radius (miles) 0.1 Submit

Agency ID	Agency Name *	Phone #	Total Routes	Total Stops	Population Served	Fare URL
159	Baker City Community Connection	1-800-823-6591	1	4	44	Fares
131	Basin Transit Service	(541) 863-2877	3	89	4,577	Fares
148	Benton County Transportation	(541) 754-1748	2	28	1,155	-
169	Blue Star Bus	1-800-247-2272	1	24	4,132	-
58	Caravan Airport Transportation	(541) 994-9645	1	21	1,316	-
61	Cascades East Transit	(541) 365-8680	14	135	5,432	Fares
179	Cascades POINT	(800) 872-7245	1	6	784	-
gherriots-selem-or-us	Cherriots	503-588-2877	82	774	40,129	-
ATS	City of Albany / Linn Benton Loop	(541) 917-7667	5	98	6,884	-
mf	City of Milton-Freewater Oregon	541-938-8243	1	29	1,501	-
37	Columbia Area Transit	(541) 386-4202	2	13	2,059	Fares
57	Columbia County Rider	(503) 366-0159	5	43	2,546	-
ctuir	Confederated Tribes of the Umatilla Indian Reservation	(541) 429-7519	7	71	1,329	-
84	Coos County Area Transit	541-267-7111	5	27	1,947	-
CTS	Corvallis Transit System	(541) 766-6998	16	386	24,270	-
curry	Curry Public Transit	(541) 412-8806	2	11	434	-
187	Eugene to BEND	(541) 382-2151	1	5	637	Fares
158	Greshound	1-800-231-2222	4	23	1,238	-
48	HUT Airport Shuttle (Phone or web reservations required for this service)	(888) 257-0126	1	7	240	-
186	HighDesert POINT	1-800-872-7245	1	6	61	Fares
94	Josephine Community Transit	541-474-5452 x2	2	19	140	-
null	Lane Transit District	null	34	1243	61,362	-
89	Lincoln County Transit	541-265-4900	5	144	3,934	Fares
91	Linn Shuttle	541-367-4775	2	12	641	-
178	NorthWest POINT	(800) 442-4106	1	9	71	Fares
157	Northeast Oregon Public Transit	(541) 936-BUSS	2	5	193	-
59	People Mover	(541) 575-2370	1	10	160	Fares
60	Porter Stage Lines	(541) 269-7183	2	5	758	Fares
133	Ride Connection	(503) 226-0700	2	8	1,105	-
SS	Sage Stage	(530) 233-6410	8	21	0	Fares
86	Sandy Area Metro	503-668-3466	1	46	906	-

Figure 4.15: "Transit Agency Summary" report

4.5.2 Transit Agency Extended Report

The "Transit Agency Extended" report is generated only for a single transit agency. This report was developed to enable the "Transit Agency Summary" report to load more rapidly. The "Transit Agency Extended" report is composed of the following data elements (see Figure 4.16):

- **Agency ID.** This is the identification number reported in the transit agency's GTFS feed.
- **Agency Name.** This is the agency name reported in the transit agency's GTFS feed.
- **Route Miles.** The summation of the lengths of the longest route variant for all routes that the transit agency serves.
- **Route Stops.** Total number of stops served by the transit agency.
- **Stops Per Route Mile.** Route stops divided by Route Miles.
- **Population Served.** The summation of unduplicated population within an *X*-mile radius (i.e., stop distance) of all stops that the transit agency serves.
- **Service Miles.** Total miles driven by a transit agency over all round trips of a route. Service miles may be calculated for a specific day of the week, a combination of days of the week (e.g., Monday and Wednesday) or for a seven day week.
- **Service Stops.** The number of trips scheduled at a stop in a route. The *service stops* for a route is calculated as its stop count multiplied by the number of visits per stop.
- **Population Served By Service.** Total unduplicated population impacted within an *X*-mile radius (i.e., stop distance) of all stops on trips served by the transit agency. *Population served by service* is calculated as *service stops* multiplied by the unduplicated population within an *X*-mile radius (i.e., stop distance) of all stops for every trip.

The "Transit Agency Extended" report includes a combo box labeled "Day of the Week" to allow the user to select a specific day of the week (or a combination of days of the week) for which the parameters *Service Miles*, *Service Stops*, *Population Served By Service*, *Service Stops*, and *Population Served By Service* can be calculated.

