DEPARTMENT OF TRANSPORTATION

Enhanced Capabilities of BullReporter and BullConverter

Taek M. Kwon, Principal Investigator
Department of Electrical Engineering
University of Minnesota Duluth

September 2017

Research Project
Final Report 2017-34

To request this document in an alternative format, such as braille or large print, call 651-366-4718 or 1-800-657-3774 (Greater Minnesota) or email your request to ADArequest.dot@state.mn.us. Please request at least one week in advance.

Technical Report Documentation Page

1. Report No. MN/RC 2017-34	2.	3. Recipients Accession No.
4. Title and Subtitle Enhanced Capabilites of BullReporter and BullConverter		5. Report Date September 2017
		6.
7. Author(s) Taek M. Kwon		8. Performing Organization Report No.
9. Performing Organization Name and Address Department of Electrical Engineering University of Minnesota Duluth 1023 University Drive Duluth, MN 55812		10. Project/Task/Work Unit No. CTS #2016012
		11. Contract (C) or Grant (G) No. (C) 99008 (wo) 202
12. Sponsoring Organization Name and Address Minnesota Department of Transportation Research Services & Library 395 John Ireland Boulevard, MS 330 St. Paul, Minnesota 55155-1899		13. Type of Report and Period Covered Final Report
		14. Sponsoring Agency Code
15. Supplementary Notes http:// mndot.gov/research/reports/2017/201734.pdf		

16. Abstract (Limit: 250 words)

Bull-Converter/Reporter is a software stack for Weigh-In-Motion (WIM) data analysis and reporting tools developed by the University of Minnesota Duluth for the Minnesota Department of Transportation (MnDOT) to resolve problems associated with deployment of multi-vendor WIM systems in a statewide network. These data tools have been used by the MnDOT Office of Transportation System Management (OTSM) since their initial delivery in 2009. The objective of this project was to expand the current conversion capabilities of BullConverter to include more raw data formats from different companies and the current BullReporter functions to include new analysis and reporting capabilities. Data analysis needs change over time, and the members of the OTSM WIM section identified several new functions that would increase efficiency and improve quality of WIM data. This report describes the new reporting and conversion functions implemented in this project.

17. Document Analysis/Descriptors Weigh in motion, Histograms, Data analysis, Data fusion		18. Availability Statement No restrictions. Document available from: National Technical Information Services, Alexandria, Virginia 22312	
19. Security Class (this report) Unclassified	20. Security Class (this page) Unclassified	21. No. of Pages 60	22. Price

Enhanced Capabilities of BullReporter and BullConverter

FINAL REPORT

Prepared by:

Taek M. Kwon
Department of Electrical Engineering
University of Minnesota Duluth

September 2017

Published by:

Minnesota Department of Transportation
Research Services & Library
395 John Ireland Boulevard, MS 330
St. Paul, Minnesota 55155-1899

ACKNOWLEDGMENTS

The author would like to express sincere thanks to the members of the MnDOT WIM group, Ben Timerson, Josh Kuhn, Greg Wentz, and Ian Vaagenes. All of them voluntarily served on the Technical Advisory Panel (TAP) of this project and provided assistance from inception of the original project ideas through the many software tests during the stages of implementation. Ben Timerson, technical liaison of this project, provided effective leadership in guiding this project to a successful completion. Josh Kuhn installed and maintained the newly written software for Kistler WIM data archiving program at MnDOT. Greg Wentz suggested how to represent vehicles with more than 12 axles on Bull-CSV files and provided software function tests on new BullConverter and BullReporter implementations. Ian Vaagenes provided several inputs following trials of many new software functions; in particular, he contributed in discovering bugs on Export tab in BullConverter. Last but not least, Nelson Cruz served as project coordinator and provided assistance related to administrative procedures.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

EXECUTIVE SUMMARY

The Bull-Converter/Reporter software stack was developed by the University of Minnesota Duluth (UMD) and the Minnesota Department of Transportation (MnDOT) to solve a heterogeneous data problem associated with multi-vendor systems deployed in a statewide network, and the software stack has been successfully deployed at MnDOT as a weigh-in-motion (WIM) data analysis and reporting tool since release of the first version in 2009. The objective of this project was to expand the current BullConverter functions to include more vendor data formats and the current BullReporter functions to include newly-identified retrieval and statistical functions. The project was divided into six tasks in which the last two are writing a final report.

The MnDOT WIM group was exploring and evaluating new company WIM sensors and controller combinations, e.g., Intercomp sensors on Intercomp controller, Intercomp sensors on International Road Dynamics (IRD) controller, Kistler sensors on Kistler controller, Kistler sensors on IRD controller, etc., through another University project entitled "Weigh-in-Motion Sensor and Controller Operation and Performance Comparison" in 2015. This required integration of two more data formats to be converted to Bull-CSV in BullConverter and writing the conversion routines was assigned as Task-1. For integration of Kistler controllers, a new software downloading tool called KisDA (**Kis**tler **D**ata-logger **A**rchiving) was developed and installed at MnDOT to capture daily raw WIM data from Kistler DL. A new tab was implemented in BullConverter to specifically translate Kistler archived files to Bull-CSV files. Another new tab was also implemented for translation of Intercomp WIMLOGIX files to Bull-CSV files.

A data retrieval function called "View Vehicles" is a new concept for retrieving WIM data. This function allows users to see individual vehicle records along with a graphical representation and a digital photo of the vehicle captured at the crossing of WIM sensors. This type of report helps visual identification and verification of WIM data such as classification errors and abnormal axle configurations. MnDOT asked the UMD research team to implement this new function in BullReporter, and it became Task-2. This function was successfully implemented and tested with user selectable capabilities of complex retrieval conditions. User selectable conditions for retrieving vehicle records include lane numbers, hour ranges, class numbers in multiple ranges, classification scheme, gross vehicle weight (GVW) range, speed range, steer axle weight range, and individual control of error/warning flags. Outputs are provided in two formats, pdf and webpage. Each thumbnail in the webpage is hyperlinked to a full-sized digital photo to allow viewing of the full-sized photo using a simple click of the mouse.

There were two computational functions added to BullReporter in this project, GVW and equivalent single axle load (ESAL) histograms, and its implementation was assigned as Task-3. The user settings for vehicle filtering in GVW and ESAL histograms were designed to share with the View Vehicles report, but one main difference was that it offered three different choices of data period selections, i.e., users could select to produce one histogram between the start and end dates and times, or a day-by-day option in which one histogram per day is produced, or a month-by-month option in which one histogram per month is produced.

Few miscellaneous improvements were assigned as Task-4. It included treatment of vehicles with more than 12 axles on 12 column weight representation in Bull-CSV and respective modifications in BullConveter. Task-4 also included a requirement to provide a document and training of editing procedures of the classification schemes used in BullConverter and BullReporter.

Overall, the project was successfully completed, delivering all of the required deliverables in each task. With this project, the Bull-Converter/Reporter software package became more powerful and complete, which in turn will provide efficiency and better quality for WIM data analysis.

CHAPTER 1: INTRODUCTION

1.1 BACKGROUND

BullReporter is a Weigh-in-Motion (WIM) data analysis and reporting software tool developed for the Minnesota Department of Transportation (MnDOT) by the University of Minnesota Duluth (UMD), and it is published on a UMD web server as Microsoft Installer (MSI) packages. It is part of the Bull-Converter/Reporter software stack, which is designed and implemented to solve incompatibility issues caused by deployment of multi-vendor WIM systems in a statewide network. This tool has been used by the Office of Transportation System Management (OTSM) at MnDOT in the St. Paul central office since its first version become available in 2009. The objective of this project was first to expand the existing conversion functions of BullConverter to include more vendor WIM formats and then to add newly-identified reporting functions in BullReporter.

A brief historical background is described. BullReporter was one of the tools (software packages) developed under a MnDOT-sponsored research project entitled "Development of Data Warehouse and Applications for Continuous Vehicle Class and WIM Data" [1]. This project was started in 2006 and sponsored by the Office of Transportation Data & Analysis (TDA, presently part of OTSM), and Dr. Taek Kwon at UMD was the principal investigator (PI). This project was formulated to solve the following issue. MnDOT was operating multiple types of WIM systems manufactured by different companies, which resulted in collecting different types of raw WIM data with completely different formats. Some manufacturers did provide a decent software tool for data analysis and reporting, but others did not. The tools provided by one vendor did not work for WIM data produced by other vendor's machines. Each company was using a proprietary data format, which resulted in prevention of sharing data files. At MnDOT, it became a problem when the trial periods of new systems ended, and a statewide data analysis was needed, requiring integration of data from all vendor machines. Therefore, there was a need to create a single WIM analysis tool that could accept data formats from all vendor machines, regardless of differences in manufacturers or different models from the same company.

To achieve the goal of creating one tool for all WIM data, a three-layered software stack architecture, shown in Figure 1, was devised and eventually implemented. The main advantage of layered architecture is that each layer can evolve independently of other layers as long as the layer-to-layer interface is clearly defined and conforms to a standard. Figure 1 illustrates this three-layered software stack referred to as a Bull-Converter/Reporter stack. At the bottom layer (Layer 0), WIM hardware systems from different vendors translate raw analog signals from sensor-crossing vehicles to digital vehicle records. This hardware is typically called a WIM controller, and it consists of charge amplifiers, analog-to-digital converters, and a microcomputer. The raw WIM data is captured by the first layer software that continuously runs and accumulates data to a daily data file that spans from zero hour to the 23rd hour of the day. This software is often embedded in the WIM controller. The format of this data is vendor dependent, and each vendor generally provides this function as part of its WIM system. The daily WIM data files are archived by MnDOT due to the following reasons. Many diagnostic details are vendor hardware dependent, and an effective way of keeping this information is through daily archiving of these raw vendor-dependent data files, because these raw files can tell a story of how the machines have performed. In addition, the raw files can be later used for extracting more detailed WIM information that was not present in standard comma separated values (CSV) files such as Bull-CSV.

The second layer of this stack translates the vendor-dependent raw daily WIM files to standard CSV files called Bull-CSV, which is served by BullConverter. After raw vendor-dependent daily WIM files are converted to a standard CSV format (Bull-CSV), the differences of data formats from vendor to vendor no longer exist.

The third layer is the application layer, and it is served by BullReporter, which only needs to understand the Bull-CSV format. This tool contains a wide range of analyses and reporting tools. Since Layer-3 is an application layer, other types of analysis tools may be integrated such as a query tool based on a geographical information system (GIS).

With this three-layered software stack, adding a new vendor machine to a statewide WIM system becomes simple, and two more types of format translation were implemented in this project by adding two new tabs. In general, BullConverter simply adds a new conversion tab for translation of a new format. For BullReporter, since the software stack follows a well-defined layered architecture, it can add more reporting functions independently of BullConverter evolution, providing seamless integration of new vendor machines.

On completion of this project, BullConverter was capable of conversion of raw WIM files from four different vendor formats: IRD iSINC binary, Kistler DL, Intercomp WIMLOGIX, and Peek PVR. Several new functions were also added to BullReporter without any discernable errors or complications that can be caused by differences in vendor data formats.

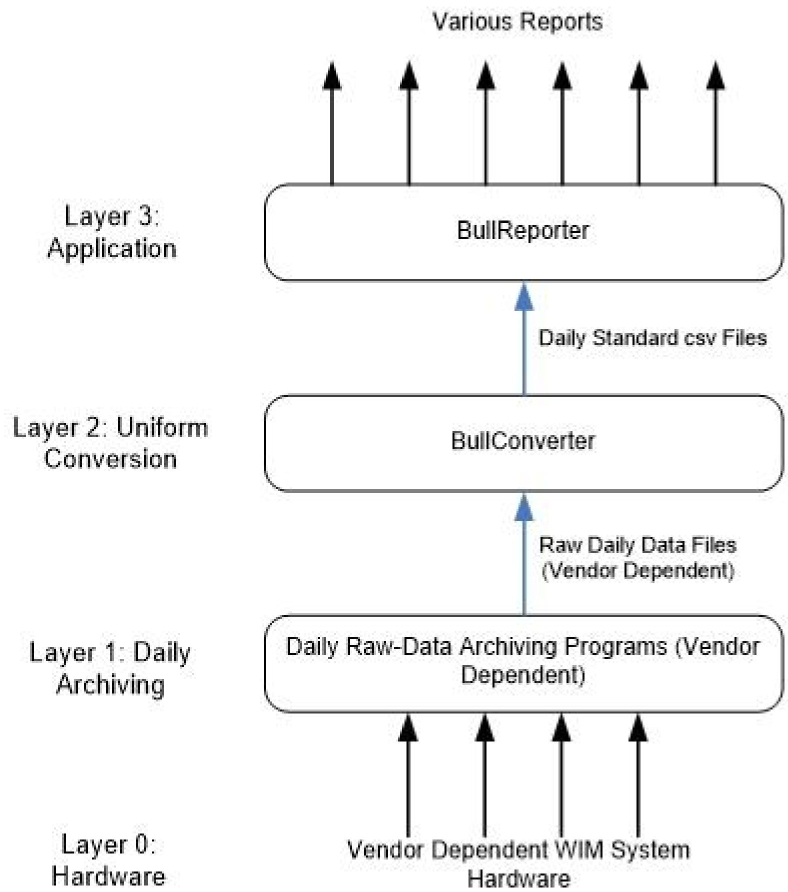

Figure 1 Bull-Reporter/Converter software stack.

1.2 PROJECT GOALS

BullReporter and BullConverter are two important software packages for WIM data processing and analysis at MnDOT. These data tools have been used by the MnDOT OTSM since their initial delivery in 2009 for analysis, forecasting, and reporting from the statewide WIM system. The objective of this project was to expand the existing conversion and reporting capabilities of BullReporter and BullConverter to meet new needs. Data analysis needs change over time, and the members of the OTSM WIM section identified several new functions that would meet new emerging data services for their stakeholders. This upgrade includes implementation of new reporting and conversion functions, as well as modification or addition of options to the current version. The tasks defined in the project work plan are summarized below.

Task 1: Addition of Intercomp Data Conversion Capability in BullConverter. This task requires implementation of a new conversion capability in BullConverter in the form of a new tab, i.e., add a new tab that is dedicated to conversion from Intercomp WIMLOGIX files to Bull-CSV files.

Task 2: Implementation of "View Vehicles" Report Function in BullReporter. "View Vehicles" report is a large table of vehicle records consisting of a thumbnail of the vehicle's digital picture and vehicle parameters drawn in a graphical vehicle model. The objective of this task was to implement an improved version of the current "View Vehicles" function available in iAnalyze. The improvements are mostly in data filtering conditions, which include start/end date and time, lanes, vehicle types, classification scheme, GVW, speed, and steer axle weights in certain ranges, as well as specific multiple hour ranges.

Task 3: Histogram of GVW and ESAL Data. MnDOT OTSM presently produces histogram of vehicles over 88 kips in a monthly table. BullReporter does not have this capability, and thus this data table is manually produced through multiple steps. The objective of this task was to implement GVW histogram functions in BullReporter with monthly reporting options as well as more detailed query conditions. A similar histogram function was asked to be implemented for ESAL values.

Task 4: Miscellaneous Functions. The original Task 4 was to add functions related to Portable WIM (PWIM) data analysis, but that option was later dropped due to ending of PWIM system deployments at MnDOT. Task 4 was modified by TAP as follows.

