



Integrated Mobile Observations 2.0

Project Final Report

August 1, 2014

Prepared by:

Bruce M. Belzowski
Principal Investigator
University of Michigan
Transportation Research Institute
2901 Baxter Road
Ann Arbor, MI 48109

Steven J. Cook, P.E.
Program Manager
Michigan Department
of Transportation
6333 Lansing Road
Lansing, MI 48917

Technical Report Documentation Page

1. Report No. UMTRI-2014-19	2. Government Accession No. N/A	3. MDOT Project Manager Steven J. Cook, P.E.
4. Title and Subtitle Integrated Mobile Observations 2.0 (IMO 2.0)		5. Report Date July 4, 2014
		6. Performing Organization Code N/A
7. Author(s) Bruce M. Belzowski		8. Performing Org. Report No. UMTRI-2014-10
9. Performing Organization Name and Address The Regents of the University of Michigan		10. Work Unit No. (TRAIS) N/A
		11. Contract No. 2009-0747
		11(a). Authorization No. Z1 / R1
12. Sponsoring Agency Name and Address Michigan Department of Transportation (MDOT) Research Administration 425 West Ottawa Street Lansing MI 48933		13. Type of Report & Period Covered Final Report 11/1/2012 to 3/31/2014
		14. Sponsoring Agency Code Federal Highway Administration (FHWA)
15. Supplementary Notes		
16. Abstract The key objective for this project was the implementation of a low cost data acquisition system that provides weather-related road information to weather analysts throughout the country in near-real time using a fleet of 60 vehicles along the I-94 corridor in southern Michigan. This work is part of an on-going research initiative to identify how state DOTs will use and benefit from the large quantities of data generated by future connected vehicle programs and to assist in refining connected vehicle system requirements. The project employed an Android-based customized smartphone system called DataProbe to gather information from the phone (date, time, latitude and longitude, altitude, number of satellites, speed, accelerometer data, and compass heading); the vehicle through its controller area network (CAN) (air and coolant temperature, odometer, barometer, tachometer, speedometer, throttle, brakes, anti-lock braking system (ABS), electronic stability control (ESC), engine traction control and braking traction control); and through external sensors, Surface Patrol, that measure road and air temperature, humidity, and dew point. This data is collected by the phone in one second intervals [except for accelerometer data which is gathered at 100 second intervals on three axis (x, y, and z)], written to a comma separated values (CSV) file for a period of five minutes, and sent via cell phone to a University of Michigan Transportation Research Institute (UMTRI) server where it is validated, stored, and sent to weather analysts in five locations throughout the U.S. The Android-based smartphone was also designed to take photos of the road manually or via a web portal designed to track vehicles in use and potentially send messages to drivers through the phone. These photos were also uploaded to the UMTRI server and sent to the weather analysts throughout the U.S.		

Over a period of 17 months, the project saw tremendous change in every part of the process including changing nearly half of the project vehicles, re-writing DataProbe source code, changing hardware, and changing the process for working with drivers and updating software. The project resulted in the delivery of 172 gigabytes of valid data representing 196,204 valid files transferred to weather analysts, with MDOT operators driving nearly 400,000 miles and taking 44,594 photos. The IMO 2.0 project has been extended to accommodate an IMO 2.0 demonstration at the 2014 ITS World Congress – Detroit in September to demonstrate how the IMO 2.0 data can be used for traveler information in the form of motorist advisory warnings posted to dynamic message signs and the MDOT MiDrive website. Project completion for this phase is November, 2015.

17. Key Words CAN bus, IMO, GPS, CSV, DataProbe, surface patrol,		18. Distribution Statement No restrictions. This document is available to the public through the Michigan Department of Transportation.	
19. Security Classification - report Unclassified	20. Security Classification - page Unclassified	21. No. of Pages	22. Price N/A

Integrated Mobile Operations 2.0

Table of Contents

Introduction	6
Scope	7
Project Overview	8
Technology Overview	10
Software	10
Hardware.....	13
The Phone.....	13
The OBD Key.....	14
The Surface Patrol Sensors.....	15
Data Collected	17
Data Flow... ..	17
DataProbe File Naming Conventions.....	18
DataProbe File Format	18
IMO Photos	20
IMO Data Server	21
DataDroid Web Portal	22
Data Quality	24
Driver Engagement/Project Momentum.....	24
The IMO User Survey	25
Vehicle Activity Tracking.....	25
The UMTRI/MDOT IMO Team Meetings	27
Lessons Learned	28
Vehicles.....	29
Hardware.....	30
Software.....	31
Communication	33
Users/Drivers.....	33
Continuous Improvements.....	34
Conclusions	35
Recommendations	35

Appendix	38
1. DataProbe File Naming Conventions	38
2. DataProbe Data Dictionary.....	39

Table of Figures

Figure 1 - IMO 2.0 System Architecture Overview	8
Figure 2 - The IMO 2.0 southern Michigan I-94 Corridor (9-counties shown)	10
Figure 3 – DataProbe Screen Example in Phone and Surface Patrol Mode Only	12
Figure 4 – DataProbe Screen Example in Phone, Surface Patrol, and CAN Mode.....	12
Figure 5 – DataProbe Screen Example in Phone and CAN Mode Only	13
Figure 6 – Bluetooth enabled OBD Key for CAN data.....	15
Figure 7 – Surface Patrol Road Temperature Sensor on a winter maintenance truck (WMT).....	16
Figure 8 – Surface Patrol Humidity Sensor Mounted on a Car Trunk	17
Figure 9 – IMO 2.0 Data Flow Diagram	18
Figure 10 – DataProbe File Format Example.....	19
Figure 11 – DataProbe Photo Examples.....	20
Figure 12 – DataDroid Web Portal Page.....	23
Figure 13 – IMO Vehicle Activity Report	26
Figure 14 – IMO Vehicle Activity Graph	27
Figure 15 – IMO Issues Report	28

Introduction

The Integrated Mobile Observations 2.0 (IMO) project was designed as a practical extension of the technology tested in MDOT/UMTRI's Slippery Road project (2012), where Android-based smartphones were programmed and installed in vehicles to gather information from the phone (e.g. latitude/longitude), vehicle (e.g. speed, rpm), and external sensors (e.g. surface temperature). For the IMO 2.0 project, with funding from the U.S Department of Transportation's Federal Highway Administration (FHWA) and in-kind funding from the Michigan Department of Transportation (MDOT), the University of Michigan Transportation Research Institute (UMTRI) focused on gathering data to support weather-related analysts throughout the U.S., who were charged with designing applications that make use of the data collected. MDOT along with the Minnesota Department of Transportation and the Nevada Department of Transportation worked independently using different strategies to gather weather-related data and send it to weather analysts.

One area of prior research on the use of smartphones to collect data focused on measuring road roughness where the accelerometer in the smartphone was used to measure roughness, with varying degrees of success. For weather-related data collection via smartphone, a number of researchers tried to develop work-arounds (some successfully) for the lack of ABS data from the CAN bus. They tend to focus on wheel slip as a proxy for ABS.

For this project, an Android-based smartphone was combined with customized software, called DataProbe, and Bluetooth-enabled external temperature sensors (called Surface Patrol) and internal vehicle data collectors [On Board Diagnostics (OBD keys)]. The DataProbe software was designed to collect data from the phone and all available sensors and data collectors in one second intervals. Data files were created in five minute intervals and sent via cell phone to an UMTRI server that sorted files into valid and invalid files and sent valid files to five weather analysts throughout the U.S.

From an overall project perspective, the goals for MDOT/UMTRI were

- 1) to instrument 60 MDOT vehicles with DataProbe smartphones and internal and external data collection devices/sensors and collect data for seventeen months
- 2) to collect accurate and timely data from each of the devices, including photos taken by the cameras on the smartphone
- 3) to deliver all data to the weather analysts in a timely manner
- 4) to keep track of vehicles in service through a web portal, including the ability to trigger photos and send messages from the portal

This report details the efforts of the combined UMTRI, MDOT, and FHWA team to accomplish these goals. The past seventeen months has led to many improvements in the technology, but also in the interactions with maintenance supervisors and IMO 2.0 drivers (MDOT fleet vehicles) in the field. To call this a team effort is an understatement. Without the continued support of MDOT maintenance supervisors and IMO 2.0 drivers, this project would not have accomplished all that it has in the past seventeen months, with nearly 400,000 miles logged and over 172 gigabytes of valid data collected. Despite the success of the data collection effort, the real test

will be how the data collected is used by the weather analysts to provide near time information to drivers concerning weather-related events.

Scope

The IMO 2.0 project will display the capability of an inexpensive Android-based smartphone to gather weather-related data from the phone, the vehicle, and external sensors from a fleet of 60 MDOT vehicles. The purpose of the project is to gather the information from the vehicles, transfer it to the University of Michigan Transportation Research Institute for processing and dissemination to a group of five weather analysts organizations throughout the U.S. in order for them to develop useful weather-related applications.

This project included the following:

- Supervise the development of the software design and development needed for the project
- Implement and deploy the system on 40 light duty cars and trucks vehicles and 20 winter maintenance trucks (WMTs) including sensor equipment as needed
- House and manage the server that will act as the intermediary between the data collected via Android-based smartphones and the weather analysts who will receive the data.
- Support data monitoring, analysis, and vehicle interactions as needed

The main goals of the project included:

- 1) All eligible vehicles are instrumented with Android phones with the latest DataProbe software, OBD keys, and Surface Patrol sensors (where applicable)
- 2) All data available to the DataProbe phone must be captured, including automatic photos taken during ABS, ESC, or Traction Control events
- 3) All data available (Basic, CAN, and Surface Patrol) from the DataProbe phone must be accurate¹
- 4) UMTRI must receive the data in a timely² manner from the DataProbe phone
- 5) All valid files must be sent in a timely manner from UMTRI to the weather analysts throughout the country
- 6) The DataDroid web portal must recognize the vehicles in service and be able to take photos and send messages from the site.

