Freight Advanced Traveler Information System (FRATIS) – Dallas-Fort Worth
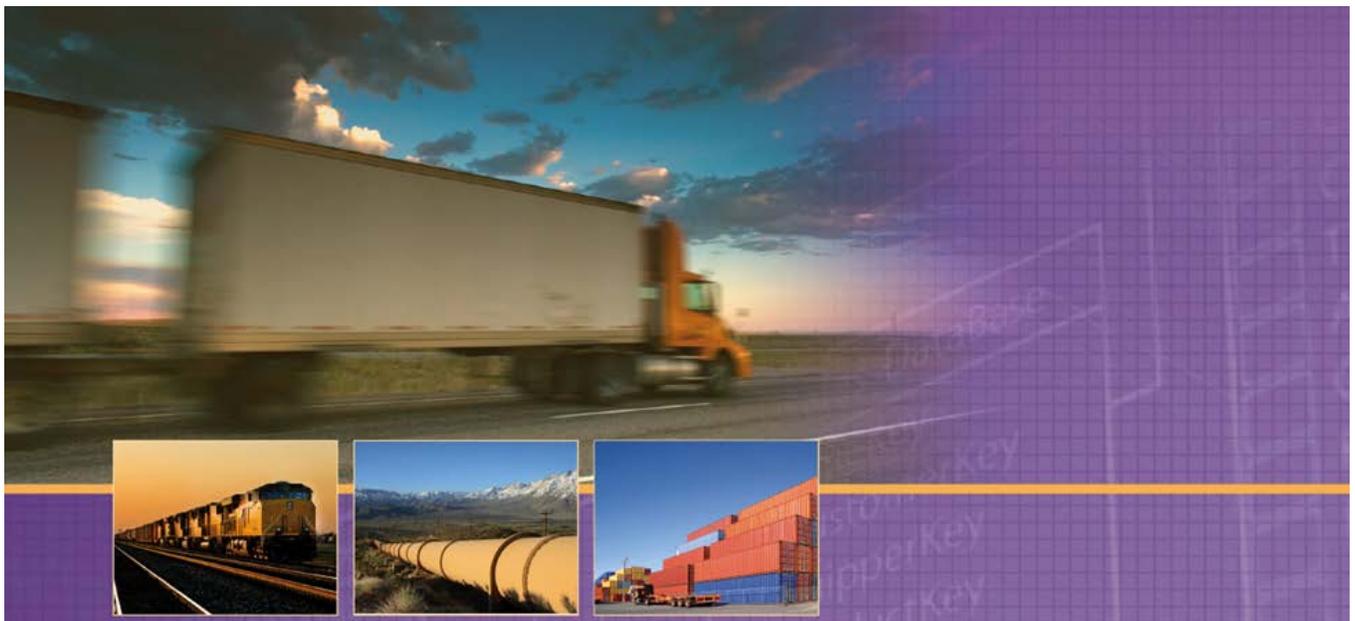
As-Built System Architecture and Design

Leidos

U.S. Department of Transportation
Federal Highway Administration

Produced by:

Contract DTFH61-11-D-00015, Task Order 12003, "Prototype Development and Small-Scale Demo for Freight Advanced Traveler Info System (FRATIS) – Dallas-Fort Worth"

Office of Operations, Office of Freight Management and Operations, Federal Highway Administration

Notice

TECHNICAL REPORT DOCUMENTATION PAGE

1. Report No. FHWA-JPO-15-219	2. Government Accession No.	3. Recipient's Catalog No.	
4. Title and Subtitle Freight Advanced Traveler Information System (FRATIS) Dallas-Fort Worth – As-Built System Architecture and Design		5. Report Date March 20, 2015	
		6. Performing Organization Code	
7. Authors Diane Newton		8. Performing Organization Report No.	
9. Performing Organization Name and Address Leidos 11951 Freedom Drive Suite 100 McLean, VA 20190		10. Work Unit No. (TRAIS) 0262	
		11. Contract or Grant No. DTFH61-11-D-00015, TO 12003	
12. Sponsoring Agency Name and Address United States Department of Transportation Federal Highway Administration 1200 New Jersey Ave., SE Washington, DC 20590		13. Type of Report and Period Covered	
		14. Sponsoring Agency Code FHWA – HOFM	
15. Supplementary Notes			
16. Abstract This document describes the As-Built System Architecture and Design for the FRATIS Dallas-Fort Worth DFW prototype system. The FRATIS prototype in DFW consisted of the following components: optimization algorithm, terminal wait time, route specific navigation/traffic/weather, and advanced notice to terminals. The high level architecture is described for each of the recommended component options.			
17. Key Words Freight Advanced Traveler Information System (FRATIS), system architecture, system design, prototype		18. Distribution Statement No restrictions	
19. Security Classify. (of this report) Unclassified	20. Security Classify. (of this page) Unclassified	21. No of Pages 43	22. Price N/A

Form DOT F 1700.7 (8-72)   **Reproduction of completed page authorized.**

Table of Contents

List of Figures

List of Tables

List of Acronyms

Acronym	Definition
4G/LTE	4th generation long-term evolution (4G/LTE)
AC	alternating power (AC)
ASD	after-market safety devices (ASDs)
BNSF	Burlington Northern-Santa Fe (BNSF)
BSM	Basic Safety Message (BSM)
C-TIP	Cross-Town Improvement Program
DFW	Dallas Forth Worth
DLWC	drop load with chassis" (DLWC)
DMA	Dynamic Mobility Applications (DMA)
DMS	Data Management System (DMS)
DOS	disk operating system (DOS)
DSRC	Dedicated Short-Range Communication (DSRC)
ECL	empty container leg (ECL) i
ED	export dray (ED),
FCL	full container leg (FCL)
FRATIS	Freight Advanced Traveler Information System (FRATIS)
FTP	file transfer protocols (FTPs)
Gmail	Google mail
GPS	Global positioning system
HOD	Hours of Duty (HOD)
HOS	Hours of Service (HOS)
HTML	HyperText Markup Language (HTML)
HTTP	Hypertext transfer protocol (HTTP)
IMAP	Internet message access protocol (IMAP)
IMCG	Intermodal Cartage, (IMCG)
IP	Internet Protocol
JRE	Java Runtime Environment (JRE)
LL	live load (LL)
MAC	media access control (MAC)
OSADP	Open Source Application Development Portal (OSADP)
PAI	Productivity Apex, Inc. (PAI)
PEWC	pickup empty with chassis" (PEWC
PII	personal identifiable information (PII).
PU	"Acknowledged Order to Pick Up" (PU)
RDE	Research Data Exchange (RDE)

Acronym	Definition
RESTful	Representational State Transfer (RESTful)
RSU	roadside unit (RSU)
SMTP	Simple mail transfer protocol (SMTP)
SQL	Structured Query Language (SQL)
TCP	transmission control protocol (TCP)
TFHRC	Turner-Fairbank Highway Research Center (TFHRC)
USDOT	United States Department of Transportation (USDOT)

1 Introduction

The United States Department of Transportation (USDOT) Dynamic Mobility Applications (DMA) program, undertaken in 2009, consistently sought to take advantage of the increasing volume of data generated within the transport system and to build on recent success stories in using these data to improve mobility and efficiency. Through the DMA Program - the Freight Advanced Traveler Information System (FRATIS) Concept Development and Functional Requirements task order was undertaken to further engage the freight transportation community in the process of documenting the concept, user needs, and high-level requirements for the next generation FRATIS.

The FRATIS Dallas-Fort Worth (DFW) Prototype was one of three regional prototypes funded by the DMA program in 2012. Prototype design began in the fall of 2012, and the prototype components were implemented via a r olling deployment beginning in January 2014. Baseline data collection began in June 2013; prototype data collection began in December 2013.

This document describes the as-built system architecture and design for the FRATIS prototype as installed in the DFW Region. Combined into a s ingle document, the Architecture and Design for the FRATIS DFW prototype includes the following:

- Section 2 pr ovides an overview of the FRATIS DFW prototype purpose and scope. This section also identifies the stakeholders who provided input for and expressed an interest in the prototype system design.
- Section 3 provides a general system description and architecture overview, including the design rationale, or the reasoning and justifications that led to the system as designed.
- Section 4 details the design of each prototype system component. This section includes the component-level diagrams.

This document is directed to readers who are familiar with the FRATIS DFW Prototype project and the work that has been conducted within this task order.

2 System Purpose

2.1 GOALS FOR FRATIS DFW

Broad goals for the FRATIS bundle of applications are identified within the FRATIS Concept of Operations published in 2012 by the USDOT (written by Cambridge

Systematics).[1] High-level in nature, these goals were presented to the local stakeholders in Dallas for review; subsequently, the Leidos prototype development team (henceforth referred to as "the development team") articulated a subset of more specific, targeted goals for the FRATIS-DFW prototype test:

- Reducing empty moves and bobtail miles
- Reducing the manual effort required for dispatchers to schedule orders
- Increasing fleet productivity due to improved efficiency – i.e., completing more orders with the same or fewer resources.
- Providing insight into queue times at local freight facilities
- Resolving sporadic, unpredictable queue times at gates
- Providing higher visibility for equipment information from steamship lines in terms of what types of containers were available for return to the participating container yard
- Giving advance notice to terminals regarding dray traffic destined for their facility
- Providing truck-specific routing tied to a driver's work orders and order origin, including destination and route
- Establishing a link between traffic, weather, and routing

2.2 STAKEHOLDERS

2.2.1 Users

Primary users of the FRATIS DFW System included a small group of freight transportation providers in the DFW Region.

- Optimization program:
  o Associated Carriers (Dispatch and Management)
  o Southwest Freight (Dispatch and Management)
- Terminal queue time:
  o Intermodal Cartage (IMCG), Wilmer, TX
  o Associated Carriers (Dispatch and Management)
  o Southwest Freight (Dispatch and Management)
- Advance notice to terminals:
  o Associated Carriers
  o Southwest Freight
  o IMCG-Wilmer
  o Burlington Northern-Santa Fe (BNSF)
- Routing, scheduling, traffic, and weather:

---

[1] http://ntl.bts.gov/lib/54000/54100/54104/12-065.pdf

- o Associated Carriers (Drivers)
- o Southwest Freight (Drivers)

2.2.2 Providers and Vendors

The FRATIS DFW Prototype relied on s everal external vendors to provide multiple components, as outlined below:

- Optimization program (Associated Carriers):
    - o Productivity Apex, Inc. (PAI)
- Bluetooth/Wi-Fi wait time hardware and software:
    - o Acyclica
- Dedicated short-range communication (DSRC) equipment:
    - o Loaned from the USDOT connected vehicle test bed
    - o Radios and antennas: Arada Systems (Arada)
- Dispatching software (inputs to optimization program):
    - o Trinium Technologies
- Real-time work order, routing, scheduling, traffic, and weather:
    - o Trinium Technologies

3 Prototype Description

3.1 PROTOTYPE ARCHITECTURE

Figure 1 illustrates the prototype architecture, and the relationship between the following components. Section 4 provides a detailed description of each component.