- Calculate GVW and ESAL based on all axle weight data available in the binary or raw data file when a vehicle has more than 12 axles (>12). Then add Error #6 at the error column, indicating that this vehicle has more than 12 axles. Individual axle weight columns should remain the same, i.e., it only displays up to 12 axles.
- Modify the IRD single file conversion utility to "Convert Selected IRD Binary files to CSV", i.e., multiple selected files should be converted instead of one file at a time.
- Increase accuracy of all ESAL values to three decimal places.
- Provide training and documentation on how to edit classification schemes.

Task 5 and 6: Final Report. Write and submit a final report, following the MnDOT publication guidelines. Follow the process of TAP review, PI edit, and final approval. Go through a contract editor review process.

CHAPTER 2: ADDITION OF INTERCOMP AND KISTLER DATA CONVERSION CAPABILITES IN BULLCONVERTER

MnDOT OTSM was sponsoring another University project entitled “Weigh-in-Motion Sensor and Controller Operation and Performance Comparison” in which performance of multiple WIM system configurations was studied. The potential test configurations include Intercomp sensors on an Intercomp controller, Intercomp sensors on an IRD controller, Kistler sensors on a Kistler WIM Data Logger (DL), and Kistler sensors on an IRD controller. In order to analyze the data collected by all possible different configurations, it came down to adding conversion functions for Intercomp and Kistler raw data to Bull-CSV. Thus, Task-1 of this project was formulated to integrate conversion utilities for InterComp and Kistler controllers in the BullConverter that would convert respective raw data formats to Bull-CSV format.

2.1 INTERCOMP WIM DATA CONVERSION

Intercomp company is headquartered in Medina, Minnesota and specializes in portable weighing and measurement products. Recently, they added high-speed WIM sensors in their product line and MnDOT was interested in trying out and checking out its performance, due to a low-cost advantage of their sensors.

Intercomp WIM sensor strips (shown in Figure 2) were designed using strain gage load-cell sensors which work based on the principle of changing resistance as they are elongated in relation to the strain of the base. This is different from the piezoelectric sensors in that they work based on the principle of piezoelectricity, i.e., squeeze of certain crystals by force makes imbalanced charges flow out of the material. Intercomp sensor strips are available in lengths of 1.5, 1.75, and 2 meters. These sensors are known to have temperature sensitive and require compensation.

Intercomp sensors are installed on a pavement by cutting a groove width slightly bigger than the sensor width 2.8 inch (70 mm) across a lane. The lead cables from sensors are then connected to an Intercomp WIMLOGIX controller which converts the raw signals from sensors to WIM parameters. In the MnROAD setup, a PC was used to interface with the WIMLOGIX controller which captures daily WIM data from I-94.

Figure 2 Intercomp WIM sensor strip

The “WIMLOGIX Pro Detailed Data Log” contains all detected vehicles for a given day, available to download via HTTP. The log is semicolon-delimited and some fields can be empty, allowing for consecutive delimiters. ‘String’ fields cannot contain the semi-colon character, or any line breaks. The file is generated in near-real time, with a new log file beginning each day. The detailed column formats of WIMLOGIX Data Log are shown in Appendix-A. This format supports vehicle records up to 15 axles and can record left and right axles. The naming scheme for the file is given by “YYYYMMDD-ID-WIMLOGIX.txt”, where the first eight characters are date, and the ID field after a dash is Station ID as defined in the device’s configuration file.

Violation codes consist of two letters. The field can contain multiple violation codes, separated by commas. They indicate that one or more factors may be interfering with the measurement of a specific vehicle. A blank field indicates that no problems were detected.

- Axle Count Mismatch (‘AM’) – Applied if the sensors covering the lane detected differing numbers of axles.
- Weight Balance (‘WB’) – Applied if the variance between sensors for any axle in a vehicle passes the acceptable threshold. This likely indicates that part of a wheel did not fully pass over a sensor.
- Over Speed Violation (‘OS’) – Applied if the detected speed of the vehicle was over the configurable threshold.
- Under Speed Violation (‘US’) – Applied if the detected speed of the vehicle was under the configurable threshold.
- Acceleration Violation (‘AC’) – Applied if the (positive) acceleration of the vehicle during its crossing of the sensors surpasses a configurable threshold.
- Deceleration Violation (‘DE’) - Applied if the (negative acceleration) deceleration of the vehicle during its crossing of the sensors surpasses a configurable threshold.
- No Loop (‘NL’) – Applied to every vehicle that was split apart because of a missing loop trigger.
- Loop Violation (‘LV’) – Applied to a vehicle that never received a begin or end loop trigger.

Conversion from above Intercomp error codes to Bull-CSV code is summarized in Table 1. The rest of parameters were converted based on mapping between Bull-CSV and WIMLOGIX columns in Appendix A. A tab in BullConverter was allocated for conversion of Intercomp WIMLOGIX files to Bull-CSV, and a screen capture of the tab is shown in Figure 3.

Table 1 Mapping from Intercomp Error Codes to Bull-CSV Error/Warning Codes

Intercomp Error	Description	Bull-CSV Error/Warning Code
AM	Axle count mismatch	8
WB	Weight balance	31
OS	Over speed	17
US	Under speed	13
AC	Acceleration (positive) violation	33
DE	Deceleration violation	33
NL	No loop	3
LV	Loop violation	3

*OS, US, AC, and DE may contain valid axle weight data, depending on the user threshold setup.

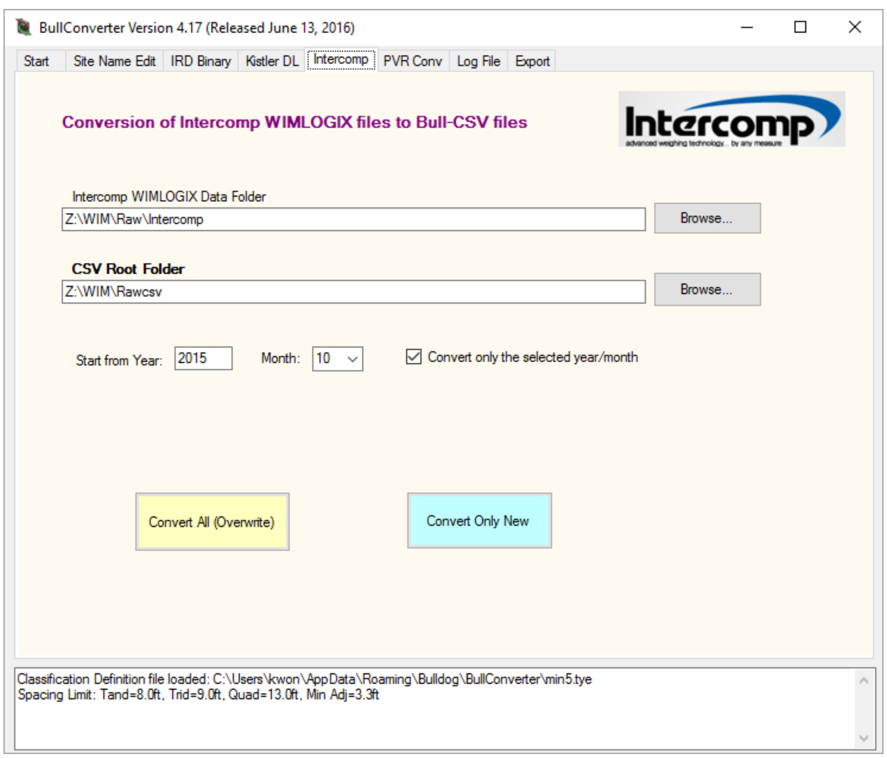

Figure 3 BullConverter tab for conversion from Intercomp WIMLOGIX to Bull-CSV files

2.2 KISTLER WIM DATA CONVERSION

Kistler corporation produces Lineas® Quartz sensors which are the most popularly used WIM axle-load sensors in state DOTs in US. Quartz load sensors use quartz crystals and are in general more expensive than polymer film based piezoelectric sensors but they produce much higher accuracies in axle-load measurements. Kistler also manufactures charge amplifiers that convert charge signals from their quartz axle sensors to voltage signals.

Recently (in 2015), Kistler expanded their product line to include WIM controllers, i.e., "Kistler WIM datalogger (DL), 5204A." This controller was designed using an architecture called Representational State Transfer (REST). It is a popularly used web-based client-server architecture, in which clients are served through RESTful web services built using Hypertext Transfer Protocol (HTTP), Uniform Resource Identifier (URI), markup languages such as HTML and XML, and web friendly formats like JASON and Atom. In Kistler DL, resources are provided through RESTful Application Programming Interface (API) services [2]. It is a secure and powerful web-based platform, but getting daily WIM files from Kistler DL is not easy for general users. It requires an application program that automates interactions of the RESful API through web and retrieve correct data items from the respective API.

MnDOT wanted to evaluate Kistler DL on Lineas Quartz sensors installed at the MnRoad test site. Since Kistler does not provide a software tool for automatic downloading and archiving, the PI created an application program for automatic downloading and archiving by working together with the Kistler US office. A new application program named Kistler-Datalogger Downloading and Archiver (KisDA) [3] was developed as a result. The main Graphical User Interface (GUI) of this new program is shown in Figure 4, and it provides automated as well as manual archiving of raw WIM data. For archiving, there are three options: single day of data at a time, 10 days of data at a time, or daily scheduled run for automated archive operations. This application program automatically accesses the designated Kistler DL through Internet with the destination IP address and password set by user and downloads WIM data from the respective DL. This program has been installed at MnDOT and continuously run through an automated daily run setting. The downloaded data are packaged as a single archived file per day with *.kis extension and its filename follows the convention given below.

yyyy.mm.dd.###.kis

where

yyyy = 4 digit representing year
mm = two digit representing month
dd = two digit representing date
= three digit site ID
kis = extension field indicating Kistler WIM data

The *.kis data files are archived in a comma separated values (CSV) format with the columns described in Table 2. Error/warning code mappings to Bull-CSV format are defined in Table 3.

A tab in BullConverter is allocated for conversion from *.kis files to Bull-CSV format, and its screen shot of the Kistler tab is shown in Figure 5. It provides conversion options of a selected month of the year only or conversion of all files from any month of the year in the past to present. Date selection option is available only up to month. If some of dates in a month are selected such that it includes some files that are already converted, existing files can be overwritten by converting again or skipped. The skip option allows conversion of only new files and ignores already converted files. These two options allow users to select the period only up to month of the year. If the check mark labeled “Convert only the selected month/year” is unchecked, all dates starting from the selected month/year to the latest date available are selected. The same, uniform GUI is used in all vendor tabs in BullConverter.

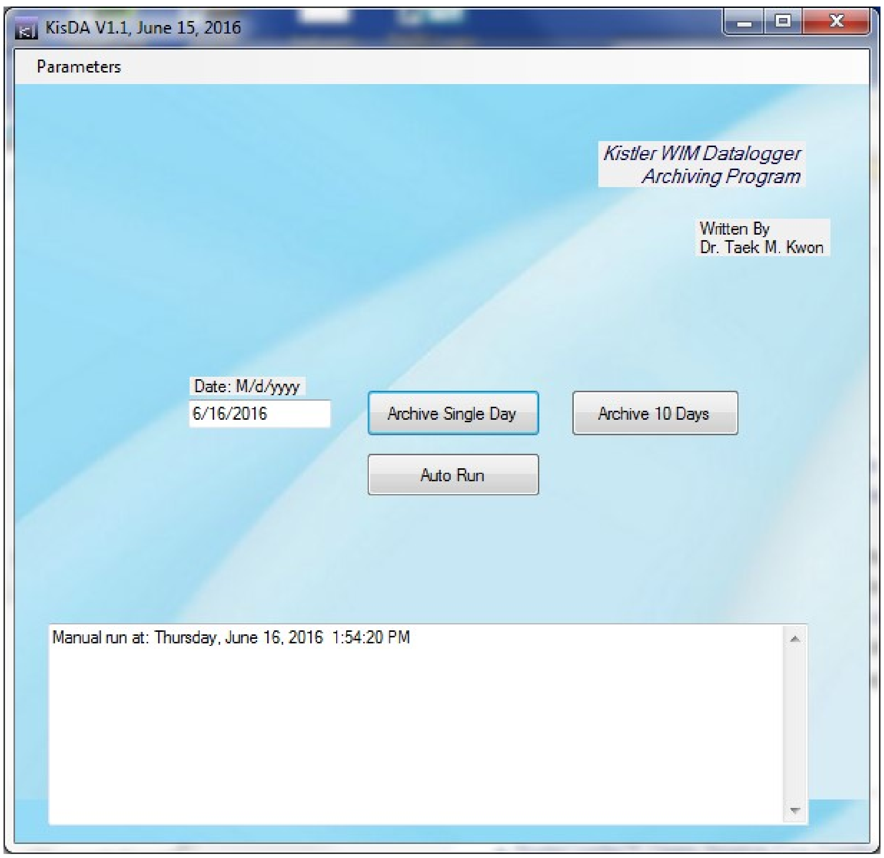

Figure 4 Main screen of the KisDA utility

Table 2 Columns of *.kis File Format

Col Num	Column Heading	Data Description	Example
1	VehID	Unique vehicle ID, integer	2236784
2	StartTime	Time of vehicle crossing the first sensor. hh:mm:ss	15:19:35
3	Millisec	Millisecond portion of StartTime	198
4	LaneNo	Lane number in integer	1
5	ErrWarning	String. Multiple error/warning codes are combined through "|" character. See Table 2 for defined codes.	2|72|80
6	MoveStatus	0=constant speed 1=acceleration -1=deceleration	0
7	FrontToFront	Headway. Time distance to the leading car on the same lane. Seconds	18.178
8	BackToFront	Gap. Time distance to the leading car on the same lane. Seconds	17.998
9	Duration	Time between entering the first sensor and leaving the last sensor	0.87
10	VehLength(in)	Total vehicle length including trailers. Integer	893
11	GVW(lb)	Gross Vehicle Weight. Integer	63380
12	Speed(mph)	Vehicle speed. Floating point	68.8
13	AxleCount	Number of axle of the vehicle. Integer	5
		Axle number=N	N=0,1,2,…
14	AxLW1(lb)	Weight of left wheel of axle #1	5940
15	AxRW1(lb)	Weight of right wheel of axle #1	6280
16	AxD1(in)	Distance to previous wheel	0
17	AxLW2	Weight of left wheel of axle #2	6480
18	AxRW2	Weight of right wheel of axle #2	7840
19	AxD2	Distance to previous wheel (#1)	187
20	AxLW3		
21	AxRW3		
22	AxD3		
23	AxLW4		
24	AxRW4		
25	AxD4		
26	AxLW5		
27	AxRW5		
28	AxD5		
29	AxLW6		
30	AxRW6		
31	AxD6		
32	AxLW7		
33	AxRW7		

34	AxD7		
35	AxLW8		
36	AxRW8		
37	AxD8		
38	AxLW9		
39	AxRW9		
40	AxD9		
41	AxLW10		
42	AxRW10		
43	AxD10		
44	AxLW11		
45	AxRw11		
46	AxD11		
47	AxLW12		
48	AxRW12		
49	AxD12		
50	AxLW13		
51	AxRW13		
52	AxD13		