¹ “Accurate” in this sense means that the data collected must be within a range of accepted values.

² “Timely” in this sense means that the data collected must continually be delivered to the weather analysts within the minimal amount of time needed to collect the data, send it to UMTRI for verification, and have the data received by the weather analysts. Our research in this area showed that “timeliness” for this project meant seven to nine minutes from the time the file was created to the time it was received by the weather analysts.

Project Overview

The IMO 2.0 project is a closed loop data collection project where data is collected at the vehicle level and data is provided to drivers based on the knowledge gained from the original data collection. This system architecture is shown visually in Figure 1.

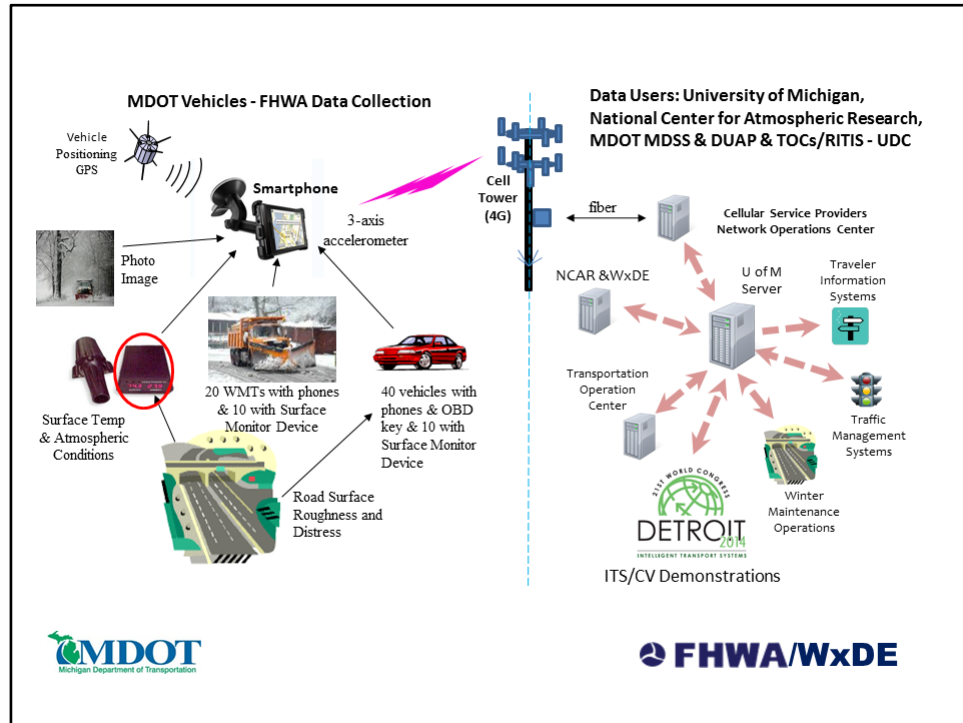


Figure 1 - IMO 2.0 System Architecture Overview

Forty MDOT cars and trucks and twenty MDOT winter maintenance trucks (WMTs) were equipped with Android-based smartphones using UMTRI's DataProbe software that generated the following data:

- time
- latitude, longitude: GPS coordinates
- altitude via GPS
- number of visible Global Positioning System (GPS) satellites
- compass heading
- the speed of the vehicle based on GPS
- 3-axis (x, y, and z) accelerometer readings sampled at 100 Hz per axis
- photos taken by the driver, through the project web portal, or when an autonomous event such as ABS or traction control activation occurs.

Ten cars and trucks and ten WMTs (20 vehicles total) were equipped with Bluetooth-enabled Surface Patrol external environmental sensors that measured

- road surface temperature
- dew point
- ambient air temperature
- humidity

Forty cars and trucks were outfitted with Bluetooth-enabled OBD keys that collected

- tachometer (RPM)
- vehicle speed
- throttle (position)
- brake activation
- anti-lock braking system activation (ABS)
- electronic stability control activation (ESC)
- traction control system activation (TSC)
- windshield wiper activation

Data is collected in the form of five minute locally-cached segments of text-based sensor data stored in a comma separated value (CSV) file and in photographic data collected from the phone's camera. Text data is continuously collected while the system is in operation while photographs are taken either manually by drivers, manually by administrators through a remote-access web portal, or automatically via a triggered event such as ABS, traction control, or stability control events. After local collection and caching, data is uploaded via a cellular 3G/4G data network and the internet to University of Michigan servers where the data is validated and sent to analysts from the following groups:

- National Center for Atmospheric Research (NCAR)
- Mixon-Hill / Data Use and Analysis Processing (DUAP)
- Atkins / Regional Integrated Transportation Information System (RITIS)
- Iteris / Maintenance Decision Support System (MDSS)
- Leidos & Synesis Group / FHWA Weather Data Environment (WxDE)

These analysts were charged with using the data to develop weather models and other uses of the data related to traffic management, traveler information systems, winter maintenance operations, and state departments of transportation asset management systems. The IMO system will also provide data for a micro-level weather reporting demonstration at the 2014 ITS World Congress in Detroit, Michigan.

The instrumented vehicles traveled primarily along the I-94 corridor in southern Michigan, as seen in Figure 2.

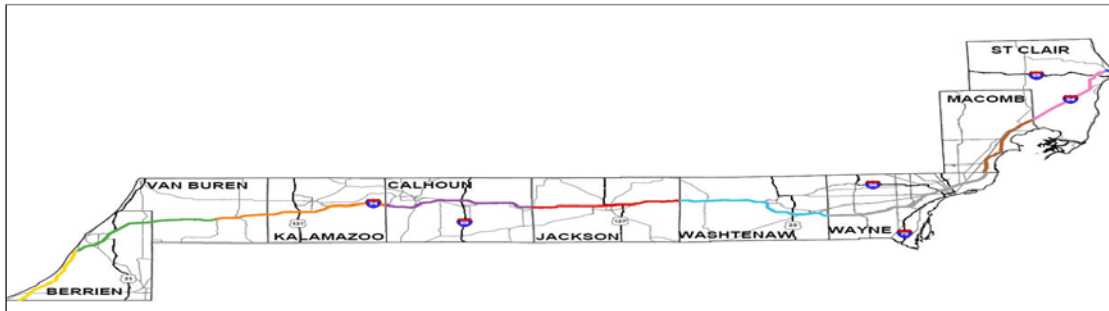


Figure 2 - The IMO 2.0 southern Michigan I-94 Corridor (9-counties shown)

Technology Overview

The IMO 2.0 project relied on a combination of custom software and commercially available hardware to gather and distribute data from the vehicles in the study.

Software

The software used to gather the information from the smartphones in the study was called DataProbe. This software was developed during the Slippery Roads study, and began as the base software for the IMO 2.0 project. Early in the study it was found that the software needed substantial customization to meet all the requirements for the project. A software programming firm, Intersog, the original designers of the DataProbe software, was hired to support the changes needed in the software. After five months of off-site development and troubleshooting with Intersog, it was decided that UMTRI needed on-site staff to work on continual maintenance of the DataProbe software including programming and managing the testing and rollout of the program. Subsequently, a staff engineer was hired to support the project.

Over the 17 months of the project, the software was updated approximately 15 times to meet the demands of the project. The carryover version from the Slippery Roads project was version 2.20, and the version used by the end of the project was version 3.7.1. Two important software issues arose that made the project more challenging than originally expected:

1. Every time a new vehicle type was brought into the project, a set of automaker-specific CAN messages needed to be programmed into DataProbe for each specific model year and model of vehicle.

Working with the two major automotive manufacturers chosen for the study provided very different results. UMTRI signed a Non-Disclosure Agreement with one of the companies and

had contacts within the company who provided insight into how to correctly read the proprietary and not publicly available CAN messages that differed for each vehicle.

Attempts to generate a Non-Disclosure Agreement with the other automotive manufacturer were not as successful. Despite numerous attempts, including a request by the Michigan Department of Transportation Director, the company would not provide the necessary agreement or support for the project. UMTRI attempted to reverse engineer some of this company's CAN codes, but the results were not satisfactory. In addition to the inability to obtain the desired data, on some instrumented vehicles from this company, there were examples of vehicles flashing their interior lights, locking and unlocking the electronic doors randomly, and even making the automatic transmissions difficult to shift. Because of the reluctance of the company to support the project, their vehicles, which represented about half of the light vehicle fleet studied, had to be removed from the project and another set of vehicles from the other company needed to be instrumented and added to the fleet.

This change occurred four months into the project and necessitated a larger commitment by UMTRI and MDOT staffs than originally expected, as MDOT worked to find vehicles that would meet the requirements for the project and UMTRI programmed the requisite CAN message codes into the DataProbe program.

2. Every time a new version of DataProbe was issued, MDOT staff had to go to each of the 60 vehicles in the project and update the program manually on each smartphone.