1. Drayage optimization
2. Terminal queue time:
    a. Bluetooth/Wi-Fi
    b. DSRC and basic safety message (BSM, connected vehicle)
3. Routing, navigation, traffic, and weather
4. Advance notice to terminals

Leidos
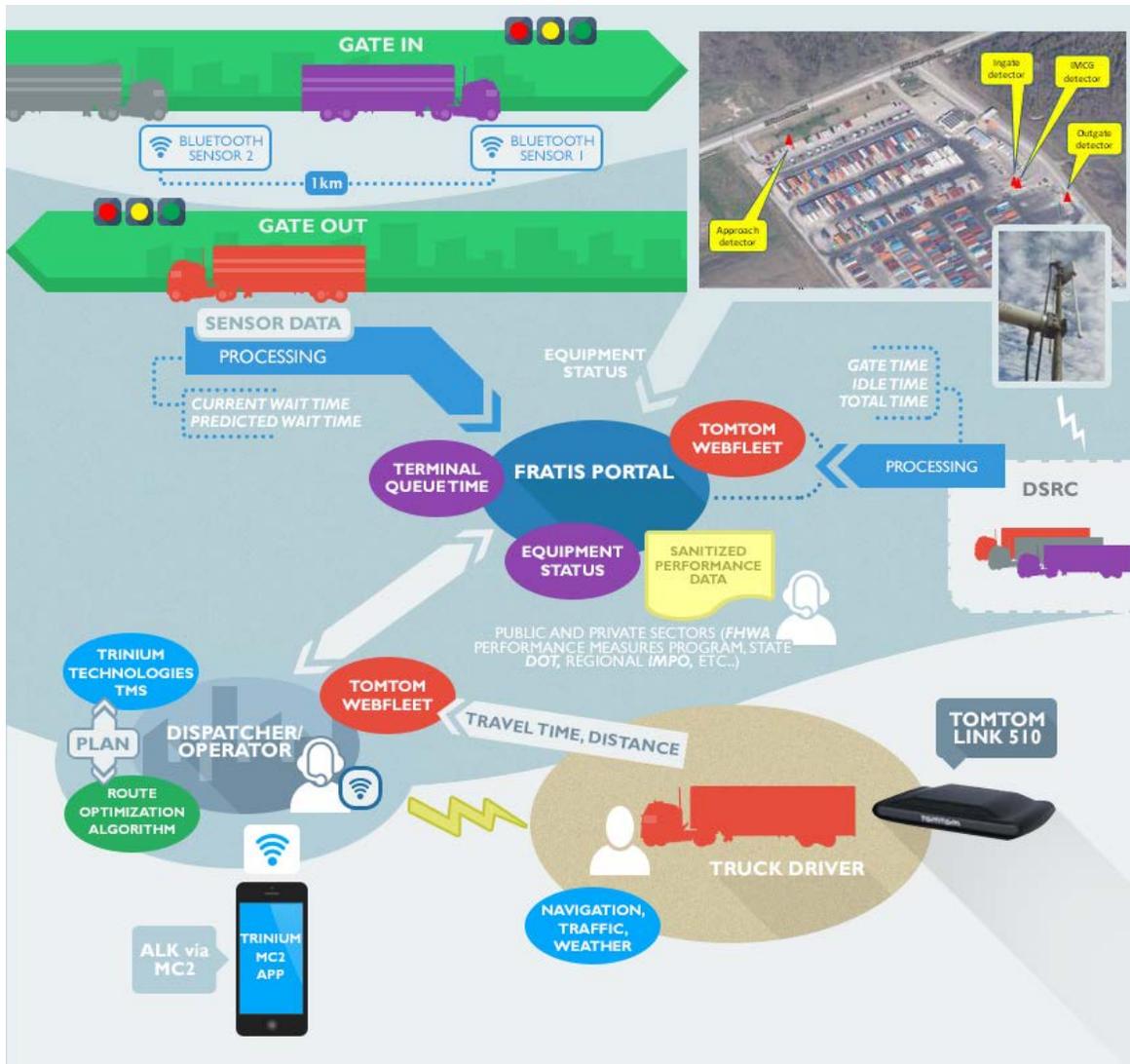
Figure 1. FRATIS DFW Architecture

3.1.1   Component Views

3.1.1.1   Optimization Program "Vesco"

The optimization program used by Associated Carriers was titled "Vesco" and consisted of the following three components:

1. Pre-optimization processor (developed by Leidos)
2. Optimization algorithm (developed by PAI)
3. Post-optimization processor (developed by Leidos)

These components are discussed in detail in Section 4.1.

Vesco takes the scheduled moves for the next day from a single carrier and creates a load plan for each driver that minimizes the number of trucks needed and the number of bobtail (tractors without a trailer or chassis) moves required; a schematic for this process is presented in Figure 2.
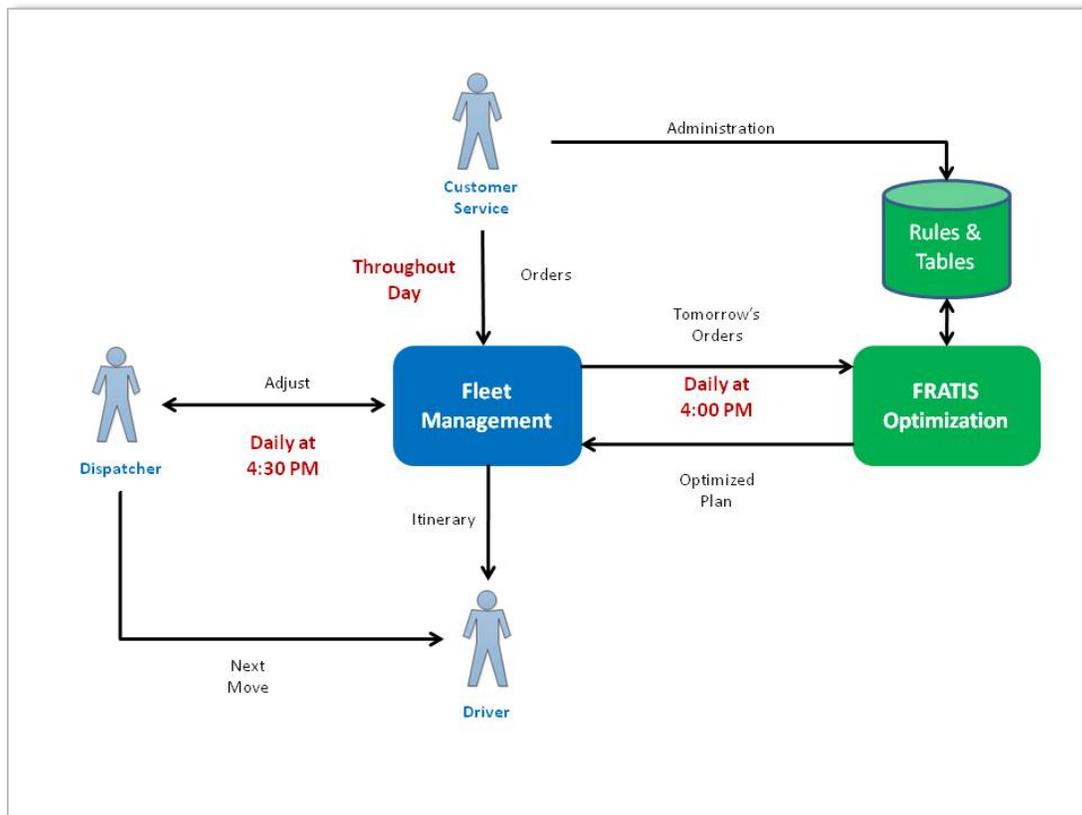
Leidos

Figure 2. Optimization Process

3.1.1.2 Alternate Optimization Program

Despite multiple attempts to make Vesco work satisfactorily for Southwest Freight between March and September of 2014, Southwest was not willing to continue using the program for the duration of the prototype. However, in noting the issues identified with Vesco by the Southwest users, the development team felt that a program which focused solely on reducing bobtails may meet the majority of the requirements for Southwest.

Identified as the "Alternate" Optimization Program, this program worked in a similar fashion as Vesco in that there was a pre- and post- optimization processor, but the

algorithm that powered the program was different. The program was run by Leidos and daily plans were emailed to Southwest Freight users. A detailed schematic of the process followed by the program is outlined in Figure 3.
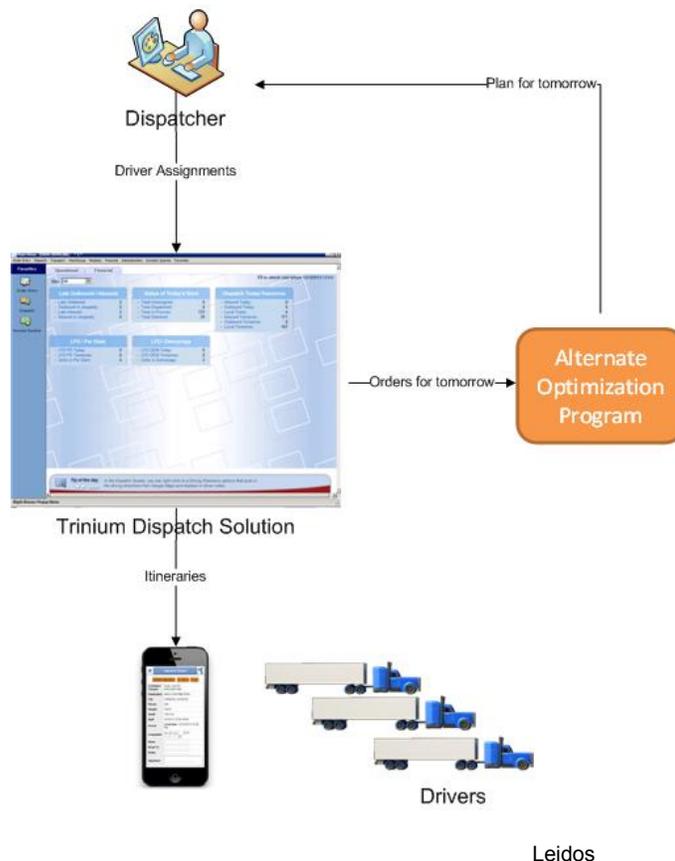
Figure 3. Southwest Freight Alternate Optimization Program

3.1.1.3   Terminal Queue Time – Bluetooth/Wi- Fi

The terminal queue component was deployed at the IMCG Container Yard in Wilmer, Texas. The objective of this component was to provide various users with the current expected time to enter a facility and to complete container drop-offs or pickups within the facility. In addition, the component provided predicted times for those activities by time of day and day of the week, based on historical data.