The screenshot shows a web browser window with the URL `mime-sin01.engr.oregonstate.edu/0088/TNA-software-tool/3/report.html`. The page title is "Transit Network Analysis Tool Reports". Below the title bar, there are buttons for "Export Report", "Print Report", and "Close Report". The main content area is titled "Transit Agency Extended Report" and includes a "Days of Week" dropdown menu set to "1 day(s) selected" and a "Population Search Radius (miles)" input field set to "0.1". Below this is a table with the following data:

Agency ID	Agency Name	Route Miles	Route Stops	Stops Per Route Mile	Population Served By Route(Unduplicated)	Service Miles	Service Stops	Population Served By Service
61	Cascades East Transit	190.45	135	0.7088	5,432	1,324.85	448	355,084

Figure 4.16: "Transit Agency Extended" report

4.5.3 Routes Report

By default, the "Routes" report is sorted by "Route Name" (notice the white arrow pointing down next to this heading). The "Routes" report is composed of the following data elements:

- **Route ID.** This is the unique route identification number reported in the transit agency's GTFS feed.
- **Route Name.** This is the route's short name as reported in the transit agency's GTFS feed.
- **Route Long Name.** This is the route's long name as reported in the transit agency's GTFS feed.
- **Route Type.** This field describes the type of transportation used on a route, as follows:
 - 0 – Tram, Streetcar, Light rail. Any light rail or street level system within a metropolitan area.
 - 1 – Subway, Metro. Any underground rail system within a metropolitan area.
 - 2 – Rail. Used for intercity or long-distance travel.
 - 3 – Bus. Used for short- and long-distance bus routes.
 - 4 – Ferry. Used for short- and long-distance boat service.
 - 5 – Cable car. Used for street-level cable cars where the cable runs beneath the car.
 - 6 – Gondola, Suspended cable car. Typically used for aerial cable cars where the car is suspended from the cable.
 - 7 – Funicular. Any rail system designed for steep inclines.
- **Route Length.** Length of the longest route variant for the given route.
- **Total Stops.** Total number of stops on the route.
- **Unduplicated Population.** Summation of the unduplicated population count within an X radius (i.e., stop distance) of each stop on a route. The default value for X is 0.1 miles. However, the value of the radius can be changed by the user on the text box shown on the upper right corner of the report and then pressing the <Submit> button.
- **Service Stops.** The number of stops scheduled on all trips in a route. The *service stops* for a route is calculated as its stop count multiplied by the number of visits per stop.
- **Population Served By Route.** Total unduplicated population impacted within an X -mile radius (i.e., stop distance) of all stops on all trips. The *Population Served by Route* for a route is calculated as route *service stops* multiplied by the unduplicated population within an X -mile radius (i.e., stop distance) of all stops on all trips.
- **Service Miles.** Total miles driven by a transit agency over all round trips of a route. Service miles may be calculated for a specific day of the week, a combination of days of the week (e.g., Monday and Wednesday) or for a seven day week.
- **Route Description.** If available, this field contains a description of the route (the value is *null* otherwise).

The "Stops" report includes a combo box labeled "Day of the Week" to allow the user to select a specific day of the week (or a combination of days of the week) for which the parameters *Service Stops*, *Population Served by Service* and *Service Miles* can be calculated.

Figure 4.17 depicts an example of a "Routes" report for the Columbia County Rider transit agency.

Route ID	Route Name	Route Long Name	Route Type	Route Length	Total Stops	Unduplicated Population	Service Stops	Population Served by Service	Service Miles	Route Description
282	null	St Helens/Scappoose-South County Flex Routes	3	25.37	25	1,986	168	283,731	168.51	Our FlexRoute service are routes that have fixed stops and run on a circulatory type schedule. The design of a FlexRoute is that there is extra time built into the schedule to allow the bus to deviate (Flex) a short distance off of its designated route to assist Elderly or Disabled passengers as well as the General Public who may have difficulty getting to a FlexRoute bus stop.
280	null	Westport/Clatskanie to St. Helens	3	84.33	10	286	49	13,549	235.25	null
281	null	Nehalem Valley Fixed Route Service	3	71.68	4	86	14	1,204	143.36	null
279	null	St. Helens-Scappoose to Downtown Portland (TriMet Transit Mall)	3	60.06	8	209	9	1,881	60.06	Transfer to MAX Red Line to Portland Airport
278	null	St Helens/Scappoose to Hillsboro/Beaverton	3	56.69	7	65	60	3,900	340.17	null

Figure 4.17: Routes report for the Columbia County Rider transit agency

4.5.4 Stops Report

The "Stops" report for a transit agency can be accessed in two different ways:

1. From the "Transit Agency Summary" report by clicking on the hyperlink inside the column labeled "Total Stops." This version of the "Stops" report lists all the unique stops operated by the transit agency.
2. From the "Routes" report by clicking on the hyperlink inside the column labeled "Total Stops." This version of the "Stops" report lists the stops operated by the transit agency on that route. A stop on this route can also exist on another route operated by the agency.

By default, the "Stops" report is sorted by "Stop Name" (notice the white arrow pointing down next to this heading). The "Stops" report is composed of the following data elements:

- **Stop ID.** This is the unique stop identification number reported in the transit agency's GTFS feed.
- **Stop Name.** Contains the name of a stop reported in the transit agency's GTFS feed.
- **Routes Stop Belongs To.** Unique route ID (or IDs) that the stop belongs to.
- **Population.** Unduplicated population count within an X-mile radius (i.e., stop distance) of a stop. The default value for X is 0.1 miles. However, the value of the radius can be changed by the user on the text box shown on the upper right corner of the report and then pressing the <Submit> button.