Updating 60 vehicles manually proved very challenging as DataProbe went through many different versions, especially as the vehicles from one company were dropped from the fleet and the new vehicles were added. It was not until a UMTRI software programmer was hired towards the end of the project to specifically design a remote update program was the problem solved. Now an updated version of DataProbe can be pushed, through the cellular network, to each of the phones in the fleet, with each driver only having to click on two buttons when prompted to install the remotely managed updates.

The DataProbe program provides some output results of its data collection to the driver via the smartphone's screen. The screen is configurable to allow for different data to be shown. Figure 3 shows a typical six box screen. In this instance it shows data received from the Surface Patrol device (ambient temperature, surface temperature, and humidity) and the phone (GPS location and time). The box labeled "MODE" determines what data appears on the boxes. In this case, it shows data in the BS mode where basic phone and Surface Patrol data is collected. The combination of data collected from the various devices can show the following codes in the "MODE" box.

- "B" if the phone is only receiving basic phone data
- "BC" if the phone is receiving only basic phone data and CAN data
- "BS" if the phone is receiving only basic phone data and Surface Patrol data
- "BCS" if the phone is receiving basic phone, CAN, and Surface Patrol data

The different combination of modes can occur for a variety of reasons. For example, if a phone is receiving only basic phone data, then its OBDKey may not be functioning properly and not sending CAN data or simply not be present in that vehicle. If a phone is not receiving Surface Patrol data, it may be normal because that particular vehicle is not equipped with Surface Patrol. All light duty vehicles, at a minimum, should be sending phone and CAN data, and all WMTs should be sending phone data. In addition to providing information to the driver on the smartphone screen, this display also helps the driver confirm proper system functionality and operation. If one or more modes are not present on the smartphone screen, system diagnostics can better be accommodated. Figure 4 shows a DataProbe screen in the BCS mode, and Figure 5 displays a DataProbe screen in the BC mode.

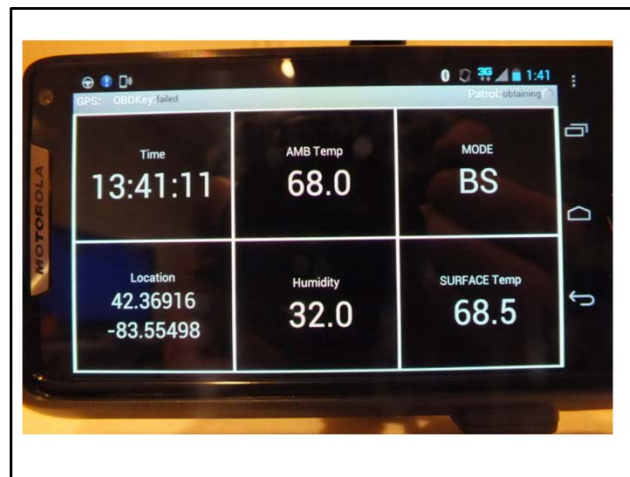


Figure 3 – DataProbe Screen Example in Phone and Surface Patrol Mode Only



Figure 4 – DataProbe Screen Example in Phone, Surface Patrol, and CAN Mode

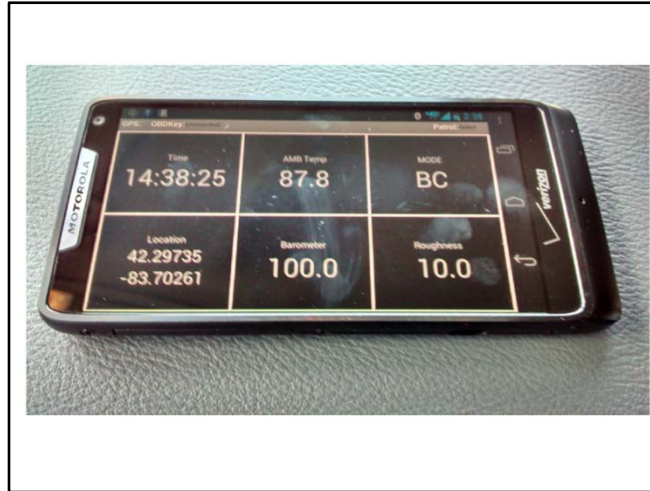


Figure 5 – DataProbe Screen Example in Phone and CAN Mode Only

Hardware

The main hardware that comprised the data collection system installed in vehicles included the phone, the Surface Patrol sensors, and the OBD key. The other project hardware consisted of the virtual server housed at the University of Michigan that received the data from the smartphones and distributed the verified data to weather analysts. The virtual server was used as a production server exclusively for receiving, sorting, and sending IMO 2.0 data. Though it held significant amounts of data, its main challenge was maintaining services without downtime caused by interruptions in service. It is described in more detail in the Data Collection section of this report.

The Phone

The smartphone used in the project was a Motorola Droid Razr M (XT907) running the Android version 4.1.2 Operating System. All the vehicles in the fleet used the same phone. Backup phones for each of the project phones were able to be procured for a very small amount after a year in service from the University of Michigan’s cellular service vendor, Verizon. Over the course of the project, seven phones were replaced primarily due to battery malfunctions, occasionally in hot weather. No phones were replaced due to excessively cold weather, despite the fact that the winter of 2013-2014 was colder than most previous Michigan winters.

Each phone had an unlimited data plan and no voice/calling options. The phone used an active Verizon 3G/4G data service to automatically transfer accumulated DataProbe data to the server at the University of Michigan.

The cost of the data plan for each phone averaged \$37 per month, so a 60 vehicle fleet averaged about \$2,220 per month. One option that we were able to employ during the summer months was to put the WMT phones on “vacation” where the project was only charged a five dollar fee to keep a phone on the plan instead of the usual \$37 per month. This saved \$3,840 over the period of six months that the WMTs were out of service.

Most phones were hardwired to the cigarette lighter/accessory fuse with a 12V to 5V voltage regulator that allowed for a key/on, key/off system, though some were connected to the cigarette

lighter via a 2-port 12 volt USB cigarette lighter adapter manufactured by Kensington. Most of the phones had a similar “dead” phone problem where the battery discharged to the point of shut down after not being used over the weekend or if the driver was on vacation or just not driving the vehicle very often. The dead phone needed to recharge by driving the vehicle to charge the phone before the operating system and the DataProbe program could be re-initiated. Due to a USB power based automatic activation of DataProbe, the program would not initiate automatically after restarting the phone while driving and the driver would have to manually restart the program. If the driver was not paying attention to the phone, it might not start at all during the trip and only startup during the next trip after the phone was recharged.

The phone was securely mounted in the vehicle by either gluing or screwing a compatible car mount adapter which holds the phone onto the instrument panel or into the headliner near the rear view mirror (which was necessary for the WMTs because an instrument panel mount blocked the driver’s view of other instruments). Mounting the phones with a clear view of the road ahead was necessary in order for the cameras in the phones to take clear and accurate photos of weather and road conditions. These photos were also sent to the weather analysts as another data modality. Each phone mount contained a magnet that triggered a signal to the Android operating system to automatically activate the DataProbe application when the phone was placed in the mount.

The OBD Key

CAN data was collected on all 40 of the light duty vehicles via the Bluetooth-enabled OBD key located under the steering wheel under the instrument panel, as seen in Figure 6. The OBD-II scan tool used is a consumer-level Bluetooth-enabled device based on the popular ELM327 OBD-interface command set. The scan tool accesses vehicle system information via the ISO-15765 high speed CAN bus available on most newer domestic vehicles. The OBD-II scan tools used, depending on the date of install, are the OBDKey130 or OBDKey140 models manufactured by KBM Systems, Ltd. (www.kbmsystems.net). This device is powered directly from a vehicle’s SAE-J1962 OBD-II connector during operation and communicates directly via Bluetooth to the DataProbe software running on the Droid Razr M.



Figure 6 – Bluetooth enabled OBD Key for CAN data

Though generally reliable, the OBD key needed, at times, to be reset by taking it in and out of its socket under the steering column. The challenge was training the drivers to notice that it was not working properly and responding accordingly. There were three ways of noticing if the device was not working properly: 1) if the blue light on the OBD key did not flash at all, 2) if the blue light on the OBD key remained on and did not flash, or 3) if the “Mode” on the phone screen did not show a “C” while the vehicle was in operation.

UMTRI also monitored each of the phones weekly, examining the data received from each phone. If the phone sent a file to UMTRI that started with only the phone number instead of the Vehicle Identification Number (VIN), this indicated that the vehicle was not collecting or sending CAN data. This applied only to the light vehicle fleet, since the WMT fleet did not send CAN data. If a vehicle was designed to send CAN data, and it did not send CAN data over 10 percent of the time, then UMTRI notified the driver and his/her supervisor that the driver needed to adjust the OBD key.

The Surface Patrol Sensors

The other hardware included on 10 light duty vehicles and 10 WMTs was the Surface Patrol sensors that measured humidity, ambient air temperature, and road surface temperature. The Surface Patrol HD unit is manufactured and sold by Vaisala Corporation (www.vaisala.com/en/products/surfacesensors/Pages/DSP211.aspx). Each Surface Patrol unit was comprised of one road surface temperature device as seen on a WMT in Figure 7 and in a variety of areas on the trucks and cars where it was not affected by the heat within the engine compartment or debris buildup from the road.