Bluetooth/Wi-Fi technology was deployed at the facility which measured the current wait time for each activity. Antennae and readers were placed at strategic locations within the facility to detect whenever a Bluetooth or Wi-Fi enabled device passes certain points. The information is transferred to a server, which correlates the times of passing trucks from the same device to calculate the desired metrics. This approach uses those

trucks that have an operating device as the sample population. It identifies and excludes devices that are not in a truck by eliminating any device that does not pass through a truck gate. The goal sample size for the queue time technology was 20% of all trucks entering the facility; Bluetooth provided 5% of the sample size, whereas Wi-Fi provided 20% of the sample size, satisfying the prototype team's target size.

This wait time information was forwarded to the FRATIS server every 15 minutes, which made it available to authorized users via the FRATIS portal. The information was also stored and used to predict expected times. The current wait times were stored on the FRATIS server and were used to calculate predicted wait time based on historical wait times. The current wait times were averaged out over each hour, and then weighted as one seventh (1/7) of a week to calculate predicted wait time by day of week.

The architecture for the terminal queue component is outlined in Figure 4.
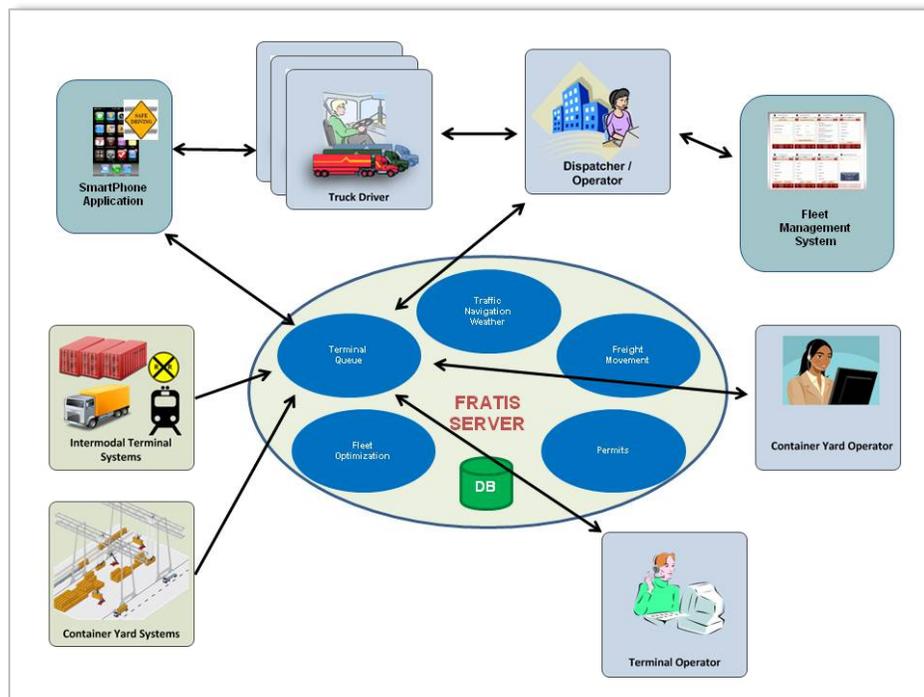
Leidos

Figure 4. Terminal Queue

3.1.1.4   Terminal Queue Time – DSRC

A unique aspect of the FRATIS DFW prototype included pilot testing of DSRC technology to calculate queue time at a facility. For research purposes, the USDOT connected vehicle program loaned the necessary equipment for a temporary installation

at the IMCG-Wilmer facility and radios to equip four of the IMCG drayage trucks. The on-board radios transmit the BSM, while the infrastructure unit receives the data necessary to calculate the terminal queue from those trucks whenever they are at the facility. Per the standard roadside architecture for DSRC technology, the DFW prototype used a roadside unit (RSU) to communicate to the in-vehicle devices. BSMs were transmitted to the data exchange service via the Internet and stored for later analysis.

3.1.1.5 Routing, Navigation, Traffic and Weather

Initially, the deployment team planned to purchase a commercially available routing and navigation product that was capable of receiving traffic and weather information and incident alerts for the participating drayage company drivers. The team evaluated the ALK Technologies, Inc. (ALK) Co-Pilot product, a mobile application, and the TomTom Pro 7150 (an in-vehicle device) for use with the prototype. As part of the assessment of implementation options for the FRATIS DFW prototype, the development team evaluated several products:

- ALK CoPilot Truck –an application from ALK Associates that can be installed on a variety of devices that provides a routing and navigation application specific to truck drivers. For an additional fee, real time weather and traffic can be integrated into the application.
- TomTom Pro 7150 Truck – an onboard device that contains a routing and navigation application specific to truck drivers. It includes real time weather and traffic information.

The development team was in the midst of exploring the ALK Co-Pilot application, when in related talks with Trinium Technologies, the dispatch software provider, it was revealed that they intended to enhance their web-based work order application MC2 to include routing, navigation, traffic and weather information via an interface with a similar ALK product. The development calendar and pl anned roll out coincided with the prototype schedule, which eliminated the need for the drivers to have multiple applications open on their smartphone. The MC2 web application integrated with the ALK Maps product as the back end tool for the routing, traffic, and weather information.

3.2 FRATIS DFW PORTAL

While the FRATIS DFW prototype consisted of several individual components, a central location was needed to consolidate information from these components for presentation to the end user. In addition, certain components such as the routing, navigation, traffic, and weather application were available only to the drivers. Creating a s eparate,

protected website provided secure access these applications to other users. In addition, it allowed presentation and information sharing regarding certain data not originally included in the scope or vision, for example, the equipment availability at IMCG-Wilmer from the steamship lines.

The FRATIS DFW website included:

- Traffic and weather information (for dispatcher use), see Figure 5
- Display of the real-time and predicted wait time at IMCG-Wilmer, see Figure 6
- Display of the IMCG-Wilmer yard status (which articulates what type of containers are being accepted at the yard that day), also visible in Figure 6
- A link to the TomTom Webfleet user interface:
  - o 50 TomTom Link 510 devices were deployed among the two drayage companies for evaluation purposes; however, at the start of the prototype, the drayage companies were given access to their Webfleet accounts for fleet monitoring purposes.
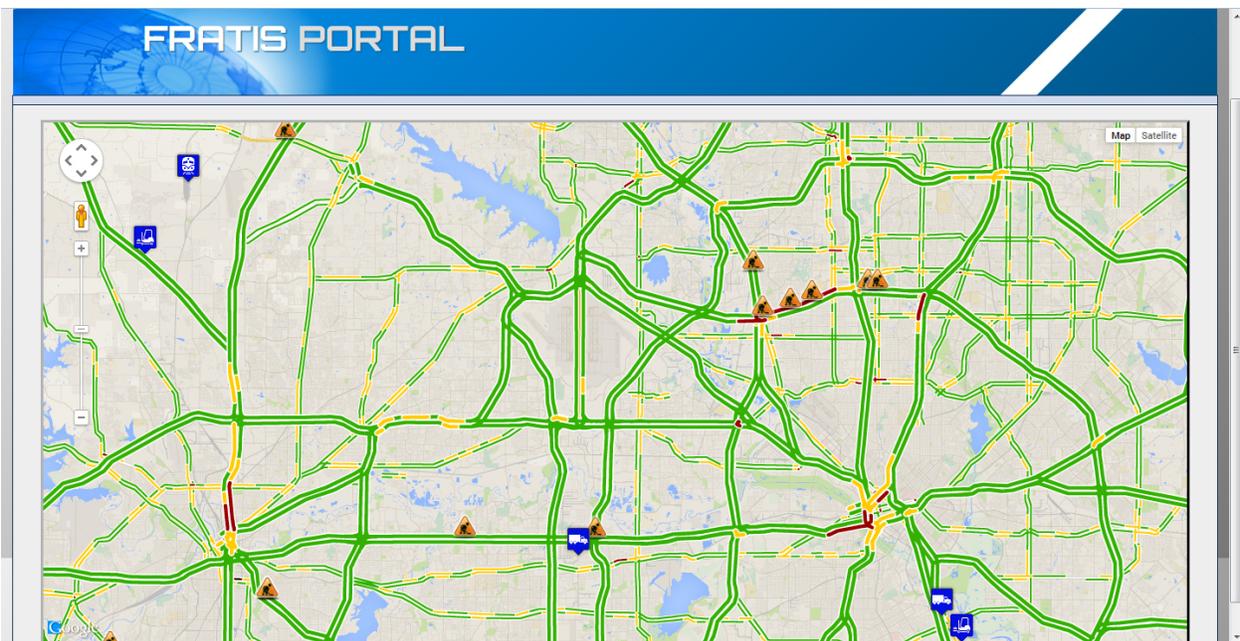
Leidos