Figure 4.18 depicts an example of a "Stops" report for the Northeast Oregon Public Transit agency.

The screenshot shows a web browser window with the following elements:

- Browser tabs: Transit Network Analysis Tool, Transit Agency Summary Report, Transit Agency Stops Report.
- Address bar: mime-sim01.engr.oregonstate.edu:8088/TNA-software-tool/stopreport.html
- Page title: Transit Network Analysis Tool Reports
- Page content: Northeast Oregon Public Transit Stops Report
- Navigation buttons: Export Report, Print Report, Close Report
- Form element: Population Search Radius (miles) 0.1 Submit

Stop ID	Stop Name	Routes Stop Belongs To	Population
28108	Elgin Depot	951	23
28072	NEOPT Hub	951,950	2
28107	Union Clinic	950	68
28129	Union County Airport	950	0
28106	Union Hotel	950	100

Figure 4.18: "Stops" report for the Northeast Oregon Public Transit agency

5.0 CONCLUSIONS AND FUTURE WORK

The TNA software tool represents a solid foundation to provide transit planners with a broader perspective of how individual transit networks and their interactions may affect a larger state-wide or region-wide transportation network. In particular, it is anticipated that this tool will provide a level of depth and richness of information about Oregon transit service that has not been available before.

However, the process of developing the TNA software tool was not free of challenges. Some of these challenges arose as a result of the requirement to develop the TNA software tool as an open source project (i.e., using OTP as the framework). OTP is a trip planner software tool, which implies that some of the OTP components such as the data structure of the "Graph File" and data retrieval procedures are optimized for finding routes between a source and destination points on a map quickly and efficiently. However, some reporting capabilities required issuing join queries and then filtering transit data in various GTFS feed files, which is basically a feature offered by GIS software packages and can be best implemented by a relational database. This issue was addressed by developing custom procedures to acquire data from the "Graph File" more efficiently.

Having access to good documentation is key in software development. OTP has beginner-level and intermediate-level documentation available to get started. However, there is no detailed documentation available to advanced developers. Also, there are modules in OTP taken from other open source projects such as *OneBusAway* (*OneBusAway undated*) that do not have any source code or documentation attached in OTP repositories. When documentation is lacking, developers have to resort to other types of help available at programming forums and mailing lists making the development process more difficult and time consuming. Finally, OTP server side components are programmed in Java whereas client side components are developed using JavaScript and various JavaScript libraries such as Leaflet and jQuery, which required different programming skills.

Plans for future enhancements of the TNA software tool include:

- Link a relational database to OTP to optimize the computation time and memory requirements needed to display stop and route information and to produce the necessary reports.
- Develop a module to allow users to perform more complex spatial and temporal analyses regarding transit networks such as identification of service gaps and common stops.
- Develop an archive of GTFS feeds so users can analyze how transit networks change over time.
- Incorporate GTFS feeds from other states.

6.0 REFERENCES

- Agafonkin, V. *Leaflet Homepage*. – undated. <http://leafletjs.com/>. Accessed September 3, 2013.
- Bivand, R. *Spatial Data in R*. 2011. <http://rspatial.r-forge.r-project.org/>. Accessed Nov. 27, 2012.
- Bishop, A. *Routino : Router for OpenStreetMap Data*. – undated. <http://www.routino.org/>. Accessed November 27, 2012.
- Bozhanov, I. *jsTree Home Page*. –undated. <http://www.jstree.com/>. Accessed August 26, 2013.
- Bruno, G., M. Gendreau, and G. Laporte. A Heuristic for the Location of a Rapid Transit Line. *Computers & Operations Research*, Vol. 29, No. 1, 2002, pp. 1–12.
- CASA. GMap Creator Homepage. *The Bartlett Centre for Advanced Spatial Analysis*. London, England. – undated. http://www.bartlett.ucl.ac.uk/casa/latest/software/gmap_creator. Accessed November 27, 2012.
- CloudMade. *CloudMade Homepage*. Menlo Park, CA. – undated. <http://cloudmade.com/>. Accessed November 27, 2012.
- clusterPy. *Universidad EAFIT: Research in Spatial Economics*. Medellin, Colombia. – undated. <http://www.rise-group.org/section/Software/clusterPy/>. Accessed November 27, 2012.
- Croizat Homepage. *Geeknet, Inc*. Fairfax County, VA. – undated. <http://croizat.sourceforge.net/>. Accessed November 27, 2012.
- Eclipse Homepage. *The Eclipse Foundation*. Ottawa, Ontario, Canada. –undated. <https://www.eclipse.org/>. Accessed Jan. 30, 2014.
- ECMA. *Introducing JSON Homepage*. *ECMA International*. Geneva, Switzerland. 2013. <http://www.json.org/>. Accessed September 14, 2013.
- ESRI. *ArcGis*. Redlands, CA. – undated. <http://www.esri.com/software/arcgis>. Accessed November 27, 2012.
- Flanagan, D. *JavaScript: The Definitive Guide*. O’Reilly Media, Inc., Sebastopol, 2011.
- GeoDa Center. *GeoDa Center & Affiliated Software*. *GeoDa Center for Geospatial Analysis and Computation*. Arizona State University. Tempe, AZ. – undated-a. <http://geodacenter.asu.edu/software>. Accessed November 27, 2012.