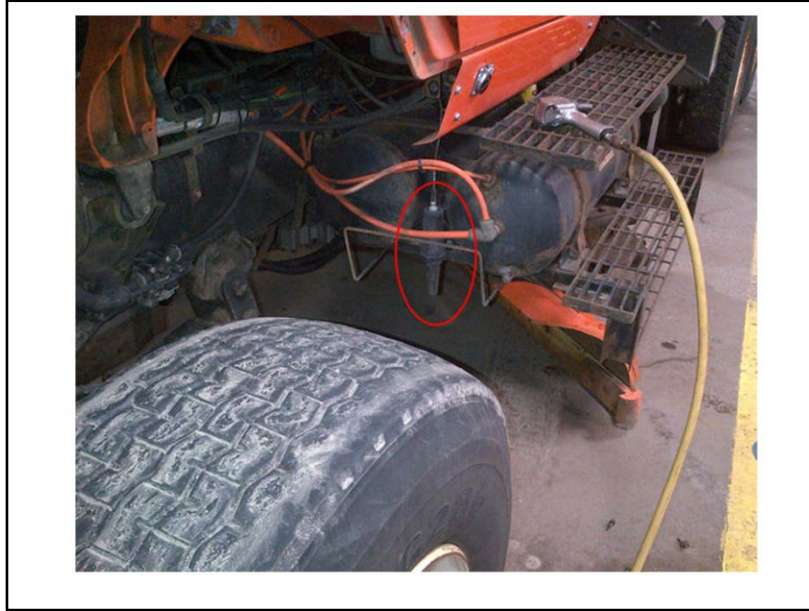


Figure 7 – Surface Patrol Road Temperature Sensor on a winter maintenance truck (WMT)

The final external Surface Patrol sensor, the humidistat shown in Figure 8, gathers ambient air temperature and humidity data. Data from the two sensors are connected to a “spreader” unit that conditions the data. The “spreader” is a unit, circled in red in Figure 1, that takes the data collected via the humidistat and the road surface temperature devices and translates it into data that is readable by the DataProbe application. The Surface Patrol unit, as a whole, is wired directly into the vehicle’s 12 volt bus. UMTRI added a Bluetooth adapter to the spreader unit in order to send the data to the phone wirelessly. The Bluetooth adapter, developed by Roving Networks is called a Firefly (RN-240M) RS-232 to Bluetooth adapter (www.microchip.com/pagehandler/en-us/products/analog/dataconverters/home.html). The Firefly requires a VKTech 12 volt to 5 volt USB regulator to supply vehicle-derived power for operation.

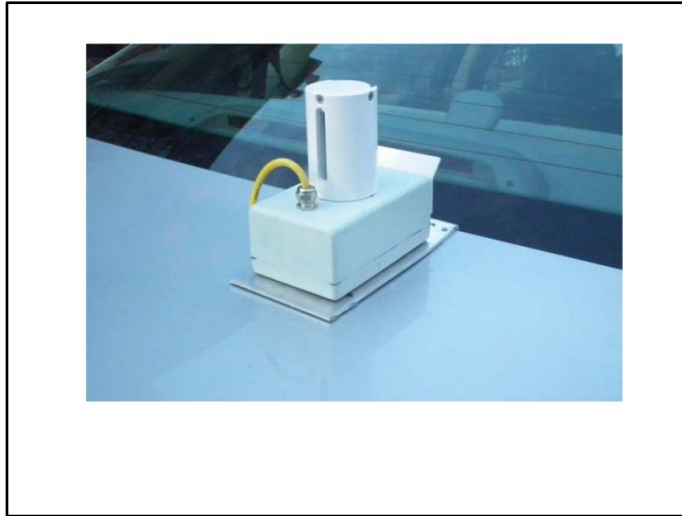


Figure 8 – Surface Patrol Humidity Sensor Mounted on a Car Trunk

During the winter months, the humidity sensor was particularly sensitive to the extreme, high pressure water used to clean the WMTs that accumulated significant amounts of snow and ice. Vehicle cleaners were given instructions not to use the high pressure water on the humidity sensor after a few of the sensors failed to send data, and it was found that water had entered the sensor enclosure.

The maintenance on the road surface sensor requires occasional re-calibration that can be time-consuming and difficult to determine when it should occur. UMTRI monitored the data received from vehicles using these sensors weekly, in order to determine if the values were out of the ordinary. Extreme values led to re-calibration requests to MDOT garages. A more consistent re-calibration process within the MDOT fleet may be required if the project continues into the future.

Data Collected

Over the course of the 13 months of data collection (started collection of data in March 2013 once vehicle instrumentation was completed), the UMTRI/MDOT IMO 2.0 project received 346,019 files (319 gigabytes of data). Of those files, 149,815 files (147 gigabytes of data) were considered invalid because of flaws in the data collected. Valid data was collected, stored, and sent to the six weather analyst groups throughout the country, resulting in 196,204 files (172 gigabytes of data). Of those valid files that were shared with the six weather analysts across the U.S., 151,610 (152 gigabytes) were data files and 44,594 (20 gigabytes) were photos. These data files represent 371,455 miles driven by MDOT vehicle operators over a 13 month period.

Data Flow

Figure 9 shows the data flow for the project with vehicles using DataProbe on the smartphones to capture and assemble the data files and photos, and transfer them to the server at the University of Michigan, where the server is programmed to sort the files into valid and invalid files

(anomalies). The valid files make up the project database and are sent to the weather analysts, while UMTRI backs up the data and summarizes the data received on a weekly basis to review progress and identify issues that need improvement/trouble shooting.

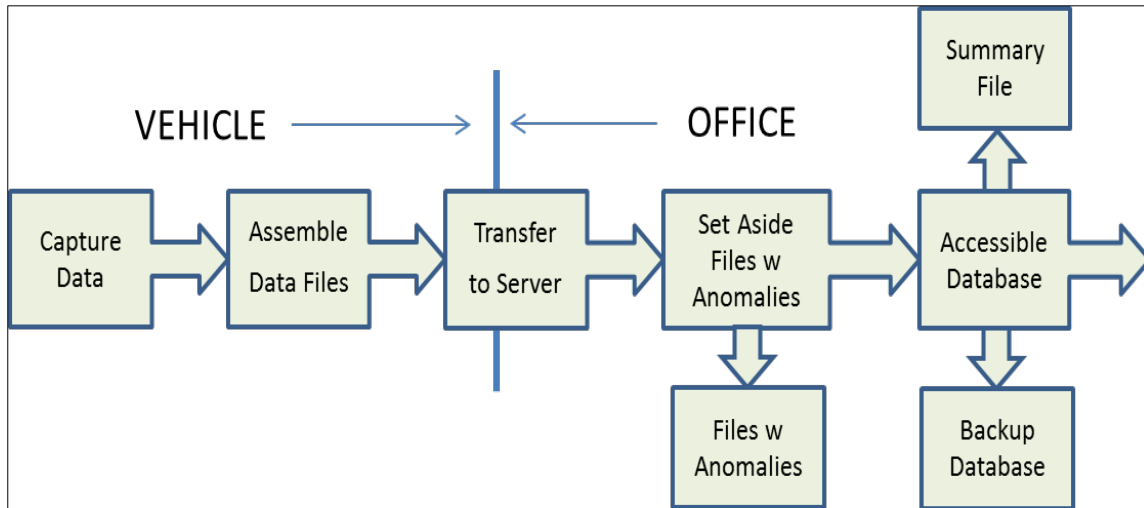


Figure 9 – IMO 2.0 Data Flow Diagram

DataProbe File Naming Conventions

In capturing the data from the various sources on each vehicle, DataProbe creates different file names based on the type of data received. Appendix 1 shows the list of file name conventions used throughout the project, depending on the version of DataProbe in use. In the final version of DataProbe used in the project, all files from WMTs begin with the ten digit phone number, followed by the month, day, year, the hour (military time), minutes, seconds, and the version of DataProbe used in generating the file.

For vehicles with CAN data, the first seventeen digits are the VIN followed by the month, day, year, the hour (military time), minutes, seconds, and the version of DataProbe used in generating the file. As noted earlier, if a vehicle that should send CAN data does not, the phone number will appear at the beginning of the file name instead of the VIN. Photo files names follow the same rules for data file names.

DataProbe File Format

Files created by DataProbe can differ based on the three types of data collected by a particular vehicle: phone data, vehicle CAN data, and Surface Patrol data. As noted earlier, only 10 light vehicles and 10 WMTs of the 60 vehicles in the study were configured with the Surface Patrol sensors to measure road surface temperature, humidity, and ambient air temperatures. These vehicles have data fields in their files that are not in the files of vehicles that are not instrumented with Surface Patrol sensors. Appendix 2 shows the final data dictionary for the project by showing the data source, data field, field description, data type, format, and field name (Primary Alias).

Figure 10 provides an example of the eight lines in the header and the first 32 seconds of a 300 second (five minute) CSV data file that combines phone, CAN, and Surface Patrol data. Each line in the CSV data file represents a one second reading. Note that the accelerometer data from the phone shown is the first two X, Y, and Z readings of the 100 readings measured per second in each axis direction.