Table 3 Error/Warning Code Mapping to Bull-CSV

*.kis Err-Warning Code	Name	Description	Bull-CSV error code
1	Out of Spec	Vehicle driving out of specifications – outside the specified operating range of the set certified mode	15
2	Vehicle Processing Error	Cannot process vehicle	14
70	Velocity above max	Driving above specified velocity	46
71	Velocity below min	Driving below specified velocity	0
72	Strong acceleration max	Strong acceleration above specified max	34
73	Strong deceleration min	Strong deceleration below specified min	34
74	High imbalance	High left/right weight imbalance	35
75	Sensor missing	Force or presence sensor signal missing	70
76	ADC overload	Charge amplifier output voltage fed to ADC (A/D Converter) is outside the ADC voltage limit	66
77	High vehicle dynamics	Highly dynamic driving behavior	67
78	Acceleration change	Strong changes in acceleration	34
79	Driving between two lanes	Driving between two adjacent lanes	68

80	Single-track vehicle	Single-track vehicle or vehicle driving on only one side of the layout	69
81	Force record missing	Force record missing	7
82	Single axle vehicle	Single axle vehicle	19
83	Stop and go	Vehicle stopped while driving through the WIM site	16
84	GVW above max	Gross vehicle weight above specified value	44
85	GVW below min	Gross vehicle weight below specified value	65
86	Axle load above max	Axle load above specified value	43
87	Axle load below min	Axle load below specified value	65
90	Undefined Kistler DL error/warning	Undefined or unknown DL error or waning messages, i.e., error or warning messages that are not included in 1, 2, 70-87.	65

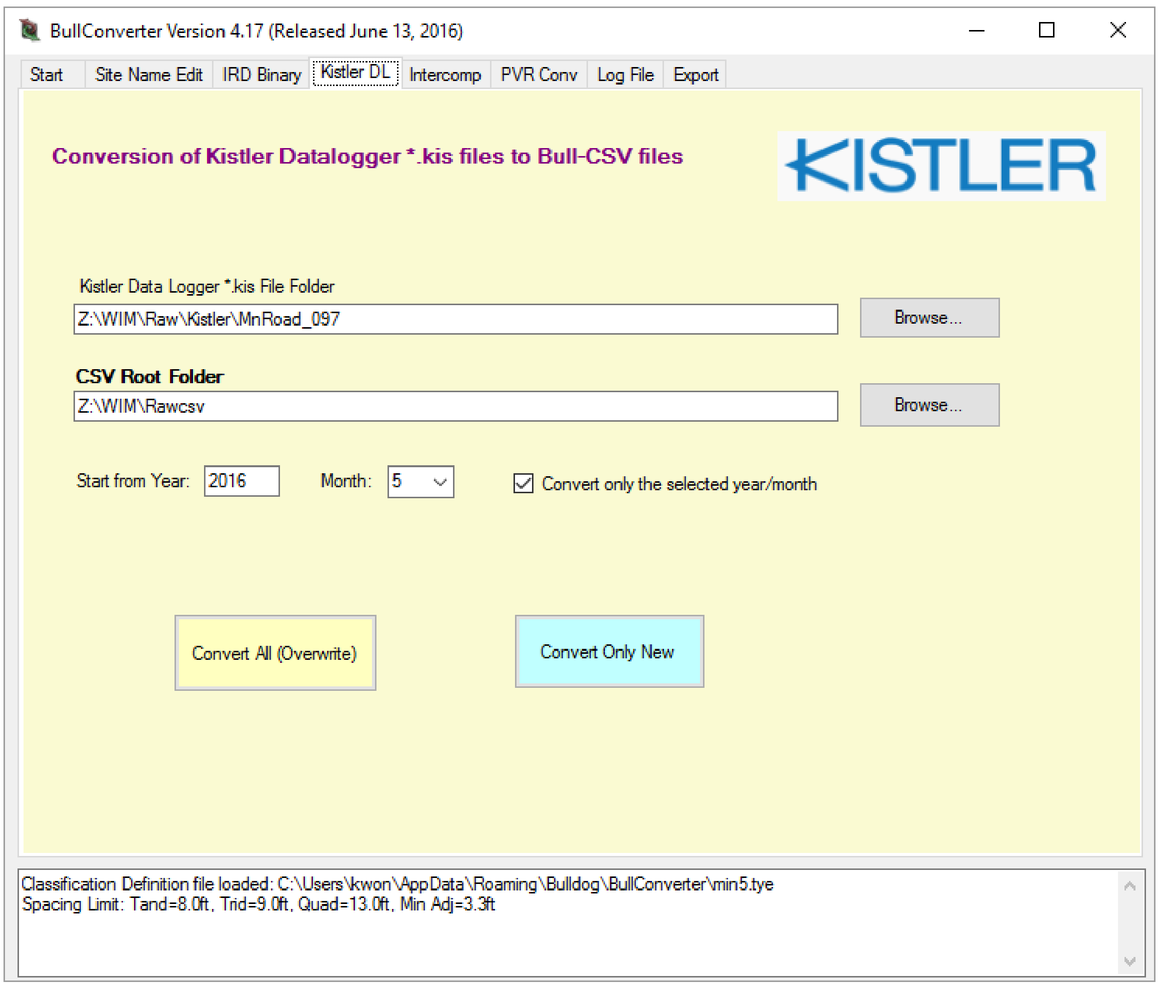

Figure 5 BullConverter Kistler DL tab for converting from *.kis files to Bull-CSV files

CHAPTER 3: VIEW VEHICLES REPORT IN BULLREPORTER

"View Vehicles" is one of the reporting tools in the software called iAnalyze [4] developed by the International Road Dynamics (IRD) Inc. This tool allows users to see individual vehicle records along with a graphical representation and/or a digital photograph of the vehicle. This report helps visual identification and verification of WIM data such as classification errors and unusual weights or axle configurations. This function is presently not available in BullReporter, and MnDOT proposed implementation of an enhanced version of the "View Vehicles" report in BullReporter. This enhanced version must include an ability to look at specific hours of the day, lane, and class. In addition, the user may be able to specify vehicles with the gross vehicle weight (GVW) or steer axle weight above or under certain limits.

3.1 IRD VIEW VEHICLES FUNCTION

iAnalyze is the main data analysis tool provided for IRD WIM systems and it implements a "View Vehicles" function with several user selectable parameters in three tabs using a Vehicle Display Wizard. The three tabs are shown in Figure 6, which consists of Site/Data, Class/Lanes, and Display Settings tabs. The Site/Data tab is the first one, and it is used to select a site in the drop down list and also to set the start and end date-time using two Microsoft date-time-picker components. The second tab (Figure 6) is the Class/Lanes tab and it is used to select the range of classes and lanes of the vehicle records.

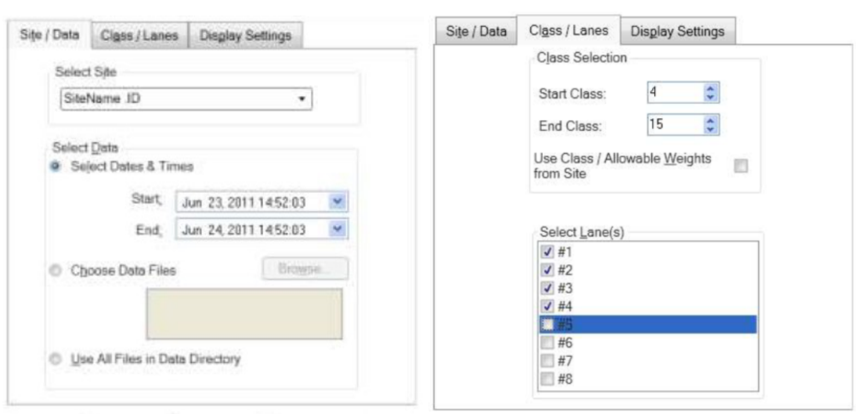

Figure 6 Site/Data and Class/Lanes tabs

The last tab is the Display Settings tab, which is shown in Figure 7. This tab provides selection of six types of vehicle ranges, i.e., (1) All vehicles, (2) Good weight vehicles only, (3) Error/Warning vehicle only, (4) Violating vehicles only, (5) Top 100 Over GVW, and (6) All over GVW. The user can select one of these six types using a radio button. There is also an option to select a display format of vehicle records, i.e., text- or graphics-based. The search conditions selected from three tabs are all applied in AND-logic relations. A sample screenshot of IRD View Vehicles report is shown in Figure 8.

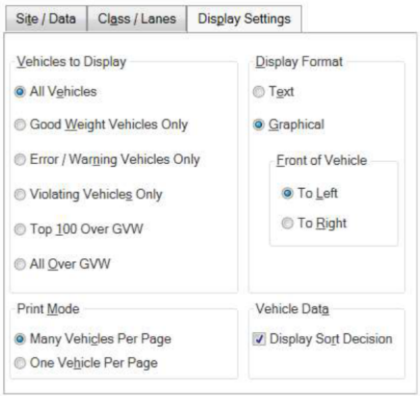

Figure 7 Display Settings tab

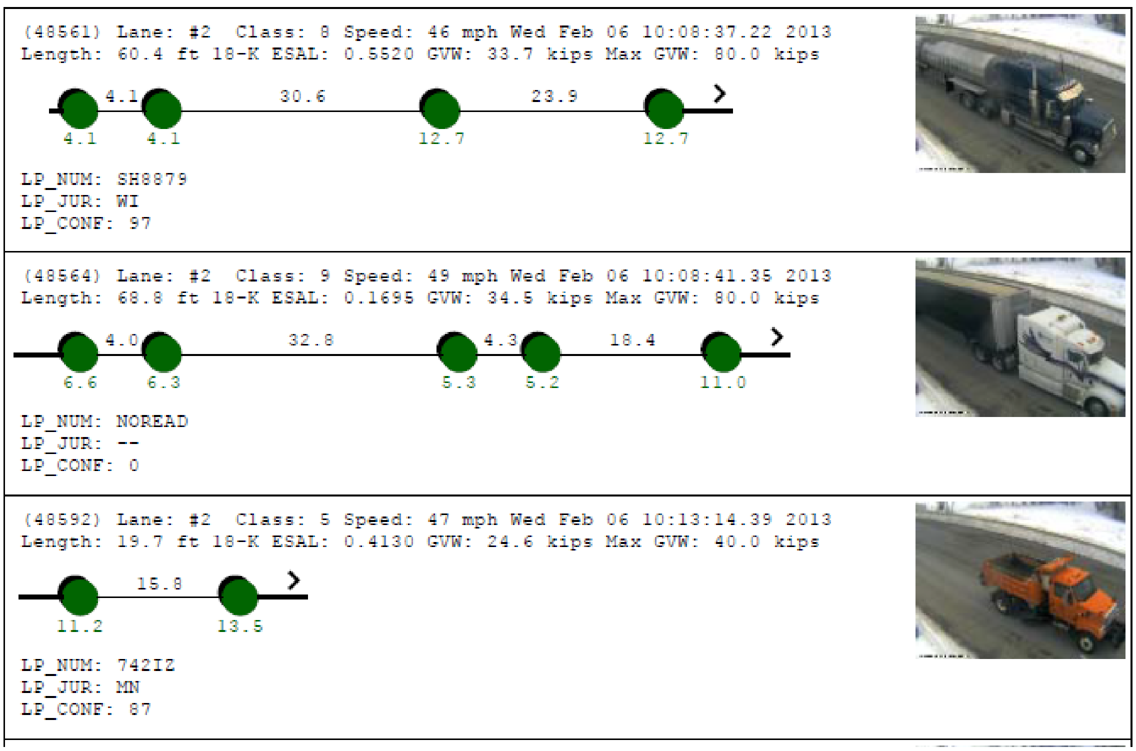

Figure 8 Screen shot of a sample iAnalyze View Vehicles report.

3.2 BULLREPORTER VIEW VEHICLES FUNCTION

Although the “View Vehicles” function in iAnalyze is a well designed for general use, there is a lot of room for improvement, especially for more advanced query of vehicle records. For example, iAnalyze does not allow for users to select vehicles based on Gross Vehicle Weight (GVW) or steer axle weight below or above a certain limit. It also does not allow to select vehicles at a certain speed range. It also lacks the capability of selecting certain types of error and/or warnings. It provides selection of the start and end date-time but not hour ranges. For example, if a user wants to look at vehicle records in the morning hours between 6:00 to 9:00 AM of a month or a whole year, he or she must retrieve the whole vehicle records between the start and end date-time and manually remove the outside the hour range of interest. Another enhancement needed was inclusion of a link that would allow display of the full original-resolution of the vehicle photo in a separate window for detailed visual inspection through a simple click of a mouse button on the thumbnail image. Mentioned above are improvements that MnDOT OTSM WIM section wanted in BullReporter, and they became one of the main tasks of this project.

For inclusion of “View Vehicles” report in BullReporter, a top-level major tab was added and it was named Report-2. The “View Vehicles” implementation in BullReporter includes several improvements over the current iAnalyze version of View Vehicles. First, all of the querying conditions are displayed on a single window, which would reduce potential errors caused by invisibility of parameters from hidden tabs of user selections. In iAnalyze, parameters are selected from three tabs but two of them are not visible at any given time. In this case, the user has to remember what parameters he or she selected in the other two invisible tabs.

In the current version of “View Vehicles” in BullReporter, user selectable parameters are listed below, which are shown in the screen shot of the Report 2 tab in Figure 9. Notice that all of them are easy to set GUI and visible in a single window.