GeoDa Center. *Open GeoDa/GeoDa Center for Geospatial Analysis and Computation*. Arizona State University. Tempe, AZ. –undated-b. <http://geodacenter.asu.edu/ogeoda>. Accessed November 27, 2012.

GeoGrouper. *Universidad EAFIT: Research in Spatial Economics*. Medellín, Colombia. – undated. <http://www.rise-group.org/section/Software/Geogrouper/>. Accessed November 27, 2012.

GitHub, Inc. *OpenTripPlanner Homepage*. San Francisco, CA. –undated-a. <https://github.com/openplans/OpenTripPlanner/wiki>. Accessed October 29, 2012.

GitHub, Inc. *Two Minutes: Minimal Introduction to OTP*. San Francisco, CA. – undated-b. <https://github.com/openplans/OpenTripPlanner/wiki/TwoMinutes>. Accessed July 24, 2013.

GitHub, Inc. *OTP Tutorial*. San Francisco, CA. – undated-c. <https://github.com/openplans/OpenTripPlanner/wiki/Tutorials>. Accessed September 10, 2013.

GitHub, Inc. *Getting Started with OTP in Eclipse*. San Francisco, CA. – undated-d. <https://github.com/openplans/OpenTripPlanner/wiki/GettingStartedEclipse>. Accessed September 10, 2013.

GitHub, Inc. *Leaflet.markercluster: Marker Clustering plugin for Leaflet*. San Francisco, CA. – undated-e. <https://github.com/Leaflet/Leaflet.markercluster>. Accessed September 3, 2013.

GitHub, Inc. *Leaflet.encoded: Plugin to support Google's polyline encoding in Leaflet*. San Francisco, CA. – undated-f. <https://github.com/jieter/Leaflet.encoded>. Accessed August 25, 2013.

GitHub, Inc. *MultipleGraphs*. Open Plans. - undated-g. <https://github.com/openplans/OpenTripPlanner>. Accessed August 30, 2013.

GitHub, Inc. *Extensions to ws/transit/ API #992*. San Francisco, CA. - undated-h. <https://github.com/openplans/OpenTripPlanner/pull/992/files>. Accessed August 30, 2013.

GitHub, Inc. *GraphBuilder*. San Francisco, CA. – undated-i. <https://github.com/openplans/OpenTripPlanner/wiki/GraphBuilder>. Accessed September 4, 2013.

Google, Inc. *OpenTripPlanner Developers Homepage*. Mountain View, CA. – undated-a. <https://groups.google.com/forum/#!forum/opentripplanner-dev>. Accessed July 24, 2013.

Google, Inc. *Encoded Polyline Algorithm Format*. Mountain View, CA. - undated-b. <https://developers.google.com/maps/documentation/utilities/polylinealgorithm>. Accessed August 25, 2013.

- Google, Inc. *FeedValidator*. Mountain View, CA. – undated-c. <http://code.google.com/p/googletransitdatafeed/wiki/FeedValidator>. Accessed July 26, 2013.
- Google, Inc. *Google Earth Homepage*. Mountain View, CA. – undated-d. <http://www.google.com/earth/>. Accessed November 27, 2012.
- googletransitdatafeed, *Homepage*. Google, Inc. Mountain View, CA. – undated. <http://code.google.com/p/googletransitdatafeed/wiki/PublicFeeds>. Accessed July 24, 2013.
- GRASS GIS Homepage. *Open Source Geospatial Foundation (OSGeo)*. Beaverton, OR. – undated. <http://grass.osgeo.org/>. Accessed November 27, 2012.
- HawthsTools. Hawth's Analysis Tools for ArcGIS. – undated. <http://www.spatial ecology.com/htools/tool desc.php>. Accessed November 27, 2012.
- Holt, J.B., C.P. Lo, and T.W. Hodler. Dasymetric Estimation of Population Density and Areal Interpolation of Census Data. *Cartography and Geographic Information Science*. Vol. 31, No. 2, 2004, pp. 103–121.
- Hynds, E. *jQuery UI MultiSelect Widget*. 2010. <http://www.erichynds.com/blog/jquery-ui-multiselect-widget>. Accessed October 14, 2013.
- ILWIS. *ILWIS: The Free, User-Friendly Raster and Vector GIS*. World Institute for Conservation & Environment. – undated. <http://www.ilwis.org/>. Accessed November 27, 2012.
- jQuery. *Java Library Homepage*. The jQuery Foundation. – undated-a. <http://jquery.com/>. Accessed September 1, 2013.
- jQuery *User Interface Homepage*. The jQuery Foundation. –undated-b. <http://jqueryui.com/>. Accessed September 3, 2013.
- Kamatu, J.N., T.N. Hutchins, A.G. Wagner, and J.H. Lambert. Mapping and Accessibility Analysis of Virginia Transit Systems. Presented at the IEEE Systems and Information Engineering Design Symposium, Charlottesville, VA. 2007, pp.1-4.
- Langridge, S. *Sortable: Make All Your Tables Sortable*. 2003. <http://www.kryogenix.org/code/browser/sortable/>. Accessed September 3, 2013.
- Laporte, G., J.A. Mesa, and F.A. Ortega. Locating Stations on Rapid Transit Lines. *Computers & Operations Research*, Vol. 29, No. 6, 2002, pp. 741–759.
- Levine, N. *CrimeStat® III: A Spatial Statistics Program for the Analysis of Crime Incident Locations (v 3.3)*. Ned Levine & Associates, Houston, TX, and the National Institute of Justice, Washington, DC. 2010. <http://www.icpsr.umich.edu/CrimeStat/about.html>. Accessed November 27, 2012.