TIME	LAT	LONG	ALT	SAT	HEAD_G	SPD-G	BRK	RPM	SPD-C	ACC	ABS	TCSE	TCSB	ESP	STMP	DPNT	TMP-SP	HUMD	X1	Y1	Z1	X2	Y2	Z2
16:33:51	42.25392	-84.4774	898.39	11	0	0	0	1368	0	0	0	0	0	0	34.5	6.7	32	33	10.04201	-0.15691	-0.17651	9.747803	-0.21574	0
16:33:52	42.25394	-84.4774	894.78	11	0	0	0	1368	0	0	0	0	0	0	34.5	6.7	32	33	9.747803	-0.09807	-0.05884	10.17931	-0.09807	-0.27458
16:33:53	42.25397	-84.4774	884.62	9	0	0	0	1368	0	0	0	0	0	0	34.5	6.7	32	33	10.53235	-0.35304	-0.58839	9.924332	-0.13373	-0.09807
16:33:54	42.25397	-84.4774	884.62	9	0	0	0	1368	0	0	0	0	0	0	34.5	6.7	32	33	10.12047	-0.11768	-0.47072	9.571289	-0.07846	-0.05884
16:33:55	42.25397	-84.4774	884.62	9	0	0	0	1368	0	0	0	0	0	0	34.3	6.7	32	33	10.37544	-0.23537	-0.45111	9.728195	-0.21574	0.03923
16:33:56	42.25397	-84.4774	884.62	9	0	0	0	1368	0	0	0	0	0	0	34.5	6.7	32	33	10.12047	-0.03923	-0.56879	9.787033	-0.11768	-0.25497
16:33:57	42.25396	-84.4774	883.63	12	234.8	1.1	0	1368	0	0	0	0	0	0	34.5	6.7	32	33	10.15968	-0.11768	-0.25497	10.12047	-0.17651	-0.33342
16:33:58	42.25396	-84.4774	878.38	12	234.8	0	0	1368	0	0	0	0	0	0	34.5	6.7	32	33	10.27737	-0.33342	-0.35304	10.0224	-0.19614	-0.23537
16:33:59	42.25396	-84.4774	878.71	9	234.8	0	0	1368	0	0	0	0	0	0	34.8	6.8	32	33	9.963562	-0.19614	-0.13373	9.963562	-0.19614	-0.13373
16:34:00	42.25396	-84.4774	878.71	9	161.5	0.6	0	1184	0	0	0	0	0	0	34.8	6.8	32	33	10.03224	-0.13373	-0.25497	10.23814	-0.09807	-0.39227
16:34:01	42.25396	-84.4774	878.71	9	161.5	0.6	0	1121	0	0	0	0	0	0	34.8	6.8	32	33	10.00278	-0.25497	-0.17651	9.98317	-0.01961	-0.23537
16:34:02	42.25394	-84.4774	881.99	12	179.4	1.7	0	1101	0	0	0	0	0	0	34.8	6.8	32	33	10.00278	-0.33342	-0.31381	10.17931	-0.07846	-0.27458
16:34:03	42.25395	-84.4774	885.27	9	179.4	0	0	1065	0	0	0	0	0	0	34.8	6.8	32	33	10.15968	-0.25497	-0.19614	10.15968	-0.25497	-0.19614
16:34:04	42.25394	-84.4774	887.24	9	179.4	0	0	1055	0	0	0	0	0	0	34.8	6.8	32	33	10.15968	0.117676	-0.39227	9.963562	-0.13373	-0.35304
16:34:05	42.25393	-84.4774	889.54	9	179.4	0	0	1030	0	0	0	0	0	0	34.8	6.9	32	33	10.19891	-0.05884	-0.39227	10.21854	0.196136	-0.37265
16:34:06	42.25393	-84.4774	889.54	9	179.4	0	0	1018	0	0	0	0	0	0	34.8	6.9	32	33	10.04201	-0.39227	-0.21574	9.767426	-1.43178	-0.15691
16:34:07	42.25393	-84.4774	889.54	9	179.4	0	0	1008	0	0	0	0	0	0	34.8	6.9	32	33	9.728195	-1.56906	-0.13373	9.865494	-0.64723	-0.13373
16:34:08	42.25393	-84.4774	888.88	12	179.4	0	1	1010	0	0	0	0	0	0	34.5	6.9	32	33	9.98317	-0.68646	-0.31381	9.904724	-1.00027	-0.35304
16:34:09	42.25392	-84.4774	889.86	12	179.4	0	1	988	0	0	0	0	0	0	34.5	6.9	32	33	10.06163	-0.09807	-0.27458	9.943939	-0.07846	-0.21574
16:34:10	42.25392	-84.4774	886.58	9	179.4	0	1	990	0	0	0	0	0	0	34.5	6.9	32	33	9.98317	-0.03923	-0.21574	9.904724	-0.23537	-0.15691
16:34:11	42.25392	-84.4774	886.58	9	179.4	0	1	981	0	0	0	0	0	0	34.8	6.9	32	33	9.904724	-0.05884	-0.13373	10.21854	-0.27458	-0.27458
16:34:12	42.25392	-84.4774	887.9	12	179.4	0	1	976	0	0	0	0	0	0	34.8	6.9	32	33	10.31659	-0.17651	-0.41188	9.943939	0	-0.23537
16:34:13	42.2539	-84.4774	892.49	12	179.4	0	1	968	0	0	0	0	0	0	34.8	6.9	32	33	9.98317	-0.11768	-0.27458	10.33621	-0.35304	-0.54918
16:34:14	42.2539	-84.4774	892.49	12	179.4	0	1	961	0	0	0	0	0	0	34.5	7	32	34	9.747803	-0.07846	0.03923	9.747803	-0.07846	0.03923
16:34:15	42.2539	-84.4774	892.49	12	179.4	0	1	968	0	0	0	0	0	0	35.3	7	32	34	10.31659	-0.35304	-0.37265	10.14008	-0.27458	-0.37265
16:34:16	42.25391	-84.4774	891.83	11	179.4	0	1	959	0	0	0	0	0	0	35	7	32	34	10.4735	-0.25497	-0.43149	9.924332	-0.09807	-0.21574
16:34:17	42.25391	-84.4774	891.83	11	179.4	0	1	953	0	0	0	0	0	0	35.3	7	32	34	10.25775	-0.19614	-0.39227	10.10085	-0.21574	-0.37265
16:34:18	42.25391	-84.4774	891.83	11	179.4	0	1	948	0	0	0	0	0	0	35	7	32	34	9.904724	-0.27458	-0.23537	9.904724	0.078461	-0.17651
16:34:19	42.2539	-84.4774	892.16	11	179.4	0	1	938	0	0	0	0	0	0	35	7	32	34	9.98317	-0.05884	-0.31381	9.924332	0.019608	-0.19614
16:34:20	42.2539	-84.4774	892.16	11	179.4	0	1	927	0	0	0	0	0	0	35.3	7	32	34	10.14008	-0.21574	-0.41188	9.943939	0	-0.39227
16:34:21	42.2539	-84.4774	892.16	11	179.4	0	1	960	0	0	0	0	0	0	35.3	7.1	32	34	10.12047	0.019608	-0.33342	10.12047	0.019608	-0.33342
16:34:22	42.2539	-84.4774	892.16	11	179.4	0	1	843	0	0	0	0	0	0	35.8	7.1	32	34	10.43428	-0.33342	0.019608	9.845871	-0.50995	0.274582

Figure 10 – DataProbe File Format Example

One major change we made in the way data was displayed in the data files was the inclusion of a missing data code (10001) for all the fields in a file. Originally, programmers used the previous valid data for a field that was missing. This caused many parts of files to look like the DataProbe program was repeatedly recording the same values when in fact it was not sending valid data for a certain amount of time. This delay could be caused by interference in the data transmission from one of the devices in the vehicle or a temporary failure by a device. The introduction of a missing data code provided a better diagnostic about how data was collected, and it also provided more accurate data for the weather analysts.

Because of the many iterations of the DataProbe program, it was difficult to keep track of the changes in the data dictionary across the different versions of DataProbe. The combination of having multiple versions of DataProbe running at the same time in the fleet and programmers making changes among the different versions created confusion in developing a final data dictionary. For example changing the name of the throttle field from THROT to ACC (accelerator position) caused confusion because it might occur only for one version of DataProbe instead of all versions. It was not until the end of the project that the data dictionary was standardized across all the vehicles in the project.

IMO Photos

Besides data collected via the phone, OBD key, and Surface Patrol devices, one of the unique features of the IMO project was the ability to capture a roadway weather situation using the camera on the phone. The ability to use the camera to substantiate what weather analysts see in the data is an invaluable tool for near time weather analysis. Figure 11 displays four examples of photos taken during the winter of 2014. Notice how clearly one can make out the condition of the road.