Figure 5. FRATIS DFW Screen Capture of Traffic Information
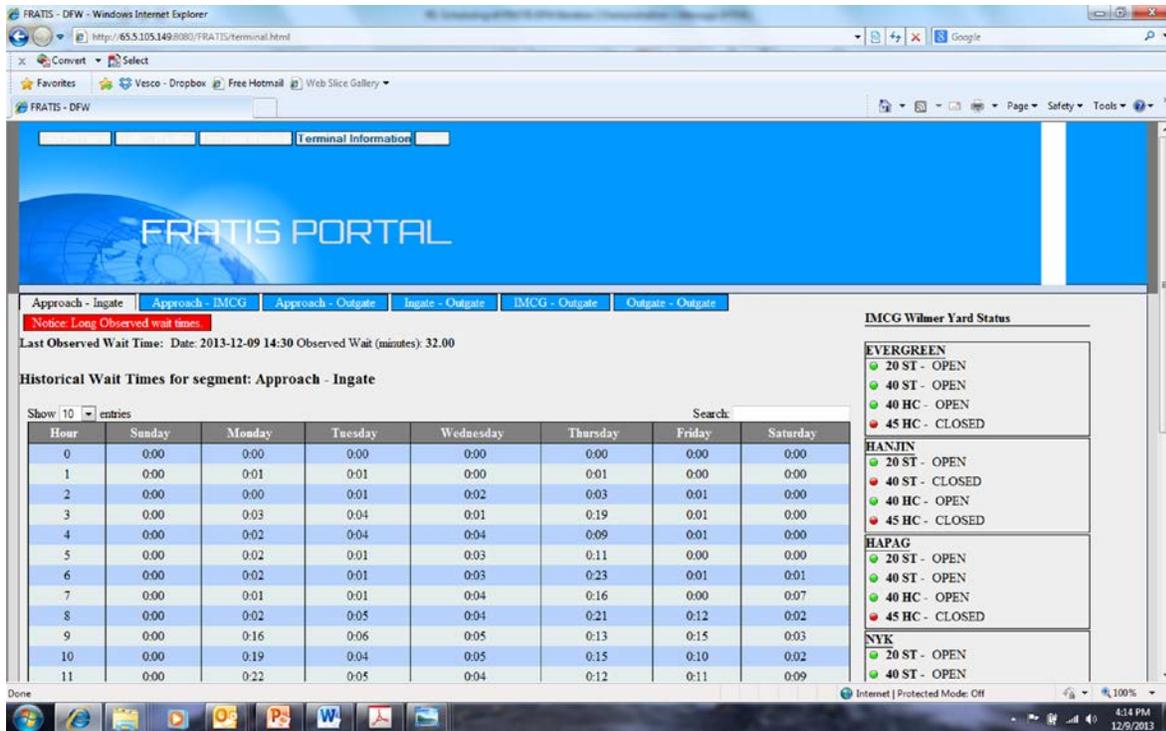
Leidos

Figure 6. FRATIS DFW Website – Display of Terminal Wait Time and Yard Status

Specifications for the FRATIS DFW website were:

- Platform and content:
    - Java, Javascript
    - Hypertext Markup Language (HTML)
    - Structured Query Language (SQL)
- Application server and data storage:
    - Glassfish Application Server
    - MySQL
- Communication protocols:
    - Hypertext transfer protocol (HTTP)
    - Simple mail transfer protocol (SMTP)
    - Internet message access protocol (IMAP)

3.3 DESIGN RATIONALE

While the development team began the effort with a design in mind, as the project evolved, decisions were made based on the activities of Leidos' partners and involved stakeholders with respect to deploying individual components, as outlined below.

3.3.1 Optimization Programs

3.3.1.1 Vesco

The original scope of work for the FRATIS DFW prototype specified using the PAI optimization program; however, this code existed mainly in a "black box" and the development team did not seek to alter this code for use with the DFW prototype. PAI's original code and documentation is housed on the USDOT's Open Source Application Development Portal (OSADP).[2] Given the fact that the optimization program was developed by PAI and documentation was limited, the development team's efforts focused on designing a front- and back-end interface for the program per the requirements of the drayage company users as opposed to altering PAI's code.

Once limited prototype testing began, the drayage company users identified issues with the program's algorithm and a subcontract was executed with PAI to incorporate these custom code changes to the algorithm. The rolling deployment of the optimization program began in February 2014, yet code changes to the optimization program and the pre- and post- optimization processing components continued through the summer of 2014.

In the fall of 2014, Associated Carriers consolidated their Houston operations into their DFW office, effectively doubling the amount of work to be handled by the same amount of staff. Since staff were busier, this impacted their ability to reliably generate optimized plans daily, yet they were pleased with the optimized plans that were being created. At that time, the delivery method for the program was changed so that the receipt of the Trinium Technologies input files would trigger the program to run remotely and email optimized plans back to the designated list of associated users.

3.3.1.2 Alternate Optimization Program

The other participating dray, Southwest Freight, made the decision that regardless of the constraints that were added to the algorithm, the daily plans created by Vesco were not suitable for their needs. In particular, Southwest Freight noted that Vesco:

---

[2] Federal Highway Administration, "Open Source Application Development Portal" web page. Available at: www.itsforge.net

- Did not prioritize loaded moves over empty moves
- Still contained many bobtails
- Did not evenly distribute the day's moves among available drivers

Southwest Freight agreed to participate in testing of a second optimization program, referred to as the "Alternate Optimization Program" which utilized optimization code initially created for the Kansas City Cross-Town Improvement Program[3] drayage company stakeholders. The development team started with this code, but carefully considered the issues that Southwest Freight had noted with Vesco (identified above) to re-write and customize the code. Consequently, where the objective function of the PAI algorithm in Vesco was to minimize total miles traveled as a fleet, the Alternate Optimization Program strictly sought to minimize bobtails.

3.3.2 Terminal Queue Time

With the exception of railroad yards, the profit margin at intermodal facilities and carriers is small. Technology will not be adopted if the cost is too high. Bluetooth and Wi-Fi have the lowest cost to facilities and carriers. An estimated $10,000 is a price that facilities could realistically afford. The Wi-Fi readers are able to detect four times the number of devices, which gives a more statistically significant sample size than Bluetooth readers.

Relatedly, DSRC probes would provide the most accurate results and could be enhanced to communicate additional information between the facility and the driver. However, at this time there are no trucks with this capability and the cost to install DSRC radios on a truck is prohibitive. Given the USDOT's investment in this technology and the probability that the technology may become more widespread in the near future, the development team, in collaboration with the USDOT, made the decision to perform a 30-day pilot test of the technology as part of the FRATIS prototype. The development team will compare the wait times collected and calculated via the two methods and provide an analysis as part of the final report documentation for the prototype.

3.3.3 Routing, Traffic and Weather Application

As articulated in Section 3.1.1.5, in the course of working with Trinium Technologies on the optimization program, it was discovered that Trinium Technologies intended to deploy an enhanced work order application to the participating drayage companies which integrated real-time routing, traffic, and weather with the work order information provided via their mobile application, MC2.

---

[3] http://www.ops.fhwa.dot.gov/freight/technology/best_practice/index.htm

However, utilizing this approach meant that the road conditions, routes provided, and alternate routing information provided to the drivers via this application were not sent to the FRATIS DFW server. Trinium Technologies' application did not track the drivers' routing decision, nor did it provide archived information to a back end database for the prototype development team to access or communicate to.

Leidos elected to use this feature as the routing, traffic, and weather component of the prototype. Although the functionality was different and in some cases, not as thorough as the direct, stand-alone Co-Pilot application, several factors drove the decision:

- Familiarity with the application; drivers were already using Trinium Technologies' MC2 application. The enhanced MC2 application added the routing, traffic, and weather directly to each work order, as used by the driver.
- The rollout of the MC2 application coincided with the timing of the FRATIS prototype.
- Lack of support from the drayage companies and Trinium Technologies if Leidos had selected an alternate solution and attempted to integrate it with the work orders provided.

3.3.4 Advance Notice to Terminals

The development team's close collaboration with Trinium Technologies during the creation of the dynamic queries needed to create input files for the optimization report uncovered additional functionality for the prototype that was not initially identified. The Trinium Technologies team provided key insights into what data were available from their system and how often it could be provided. In addition, the drayage companies, in return for insights into terminal activities at the local container yard, were open to the idea of sharing data. Since the data were already being collected, the development team decided to use these data for a new, unplanned component of the prototype.

Using the dynamic queries that had been created for the optimization program, the development team scheduled a process to pick up these reports from an email inbox and created code to parse the data from these reports into an email presentation intended for IMCG's Wilmer and Haslet facilities as well as BNSF Haslet yard.

4 Component Design

4.1 OPTIMIZATION PROGRAM 'VESCO'

Vesco was used by Associated Carriers, and included the optimization algorithm developed by PAI. The development team built a pre-optimization and post-optimization

processor to complete the program, which are described in detail in the sub-sections below. The overall Vesco optimization process is outlined in Figure 7.
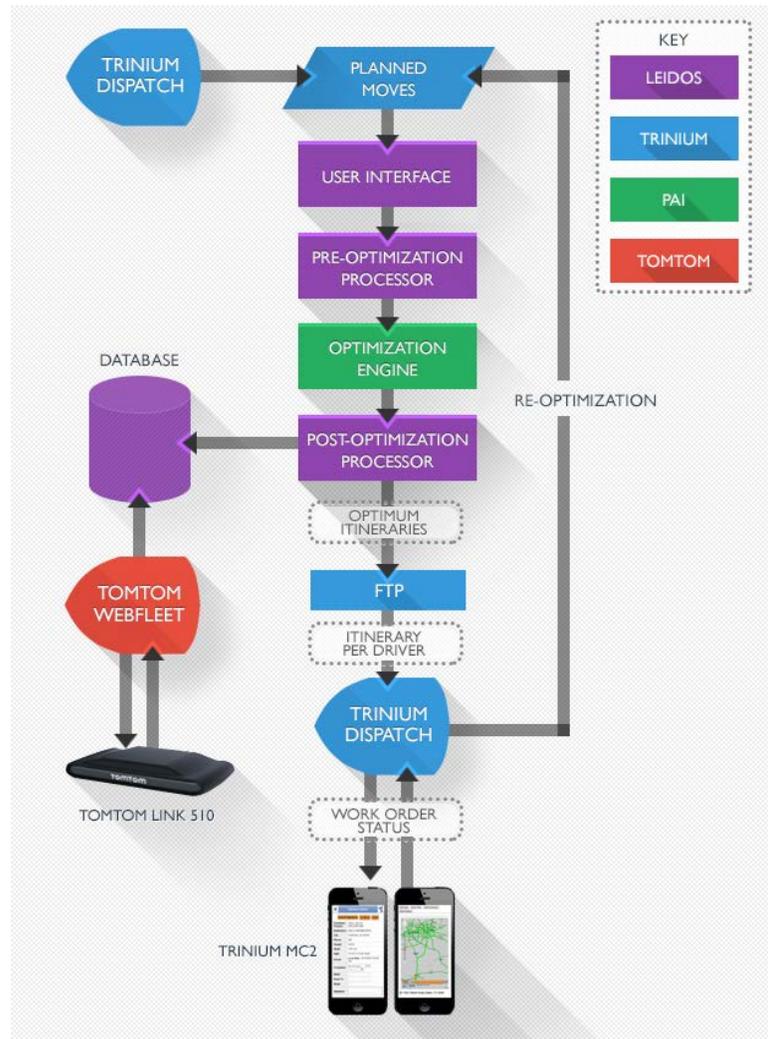
Leidos

Figure 7. Vesco Optimization Program

A scheduled process extracted a comma delimited file from the Trinium Technologies fleet management system; this file contained the following information for each scheduled move:

- Move type
- Origin
- Destination
- Pickup window

- Drop-off window

The pre-processor transformed the file and added any additional information (e.g., mileage) required by the algorithm. The optimization algorithm created the most efficient load plan by distributing the moves to drivers and specifying the sequence in which they should be m oved. The post-processor added estimated schedule times based on predictive travel time and terminal wait time. It also stored the load plan for future use.