Lindsay, J. *Whitebox Geospatial Analysis Tools (GAT)*. Centre for Hydrogeomatics. University of Guelph. Ontario, Canada. 2009.
<http://www.uoguelph.ca/~hydrogeo/Whitebox/index.html>. Accessed November 27, 2012.

Lucas-Smith, M. and S. Nuttall. *CycleStreets: UK-wide Cycle Journey Planner and Photomap*. undated. <http://www.cyclestreets.net/>. Accessed November 27, 2012.

MapQuest, Inc. *MapQuest-OSM Tiles + MapQuest Open Aerial Tiles*. Denver, CO. – undated. <http://developer.mapquest.com/web/products/open/map>. Accessed September 3, 2013.

McGarigal, K., B. Marks, E. Ene, Chris Holmes, S.A. Cushman, and M. Neel. *FRAGSTATS: Spatial Pattern Analysis Program for Categorical Maps*. Department of Environmental Conservation, University of Massachusetts. 2012.
<http://www.umass.edu/landeco/research/fragstats/fragstats.html>. Accessed November 27, 2012.

Minerva. *Minerva Homepage*. <http://www.minerva-gis.org/>. Accessed November 27, 2012
 OneBusAway Project. OneBusAway Homepage. – undated. <http://onebusaway.org/>. Accessed July 24, 2013.

ODOT *Public Transit GTFS Feeds Homepage*. – undated. Trillium Solutions, Inc. Portland, OR. <http://www.oregon-gtfs.com/>. Accessed July 24, 2013.

OpenRouteService. *OpenRouteService*. OpenStreetMap Foundation. England. – undated. <http://wiki.openstreetmap.org/wiki/OpenRouteService>. Accessed November 27, 2012.

OpenStreetRouting. OpenStreetMap Wiki contributors, *OpenStreetRouting*, *OpenStreetMap Wiki*, England, - undated.
<http://wiki.openstreetmap.org/w/index.php?title=OpenStreetRouting&oldid=250480>. Accessed November 27, 2012.

OpenTripPlanner. *OpenTripPlanner Homepage*. – undated. <http://www.opentripplanner.org/>. Accessed November 27, 2012.

OSM. *OpenStreetMap Homepage*. OpenStreetMap Foundation. England. – undated. <http://www.openstreetmap.org>. Accessed January 30, 2014.

OSRM. *Open Source Routing Machine (OSRM)*. – undated. <http://project-osrm.org/>. Accessed November 27, 2012.

PySAL. *Pysal v1.7.0 Reference Guide*. – undated. <http://pysal.readthedocs.org/en/v1.7/>. Accessed November 27, 2012.

QGIS Homepage. *QGIS Project*. – undated. <http://www.qgis.org/en/site/>. Accessed November 27, 2012.

REGAL Homepage. *Regional Analysis Laboratory*. Department of Geography, San Diego State University. – undated. <http://regionalanalysislab.org/>. Accessed November 27, 2012.

Roth, M. *How Google and Portland's TriMet Set the Standard for Open Transit Data*. Streetsblog SF. 2010. <http://sf.streetsblog.org/2010/01/05/how-google-and-portlands-trimet-set-the-standard-for-open-transit-data/>. Accessed July 26, 2013.

SAGA. *SAGA: System for Automated Geoscientific Analyses*. Hamburg, Germany. – undated. <http://www.saga-gis.org/en/index.html>. Accessed November 27, 2012.

SANET Team. *SANET. A Spatial Analysis along Networks (Ver.4.1)*. Tokyo, Japan. – undated. <http://sanet.csis.u-tokyo.ac.jp/>. Accessed November 27, 2012.

SaTScan™ Homepage. *SaTScan*. Boston, MA. - undated. <http://www.satscan.org/>. Accessed November 27, 2012.

Schneider, W. and S. Rezić. *BBBike: Your Cycle Route Planner*. – undated. <http://www.bbbike.org/en/>. Accessed November 27, 2012.