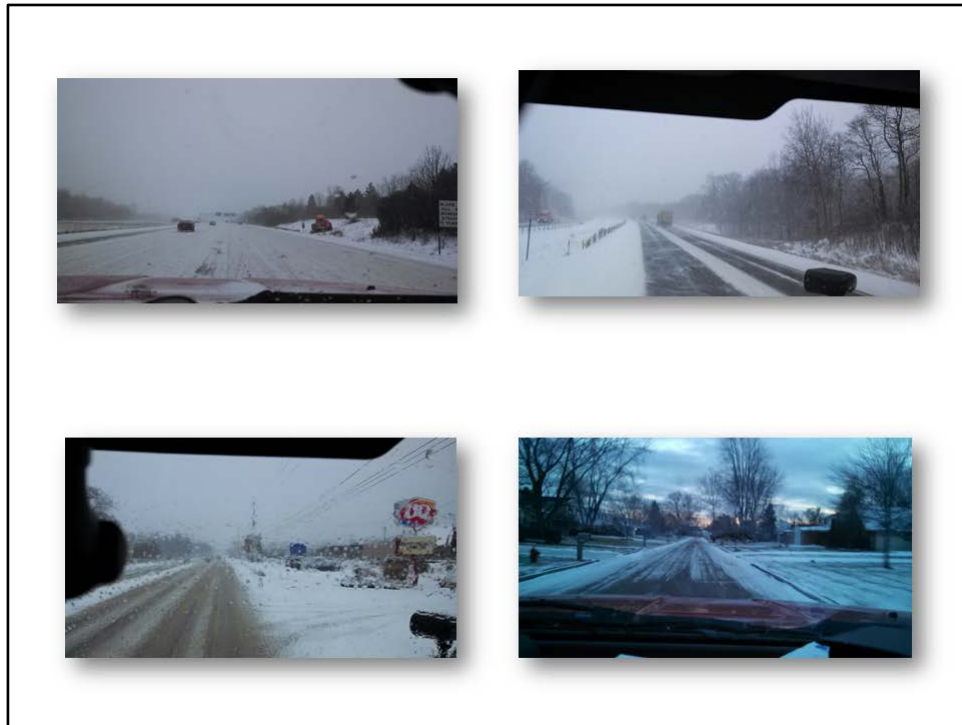


Figure 11 – DataProbe Photo Examples

For this project, photos were taken during three different scenarios:

1. The driver can take a photo when he/she sees a weather-related event occurring, such as the beginning of snow storm.
2. Anyone authorized to log onto the DataDroid web portal can take photos with any vehicle in service
3. If a vehicle with CAN data encounters an ABS or traction control event, DataProbe takes a photo.

Of the three scenarios, the ABS/traction control scenario was the most difficult to program. The CAN codes related to ABS/traction control are not easily distinguished from other similar events on the CAN bus. Also because ABS/traction control events do not happen regularly and we are unable to easily see which vehicles have these technologies onboard, it has been difficult to capture the event in data format as well as in photo format. In order to capture more photo

information about an event, all the phones were programmed to take three pictures with one push of the camera button. In all cases, three photos are taken one second apart to ensure photo clarity and reliability.

Photos in the DataProbe system differ from the data files because they do not have to wait five minutes to complete data collection before uploading to the University of Michigan server. Photos are sent as soon as they are taken (unless DataProbe is in the middle of uploading a data file), increasing the timeliness of the photos.

One can even envision taking more photos and developing a tool to interrogate and interpret what is happening in the photo from a weather perspective. The GPS data connected to the photo can be used to quickly locate where the photo was taken and signage can be updated quicker than through weather models based on vehicle and other supporting data.

IMO Data Server

The data collected via DataProbe and transmitted to the University of Michigan was stored on a virtual server within the MiServer group on the UM campus. This process allowed maximum flexibility in changing the project storage space as needed, without the cost of purchasing additional hard drives or servers. Backup is performed daily, and backup files are accessible for up to a month. UMTRI performed a monthly backup to maintain the long term status of project files.

The server was designed as a production server that receives data files, sorts the files into valid or invalid files, and then either puts them into a folder that allows only certain weather analysts servers to “pull” the data or “pushes” the data to the weather analysts servers that preferred to receive the data in this manner. The virtual server uses an Apache HTTP web server to manage the web traffic between the University of Michigan server and the IMO phones. Apache receives messages via HTTP from the phones and provides responses from the IMO service and vice versa. The Apache server also manages the transfer of files being uploaded from the phones also via HTTP.

Sending files to the weather analysts in a timely manner was considered one of the main goals of the project, and the UM server and sorting program played a key role in making that happen. At the beginning of the project, files were stacking up on the phones and not being uploaded in a timely manner, especially when the driver was turning off the vehicle and the phone needed to upload its final set of files for a given period of time. Making adjustments in the DataProbe program to upload a file as soon as it was completed instead of waiting for a group of files to send eased the congestion on the phone and created a consistent delivery time of about seven to nine minutes from the time a file was first created. This time included the five minutes of data collection, which meant that the file was uploaded, sorted, sent, and arrived at the weather analysts in about two to four minutes. Hence, the designation of project data delivered in near time rather than real time.

One can see from the number of valid and invalid files collected that there were almost as many invalid as valid files collected. This result demonstrates two things: 1) the sorting program may

have been too restrictive in determining if a file is valid, and 2) the DataProbe data collection system is very sensitive to anomalies in the system.

The original sorting program was designed to make a file invalid if any of the following conditions were met:

1. Any time during the five minutes of data collection three or less GPS satellites were visible to the phone.
2. More than 325 columns were reported in the CSV file, which would indicate a corrupted file or a data collection error.
3. The file did not show movement from one location to another as measured via the GPS coordinates.

After seven months of data collection, it became clear that some of the files contained valid data that was being discarded because some of the data in the same file was invalid. Because of this, we changed the sorting program to use the following criteria to make a file invalid:

1. If less than three GPS satellites are used for the first sixty seconds of a file
2. If after the initial sixty seconds there are less than three GPS satellites used for more than fifteen seconds
3. If the speed generated from the GPS is less than five miles per hour for more than 295 of the 300 maximum number of seconds in a file
4. If any of the accelerometer values in the 5th row of data are marked as missing (10001)
5. If there are more than 335 columns in the file

The sensitivity of the overall DataProbe data collection system also played a role in the large number of invalid files generated and collected. Issues related to smooth and consistent phone startup and uploading of data, the connectivity of the DataProbe software with GPS satellites, the OBD key and Surface Patrol devices, and the operation of the OBD key and Surface Patrol devices themselves all play a role in affecting the validity of the data files. One can also see a large number of vehicles idling for long periods of time as a driver-related aspect of file validity. However, the number of invalid files does not affect the quality of the data received by the weather analysts; in fact, keeping invalid files from reaching the analysts improves their ability to use the valid data. It keeps them from having to sort through files that have little or no value to support application development.

DataDroid Web Portal

One of the unique additions to the IMO 2.0 project over the original Slippery Road project was the development of a web portal to track vehicles in service, take photos remotely, and send text messages to drivers. The PHP-based web portal, named DataDroid, was developed using the Apache web server software, making its links to the vehicles in service based on their uploading

of data through the Apache web service. Figure 12 displays what a user sees after logging onto the website.

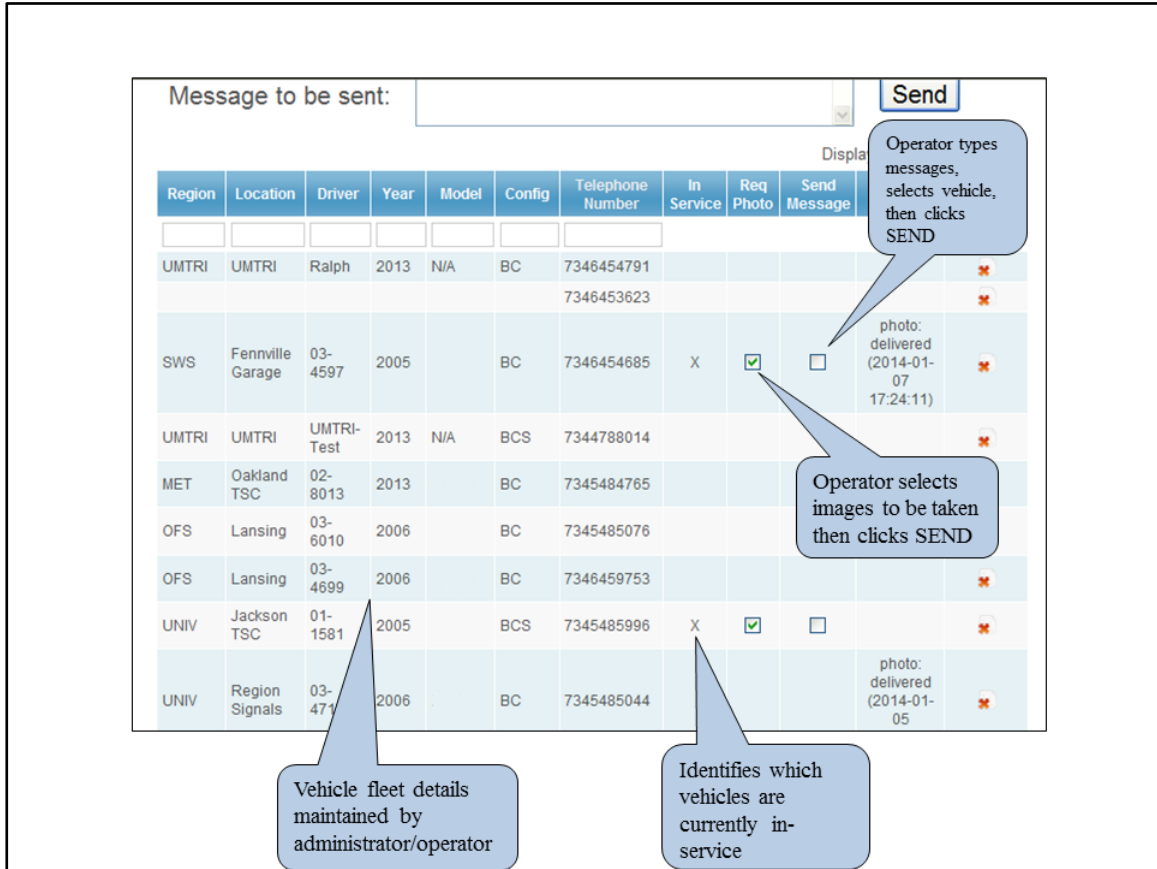


Figure 12 – DataDroid Web Portal Page

The web portal was tested continually throughout the project to see how well it identified vehicles in service, took photos remotely, and sent messages to drivers. In almost all cases, the portal was able to take photos remotely and send messages. Its weakness was its inability to identify all the vehicles in service. It would identify some vehicles in service but not all of them, though it would not provide false positives, saying a vehicle was in service when in fact it was not.