4.1.1 Pre-Optimization Processor

The pre-optimization processor included the following functions:

- Coordinated with the Trinium Technologies' dispatching software on q uery development
- Used macro to translate Trinium Technologies queries into format required by PAI optimization engine
- Created an interface to automatically trigger the optimization program and generate an optimized plan

4.1.1.1 Query Development (Optimization Inputs)

The development team worked with the drayage company dispatching software vendor, Trinium Technologies, to develop system queries which identified moves planned for the next day. These were created as "dynamic queries" and required tweaking, as Trinium Technologies identifies moves by "legs," and in some cases, Leg 1 of an order is completed on one day, while the remaining leg or legs may be completed sometime later, not necessarily on that same day. In addition, both drayage companies perform live loads (LLs) and unloads. The queries had to be carefully constructed in order to keep all legs of an order that contains an LL/unload together, since the driver of these loads waits while this activity takes place.

The output of the Trinium Technologies queries includes comma separated files that are emailed to designated recipients at specific times of day. One set of queries was run at 4 p.m. Eastern Standard Time (EST) containing all orders planned for the next day. In addition, a second set of queries was run at 10 a.m. EST. The first query contained all orders planned for the remainder of the day, while the second query contained all orders that had been completed up to that point for the day. This second set of queries was used to "re-optimize" the orders for the same day.

Where the Trinium Technologies system used a leg-based approach, the PAI optimization algorithm used a "stop" based approach, which assumes that an order will

make a specified number of stops and that a freight action occurs at each stop. The freight actions used by PAI are identified in Figure 8:

FREIGHT ACTIONS	STATES
Pickup Loaded (PL)	Loaded
Pickup Loaded with Chassis (PLWC)	Loaded
Live Loading (LL)	Loaded
Pickup Empty (PE)	Empty
Pickup Empty with Chassis (PEWC)	Empty
Live Unloading (LU)	Empty
Drop-off Empty (DE)	Chassis
Drop-off Loaded (DL)	Chassis
Pickup Chassis (PC)	Chassis
Drop-off Empty with Chassis (DEWC)	Bobtail
Drop-off Loaded with Chassis (DLWC)	Bobtail
Drop-off Chassis (DC)	Bobtail

Figure 8. Optimization Program Freight Actions (PAI)[4]

4.1.1.2   Macro

The macro built by the development team translated the Trinium Technologies comma separated files from the leg-based order structure to the "route-stop" format required by the PAI algorithm. This new route-stop based file included all fields required by the algorithm as well as the requested fields from the Trinium Technologies files that the drayage companies wanted to view within the optimized plan. The following data elements were associated with each order: container number, steamship line, leg type (full or empty), dispatch sequence number, and whether the load was an LL or unload (indicated by a "yes" or "no" entry).

The PAI algorithm needed:

- Job number – this was the same as the order number from the Trinium Technologies software
- Sequence number – this was the same as the leg number in the Trinium Technologies software
- Stop action (per Figure 8)
- Location name
    o Street address

---

[4] Productivity Apex, Incorporated, "Design and Logic Review", November 2012.

- Stop delay
- Appointment times:
  - o Window start
  - o Window end

The macro built by the development team relied on the mapping of the Trinium Technologies software order structure to the order structure required by the optimization program. The development team built one mapping file for Associated Carriers and a second for Southwest Freight, since their operations were slightly different. The macro built by the development team translated each order from a structure used by the drays and Trinium Technologies software to the order structure required by the PAI algorithm.

In Table 1, the Trinium Technologies software order on the left would be described as an export dray (ED), where the empty container leg (ECL) is picked up at location A and taken to location B, where an LL occurs. Then, the full container leg (FCL) is taken from location B and delivered to Location C, its final location. However, the PAI algorithm would describe this same type of order as a series of freight actions. The first action that the driver executes is a "pickup empty with chassis" (PEWC) at Location A. Then, the empty container is loaded at location B (LL). Finally, the driver will execute a "drop load with chassis" (DLWC) freight action at Location C.

Table 1. Sample Mapping

TRINIUM ORDER STRUCTURE					ALGORITHM MAPPING			
Dispatch	Pickup	Delivery	Leg	LL	Stop Action	Location	Name	Description
ED	A	B	ECL	Yes	PEWC	A	Export 1	Export with Live Load
ED	B	C	FCL	Yes	LL	B		
					DLWC	C		

The development team had to create mappings for every potential order type for both carriers and implement these mappings within the macro. Each mapping was identified with a unique name (in the sample above, this order type is called "Export 1"). If an order did not follow one of these mappings, Vesco would not assign that order within the optimized plan.

4.1.1.3 Running Vesco

Vesco is a server application written to generate optimization plans. Vesco checks a generic email account for new queries sent by the Trinium Technologies software. When one is received, Vesco formats the plan using the macro mentioned above, and runs it through the PAI optimization algorithm to generate an optimized plan. Vesco then emails the optimized plan to the corresponding freight company and uses file transfer

protocols (FTPs) to send a stripped down version of that plan to the Trinium Technologies software. Executed moves are also run through the optimizer to generate re-optimized plans.

Parameters for the program are outlined below:[5]

- **Tools and language:**
    o Microsoft Visual Studio 2010
    o C#
- **Prerequisites:**
    o .NET Framework 4.5 or newer
    o Microsoft Excel
    o Access to the Internet to receive daily plans from an email inbox (in the case of the FRATIS DFW, the team used a dedicated email account in Google mail [Gmail])
    o Outgoing access across transmission control protocol (TCP) Port 587 (email port for Gmail)
- **Running Vesco:**
    o From disk operating system (DOS) prompt, navigate to the location of "VescoConsole.exe" and type in the program name "VescoConsole.exe" and hit enter
    o A user may also double click "VescoConsole.exe" to execute the program

4.1.2 Optimization Algorithm

The optimization algorithm built by PAI is an "ant colony" algorithm, which is an algorithm for finding optimal paths that is based on the behavior of ants searching for food.[6] The theory behind this is that the base or initial state is such that the "ants" wander randomly as they search for food but over time, the shortest path is found and all ants eventually follow this path. In the case of the PAI algorithm, the objective function of the optimization is to maximize drayage operations efficiency through:[7]

- Maximizing value added moves (loaded)
- Minimizing non-value added moves (bobtail, empty, and chassis)
- Maximizing load matching and backhauls

---

[5] Version numbers were not included in the compilation of the name of the .exe so therefore are not relevant.
[6] http://mathworld.wolfram.com/AntColonyAlgorithm.html
[7] Productivity Apex, Incorporated, "Design and Logic Review," November 2012.

As-is, the algorithm's constraints include:[8]

- Time window at each stop
- Driver hours of service (HOS)
- Driver hours of duty (HOD)
- Equipment owner
- Equipment size
- Equipment availability
- Equipment acceptance
- Permission to reload

In addition to the constraints built into the PAI code, Associated Carriers requested additional customization to the program, as outlined below:

- Hazardous orders needed to be assigned to hazardous-certified drivers.
- Associated Carriers provides local and regional trips. Certain drivers prefer the regional trips, and therefore requested that the algorithm understand and identify these regional orders so that they could be assigned to drivers who prefer to complete these moves.
  - o Within the pre-optimization processor, the development team added fields to indicate long-haul and short-haul identifiers to the location/customer and driver information contained in the input file.
  - o The algorithm itself then sought to "match" long-haul orders to long-haul drivers, and vice versa as a preference. If long-haul orders remained with no long-haul drivers available to execute that order, it would then be assigned to the next remaining driver.
- Associated Carriers has important, "high priority" customers whose work requires reliable drivers. As with the long-haul/short-haul specification, the development team again worked with PAI to add a "priority" identifier within the location/customer and driver fields. Again, with these orders and drivers properly identified, the algorithm then sought to match high priority orders to the most reliable orders. The priority identifier was an indicator of 1-3:
  - o "1" indicates a high priority customer and extremely reliable driver.
  - o "2" indicates an average priority and reliability ranking.
  - o "3" indicates a lower priority and reliability.
  - o Orders identified with a "1" could only be assigned to drivers also identified by a "1." However, orders identified by a "3" could be as signed to any driver.

---

[8] Productivity Apex, Incorporated, "Design and Logic Review," November 2012.

The development team worked with PAI to articulate a priority among these additional constraints, with the hazardous assignment being the most critical and t he long-haul/short-haul identifier being the least critical.

In order to create the optimized plan, the PAI algorithm uses the information from the input file to understand the elements below with respect to the entire group of orders planned for the day. These elements are evaluated by the algorithm to optimize the assignment of all orders.

- Travel distance between stops
- Travel time between stops
- Feasible dynamic points
- Driver's estimated earliest start time

After the PAI algorithm runs, it calculates and stores the following data elements:

- Miles driven
- Total miles driven
- Total miles driven per state
- Total miles driven per truck
- Total miles driven per truck per state
- Total estimated driving time and stop time

As part of the post-optimization process developed by Leidos, the development team worked with PAI to expose some of these elements for presentation on the optimized plan. In particular, the drayage companies were interested in understanding the estimated driving time and stop time, in addition to the total miles driven for the entire fleet and total miles driven per individual truck.

In summary, Vesco created a daily plan that was used by Associated Carriers to schedule their drivers. The algorithm focused on minimizing the number of bobtail moves and bobtail miles within a single carrier. The PAI algorithm did not try to match loads between two different carriers. It should be noted that the optimizer may reduce the number of trucks and/or drivers utilized by each carrier during the day; however, this may correspond to an increase in the number of bobtails/bobtail miles.