Stamen Design. *Dotspotting's Toner Cartography Download Page*. San Francisco, CA. 2011. http://content.stamen.com/dotspotting_toner_cartography_available_for_download. Accessed September 3, 2013.

Tiger. *TIGER Products*. US Census Bureau. Washington, D.C. – undated. <http://www.census.gov/geo/maps-data/data/tiger.html>. Accessed September 3, 2013.

US Census Bureau. *US Census Bureau Homepage*. Washington, D.C. – undated. <http://www.census.gov/>. Accessed July 26, 2013.

W3C®. *World Wide Web Consortium Homepage*. Cambridge, MA. – undated-a. <http://www.w3.org/html/>. Accessed January 30, 2014.

W3C®. *Cascading Style Sheets Home Page*. Cambridge, MA. – undated-b. <http://www.w3.org/Style/CSS/Overview.en.html>. Accessed January 30, 2014.

Wiki. *WhereAmI/OSM Binary Format*. *OpenStreetMap Foundation*. England. – undated. http://wiki.openstreetmap.org/wiki/OSM_Binary_Format. Accessed September 3, 2013.

Xie, F., and D. Levinson. *Evolving Transportation Networks*. Springer, New York, 2011.

YOURS. *OpenStreetMap Wiki contributors, "YOURS," OpenStreetMap Wiki*. – undated. <http://wiki.openstreetmap.org/w/index.php?title=YOURS&oldid=1009353>. Accessed November 27, 2012.

**APPENDIX A: LIST OF REVIEWED COMMERCIALY AVAILABLE TNA
SOFTWARE TOOLS**

Package	Free?	Operating Systems	Developer institution/person	Field	Main Features	Platform	License Type	Citation	Visualization		
									open source	GTFIS	Yes
ArcGIS	No	Linux, Windows, Unix, iOS, Android, Windows Phone	Esri	GIS	In addition to spatial data editing and visualization, ArcGIS provides spatial analysis and modeling features including overlay, surface, proximity, suitability, and network analysis, as well as interpolation analysis and other geostatistical modeling techniques.	Python, Web API, .NET	Proprietary	(<i>ESRI undated</i>)	No	Yes	Yes
ClusterPy (non-GUI, open source)	yes	Linux, MAC OS, Windows	RiSE group	Spatial clustering	Library of algorithms to aggregate areas into regions, where each region is geographically connected, while optimizing a predefined objective function.	Python	BSD license	(<i>clusterPy undated</i>)	Yes	No	No
GeoGrouper (GUI, free)								(<i>GeoGrouper undated</i>)	Yes	No	No
R-Analysis of Spatial Data	yes	Linux, MAC OS, Windows	Roger Bivand (maintainer)	Analysis	Full integration of spatial data analysis tools with the R: classes for spatial data; handling spatial data; reading and writing spatial data; point pattern analysis; geostatistics; disease mapping and areal data analysis; spatial regression and ecological analysis.	R	GPL-2	(<i>Bivand 2011</i>)	Yes	No	No
Google Earth	yes (Pro version is \$400, free to educators)	Linux, MAC OS, Windows	Google	3D visualization	Easy to use, dynamic graphics, historical maps, users can create dynamic "tours," integrated with the Web.	-	Freeware	(<i>Google undated d</i>)	No	No	Yes
PySAL	yes	Linux, MAC OS, Windows	GeoDa Center	Analysis	Modules for: computational geometry, spatial weights, ESDA, inequality, mobility and spatial econometrics	Python	BSD license	(<i>PySAL undated</i>)	Yes	No	No

Package	Free?	Operating Systems	Developer institution/ person	Field	Main Features	Platform	License Type	Citation	Visualization		
									open source	GTFS	
Minerva	yes	Linux, MAC OS, Windows	Fulton High Performance Computing Initiative (Arizona State University)	Visualization (3D)	High performance; ability to display large amounts of raster and vector from multiple sources	C++	BSD license	(Minerva undated)	Yes	No	Yes
GMap Creator	No	Linux, MAC OS, Windows	CASA	Web Mapping	Thematic mapping. Creates image pictures from shapefiles and creates Google Maps websites with the data linked to the shapefile	Java	UCLB Licence CGM-C 1.00	(CASA undated)	No	No	Yes
DynTM	yes	Linux, MAC OS, Windows	GeoDa Center	Web mapping	Scalable Tile Map Service that distributes dynamic choropleth maps in real time through a new caching methodology. It is implemented to run on the Amazon Cloud.	Python		(GeoDa undated a)	Yes	No	Yes
Quantum GIS	yes	Linux, MAC OS, Windows	QGIS Development Team	Visualization	Easy to use, ability to expand functionality with Python plugins. Geoprocessing functions included.	C++	GPL	(QGIS undated)	Yes	No	Yes
GRASS	yes	Linux, MAC OS, Windows	GRASS Development Team	GIS	Extensive set of GIS tools for both raster and vector data, SQL, visualization, voxel	C, C++, Python, Tcl	GPL	(GRASS GIS undated)	Yes	No	Yes
Open GeoDa	yes	Linux, MAC OS, Windows	GeoDa Center	Analysis	Interactive exploratory analysis, dynamic linking and brushing for cross-section data, basic spatial regression capabilities. Cross-platform version of Legacy GeoDa	C++	GPL	(GeoDa undated b) (51)	Yes	No	Yes