One of the major challenges for the project was knowing when a vehicle was in service. UMTRI researchers had no way of knowing if the DataProbe units were not sending data because the vehicle was not in service or if the DataProbe unit was inoperable. The web portal was an attempt to help sort out this problem, but unfortunately, it turned out not to be as robust as was hoped. Because of this weakness, tracking DataProbe performance was always performed after the fact, reviewing the previous week’s data to see which vehicles were not sending data and contacting the drivers and their supervisors to see if the vehicle was indeed out of service the previous week. One of the continuous improvement tasks for the IMO 2.0 team is to improve the web portal to become the ground truth about what is currently occurring on the road. Having

ground truth for vehicles in service will allow researchers to be more proactive in addressing hardware and software problems quicker, getting the systems back up and running when they fail.

Data Quality

Another goal for the project was to provide accurate data to the weather analysts. Though MatLab programs designed specifically for this project to track vehicle data provided good information about how many miles were traveled by each vehicle, UMTRI researchers did not have the database support to develop programs to analyze the quality of the data gathered. Researchers mostly relied on manual spot checks of files to see if the data met a test of “reasonableness,” that is, were the data values recorded within a normal range of values for a specific data field such as RPM or road temperature.

The weather analysts were also expected to develop programs to test the data against other local measures of weather. The National Center for Atmospheric Research was the most thorough in testing data against other weather metrics in the area where IMO vehicles travelled. In general, DataProbe weather-related data seemed to pass their tests, except for some surface temperature data. Upon inspection, it turned out that the surface temperatures failed their tests because there were no other weather points in the area with which to compare the data. Mixon-Hill also provided insight into where DataProbe data was missing or inconsistent. The comments of both groups helped improve the system. More interaction with the groups while they developed their monitoring systems would have allowed UMTRI researchers to make adjustments to the data collection sooner in the project.

Driver Engagement/Project Momentum

The early challenges the project faced with the need to implement multiple DataProbe versions, the inability to generate support from one of the automotive manufacturers or to reverse engineer their CAN codes (which led to dropping their vehicles from the study), and some other technology challenges all coalesced to drain some momentum from the project. There was a need to re-build engagement in the project, and during the summer of 2013 seven major actions helped move the project in the right direction:

1. Because there seemed to be so many different challenges, a survey was developed to get driver feedback about how the DataProbe system was working and where the main challenges lay.
2. UMTRI hired an electrical engineer versed in hardware and software to provide the necessary support for the project technology.
3. An “all hands on deck” call-in meeting led by MDOT brought together drivers and supervisors to reiterate the importance of the project.
4. A weekly vehicle activity report was developed by UMTRI and emailed to IMO team members including MDOT supervisors that detailed how many miles and photos were taken by each driver during the previous week.
5. MDOT took the vehicle activity report data and created weekly and cumulative graphs and emailed them to each IMO driver, comparing how many miles each driver had driven and how many photos each driver had taken during the previous week and compared them to their colleagues.

6. MDOT designated a supervisor from four areas (Southwest Region office, Lansing central Office of Field Services, Metro Region office, and University Region office) to take the lead in supporting the IMO project in their area.
7. The IMO team initiated weekly calls/meetings that included UMTRI, MDOT, FHWA, Booz Allen Hamilton consultants, Mixon-Hill, NCAR, Vaisala, and other stakeholders as needed to discuss progress on a clear set of issues in prioritized order that focused the team on what needed to be done in the near term (weekly), while also focusing on any longer term needs like the 2014 ITS World Congress – Detroit in September.

The IMO User Survey

UMTRI/MDOT IMO team developed a user survey to get some feedback from drivers about the DataProbe system. The survey confirmed what was discussed anecdotally that the Surface Patrol devices would not work unless we found a Bluetooth interface for the Surface Patrol spreader system, that the CAN codes from one of the manufacturers were not working and actually causing alarm for some of the drivers, and that the multiple versions of DataProbe created inconsistent file formats for the weather analysts.

UMTRI prioritized the list of issues, and armed with this information, UMTRI hired an electrical engineer to support the hardware and software challenges the project faced. MDOT called an “all hands on deck” meeting to reiterate the importance of the project, and quickly followed up with the vehicle activity tracking report published weekly by UMTRI and individual graphs for each driver comparing the number of miles driven and the number of photos taken weekly and cumulatively published weekly by MDOT.

Vehicle Activity Tracking

As noted in the previous section, tracking vehicle activity for a sixty vehicle fleet is a daunting task, especially if there is no automated process that tracks vehicles in service in real time. The best process available was to track each vehicle on a weekly basis by using MatLab programs developed by UMTRI researchers to summarize all the data for a particular vehicle including the number of miles traveled and the number of ABS and traction control events for the previous week. As shown in Figure 13, the weekly IMO vehicle activity report that was the result of the analyses provided MDOT managers and supervisors an overview of the previous week’s driving by each driver/vehicle in the fleet. This spreadsheet also provided an analysis of the Surface Patrol devices for each vehicle, and the current list of IMO 2.0 vehicles in use.

A quick review of the data shows the number of miles driven, the number of photos taken, and the number of ABS and traction control events reported by the DataProbe program. It also shows the total number of miles driven and photos taken over a designated number of weeks (in this case since the beginning of 2014) for each driver and for the entire IMO 2.0 sixty vehicle fleet.

This spreadsheet was sent to MDOT/IMO supervisors and team members, making it one of the discussion topics for a weekly IMO meeting. This meeting became one of the points of contact for the drivers as they were able to compare discrepancies between the number of miles DataProbe reported and their own mileage log from the previous week. This process alerted UMTRI to any inconsistencies between DataProbe miles reported and the actual number of miles driven, allowing for corrective action to be taken.

			= Received data or photos last week		= No data or photos received last week							
Number of Vehicles	Vehicle Number	Driver	Location	Version	Surface Patrol	Miles Driven 3.9.14 to 3.15.14	Photos	ABS	TSC	Total Miles Driven (12 Weeks)	Total Number of Photos (12 Weeks)	
	Southwest											
1	03-4745	Ed Martin	South Haven Garage	3.7.1	1	128	1			985	251	
2	03-4766	Glenn Ingold	Plainwell Garage	3.7.1	1					844	0	
3	03-4767	Mike Freeman	Coloma Garage	3.7.1	1	284	9	1		1090	402	
4	03-4817	Zack Clothier	Kalamazoo Garage	3.7.1	1	133	3	2		1491	193	
5	03-4847	Mike Freeman	Coloma Garage	3.7.1	1					478	185	
6	03-4597	Rick Weaver	Fennville Garage	3.7.1		427	1			1982	196	
7	03-4717	Mike Streeter	Kalamazoo TSC	3.7.1a		272				2527	117	
8	03-4737	Brian Reiter	Hastings Garage	3.7.1		8		1		287	85	
9	03-4718		Coloma TSC	3.7.1						354	69	
10	03-4851	Trace Plummer	Jones Garage	3.7.1	1	132	3			1198	326	
11	03-4719	Mark Georgopoulos	Kalamazoo TSC	3.7.1a						1635	120	
	WMTs											
	04-1602	5070	Sawyer Garage	3.7.1w						5406	549	
	04-1604	5104	Kalamazoo Garage	3.7.1w	1					2832	701	
	04-1605	4746	Sawyer Garage	3.6.5w		51				4339	298	
	04-1629	5150	Kalamazoo Garage	3.7.1w						2894	87	
	04-1630	6014	Kalamazoo Garage	3.7.1w		6	42			2473	404	
	04-1635	0495	Marshall Garage	3.7.1w						3723	524	
	04-1636	5483	Coloma Garage	3.7.1w						3487	645	
	04-1637	7726	Marshall Garage	3.7.1w						3221	3	
	04-1640	5973	Kalamazoo Garage	3.7.1w						5564	1075	
	04-1644	0908	Kalamazoo Garage	3.7.1w	1					241	103	
	04-1645	5150	Marshall Garage	3.7.1w	1					3582	217	
	04-1661	5872	Kalamazoo Garage	3.7.1w	1					3877	289	
	04-1676	5356	Coloma Garage	3.7.1w						920	112	
	04-1678	1579	Coloma Garage	3.7.1w	1					5660	421	
	04-1680	3772	Marshall Garage	3.7.1w	1					2158	410	
	04-1696	8073	Sawyer Garage	3.7.1w	1					443	231	
	04-4006	3996	Coloma Garage	3.7.1w	1					4736	569	
	04-4011	6448	Coloma Garage	3.7.1w	1					1759	249	
	04-4014	7578	Sawyer Garage	3.7.1w	1					3038	41	
	04-4015	8451	Coloma Garage	3.7.1w	1					3039	108	
	Lansing											
12	03-6010	Matt Chenoweth	Lansing	3.7.1	1	129	15			306	27	
13	03-4667	Chuck Bergmann	Lansing	3.7.1	1	994	24	4		4661	1084	
29	02-9133	Matt Pratt	Lansing	3.7.1	1					4477	3568	
15	03-4721	Cory Rogers	Lansing	3.7.1		176				361	7	
	Metro											

Figure 13 – IMO Vehicle Activity Report

MDOT also created and emailed to all the drivers its own graphs of the spreadsheet data to track the mileage and photos taken by each driver in the study for each week and cumulatively over a certain period of time. Figure 14 shows an example of the cumulative number of miles and photos taken by each individual driver.