4.1.3   Post-Optimization Processor

Similar to the pre-optimization processor, the development team wanted to tailor the optimization algorithm's output for the drays' specific needs. As such, on the back end, the development team worked with PAI to uncover necessary data elements for

presentation to the dispatchers as part of the optimized plan. At the top of the plan were summary statistics, including:

- Total time
- Total jobs (orders)
- Jobs that could not be mapped (for manual assignment)
- Unassigned jobs (for manual assignment)
- Any jobs with invalid locations (not included in the template file used by the algorithm file)

Below the summary were the individual route solutions for each driver. With respect to each driver's route, the following data is presented:

- Order number
- Freight action (from the list of freight actions recognized by the PAI algorithm, and listed in Figure 8)
- Stop delay (in minutes; this is the estimated time each freight action would take):
    - The development team worked with the drays to create estimates for each of the freight actions used by the PAI algorithm
- Hazardous (whether the load was hazardous classified)
- Oversize/overweight (whether the load was oversize/overweight)
- Leg number (from Trinium Technologies software)
- Steamship line
- Whether the load was an LL or unload
- Trinium Technologies software dispatch sequence number (for dispatcher use when assigning orders).

A screen capture of a sample optimized plan is presented below in Figure 9.
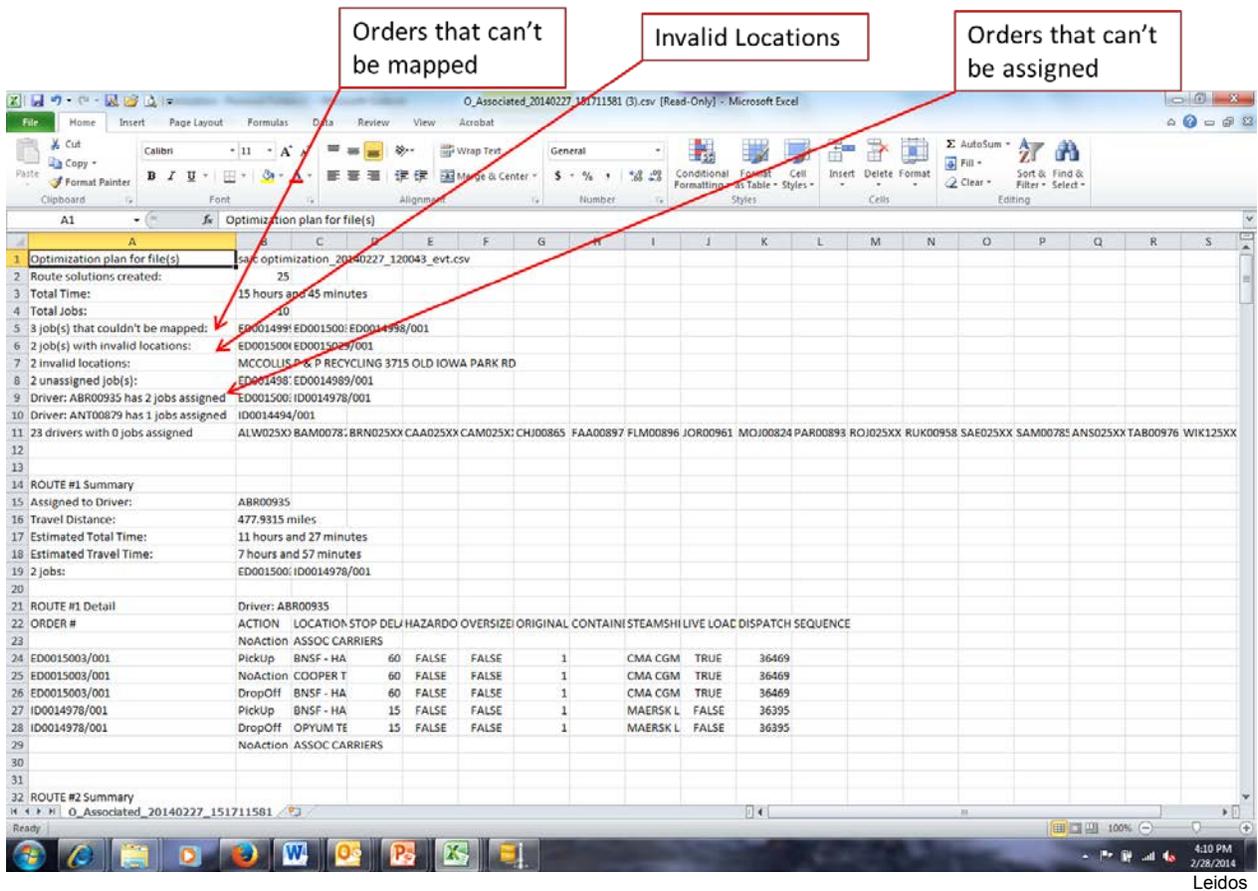
Figure 9. Associated Optimized Plan

The Vesco program for Associated Carriers also included an automatic re-optimization which took place daily at 10 a.m. EST. The process for executing the program was slightly different from the optimization that was conducted in the afternoon for the next day. For a re-optimization, two Trinium Technologies queries were needed. The first, like an optimization for the night before, included a file containing the orders to be completed for the remainder of the day. The second file contained a list of orders that had been completed for the day, including a driver identifier for which driver completed the order. From this file, the optimization program updated each driver's starting location with the destination of his last order.

As with an initial optimization, the program is triggered to run automatically upon receipt of the necessary files to an email inbox. The program runs automatically, generates an optimized plan and emails the optimized plan (in the format specified in Figure 9) to the designated list of recipients at Associated Carriers.

4.2 ALTERNATE OPTIMIZATION PROGRAM (SOUTHWEST FREIGHT)

The optimization engine and user interface were both written in Java code. The objective of the code was similar to the optimization generated by the development team as part of the Cross-Town Improvement Program (C-TIP) and was the basis for this Alternate Optimization Program. The process followed by the program is depicted in Figure 10.
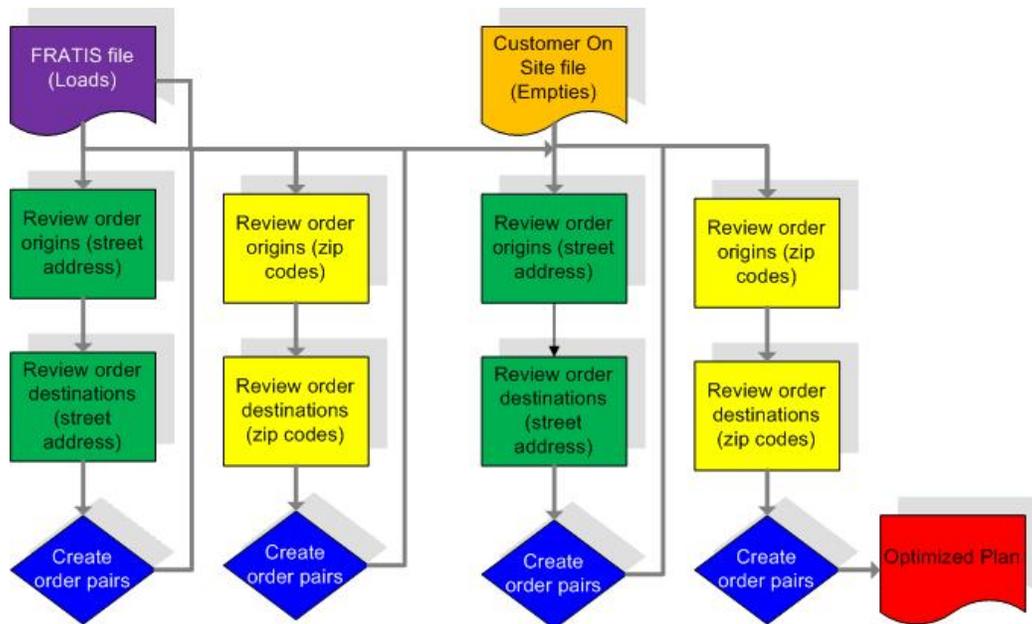
Leidos

Figure 10. Alternate Optimization Program Process

To account for Southwest Freight requirements, the program limits the number of orders that can be assigned to a single driver (route) to four, although this can be altered in the code. In addition, the program offers the user the flexibility to determine the number of drivers/routes that should be included in the daily plan. For Southwest Freight, they wanted to make sure a certain number of drivers would have work each day. This can also reduce the number of orders per route/driver. As described above, both the user interface and code were written in Java.

The alternate optimization program was deployed to Southwest Freight in October 2014. For the remainder of the prototype period, the development team used the program to generate optimized plans nightly, which were provided to Southwest Freight via email. The development team discussed having the Southwest Freight users install and run the program; however, to guarantee the creation and archival of daily optimized

plans, the development team decided to continue running the plans and sending them out. The program, however, is easily installed as a .jar file.

A summary of the program

- **Tools and language:**
    - o Eclipse
    - o Java 1.6
- **Prerequisites:**
    - o Java Runtime Environment (JRE) 1.6 or higher
    - o Microsoft Excel
- **Running the Alternate Optimization program:**
    - o From DOS prompt, navigate to the location of optimizer.jar enter the command java –jar optimizer.jar
    - o A user may also double click the optimizer.jar file

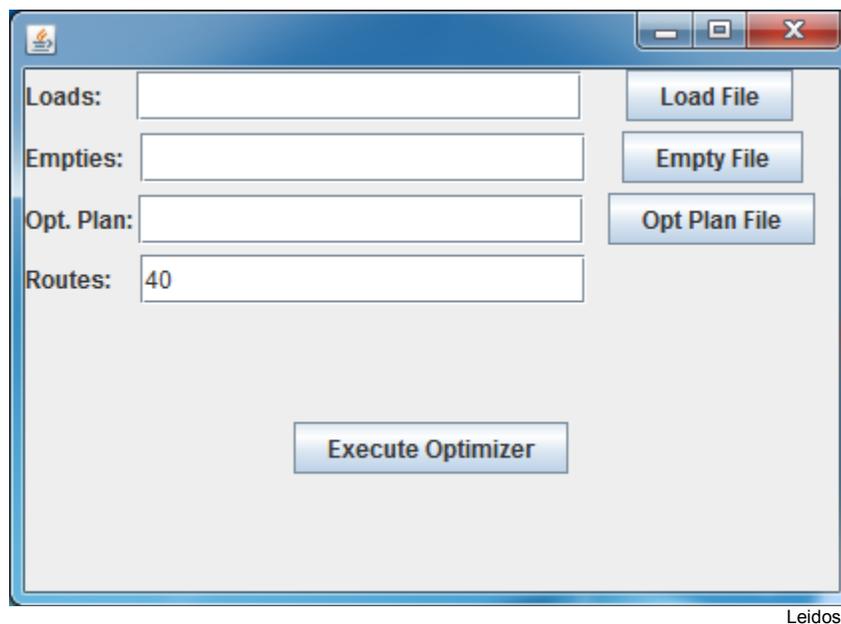Figure 11 captures the user interface for the program.