Package	Free?	Operating Systems	Developer institution/ person	Field	Main Features	Platform	License Type	Citation	Visualization		
									open source	GTFS	
STARS	yes	Linux, MAC OS, Windows	Regal Analysis Laboratory	Analysis	ESTDA (Exploratory Space-Time Data Analysis). Interactive linking, brushing and travelling for space-time data	Python	GPL	(Regal Undated)	Yes	No	No
GeoDaSpace	yes	MAC OS, Windows	GeoDa Center	Spatial Econometrics	Advanced spatial models for cross section (GMM, IV and spatial HAC)	Python	Freeware	(GeoDa undated a)	NA	No	No
GeoDaNet	yes	MAC OS, Windows	GeoDa Center	Point patterns in networks	Network distances, basic network statistics, global and local K-functions in networks.	Python	Under development	(GeoDa undated a)	NA	No	No
SANET	yes	Windows	Atsu Okabe	Point patterns in networks	Analysis of events that occur on networks or alongside networks. Includes 14 tools for point patterns in networks	-	Freeware (terms)	(SANET undated)	No	No	No
CrimeStat	yes	Windows	Ned Levine and Associates	Spatial analysis of points, zones, and lines	Basic spatial description, hot spot analysis, kernel density interpolation, space-time interaction, journey-to-crime analysis (Geographic profiling), crime travel demand modeling, and spatial regression.	-	Freeware	(Levine 2010)	No	No	No
SaTScan	yes	Windows, Linux, MAC OS	Martin Kulldorff and Information Management Services Inc.	Epidemiology	Spatial, temporal and Space-Time Scan Statistics	-	Freeware	(SaTScan undated)	No	No	No
Croizat	yes	Linux, MAC OS, Windows	Mauro J. Cavalcanti	Panbiogeography	Ability to integrate and analyze spatial data on species or other taxa and to explore geographical patterns in diversity under a panbiogeographic and graph-theoretic approach	Python	GPL	(Croizat undated)	Yes	No	Yes

Package	Free? yes (Requires ArcGIS, which is not free)	Operating Systems	Developer institution/ person	Field	Main Features	Platform	License Type	Citation	open source	GTFS	Visualization
Hawths Tools	yes (Requires ArcGIS, which is not free)	Windows	Hawthorne Beyer	Ecology	Analysis, sampling, animal movements, kernel, raster, table, vector editing, specialist and other tools.	-	Freeware	(<i>HawthsTools undated</i>)	No	No	No
FRAGSTATS	yes	Windows	Dr. Kevin McGarigal	Landscape metrics	Area, density, edge metrics. Shape metrics (Fractal dimension index) and much more.	Visual C++	Freeware	(<i>McGarigal et. al 2012</i>)	No	No	No
SAGA	yes	Windows, Linux	Institute of Geography at the University of Hamburg, Germany	GIS, Analysis	Grid analysis, Geostatistics, Terrain Analysis, Hydrology simulation, TIN tools.	C++	OpenSource	(<i>SAGA undated</i>)	Yes	No	Yes
Whitebox GAT	yes	Windows	John Lindsay - Centre for Hydrogeomatics, University of Guelph, Canada	Remote sensing & GIS	GIS analysis, hydrological tools, image processing tools, LiDAR tools, statistical analysis, stream network analysis, terrain analysis	.NET framework 3.5	GPL	(<i>Lindsay 2009</i>)	Yes	No	Yes

Package	Free?	Operating Systems	Developer institution/ person	Field	Main Features	Platform	License Type	Citation	Visualization		
									GTFS	open source	Visualization
ILWIS Open	yes	Windows	ITC - Netherlands	Remote sensing & GIS	Integrated raster and vector design; import and export of widely used data formats; on-screen and tablet digitizing; image processing tools; orthophoto, image georeferencing, transformation and mosaicing; Advanced modeling and spatial data analysis; 3D visualization; projection and coordinate system library; geo-statistical analyses; production and visualization of stereo image pairs; spatial Multiple Criteria Evaluation; WMS; Surface Energy Balances.	-	GPL	(<i>ILWIS undated</i>)	Yes	No	Yes
Open Trip Planner	Yes	Windows, Linux	TriMet, OpenPlans, and the developers of FivePoints, OneBusAway and Graphserver	Multi modal trip planner	Plans multi-modal walking, biking and transit trips, Imports data from GTFS, shapefiles, OpenStreetMap and the National Elevation Dataset, Exposes a RESTful API (XML and JSON), which other apps or front-ends can build on	java	LGPL	(<i>OpenTripPlanner undated</i>)	Yes	Yes	Yes
OpenRouteService	NA	NA	Pascal Neis	Routing/planning	Location Based Services : Route Service, Geocoder/Reverse Geocoder, Directory Service, Accessibility Analysis Service, Emergency Route Service	NA	Creative Commons Attribution-ShareAlike 2.0 license	(<i>OpenRouteService undated</i>)	Routing	No	Yes