- WIM Site
- Start date and hour, end date and hour
- Multiple hour ranges (example: 6:00-8:00, 12:00-14:00, 17:00-20:00)
- Lane selections
- Class selections in multiple ranges (example: 5-9, 9, 11-13)
- Classification scheme selection
- Lane-by-lane report (provide each lane in a separate report)
- GVW range defined by above and/or below
- Speed range defined by above and/or below
- Steer axle weight range defined by above and/or below
- Inclusion or exclusion of errors and warnings individually using checkmarks
- Display option of vehicle with image only
- Output in two formats, pdf and web page
- Link to full-size digital photo from thumbnail

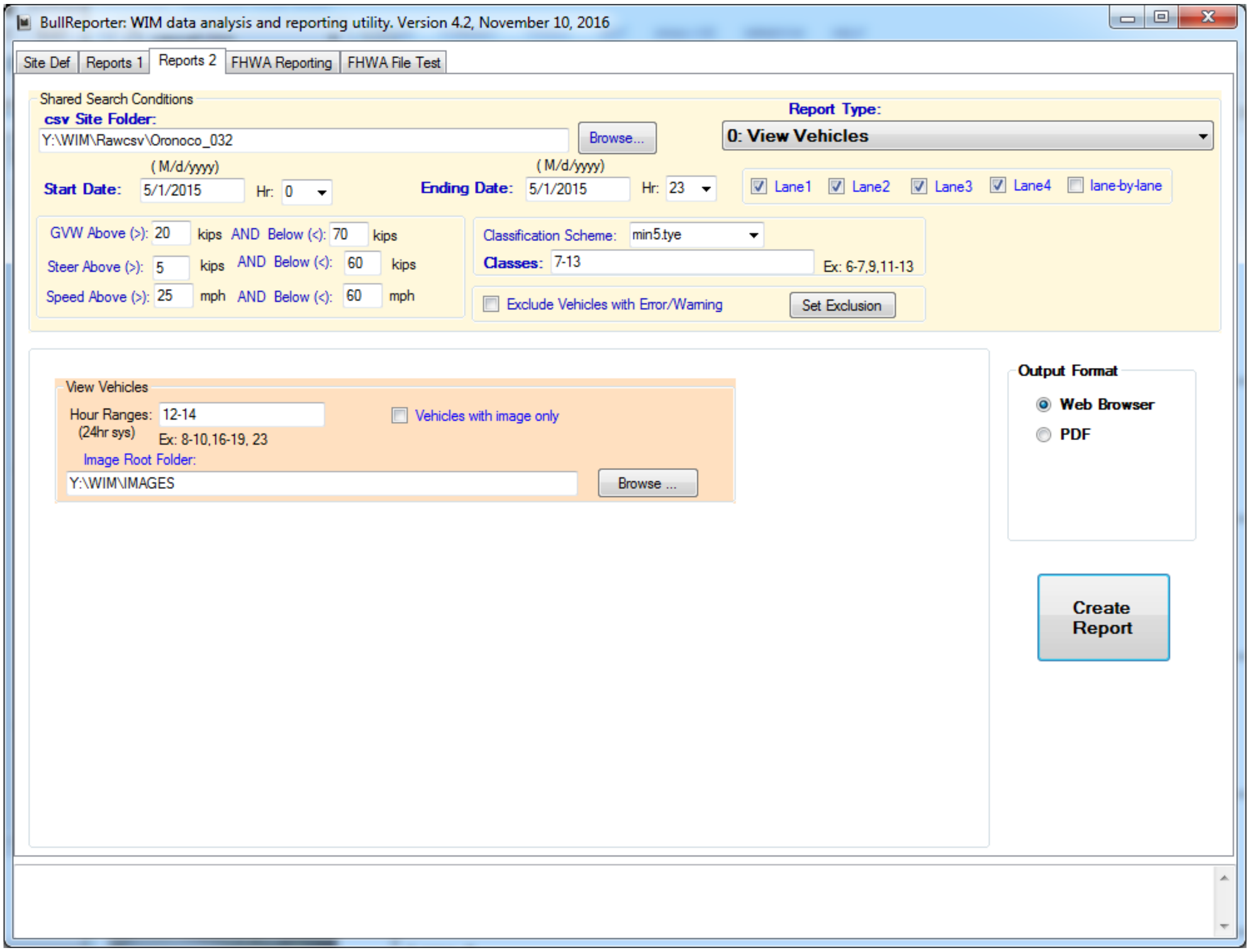

Figure 9 User interface of parameter selections in the View Vehicles Report of BullReporter

The query for “View Vehicles” is determined by logical conjunction of all user selectable parameters. The same query logic is also used for other types of reports available in the Report-2 tab selected by the “Report Type” drop-down menu. Presently, the report types include “GVW Histogram” and “ESAL Histogram”.

In the Figure 9 window, a button named “Set Exclusion” can be seen next to the check mark “Exclude Vehicle with Error/Warning.” Pressing this button opens up a window that contains a check list of all warnings and errors. Any item unchecked will be ignored from the search of exclusion condition, i.e., when the check mark “Exclude Vehicle with Error/Warning” is checked, only the vehicle records that contain unchecked errors and warnings will be made through in the vehicle filtering process. This check list allows users to individually control any error and/or warning combinations in filtering vehicle records.

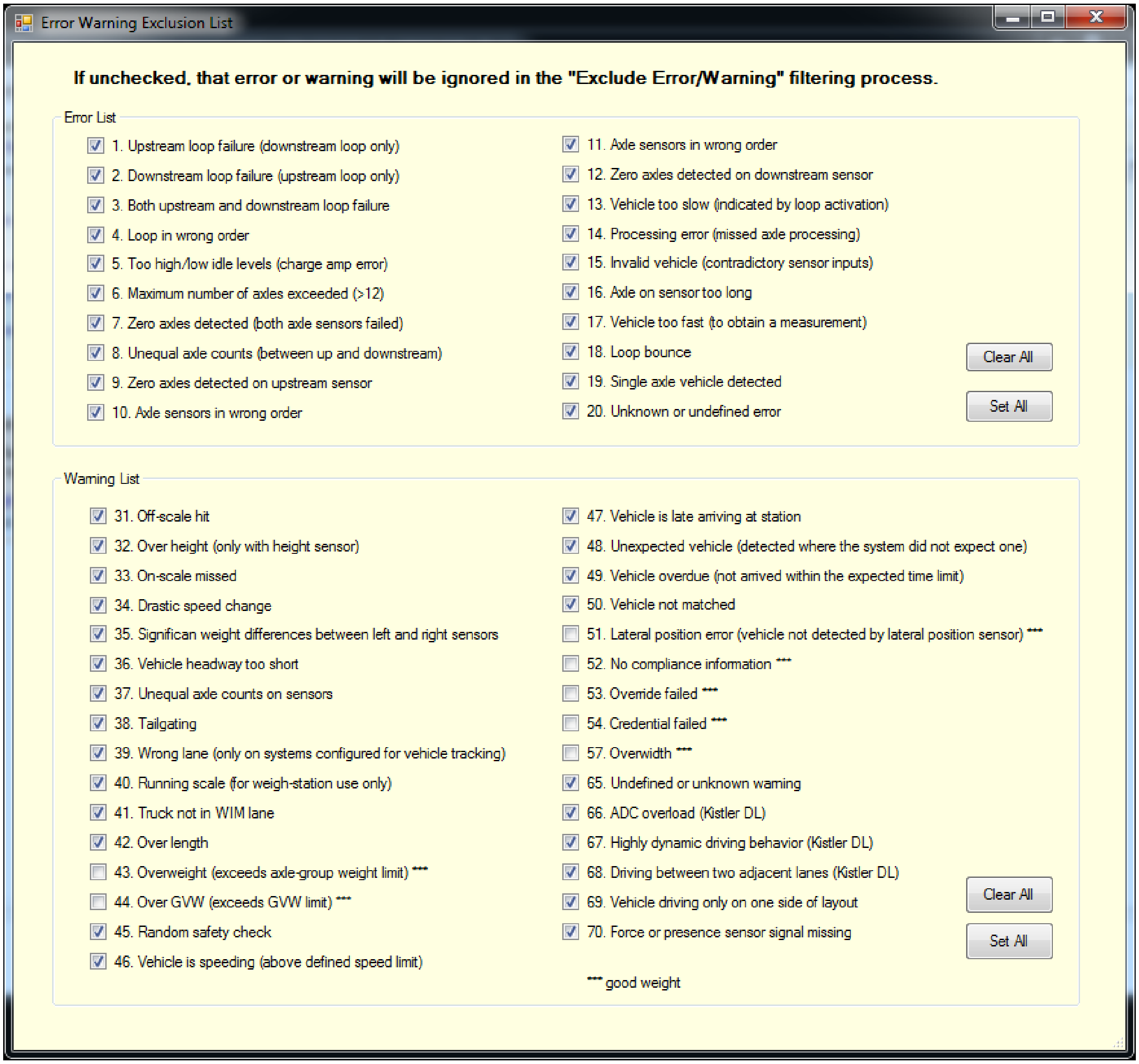

Figure 10 Masking of errors or warnings from exclusion.

For the final output formats of the "View Vehicles" report in BullReporter, web page and pdf formats are available. For both formats, reports can be generated lane-by-lane or all lanes together depending on the user selection. Web page format allows direct link to full size digital photos of the thumbnail, in addition to display of the View Vehicles format. All lanes together in a single report displayed in a web browser (Google Chrome) is shown in Figure 11 as an example. BullReporter automatically selects the default web browser set by the PC user and displays View Vehicles report in that default browser. When user presses the left mouse-button pointed on the thumbnail vehicle photo, it immediately retrieves the original digital photograph and displays in a separate window. For example, Figure 12 is the original vehicle photo retrieved when the thumbnail of the vehicle number 42006 (first vehicle of the list) in Figure 11 was pressed.

View Vehicles report can be generated using a pdf format as well, and an example is shown in Figure 13. It automatically selects a default pdf viewer of the PC and displays on that window, which in this case was Adobe Acrobat Reader. It should be noted that the hyper link to the original photo does not exist in the pdf report since the PC that runs the report may not have the original vehicle's photos.

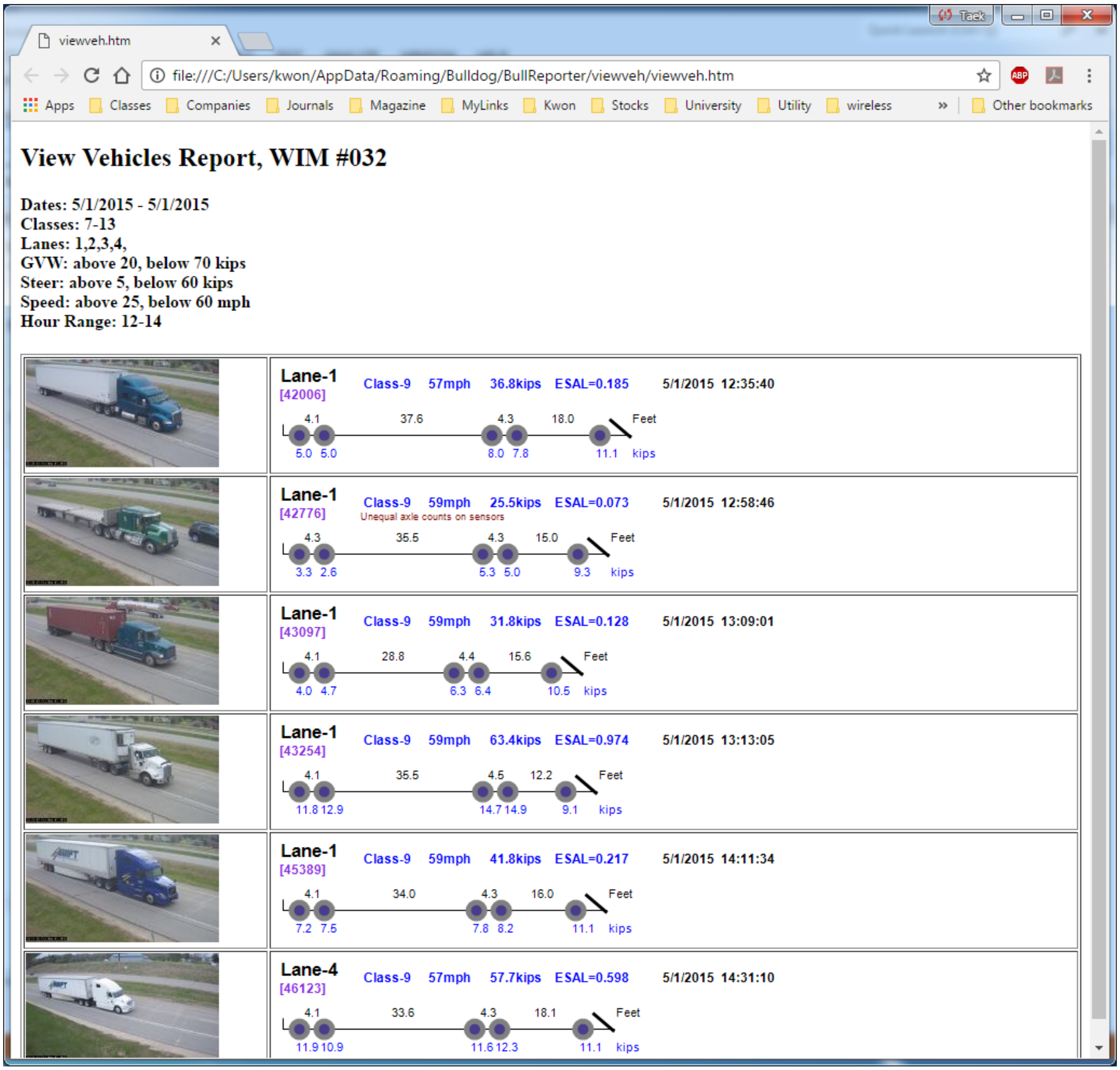

Figure 11 View Vehicles report generated on a web browser

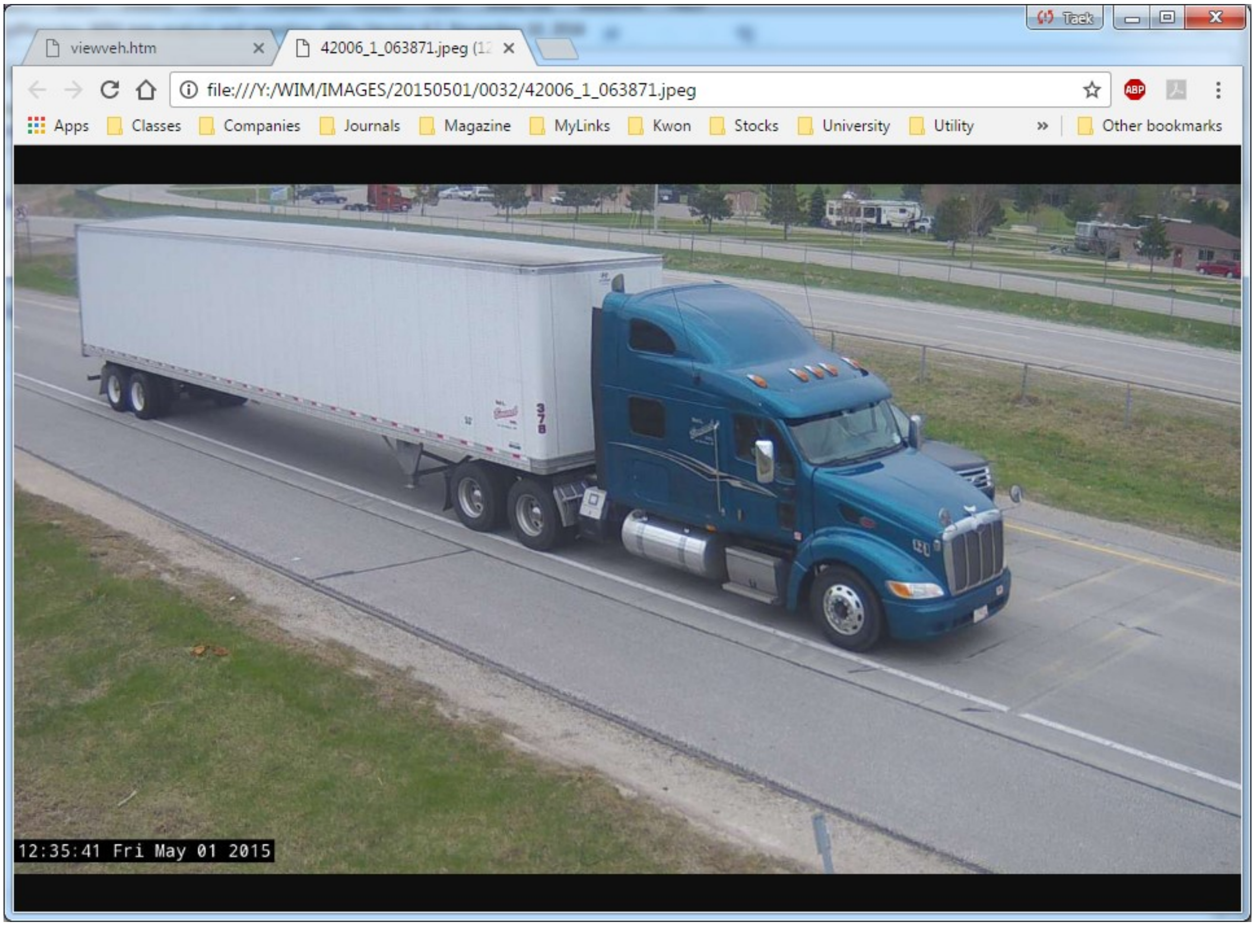

Figure 12 Full size photo retrieved when the thumbnail picture is pressed (vehicle # 42006)