Leidos

Figure 11. Southwest Alternate Optimization Program User Interface

Similarly, given the budget remaining within the prototype, the optimized plan format was kept simple, displaying the information requested by the user. Presented as a comma separated file, the program first summarized any orders that could not be

matched, so that these orders were easily identified for manual assignment. A sample is shown in Table 2.

Table 2. Excerpt of Southwest Alternate Optimization Program Plan

The Number of matched moves from COS file = 10								
The Number of matched moves from Optimization file = 13								
List of unassigned orders: 0								
Count	ID	ContainerNumber	From	From City	To	To City	Start	End
List of assigned orders:								
Count	ID	ContainerNumber	From	From City	To	To City	Start	End
Route #0								
0	CEX0141213/001	HRZU 433483-4	HAYES RETAIL SERVICES	MESQUITX	UPRR - DALLAS INTERMOD/	WILMERTX	0	0
1	CIM0140994/007	EMCU 955912-6	UPRR - DALLAS INTERM(	WILMERTX	COSTCO 288	GRANDPTX	0	0
2	CIM0140517/003	TEMU 733535-7 IMP40H	COSTCO 288	GRANDPTX	UPRR - DALLAS INTERMOD/	WILMERTX	0	0
3	CIM0141030/001	HMCU 914044-6	UPRR - DALLAS INTERM(	WILMERTX	WISTERIA	FARME1TX	700	700
Route #1								
0	CIM0139357/001	CMAU 427941-2	SOUTHWEST FREIGHT	DALLASTX	COOPER TIRES CENTER	GRANDPTX	700	700
1	CIM0140593/005	BMOU 464930-3 IMP40H	COSTCO 288	GRANDPTX	UPRR - DALLAS INTERMOD/	WILMERTX	0	0
2	CIM0141041/002	OCGU 803500-5	UPRR - DALLAS INTERM(	WILMERTX	FORTRESS IRON NANCY JOI	RICHARTX	800	1000
Route #2								
0	CIM0139358/002	CMAU 456633-0	SOUTHWEST FREIGHT	DALLASTX	COOPER TIRES CENTER	GRANDPTX	700	700
1	CIM0140593/006	TEMU 609674-6 IMP40H	COSTCO 288	GRANDPTX	UPRR - DALLAS INTERMOD/	WILMERTX	0	0
2	CIM0140999/001	EISU 185214-6	UPRR - DALLAS INTERM(	WILMERTX	SPECTRUM BAGS INC	ARLINGTX	800	1200
Route #3								
0	CIM0139358/003	INKU 637802-6	BNSF	HASLETTX	COOPER TIRES CENTER	GRANDPTX	700	700
1	CIM0140703/004	TEMU 712313-1 IMP40H	COSTCO 288	GRANDPTX	UPRR - DALLAS INTERMOD/	WILMERTX	0	0
2	CIM0140999/002	IMTU 103176-4	UPRR - DALLAS INTERM(	WILMERTX	SPECTRUM BAGS INC	ARLINGTX	800	1200

Leidos

4.3 TERMINAL QUEUE

4.3.1 Bluetooth/Wi-Fi

Acyclica, the hardware and software provider for the Bluetooth/Wi-Fi queue time monitor system at IMCG-Wilmer, installed four antennas in protective enclosures around the yard.[9] The equipment and enclosure is shown in Figure 12.

---

[9] Acyclica, Inc. web page. Available at: www.acyclica.com

Figure 12. Bluetooth/Wi-Fi Wait Time Readers (Enclosure and Reader)

Antennas were installed at the approach road, in-gate, IMCG gate, and out-gate, as illustrated in Figure 13, to detect passing Wi-Fi devices.

Figure 13. Overview of Bluetooth/Wi-Fi Wait Time Technology at IMCG-Wilmer

The media access control (MAC) addresses and detection times are passed in real time to the device server. The approach road data are passed via a cellular call, while data for the other three locations are passed through an internet connection. This configuration was selected based on facility conditions:

- The approach reader location was located on a power pole, so hardwired power was available. A cellular call was created to pass the data collected from this reader because no network connection was available. Also, due to the distance from existing network ports, there was uncertainty with respect to the signal strength of a wireless connection.
- The remaining three readers were all at existing, enclosed booths at IMCG-Wilmer, providing hardwired power and network connections.

The Acyclica data server does basic scrubbing and stores the MAC address and detection time. A scheduled process on the FRATIS server will retrieve any data since the last retrieval via a Representational State Transfer (RESTful) web service. The FRATIS server analyzes the time between locations to calculate current time to in-gate, current time to in-gate for IMCG trucks, current time for a drop-off, and current time for a pickup. The FRATIS server stores the average current listed wait times received from Acyclica for the four 15-minute reads each hour. The FRATIS server determines the current hour and displays that value from the database. For predicted times, the FRATIS server utilizes the historical time database table, which stores the historical time for each hour of the day for all seven days of the week. These times are updated with the current wait time (as described in Section 3.1.1.3 to maintain a predicted wait time).
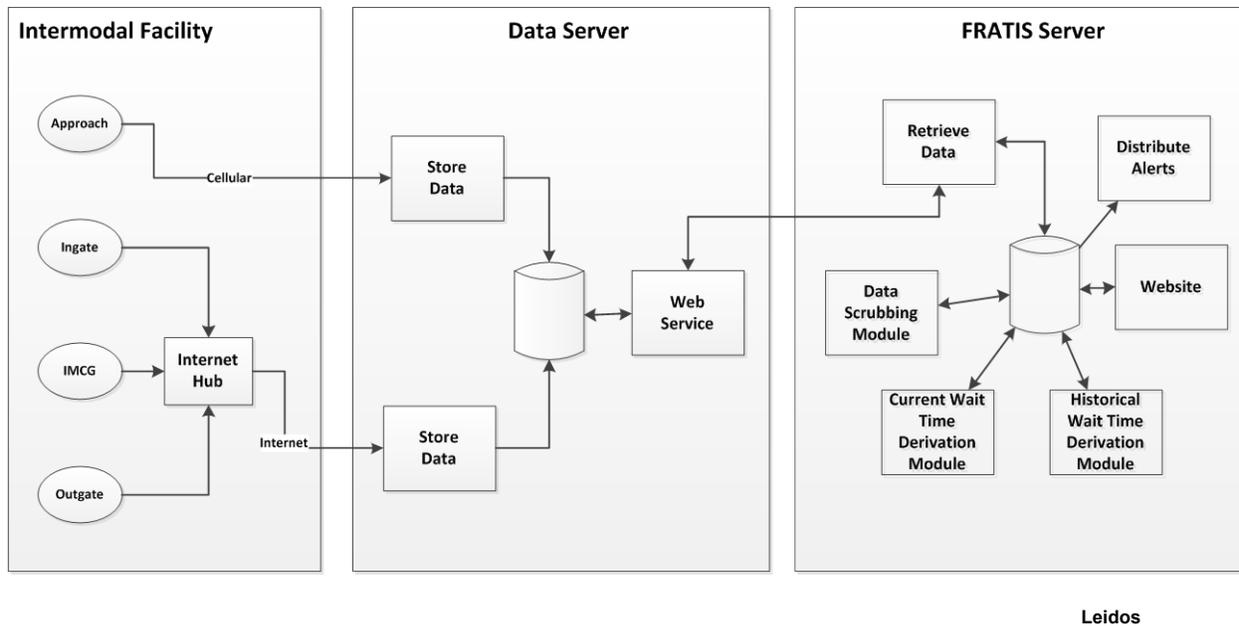
Leidos

Figure 14. Terminal Queue Architecture

One of the main concerns with respect to Bluetooth and Wi-Fi technology is the protection of personal identifiable information (PII) with respect to the anonymity of the data collected by the Acyclica system:

- A MAC address is a unique identifier assigned to Bluetooth and Wi-Fi devices; however, the device owner is unknown
- MAC addresses were encrypted for further privacy
- Devices can be matched point to point but never unscrambled to reveal the original device identification

4.3.2 Dedicated Short-Range Communication (DSRC)

The overall process for the 30-day pilot test of the DSRC equipment in Dallas was as follows:

1. Acquire and configure a RSU from USDOT Turner-Fairbank Highway Research Center (TFHRC)
2. Set up security certificates on the RSU
3. Acquire and configure five after-market safety devices (ASDs) from USDOT connected vehicle Michigan test bed – note the ASDs are the DSRC devices that are installed within the vehicles. Use of these ASDs was also contingent on:

- o Completing a loan agreement to authorize use of this equipment (between Leidos and USDOT)
- o Building cigarette lighter adapters to allow power to the units
- o Bootstrapping and uploading certificates for devices
- o Configuring devices for truck use
4. Bench testing of items 1-3 prior to site install (conducted at TFHRC)
5. Install RSU in truck yard (including backhaul)
6. Install ASDs in trucks
7. Configure existing connected vehicle data management system (DMS) to collect data from the RSU during the month long test:
   - o For this test, new code was written in Java to process BSMs received from the ASDs to calculate the wait time. This code parsed the BSMs and used the geofences (described in the next bullet) to identify where in the facility the message was transmitted.
   - o Establish geofences around the terminal access points, writing appropriate Java code and testing. These geofences were necessary to identify which of the four operational areas the truck was in when the BSM was sent (approach, in-gate, out-gate, or within the yard). These geofences defined the boundaries, i.e., latitude and longitude for each of these areas. Each BSM also contains a latitude and longitude position.
8. Decommission system and return devices to USDOT

The DSRC equipment was installed at the IMCG-Wilmer yard and in four of their dray trucks on December 18, 2013 and remained in place throughout the month of January 2014. The equipment was removed and returned to USDOT in February 2014.