Package	Free?	Operating Systems	Developer institution/ person	Field	Main Features	Platform	License Type	Citation	Visualization		
									open source	GTFS	Yes
YOURS	Yes	Linux	NA	Routing/planning	Routing, unlimited via points, GPX route file output	Gosmore, Namefinder, Route altitude profiles SRTM, OpenLayers, RouteInstructions	BSD	(<i>YOURS undated</i>)	Yes	No	Yes
Cloudmade	NA	NA	CloudMade	Routing/planning	Routing, GPX export	Cloudmade Webmap Studio	NA	(<i>CloudMade undated</i>)	No	No	Yes
Routino	Yes	linux	NA	Routing/planning	Routing	C, JavaScript	Affero GPLv3	(<i>Bishop undated</i>)	Yes	No	Yes
BBBike @ World	NA	NA	Wolfram Schneider & Slaven Rezić	Routing/planning	Bicycle routing	NA	NA	(<i>Schneider and Rezić undated</i>)	Yes	No	Yes
OSRM	Yes	linux ; macos ; FreeBSD	Dennis Luxen	Routing/planning	Very fast routing, Rudimentary driving instructions, Highly portable, Simple data format makes it easy to import custom data sets instead of OpenStreetMap data, Making use of turn restrictions	C++	AGPL	(<i>OSRM undated</i>)	Yes	No	Yes

Package	Free?	Operating Systems	Developer institution/person	Field	Main Features	Platform	License Type	Citation	Visualization	GTFS	open source
									Yes	No	No
CycleStreets	Only Client Apps	NA	CycleStreets	Routing/planning	cycle journey planner, Export of planned routes in KML and GPX, Photos geolocated along a planned route are shown on the route	Python, PHP, OpenLayers and NameFinder.	proprietary	(Lucas-Smith and Nuttall undated)	No	No	No
OpenStreetRouting	NA	NA	Awi	Routing/planning	simple online routing app	PyrouteLib 2.	NA	(OpenStreetRouting undated)	NA	NA	NA

APPENDIX B: GLOSSARY OF TERMS

Area Served

For a stop, this is the area (in square miles) within an *X*-mile radius of a stop.

For a route, this is the sum of the non-overlapping areas (in square miles) within an *X*-mile radius of all stops on a route.

For a transit agency, this is the sum of the non-overlapping areas (in square miles) served by all its routes.

Service stops

The number of trips scheduled at a stop in a route². The service stops for a route is calculated as its stop count multiplied by the number of visits per stop.

Example: A single route with 12 stops and two visits per stop would have a service stops of $12 * 2 = 24$.

Population served by service (see service stops)

Total unduplicated population impacted within an *X*-mile radius (i.e., stop distance) of all stops on a trip or trip segment specified.

Route population served by service is calculated as route service stops multiplied by the unduplicated population within an *X*-mile radius (i.e., stop distance) of all stops on a trip or trip segment specified.

Transit agency population served by service is the sum of population served by service over all routes.

Example: A route with 12 stops, two visits per stop, and an unduplicated population of 3,400 within 0.25 miles of all stops would have a population served by service of
 $12 * 2 * 3400 = 81,600$.

Region

A geographical boundary that defines a state, county, census tract, census block, census place, state congressional district, or federal congressional district used for calculating performance metrics.

Round Trip

A two-way pass of all stops in a route, accounting for *both* outbound and inbound travel.

Route

A collection of ordered stops presented publicly as a "line" with a name [2].

² Not a measure of actual times a bus stops, but a measure of possible stops.

Service Miles Per Day

Total miles driven by a transit agency in a day over all round trips of a route. Service miles may be calculated based on a seven day week as well.

Example: An agency has three routes that measure 10 miles, 12 miles, and 25 miles in length. Each route has one round trip per day. The total service miles per day for this agency is $(10 + 12 + 25) * 2 = 94$ miles.

Stop

Designated location on a route where transit vehicles pick up customers.

Trip

A one-way pass of all stops in a route, accounting for *either* outbound or inbound travel.

Trips Per Day

Number of trips in a single day for a given route.

Unduplicated Stop Population

Unduplicated population count within an X -mile radius (i.e., stop distance) of a stop. Population information is taken from 2010 census block data from the U.S. Census Bureau [1].

To calculate the unduplicated stop population, the population of a census block will be assumed to be concentrated at the geographic centroid of the census block shape. Then, the population within X miles of a stop will be calculated as the sum of the population of the census blocks whose centroids are contained within the area defined by the X -mile radius from the stop.

Visits per stop

Number of trips per stop on a route.