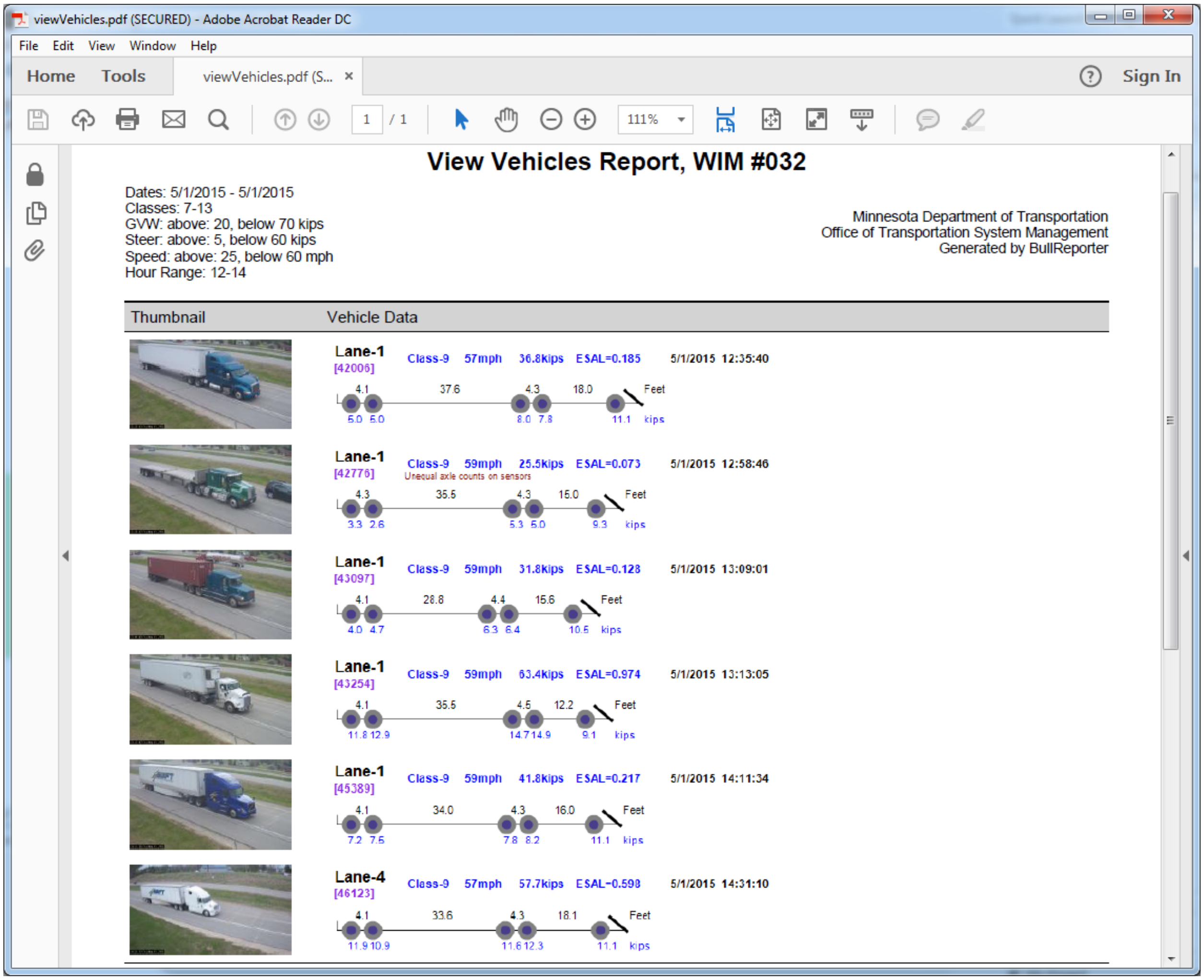

Figure 13 View Vehicles report generated in a pdf format

CHAPTER 4: HISTOGRAM OF GVW AND ESAL, AND OTHER MISCELLANEOUS FUNCTIONS

4.1 HISTOGRAM OF GVW AND ESAL

Task-3 description of this project states that the research team implements new histogram functions for Gross Vehicle Weight (GVW) and Equivalent Single Axle Loads (ESALs) to BullReporter. According to MnDOT OTSM during one of the project-formulation meetings, monthly histograms of GVW were produced manually at the end of the year one month at a time, which would result in running 12 times of repetition to produce a yearly report. MnDOT therefore added Task-3 which is implementation of new histogram generating functions in BullReporter for GVW and ESAL. For implementation, MnDOT also asked the research team to integrate multiple options to the two functions, which may include an option to select monthly reporting and options for setting GVW limits, limiting classes, setting histogram bin size and the number of bins, etc.

A histogram is constructed by splitting the data into non-overlapping intervals, called bins. Each bin contains the number occurrences (i.e., frequency) in the data set that are contained within that bin. The bin-length can be equal-length or varied, but all histograms in this project were implemented using only the equal-length bins. When the bin data are plotted using rectangular bar graph, it is called a histogram and allows inspection of data for its underlying distribution.

The research team decided to use the limiting conditions set by the same Shared Search Conditions of the “View Vehicles” report, which are very specific and meet MnDOT requirements. The limiting conditions are:

- Option to select a site
- Options to limit lanes
- Options to select a GVW range above or below user settable limits
- Options to select a steer axle weight range above or below user settable limits
- Options to select a speed range above or below user settable limits
- Options to select any combination of classes
- Option to select a classification scheme
- Options to include or exclude any number or combination of errors and warnings
- Option to select all lanes or lane-by-lane reports

The screen shot GVW histogram GUI is shown in Figure 14. For the GVW histogram, user must enter the bin size in kips and number of bins. The low-limit value of the starting bin is set by the “GVW Above” entry in the Shared Search Conditions groupbox. All bins are equally spaced based on the parameters set. The histogram data are then constructed in three different options of data periods, which are:

- Start-to-end dates and time
- Day-by-day (produces one histogram per day)
- Month-by-month (produces one histogram per month)

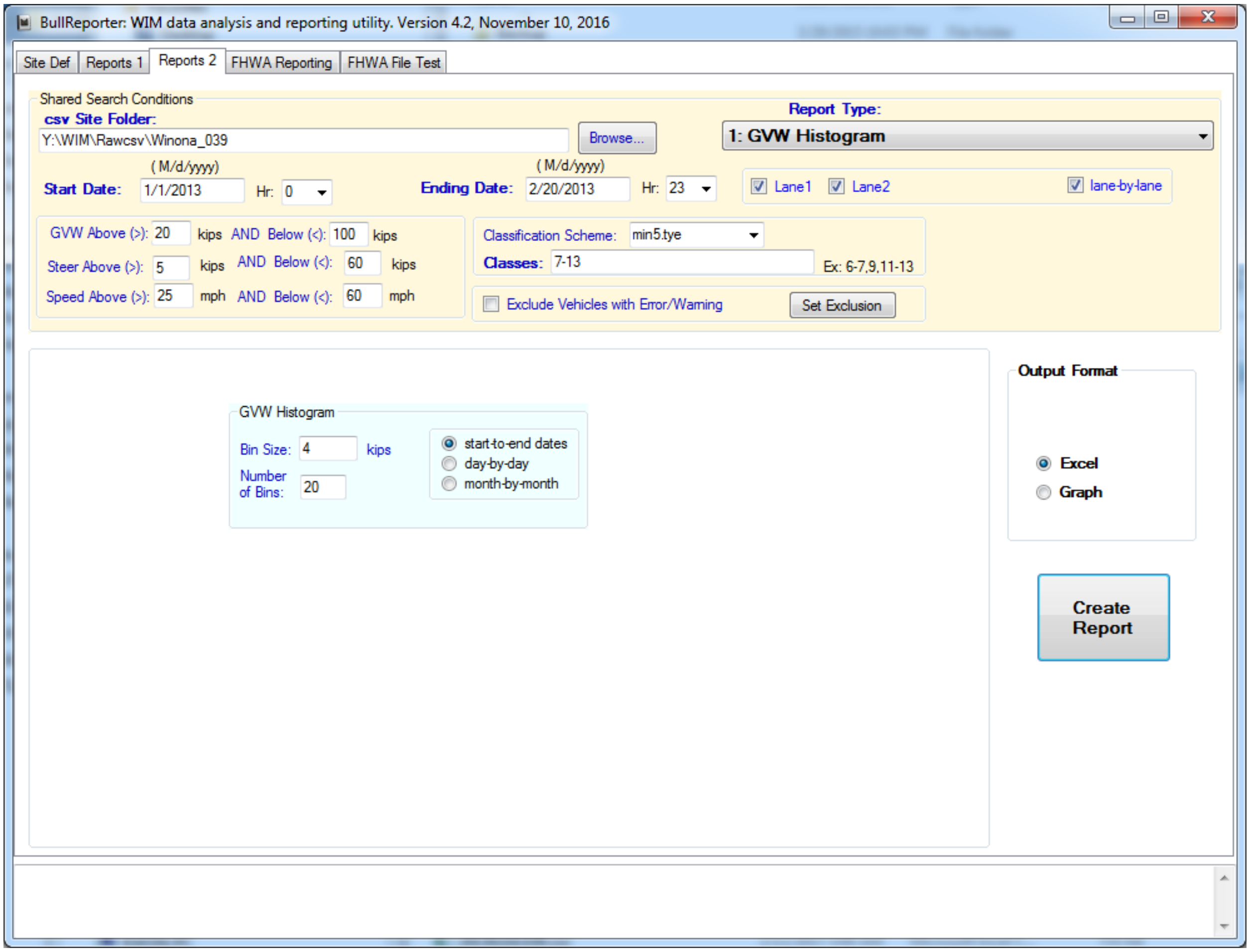

Figure 14 Graphical user interface for the GVW Histogram function

To demonstrate the implemented GVW histogram function, user parameters are set as follows (Figure 14) and the final histogram data outputted in Excel is shown in Figure 15:

- WIM Site: 039
- Dates: 1/1/2013 – 3/20/2013
- Lanes: 1 and 2
- Classification scheme: min5.tye
- GVW range: 20 - 100 kips
- Steer axle range: 5 – 60 kips
- Speed range: 25 – 60 mph
- Classes: 7 – 13

- No exclusion of errors/warnings
- Bin size: 4 kips
- Number of bins: 20
- Data periods for histogram computation: start-to-end dates

	A	B	C	D
7				
8	kips	Lane-1	Lane-2	Total
9	20->24	142	287	429
10	24->28	346	684	1030
11	28->32	716	987	1703
12	32->36	859	813	1672
13	36->40	583	602	1185
14	40->44	283	505	788
15	44->48	181	497	678
16	48->52	162	347	509
17	52->56	147	213	360
18	56->60	149	178	327
19	60->64	200	161	361
20	64->68	276	195	471
21	68->72	372	290	662
22	72->76	623	294	917
23	76->80	743	163	906
24	80->84	678	110	788
25	84->88	324	29	353
26	88->92	107	7	114
27	92->96	32	4	36
28	96->100	13	0	13
29				

Figure 15 Computed histogram data output sent out to Excel spreadsheet

When the “Graph” option is selected as the output format (see Figure 14), a histogram is produced for the data periods, start-to-end dates. When the lane-by-lane check mark is checked, the histogram is generated for each lane, which is shown in Figure 16. This option basically lets the graph to be plotted using both of the lanes, Lane-1 and Lane-2 columns from Figure 15, and the display of each lane histogram can be selected from the legends with check marks within the graph. When the lane-by-lane check mark is unchecked, the total of all lanes column in Figure 15 is plotted, which is shown in Figure 17. It should be mentioned that all plots can be edited and customized using the options available through the “Edit Graph” button, located at the right side top of the graph window.

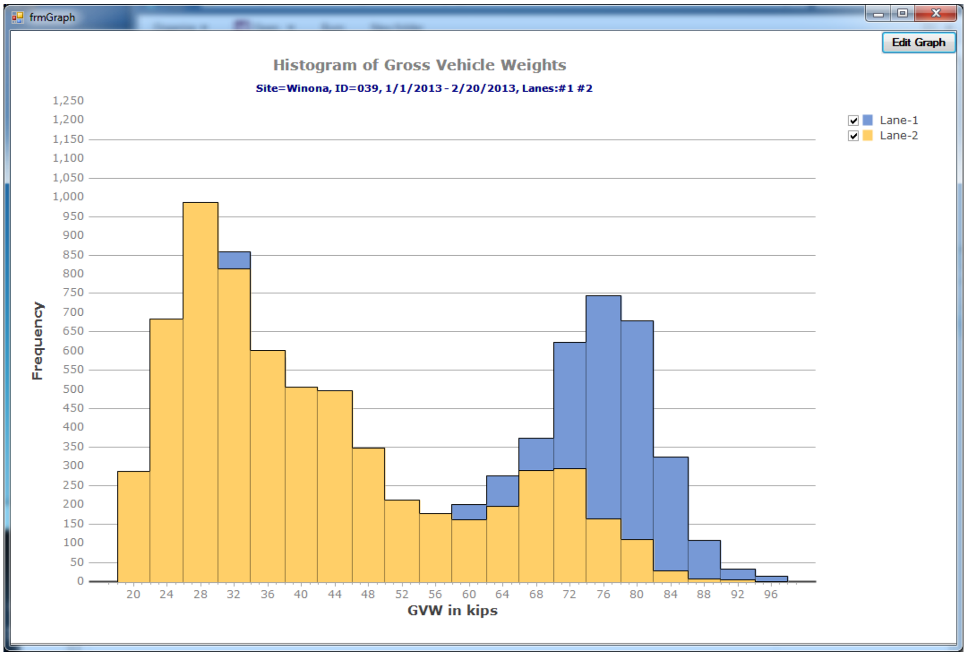

Figure 16 Histogram generated when the lane-by-lane checkmark is checked

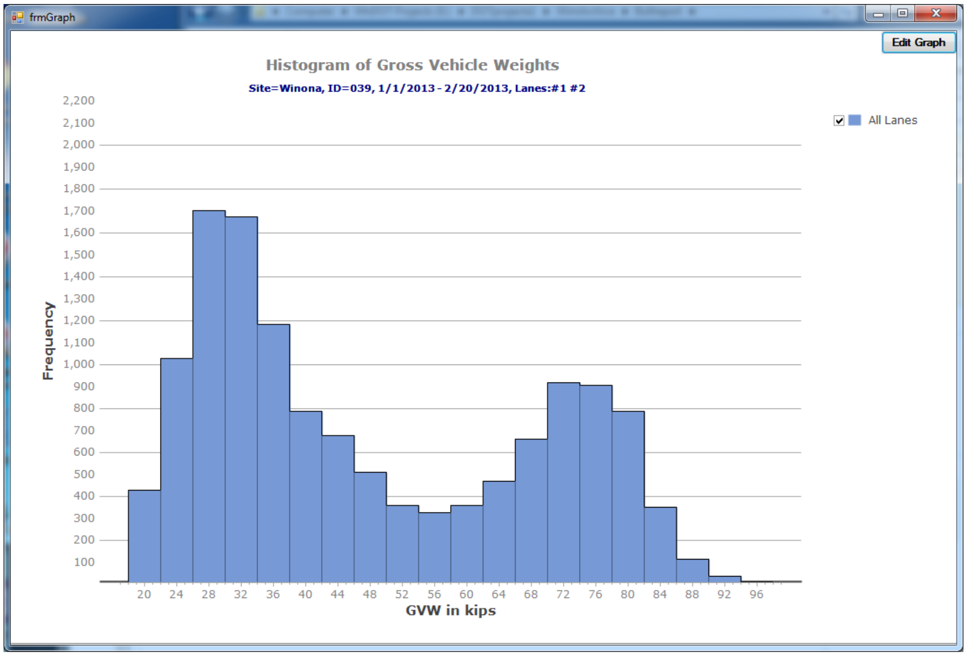

Figure 17 All-lanes combined histogram is generated when the lane-by-lane checkmark is unchecked

ESAL is computed based on empirical equations developed from the American Association of State Highway Officials (AASHO) Road Tests which established a damage relationship for comparing the effects of axle loads carrying different loads. The reference axle load is an 18,000 pounds single axle with dual tires [5,6].