The equipment installed included:

- Five Arada ASD units with 90 days security certificates:
  – Two DSRC stick antennas
  – One flat global positioning system (GPS) antenna
  – One alternating power (AC) power adapter
  – Cigarette adapters for power supply
  – Default configuration for these units broadcast BSMs automatically
- One mobile RSU by Arada
- 4th generation long-term evolution (4G/LTE) cellular modem to support the backhaul data connection (allows the RSU to access the Internet)

Communications and network protocols to support the test included:

- Security certificates for the ASD units – these were completed at USDOT TFHRC
- Global Internet Protocol (IP) v6 address and an additional IPv4 address
- IP protocol 41 must be allowed to reach the RSU from 12.168.58.0/24 range. This allowed the technical team to "tunnel" the IPv6 addresses to the RSU from the connected vehicle network

BSMs from the five ASDs were broadcast at 10 Hertz (10) times a second). The message elements and values are based on the J2735 standards and latest USDOT specifications.

The data were collected for the month of January 2014. To reduce the amount of data stored, the development team only stored the data necessary to calculate wait time from the BSM; namely, latitude, longitude, speed, heading, date and time. These were the data elements necessary to identify what area of the facility the truck was located, and a time estimate of how long the truck was there. At this time, the data has not been uploaded to the USDOT's Research Data Exchange (RDE). The data will be evaluated to determine whether they violate any PII restrictions that may prevent IMCG from authorizing its distribution. Analysis is currently underway and includes calculation of wait time at the approach to the IMCG-Wilmer facility. Additional analysis will be done to compare the wait times calculated based on the BSM data to the wait times calculated by the Acyclica Bluetooth/Wi-Fi wait time system.

The photos in Figures 15 and 16 document the equipment installed at the yard in Wilmer.

Leidos

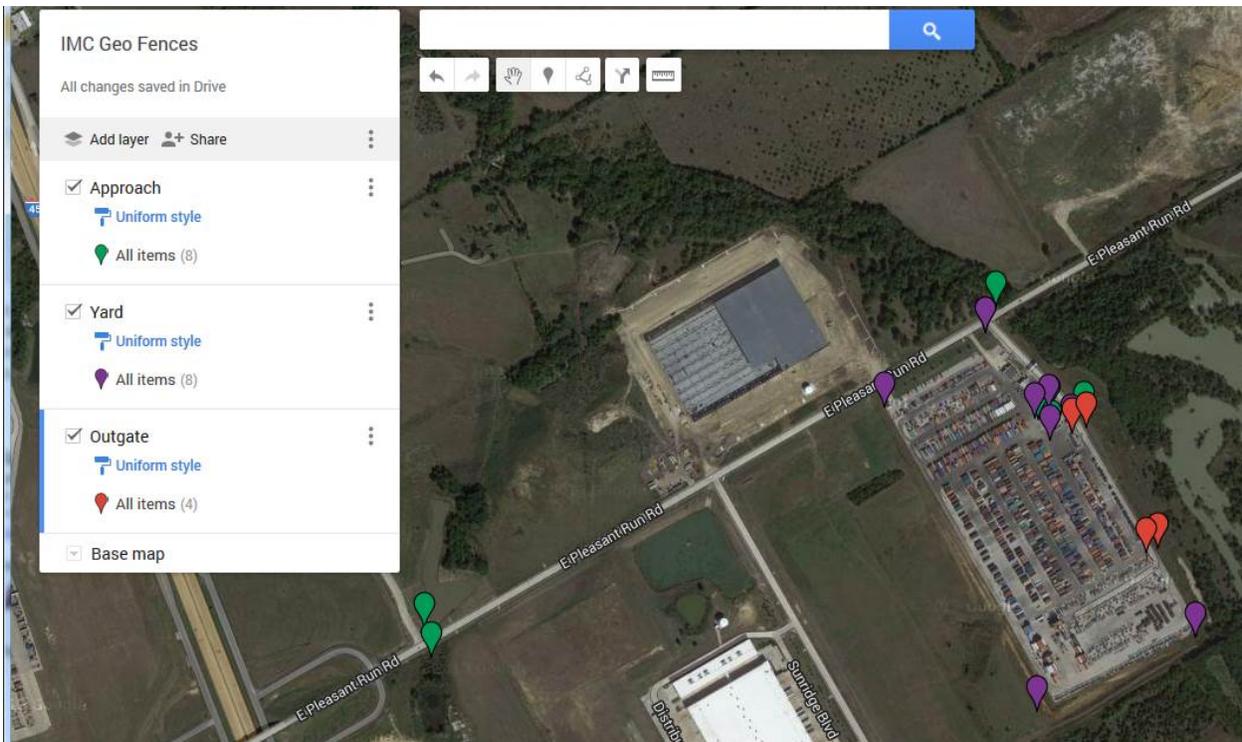**Figure 15. DSRC RSU**

Figure 16. Truck DSRC Antenna and Radio

The geofences set up by the development team in the analysis of the BSMs collected during the test period are identified in Figure 17.

Figure 17. Geofences for BSM Data Analysis

4.4 ROUTING, NAVIGATION, TRAFFIC AND WEATHER

This function of the FRATIS DFW was provided through Trinium Technologies driver-direct work order web application. It was designed with the driver in mind, to provide the most efficient route to their destination taking into account characteristics of the truck, traffic conditions, and weather conditions. Specific characteristics of this application include:

- Dynamic routing is offered through the Trinium Technologies MC2 web application.
- The origin and destination locations from the work order are used to automatically generate a route and display it on a map and as text below the work order within the application.
- The route information is powered by ALK Maps:
    - Recommended route, geocoded points, weather, and traffic are viewed as separate controllable layers (user can turn off/on)
    - Route is a "truck" route, and therefore favors truck designated routes, with special consideration/priority given to:
        - Using truck oversized routes
        - Discouraging toll road use
        - Minimizing mileage (provided optimal traffic conditions)
        - Following prescribed hazmat routes.
    - ALK Maps also takes into account the weight, height, width, and length of standard commercial vehicles to route around obstructions and restricted roads
    - The ALK Maps software includes logic for differentiated turn costs (for left and right turns) as well as methods for keeping commercial traffic out of local neighborhoods and on principal through-roads.
    - Truck route functions include:
        - Truck-restricted, truck-designated, and truck-prohibited roads
        - Hazmat-specific road classes and routing
        - Bridge heights and clearances
        - Load limits
        - Weight limits and allowances
        - One-way road designations
        - Left hand and dangerous turn restrictions
        - Urban road classifications
- The recommended route is the shortest truck route, is intended to optimize transit time, and assumes exclusion of toll roads. The ALK maps application monitors

congestion in real-time to avoid traffic incidents, but the driver ultimately has the responsibility to decide on detours around congestion they encounter mid-route.

- At a minimum, traffic and weather refresh each time the driver processes any Trinium Technologies workflow step (from "Acknowledged Order to Pick Up" (PU) in to PU out, etc.).

- The route, traffic, and weather *also* update anytime the screen is refreshed. Therefore, these factors can be upd ated by the driver at any time during the execution of the route. The route, traffic and weather will also **automatically refresh** anytime the driver exits and r e-enters the MC2 application. Therefore, the routing application can be as dynamic as the driver chooses, similar to anytime a personal driver were to "update from here" within Google Maps.

- Real-time, predictive, and historical traffic speed via Inrix (standard for all ALK products).

- Real-time traffic and weather overlay on the map-presentation of the recommended route (Figure 18):
    o Weather data are provided from ALK, with backend interface to Weather Underground as the data source.
    o Incidents and construction are indicated by icons on the map, with non-recurring congestion indicated as red along the highlighted route.

Trinium Technologies
Figure 18. Enhanced MC2 Application (Trinium Technologies)

4.5 ADVANCE NOTICE TO TERMINALS

As discussed in Section 3, the detailed review of the Trinium Technologies dynamic queries revealed data that the development team felt would be useful among other stakeholders. Given that the Trinium Technologies query comma separated files are automatically sent to a specified inbox, the data were available for analysis, consolidation, and re-distribution, with the permission of the drayage companies who authorized release to specified prototype participants.

The process to isolate and distribute these data is as follows:

- Using dynamic query data for Southwest Freight and Associated Carriers, the development team wrote a process to retrieve the queries from an email inbox.
- The development team wrote Java code to consolidate the loads from both carriers destined for the following facilities:
  - o IMCG-Wilmer
  - o IMCG-Haslet
  - o BNSF
- The code then generates an email to IMCG and B NSF identifying the loads destined for the above facilities.
- The email contains a table detailing, for each drayage company:
  - o The number of loads
  - o Load type (loaded or empty)
  - o Appointment window

A sample of this email is presented in Table 3. Please note the explanation of the fields:

- Location: this is the facility that will receive the load (IMCW = IMCG – Wilmer, IMCH = IMCG – Haslet).
- SSL: steamship line (container)
- Size: container size (20, 40, etc.)
- FCL/ECL: full container load or empty container load
- From time: start of the appointment window
- To time: end of the appointment window

Table 3. Advance Notice to Terminals

Location	Dray Company	Pickup\ Delivery	Container	SSL	Size	FCL\ ECL	Date	From Time	To Time
IMCW	███████	Delivery	HJCU 134034-8 IMP40	Hanjin Shipping Company	40	ECL	02/04/2015	08:00:00	16:00:00
IMCW	███████	Pickup	EXP40	Hanjin	40	ECL	02/04/2015	10:00:00	15:00:00

Location	Dray Company	Pickup\Delivery	Container	SSL	Size	FCL\ECL	Date	From Time	To Time
				Shipping Company					
IMCW	■■■■	Pickup	EXP40	OOCL America	40	ECL	02/04/2015		
IMCW	■■■■	Pickup	EXP40	Hanjin Shipping Company	40	ECL	02/04/2015	10:00:00	15:00:00
IMCW	■■■■	Pickup	EXP40	Yang Ming America	40	ECL	02/04/2015		

Leidos

The intent of this information is to improve the planning process at terminals. By anticipating when loads will arrive, and what level of effort will be required to process them (whether loaded or empty), the terminals may be able to better plan the labor and resources to accommodate them. In addition, it provides the terminals with insight and visibility over expected daily volumes. Last, it will help the terminal to coordinate appointments among multiple drayage companies and customers.

U.S. Department of Transportation
ITS Joint Program Office-HOIT
1200 New Jersey Avenue, SE
Washington, DC 20590

Toll-Free "Help Line" 866-367-7487
www.its.dot.gov

FHWA-JPO-15-219