GUI implementation of ESAL histogram is shown in Figure 18, which is identical to that of GVW histogram, except inclusion of ESAL parameters and replacement of variables from GVW to ESAL. Since the "Shared Search Conditions" group does not include ESAL limit conditions, setting of the start value of the first bin was placed in the Histogram setup group. Bin size and number of bins are also included as user programmable parameters in that group. ESAL values are recalculated using the parameters in the "ESAL Parameters" group which is located in the right side bottom of the "Shared Search Conditions" group. The "ESAL Parameters" group is invisible for the View Vehicles and GVW histogram reports. Similarly to GVW histograms, ESAL histograms can be generated using three different types of data periods, i.e., start-to-end dates, day-by-day, and month-by-month. The final histograms are outputted in one of two formats, Excel spreadsheet or graph.

Figure 19 shows an ESAL histogram generated using the setup shown in Figure 18. Figure 20 shows a histogram generated for all lanes by unchecking the lane-by-lane check mark while Figure 19 shows one histogram per lane. For these example, the same set used in the GVW histogram, i.e., WIM site #039 from 1/1/2013 – 2/20/2013 was used.

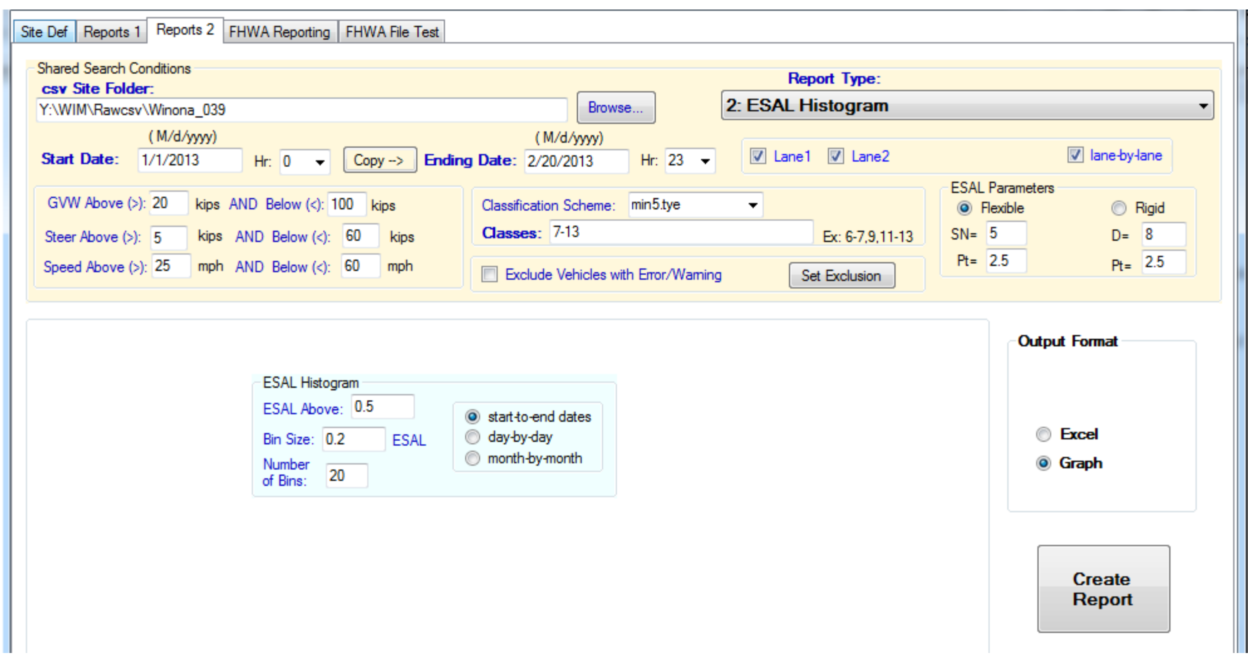

Figure 18 Setup parameters for ESAL histogram

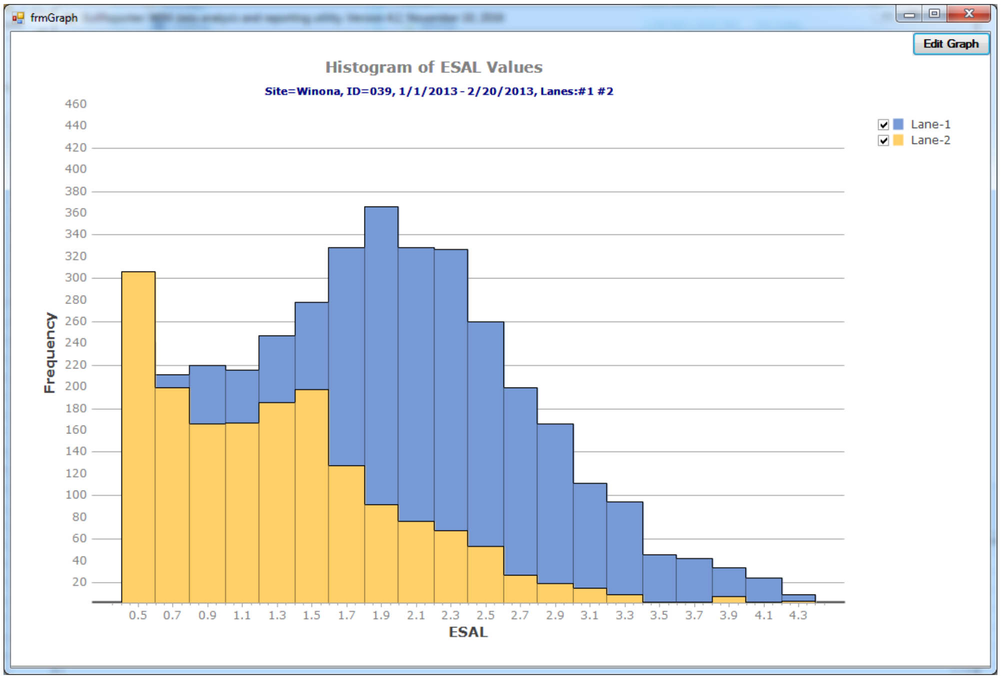

Figure 19 ESAL histogram generated for the option lane-by-lane

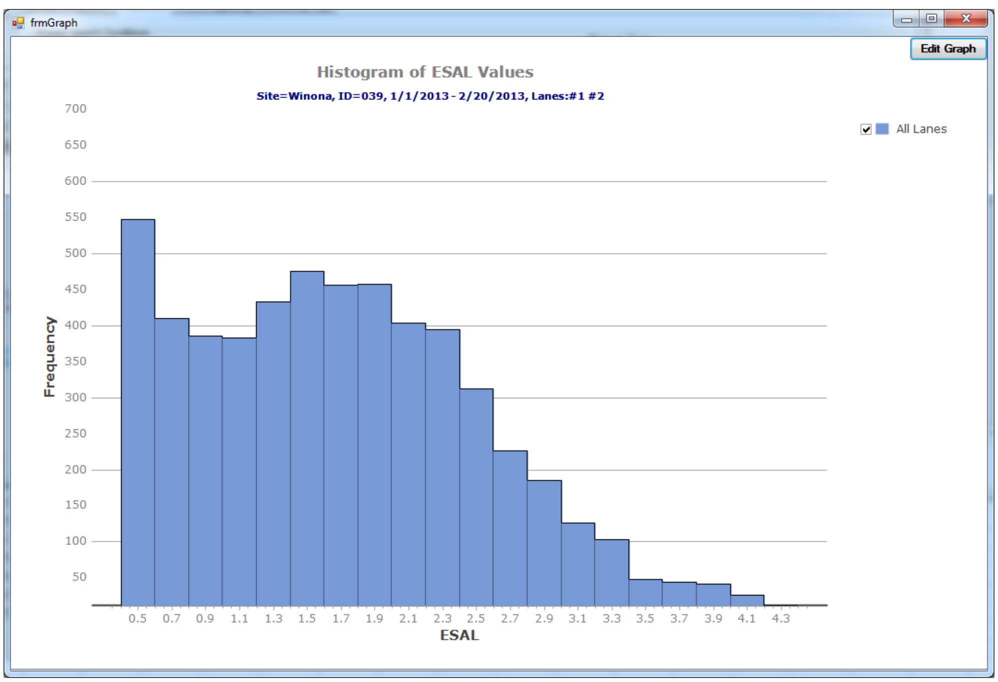

Figure 20 ESAL histogram generated for all lanes combined (uncheck lane-by-lane option)

4.2 OTHER MISCELLANEOUS FUNCTIONS

This section describes Task-4 of the project. The original Task-4 defined in the work-plan (contract) was modified at the Technical Advisory Panel (TAP) meeting held on December 15th 2016 at CO Room 360. This modification was made by the TAP after a discussion of the changing circumstances at MnDOT from the time the work-plan was written. More specifically, MnDOT decided to drop any further development or deployment of Portable Weight-in-Motion (PWIM) Systems after trials at the Dakota County in the summer of 2015. In that trial, the PWIM accuracy observed without temperature compensation was unsatisfactory. Another drawback of PWIM was that calibration for every installation was cumbersome; consequently, MnDOT decided to drop any future deployment of PWIM systems. Since this project was started on May 14th, 2015, and the work-plan was written even before that, the original Task-4 was established for PWIM data analysis without realizing future abandonment of PWIM deployments.

After dropping PWIM data analysis which required hourly data selection options, newly defined Task-4 at the December 15 TAP meeting in 2016 are summarized below.

- Calculate GVW and ESAL based on all axle weight data available in the binary or raw data file when a vehicle has more than 12 axles (>12). Then add Error #6 at the error column, indicating that this vehicle has more than 12 axles. Individual axle weight columns should remain the same, i.e., it only displays up to 12 axles.
- Modify the IRD single file conversion utility to "Convert Selected IRD Binary files to csv", i.e., multiple selected files should be converted, instead of just a single file.
- Increase accuracy of all ESAL values to 3 decimal places.
- Provide training and documentation on how to edit classification schemes.

All of the items listed above were completed and delivered. The deliverables including BullConverter Version 4.22 and the work done for Task-4 are summarized below.

- A new version BullConverter V. 4.24 (Figure 21) now calculates GVW and ESAL based on all axle weight data available in the binary or raw data file, including vehicles that have more than 12 axles (>12). If a vehicle has more than 12 axles, BullConverter adds Error #6 at the error column, which indicates more than 12 axles in the original data. Individual axle weight columns follow the same Bull-CSV format as in the past, i.e., it only displays up to 12 axles. This modification was made to all raw data types from 4 different company machines, i.e., IRD, Intercomp, PVR, and Kistler. It should be noted that ESAL values are always calculated using the raw axle weight and spacing data and the ESAL parameters set at the Start tab.
- Due to the new computational approach in GVW and ESAL for vehicles with more than 12 axles, the export utility had to be modified. Starting Version 4.22, the Export utility conforms to the following rule. If Error #6 exists in a vehicle record, ESAL and GVW values are directly copied from the ESAL and GVW columns of the original data, i.e., not recomputed based on the user defined ESAL parameters and addition of individual axle weights in GVW. If Error #6 does not exist or ESAL column is empty or not available in the original csv data for vehicles with Error #6, ESAL is recomputed base-on the user defined parameters. A similar logic is applied to GVW

overweight warnings, i.e., it uses the GVW column data instead of summation of all axle weights in the csv file if Error #6 exists.

- Starting BullConverter V 4.22, the IRD single file conversion utility in the “Log File” tab was modified to “Convert Selected IRD Binary files to csv”, i.e., all selected IRD raw binary files are now converted, instead of just a single file. Multiple files can be selected from the file open dialog using the left mouse button while holding a shift or control key. Current version of the “Log File” tab is shown in Figure 22.
- In both BullReporter and BullConverter, accuracy of all ESAL values was extended to 3 decimal places. All original variables were internally defined as double, and ESAL accuracy changes to three decimal places are only affected in the display.
- Training and documentation on how to edit new classification schemes on BullConverter and BullReporter were provided at the 12/15/2016 TAP meeting. The class definition edit manual used for this training is available on-line [7].

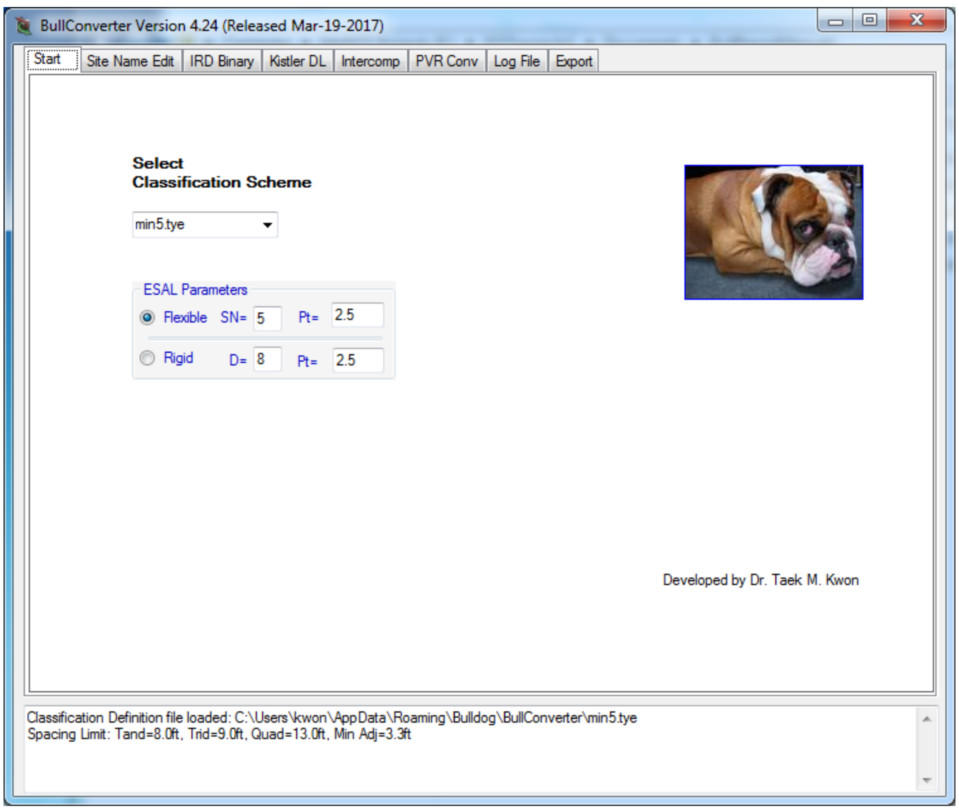

Figure 21 ESAL parameters must be set from the Start tab

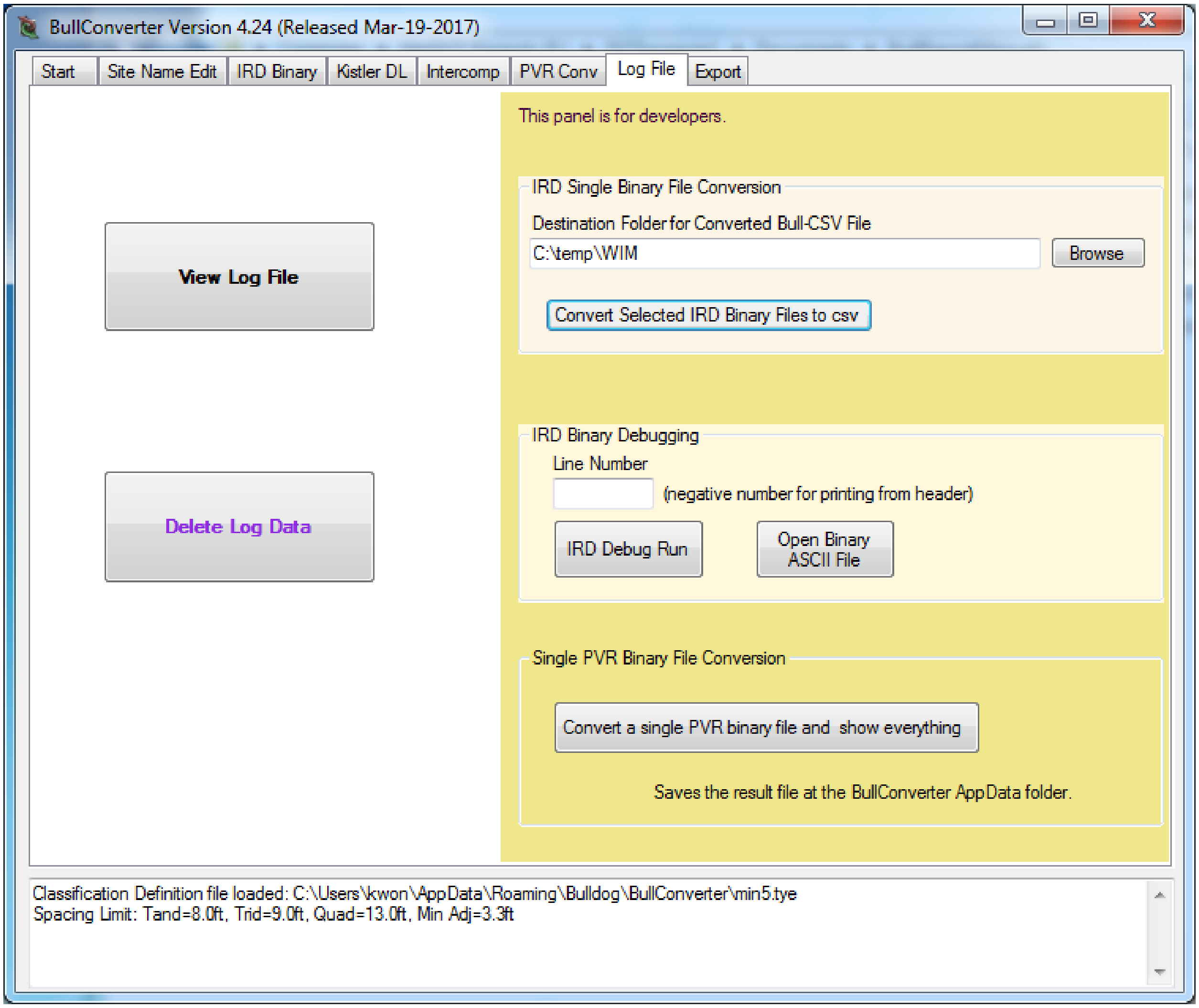

Figure 22 Log File tab in BullConverter

CHAPTER 5: CONCLUSIONS AND FUTURE RECOMMENDATIONS

The Bull-Converter/Reporter software stack has been successfully used by MnDOT OTSM for WIM data analysis and reporting since its inception in 2009. Data analysis and reporting needs change over time, and the members of the OTSM WIM section identified new functions and modifications of existing ones that qualify for a major revision to improve quality of the current version. This led to a proposal and a contract for this project, from which the UMD research team implemented the new functions and modification of the existing ones, and MnDOT OTSM tested and verified the implementation.

After completion of all tasks in this project, conversion tabs for two new WIM data formats are now available in BullConverter, which are for data conversion of Intercomp WIMLOGIX and Kistler DL controller files. In BullReporter, a new data retrieval function called “View Vehicles” is now available, which allows users to see individual vehicle records along with a graphical representation and a digital photo of the vehicle at the moment of crossing the site. Users have many selectable parameters that can be used for retrieval of a particular set of vehicles such as certain vehicle classes, weight ranges, etc. In addition, two new computational functions, GVW and ESAL histograms, have been implemented with three different options for selecting periods. Revisions also include several invisible improvements such as treatment of vehicles with more than 12 axles and internal data structures.

During the project, a total of 18 revisions (V4.7 – 4.24) were released for BullConverter and 12 revisions (V3.0 – 3.7, V4.0 – 4.3) for BullReporter. Each revision was a result of new implementation or modification of software codes, following MnDOT test results or recommendations. Overall, this project, a total team effort between UMD and MnDOT, has been successfully completed, with timely delivery of all required task deliverables.

One thing learned or demonstrated from this project is that as long as problems are clearly identified, they can be solved by a team effort between MnDOT and university researchers. If one future recommendation has to be made, it would be that MnDOT and university researchers should work together from the identification stage of a problem to formulation of a project.

REFERENCES

[1] T. M. Kwon (2009). *Development of Data Warehouse and Applications for Continuous Vehicle Class and Weigh-in-Motion Data (2009-33)*. Minnesota Department of Transportation, St. Paul, MN.

[2] S. Allamaraju (2010). *RESTful Web Services Cookbook*. Sebastopol, CA: O'Reilly.

[3] T. M. Kwon (2016). *KisDA: User Manual*. University of Minnesota Duluth, MN. Accessed from: http://www.d.umn.edu/~tkwon/download_KisDA/KisDA_UserManual.pdf

[4] IRD (2014). *Software Operator's Manual iAnalize*. Sask Canada: IRD. Accessed from: http://www.irdinc.com/public/uploads/downloads/1452713024_iAnalyze5.12_manual.pdf

[5] AASHO (1962). *The AASHO Road Test: Report 1, History and Description of the Project, Special Report 61A.* Highway Research Board, National Academy of Sciences, Washington, D.C. Accessed from: http://www.pavementinteractive.org/aasho-road-test/

[6] AASHO (1962). *The AASHO Road Test: Report 5, Pavement Research, Special Report 61E.* Highway Research Board, National Academy of Sciences, Washington, D.C.

[7] T. Kwon (2016). *BullReporter/BullConverter Vehicle Classification: How to Edit the Class Definition File.* University of Minnesota Duluth, MN. Accessed from: http://www.d.umn.edu/~tkwon/Download/ClassDefEditGuide.pdf

APPENDIX A:

WIMLOGIX DATA LOG COLUMN FORMAT

Column Number	Description	Example
1	Date (MM/DD/YYYY format)	8/17/2015
2	Time (HH:MM:SS.hh, 24 hour format, to hundredths of a second)	00:01:56.20
3	VehicleID (number, An incrementing counter for a given device of each vehicle seen)	248625
4	Lane (number, Device Specific, 1 or 2)	2
5	Speed (number, In increments of 0.1 miles per hour)	52.9
6	GVW (number, Gross Vehicle Weight, in increments of 1 pound)	49856
7	Length (number, Total Vehicle Length, in increments of 0.1 feet)	41.9
8	AxleCount (number, Total Axles Detected)	5
9	Class (string, as defined in the device's class configuration file)	9
10	WeightViolations (string, as defined in the device's class configuration file)	OS
11	ESAL (number, Graduations of 0.0001, uses values contained in device configuration). Not supported for the current version	1
12	ViolationCode (string, Intercomp Error Code)	AC
13	LicensePlate (string, the results of any license plate OCR camera)	
14	LicsensePlateState (string, two-letter state/country abbreviation, from OCR camera)	
15	USDOT (string, the results of any US-DOT OCR camera)	
16	OtherID (string, the results of any RFID, region specific ID OCR, etc.)	

17	AX1L (number, left weight of axle in increments of 1 pound)	4600
18	AX1R (number, right weight of axle in increments of 1 pound)	4542
19	AX2L (number, left weight of axle in increments of 1 pound)	5662
20	AX2R (number, right weight of axle in increments of 1 pound)	5882
21	AX3L (number, left weight of axle in increments of 1 pound)	5840
22	AX3R (number, right weight of axle in increments of 1 pound)	5310
23	AX4L (number, left weight of axle in increments of 1 pound)	5134
24	AX4R (number, right weight of axle in increments of 1 pound)	6444
25	AX5L (number, left weight of axle in increments of 1 pound)	2686
26	AX5R (number, right weight of axle in increments of 1 pound)	3756
27	AX6L (number, left weight of axle in increments of 1 pound)	
28	AX6R (number, right weight of axle in increments of 1 pound)	
29	AX7L (number, left weight of axle in increments of 1 pound)	
30	AX7R (number, right weight of axle in increments of 1 pound)	
31	AX8L (number, left weight of axle in increments of 1 pound)	

32	AX8R (number, right weight of axle in increments of 1 pound)	
33	AX9L (number, left weight of axle in increments of 1 pound)	
34	AX9R (number, right weight of axle in increments of 1 pound)	
35	AX10L (number, left weight of axle in increments of 1 pound)	
36	AX10R (number, right weight of axle in increments of 1 pound)	
37	AX11L (number, left weight of axle in increments of 1 pound)	
38	AX11R (number, right weight of axle in increments of 1 pound)	
39	AX12L (number, left weight of axle in increments of 1 pound)	
40	AX12R (number, right weight of axle in increments of 1 pound)	
41	AX13L (number, left weight of axle in increments of 1 pound)	
42	AX13R (number, right weight of axle in increments of 1 pound)	
43	AX14L (number, left weight of axle in increments of 1 pound)	
44	AX14R (number, right weight of axle in increments of 1 pound)	
45	AX15L (number, left weight of axle in increments of 1 pound)	
46	AX15R (number, right weight of axle in increments of 1 pound)	

47	SP1 (number, space between axle 1 and axle 2 in increments of 0.1 feet)	13.3
48	SP2 (number, space between axle 2 and axle 3 in increments of 0.1 feet)	3.2
49	SP3 (number, space between axle 3 and axle 4 in increments of 0.1 feet)	22.2
50	SP4 (number, space between axle 4 and axle 5 in increments of 0.1 feet)	3.3
51	SP5 (number, space between axle 5 and axle 6 in increments of 0.1 feet	
52	SP6 (number, space between axle 6 and axle 7 in increments of 0.1 feet)	
53	SP7 (number, space between axle 7 and axle 8 in increments of 0.1 feet)	
54	SP8 (number, space between axle 8 and axle 9 in increments of 0.1 feet)	
55	SP9 (number, space between axle 9 and axle 10 in increments of 0.1 feet)	
56	SP10 (number, space between axle 10 and axle 11 in increments of 0.1 feet)	
57	SP11 (number, space between axle 11 and axle 12 in increments of 0.1 feet)	
58	SP12 (number, space between axle 12 and axle 13 in increments of 0.1 feet)	
59	SP13 (number, space between axle 13 and axle 14 in increments of 0.1 feet)	
60	SP14 (number, space between axle 14 and axle 15 in increments of 0.1 feet)	

APPENDIX B:

BULLREPORTER/BULLCONVERTER VEHICLE CLASSIFICATION: HOW TO EDIT THE CLASS DEFINITION FILE

1. Introduction

Both BullConverter and BullReporter provide its own vehicle classifications based on a user selectable scheme. The text file that defines vehicle classes is called class-definition file, and it has a filename extension of either ".tye" (English units) which uses **pounds** and **feet** or ".tym" (Metric units) which uses **kg** and **cm**, respectively. These files are located in the following folders and it can be selected from a ComboBox provided within the software:
C:\Users***username***\AppData\Roaming\Bulldog\BullConverter
C:\Users***username***\AppData\Roaming\Bulldog\BullReporter

Name	Date modified	Type	Size
Log	6/30/2015 2:26 PM	File folder	
Reports	7/16/2015 9:56 AM	File folder	
min5.tye	4/23/2011 10:16 PM	TYE File	10 KB
min5.tym	4/23/2011 10:15 PM	TYM File	7 KB
min54.tye	4/23/2011 10:17 PM	TYE File	10 KB
min54.tym	4/23/2011 10:15 PM	TYM File	7 KB
min54.typ	11/2/2009 11:17 AM	TYP File	7 KB
Settings.ini	7/16/2015 7:19 PM	Configuration sett...	2 KB

Figure B-1 Typical class definition and other files in the AppData directory of BullReporter

Figure B-1 shows typical files and directories which are present in the AppData directory for BullReporter. Similar files are present in the BullConverter AppData directory. Multiple class definition files can co-exist, but only one of them can be selected by user. Vehicles are then classified according to the specification defined in the selected class-definition file.

A vehicle class is specified based on five parameters: (1) number of axles, (2) axle spacings, (3) axle weights, (4) axle groups, and (5) gross vehicle weight (GVW). Class-definition files define them and can be edited using any text editor, such as a Windows Notepad. Presently available ".tye" files developed by MnDOT TDA (Office of Transportation Data and Analyses) are

"min5.tye" → This is equivalent to FHWA
"min54.tye" → Class-16 was added, which is 5 axle semi-trucks with the last split tandem
"minn6.tye"

It is acknowledged that class-definition format and scheme were originally developed by IRD for their WIM systems, which was later adopted for MnDOT's own classification schemes. MnDOT classification schemes have been modified over the years and is expected to be continuously evolved. For reference, FHWA classification is shown in Figure B-2 at page B-7, which is the basis for MnDOT vehicle classification.

2. Format of *.tye and *.tym Files and Edit Information

One of the features added and different from the original IRD format of the class-definition file is the ability to add comment lines at the beginning of the file and in-line. The comment lines start with a “#” character at the first column (character) of the line. Comment lines are only allowed at the beginning of the file before the Header section. The actual definition starts after the comment block. In-line comments can be added at the end of data in each data line by starting again with a “#” character. Any text beyond a “#” in a line is ignored by the algorithm.

Overall, any class-definition file must consist of three sections: comment section, header section, and class-definition section. Each section must be separated by an empty line. Details of each section is described in the subsequent subsections.

Comment section: • First column of each line must start with a “#” character. • No blank lines are allows within this section, i.e., each line must have at least one character and it must start with “#”. • At the end of this section, there must be at least one blank line before the Header section.
Header section: • It consists of 9 lines, and each line must start with a numeric number which defines a parameter. • Not all parameters are used for classification. • A blank line is required before the Class-definition section.
Class-definition section: • A class is defined using a block of 10 lines, and each line within the block defines specific parameters of the class. • Multiple blocks can be used to define one vehicle class. • There must be one blank line between blocks to separate each block

2.1 Comment Section

Comment section is placed at the beginning of a class-definition file. Each line must start with a ‘#’ character, and the Comment section must be contiguous, i.e., no blank line or lines not starting with a ‘#’ character are allowed. An example of Comment Section (taken from “min5.tye”) is shown below.

```
#**************************************************************************
*
#English Unit Type definition file: Spacing=feet, weight=pound
#
# This file consists of Comment, Header, and Class-definition sections.
# Each section must be separated by one blank line.
#
# The Comment section consists of comment lines, each of which must have a '#' character at
# the first column. Comment lines must be placed at the beginning of the file and should be
# consecutive without any blank lines.
#
# The Header section consists of 9 lines. Each line defines a parameter used by the algorithm.
# The header lines must start with a numeric number, i.e., it cannot start with a space or
# a non-numeric character. The order of the lines must follow the defined order of parameters.
#
# The Class-definition section consists of many blocks, each block composed of 10 lines.
# Each block defines a vehicle class, and a blank line is required to separate blocks.
# One of more blocks can be used to define a single class.
#
# In-line comments can only be added in the Header and Class-definition section.
# In-line comments are added after numeric data by staring with a character '#'
# Any information entered after '#' character is simply ignored by the algorithm.
#
#
# History
# 12/5/2009, both Metric and English vehicle type files are accepted
# 11/28/2009, modified to allow comments at the beginning and in line
#
#
# At least one empty line is required before the start of the Header section.
#**************************************************************************
**
```

A blank line is required after Comment Section to indicate ending of the Comment section.

2.2 Header Section

Header Section consists of consecutive 9 lines, and each line must start with a numeric number. The order of the lines must follow the defined order of parameters as below.

Line #1: Compliance data available (not used)
Line #2: Maximum tandem axle spacing (feet or cm)
Line #3: Maximum tridem axle spacing (feet or cm)
Line #4: Maximum quadrum axle spacing (feet or cm)
Line #5: Minimum adjacent (tandem) axle spacing (feet or cm)
Line #6: Class number for fragmentation (0 or 1 axle vehicles)
Line #7: Class number for no matches with existing classes
Line #8: Vehicle type for auto-calibration (not used)
Line #9: Number of types defined (not used)

Among the Header lines, only Lines #2, #3, #4, #5, #6, and #7 are used by the algorithm (Lines #1, 8, and 9 are not used). An example of Header section (taken from min5.tye) is shown below. All spacing data is in feet since it is an English unit file.

```
1  # compliance data available: 1=yes, 0=no (not used)

8.0  #  maximum tandem axle spacing in feet

9.0  # maximum tridem axle spacing in feet

13.0  # maximum quadrum axle spacing in feet

3.33  # minimum adjacent (tandem) axle spacing in feet

15   # class number for totally undefineable classes, vehicle fragments (0 or 1 axle vehicles)

14   # type-14 is selected if the vehicle matches with none of the defined classes

24   # vehicle type for autocalibration (not used)

40   # number of types defined (not used)
```

In the above example, tandem axle group is defined by spacing of two adjacent axles that have less than 8 feet (maximum) and greater than 3.33 feet. Similarly, tridem axle group is defined by a group of three adjacent axles that have a total spacing less than 9 feet and each axle spacing greater than 3.33 feet. Quadrum axle group is defined by a group of four adjacent axles that have a total spacing of less than 13 feet and each axle spacing greater than 3.33 feet.

When axle group spacing limit is read in, the boundary is checked and if the value is outside the boundary, a default value is set. The boundary of maximum tandem axle spacing is greater than 6 and less than 10 feet. If user sets outside this range, it internally sets the default value 8. Similarly, the range for maximum tridem axle spacing range is greater than 7 and less than 11 feet, and the default value is 9 feet. The range of maximum quadrum axle spacing is greater than 11 and less than 15 feet, and the default value is 13 feet.

2.3 Class-definition Section

Class-definition Section consists of blocks of 10 lines. Each bock defines one vehicle class, and multiple blocks can be used to define one vehicle class. The 10 line block is defined as follows.

Line #1: Vehicle class number
Line #2: Number of axles
Line #3: Minimum axle spacing values from front to back (feet or cm)
Line #4: Maximum axle spacing values from front to back (feet or cm)
Line #5: Axle group types: s=single, d=tandem, t=tridem, q=quad, x=any
Line #6: Minimum axle weights from front to back (pounds or kg)
Line #7: Maximum axle weights from front to back (pounds or kg)
Line #8: Minimum GVW (pounds or kg)
Line #9: Maximum GVW (pounds or kg)
Line #10: Compliance (not used)

An example of class definition for class #2 in min5.tye (English unit) is shown below.

```
2   #vehicle class number
2   # number of axles
4.954   # min axle spacings from front to back
9.810   # max axle spacings from front to back
s  s  # axle group types: s=single, d=tandem, t=tridem, q=quad, x=any
0.000  0.000   # min axle weights from front to back
5008.903  6012.006   # max axle weights from front to back
0.000   # min GVW
9008.088   # max GVW
0      # FHWA compliance (not used)
```

Another example, providing a definition for class #9 (taken from min5.tye), is shown below without in-line comments.

```
9
5
6.004  1.969  8.990  1.969
29.003  10.007  67.257  8.005
s  d  d  x  x
3196.703  0.000  0.000  0.000  0.000
39994.059  39994.059  39994.059  39994.059  39994.059
0.000
213848.394
1
```

Please note that line #5 defines type of axle groups and it tells the following: the first axle is single; the second and third axles are tandem axles (a tandem axle-group); and the fourth and fifth axles can be any. Because the second and third axles are defined as tandem, the minimum axle spacing requirement of 3.3 feet and maximum of 8 feet is applied to belong to this class. In other words, axle grouping definition takes precedence over the defined spacing.

It should be noted that min5.tye file was converted from the original min5.tym. Therefore, the numbers include such as 39994.059, which was not rounded and should have been simply 40000 (pounds).

3. Classification Algorithm Implementation Based on Class Definitions

This section describes how BullReporter and BullConverter implement vehicle classification using the class-definition file.

First, all of the header information and class definitions are read in, and then they are organized based on the number of axles. For example, if a vehicle has two axles, the algorithm only searches the class-definition that has two axles, i.e., classes #1, #2, #3, #4, and #5. The details of search order follows below when axle spacing, grouping, and weight information of a test vehicle is given.

Step 1: Search and pick all class numbers that has the same number of axles as the test vehicle
Step 2: Keep only the class number candidates from Step 1 that meets the GVW range
Step 3: Test axle groupings and keep only the class numbers that meet the test vehicle's axle groupings
Step 4: Test the axle spacing of the test vehicle and keep the class numbers that meet the specified axle spacing ranges
Step 5: Test the axle weights of the test vehicle and keep only the class numbers that meet the specified axle weight ranges

If the five step test resulted in more than one class numbers, the first one in the remaining search pool is declared as the class number.

If a vehicle has zero or one axle, then the vehicle is declared class #15 without going through the five step test.

At the end of five-step test, if the vehicle belongs to none of class # in the definition, class #14 is declared.

If a vehicle has 7 or more axles, it then checks minimum requirement of axle spacing and weights. If all axle spacing is greater 2 feet and weight of each axle is greater than 400 pounds, class #13 is declared.

FHWA VEHICLE CLASSIFICATION

	CLASS GROUP		DESCRIPTION	NO. OF AXLES
	1		MOTORCYCLES	2
	2		ALL CARS CARS CARS W/ 1-AXLE TRAILER CARS W/ 2-AXLE TRAILER	2 3 4
	3		PICK-UPS & VANS 1 & 2 AXLE TRAILERS	2, 3, & 4
	4		BUSES	2 & 3
	5		2-AXLE, SINGLE UNIT	2
	6		3-AXLE, SINGLE UNIT	3
	7		4-AXLE, SINGLE UNIT	4
HEAVY TRUCKS	8		2-AXLE, TRACTOR, 1-AXLE TRAILER (2&1) 2-AXLE, TRACTOR, 2-AXLE TRAILER (2&2) 3-AXLE, TRACTOR, 1-AXLE TRAILER (3&1)	3 4 4
HEAVY TRUCKS	9		3-AXLE, TRACTOR, 2-AXLE TRAILER (3&2) 3-AXLE, TRUCK W/ 2-AXLE TRAILER	5 5
HEAVY TRUCKS	10		TRACTOR W/ SINGLE TRAILER	6 & 7
HEAVY TRUCKS	11		5-AXLE MULTI-TRAILER	5
HEAVY TRUCKS	12		6-AXLE MULTI-TRAILER	6
HEAVY TRUCKS	13	ANY 7 OR MORE AXLE		7 or more
	14	NOT USED		
	15	UNKNOWN VEHICLE TYPE		

Figure B-2 FHWA 13 Vehicle Category Classification (courtesy of FHWA)

APPENDIX C:

BULL-CONVERTER/REPORTER REVISIONS

Note: Documentations on some of the revisions are not available because they were never recorded. Below revision histories are provided as it is.

<u>BullConverter Revision History</u>

2/8/2017 (Version 4.22)

- IRD single file conversion was modified to "Convert Selected IRD Binary files to csv", i.e., all selected (multiple) files are converted
- ESAL and GVW computations are now computed based on the actual number of axles when the number of axles are greater than 12. Then, Error #6 is added.
- In the Export file, if Error #6 exists, ESAL and GVW are no longer recomputed and the original values are used, i.e. skip ESAL computation but check the original weight and GVW for overweight warnings.
- Minn6.tye file was produced from Minn6.tym and added to the Ben's Settings on both Bull Reporter and Converter

12/12/2016 (Version 4.21)

- Ian Vaagenes reported that export was not recalculating the warning based on new parameters. This error was fixed.
- It turned out to be caused by implementation of inclusive computation, i.e., it simply added the existing errors for 43 and 44 without checking the original.
- It is now changed to use the new computation results, i.e., it will now remove error 43, if it existed in the original error but it actually does not violate

11/12/2016 (Version 4.20)

- New external data type, 17, "1:SORT*", was discovered from IRD csv files. This new type threw an error so it was now taken care of.
- The software was further coded to take care of all types of future external data types, and if it is unknown, it will be printed on the Log file.
- It was tested for single and multiple image files, image + license, etc.

10/25/2016 (Version 4.19)

- If a vehicle exceeds the number of axle limit which is 12, it adds an error code 6 in the record.
- Changed to .Net Framework 4.5, VS2012.

6/30/2016, v4.18

- Bull-CSV format modification. Col#33:TypFile was modified to ESAL.
- TypFile information was entered in line 2.
- Line 2 format is changed to: -------;Version # (release date); typFile
- No error vehicle is left blank "" instead of "0"

- In export files, axle group overweight (43) and GVW overweight(44) warnings are added 0.

6/13/2016 (Version 4.17)

- Added a new tab specifically designed for Kistler DL data to Bull-CSV conversion
- HourFix algoritm for IDL applied. It fixes the problems associated with zero volumes in first part of hours in a day.

3/17/2016 (Version 4.14)

- Error: Intercomp WimLogix errors were not properly translated.
- It was caused by documentation error from Intercomp. 10th column “weightViolations” was used as errors, instead of 12th column ViolationCode. This error was fixed.
- Also changes include for single and zero-axle errors that this error was added to the errors of the existing ones in WimLogix.

1/5/2016 (Version 4.11)

- Temperature decoding was done successfully for all IRD binary files and implemented.
- More fields of IRD binary format were identified and incorporated to the documentation.
- Addition of classification based on Minnesota classification scheme (Minn6) was done.
- Moved single IRD file conversion to a single tab.

9/18/2015 (Version 4.10)

- Warnings 43, 44, 51, 52, 53, 54, and 55 are now included from this version
- Classification was updated for class-13. It now checks minimum requirement for 7 or more axles, 2ft minimum axle spacing, and 400 pounds of axle weights. This was globally adapted, which include all Bull systems

9/16/2015 (Version 4.9)

- Status code was finally cracked and incorporated into BullConverter
- Warnings 43, 44, 51, 52, 53, 54, and 55 are not included in the error since they are warnings with good data
- Excel displayed “37:38” into “37:38:00”, i.e., it added “:00” to convert to time format. It could be corrected by setting the column to text.

9/12/2015 (Version 4.8)

- Class-15 or error cases exit before translating the vehicle number, according to the binary data. This problem was found by Greg Wentz and fixed in this version, i.e., even if the vehicle record contains a fatal error, vehicle numbers are still correctly translated.

9/8/2015 (Version 4.7)

- Intercomp WIMLOGIX data conversion tab was added. Due to lack of data, the test was done with only one data file.
- IRD vehicle numbers now match with the number shown in the vehicle picture file.
- Two more columns were added: Temp, calFac. Temp column contains *pavement temperature* in Fahrenheit. The calFac column contains the multiplication factor applied to the original measurement for calibration.
- Advanced Installer was upgraded to 12.4 and used.
- .Net frame work now supports above 4.0.

BullReporter Revision History

5/4/2017 (Version 4.3)

- Modified "Rep 19: Top # GVW" and "Rep 25: GVW load spectra" to reflect changes in GVW of vehicles with more than 12 axles.
- View vehicles report: the search conditions display on top left header in web-page and pdf were not accurate due to new added conditions. This problem was corrected.
- Graphs of GVW and ESAL histograms were replaced from a bar graph to a true histogram graph. All bin widths are now equal.
- A copy button was added between the start and end dates textboxes. This allows to copy the start date text to the end date text, from which user modifies the end date. This approach is more convenient to users than typing both start and end dates.

10/25/2016 (Version 4.0)

- Users now can set low and upper limits for screening GVW
- For all histogram or load spectra, users now can set bin size and number of bins to control the histogram plot.
- All reports can utilize the options that allow exclusion of errors and warnings → It allows users to set which errors and warnings to be included or excluded from data screening or computation.
- Lane out-of-index is checked for undefined lanes and a proper error message is offered to users, instead of a crash.
- All global matrices were converted to local and passed through each procedure.
- Reorganized the ReportGen modules.
- Upgraded the current .net framework to 4.5 and VS to VS2012. This required a new version of TeeChart which was eventully purchased.
- All ESAL values are now printed up to 3 decimal places (Ben Timerson proposed up to 3 decimal places).
- View Vehicles with Good Weights Only report was removed, since it can be controlled and set using the regular View Vehicles Report with Error/Warning selection options

4/27/2016 (Version 3.7)

- Some of the pictures were not found because the vehicle number was displayed in 5 digits. This problem was fixed by extending the vehicle number range.
- Modified the viewveh.csv file to include the original image path. It now has the following columns, (Lane#, model_path, thumb_path, org_img_path) per vehicle.

4/3/2016 (Version 3.3)

- Finished the initial version of view vehicles report. This version creates structured directories for thumbnails and vehmodel pics.
- Finished the initial implementation of GVW histogram and ESAL Histogram.

2/5/2016 (Version 3.0)

- Upgraded the original ini settings to xml-based configuration. This was done by adding the Configuration project to the existing solution. It could be later changed to DLL.
- Modified vehClass to use the minimum tandem axle spacing specified in the *.tye file. This gives users a more control for vehicle classification.