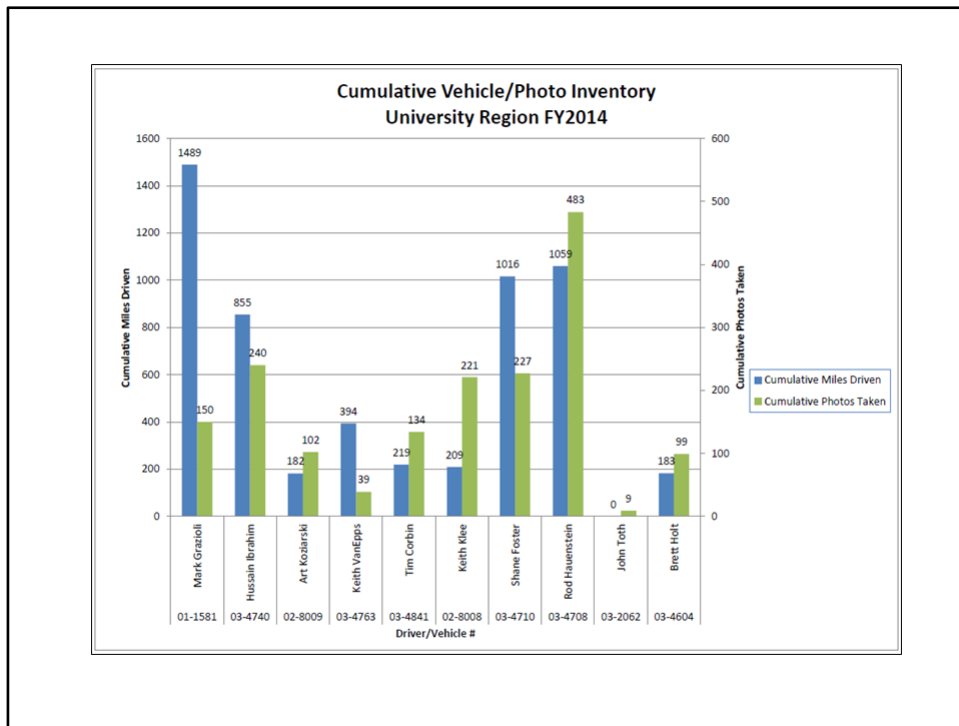


Figure 14 – IMO Vehicle Activity Graph

By keeping track of each week’s worth of data, UMTRI researchers were able to see any patterns of low or no mileage driven, which led to an email or call to a driver to see if the vehicle was in service or if DataProbe was not working properly. Though time-consuming, the direct contact with drivers reinforced the importance of their participation in the project and also reminded supervisors to ask drivers if the DataProbe unit was acting properly, reinforcing their role in the project as well. UMTRI also provided an IMO-Support email address for drivers and supervisors to use to bring up any issues they had with the phones.

The MatLab analyses also provided researchers with insight into which DataProbe units were not reporting data appropriately. As noted earlier in the report, if the file name for a vehicle began with its phone number instead of its VIN, then the vehicle was not providing CAN data. The MatLab program separated the vehicles with phone numbers in the file name from the ones with VINs, so UMTRI researchers could see the number of miles where a vehicle sent CAN data and the number of miles it sent only phone data. If a high percentage of data was phone data only, researchers contacted the driver and the supervisor to see if the OBD key was working properly or if needed to be reset. Because researchers were not able to meet in person with each driver and vehicle to inspect each phone’s problems, using the vehicle activity reports, MDOT graphs, and follow-up phone calls and emails were the most efficient/effective methods to increase driver participation, as well as monitor the DataProbe system.

The UMTRI/MDOT IMO Team Meetings

Early in the IMO team meetings, the agenda tended to focus on a variety of challenges without prioritizing what needed to be done. All issues seem to be associated with high importance. Cataloging issues and creating a risk assessment spreadsheet helped establish order and priority.

The driver survey helped focus the group on the major issues that needed to be addressed, like dropping vehicles from the study because of the lack of support and UMTRI’s inability to reverse engineer the CAN codes. Dropping vehicles from the study also meant finding and instrumenting almost 20 new vehicles in the light vehicle fleet. MDOT IMO supervisors were brought into the weekly meetings to support the transition to the new set of vehicles, and also to monitor and support the IMO drivers in their respective region. This support proved crucial to the project because it brought in supervisors trained in managing technology rollouts and dealing with all the idiosyncrasies that each rollout provided.

Each weekly IMO team meeting now had a list of prioritized “Issues” as shown in Figure 15 with each issue described, its status noted, the risk involved in not overcoming the issue, how the issue would be mitigated, the status (Open or Closed), the expected completion date, goals affected by the issue, and the number of vehicles affected. Each meeting also included representatives from UMTRI, MDOT, FHWA, Booz, Allen, Hamilton consultants, Vaisala, NCAR, Mixon-Hill, and other stakeholders as needed with a focus on meeting clearly defined project goals with discussions on how each issue reported was progressing.

3.20.14								
Issue	Description	Status	Risk	Mitigation	Issue Status	Completion Dates	Goals Affected	# of Vehicles Affected
1	IMO Report	Draft due to Steve. Received and reviewed template for draft report. April 15 draft report to FHWA. Need to set date for first draft to MDOT. March 31st due date for final report draft						
2	Equipping two 2006 vans with CAN devices and Surface Patrol devices, as well as getting CAN codes from the manufacturer	Testing has begun on one of the vans, and we are having trouble identifying ABS and Traction control signals. Testing continues. Still problems with ABS and Traction control codes.	No CAN data recorded in vans for ITS World Congress	See if we can get the CAN data from the manufacturer	Open	Spring '14	1,2	2

Figure 15 – IMO Issues Report

Each week brings more challenges because the hardware and software used for this project are not “hardened” in the sense that an automaker hardens its technology by designing and testing technology until it meets the standards necessary to compete in the marketplace. The IMO process of instrumenting vehicles and gathering information remotely has continuously improved with regard to the hardware and software necessary to generate accurate and timely data. But looking at a project like this as purely a technology project, one misses the point that so many in the information technology world have learned: that successful technology-based projects are a balance of people, process, and technology. For the IMO project through its combination of MDOT drivers and supervisors, weekly meetings with prioritized issues, and continuous improvement of the DataProbe application, this balance is now in place for its next generation of development.

Lessons Learned

The number of major changes that took place during this project provides a long list of lessons learned from a variety of perspectives: vehicles, hardware, software, communication, and users.

Vehicles

- The process for receiving the required CAN bus information from vehicle manufacturers can be a long process that requires planning before the project. Experience suggests not beginning a project until a manufacturer not only provides the necessary CAN data, but also agrees to provide technical support regarding CAN bus message codes and data acquisition.
- Vehicles come in different model years, models, trim levels, accessory, and safety packages. Automakers have a difficult time finding information for models two or three years old much less 7 or 8 years old. Engineers are focused on current and future development. As the result of staff reductions and outsourcing, there are also fewer engineers that know how to interpret the CAN bus message codes.
- One major automotive manufacturer signed a non-disclosure agreement (NDA) with UMTRI. UMTRI also had the support of one of the company's product development supervisors who works in the engine control unit area. This individual was invaluable in providing CAN code interpretation.
- Twenty fleet vehicles from one automotive manufacturer were instrumented for the IMO project. Some CAN message codes for these vehicles were available. However, due to issues with code interpretation and inadequate technical support, nine months into the project all of these vehicles were replaced with the vehicles from the manufacturer with whom we had a Non-Disclosure Agreement. Issues with the vehicles stemmed from the OBDKey/CAN data interface with the DataProbe program, causing the vehicles to perform erratically, with lights flashing, door locks opening and closing, and transmissions locking up. It took more than six months to find 20 vehicles from the other manufacturer in the MDOT fleet to replace the vehicles as only high mileage driver/vehicles were considered for the project. To date, the project still lacks proper CAN bus technical support from the manufacturer for the vehicles that were replaced.
- Maintaining a fleet of 60 vehicles with up to date hardware and software, especially if the software is changing constantly, as it did during the early part of the project, was very difficult. Constantly changing versions of DataProbe led to many different data formats, which made it more difficult for the weather analysts to manage the data they received. Having a separate set of test vehicles to allow testing would improve consistency and control of the data formats.
- Tracking whether vehicles in the fleet were in service was one of the main challenges for this project. The web portal was supposed to be designed to do this; however, it suffered a number of minor failures preventing the display of active vehicles. There were many instances where a vehicle was on the road but not shown as in service on the portal. Because the web portal was designed by programmers outside the U.S., it was difficult to test new iterations of the web portal design to fix problems, especially because of the time difference between countries (USA and the Ukraine in this instance).

Hardware

- One of the challenges faced early in the project was the Bluetooth connection between the Surface Patrol spreader box that translates data from the Surface Patrol device that allows it to be read by the DataProbe phone. We expected to use the same serial port Bluetooth adapter that was used in the Slippery Road Detection and Evaluation study that took place within the last two years, but the IOGear device used was no longer available. Numerous internet sources were tried unsuccessfully to procure the legacy adapters, requiring the investigation of new devices. Three new Bluetooth serial adapters in current production were tested. Working with each of the serial port Bluetooth adapter companies, an attempt was made to get their adapters to work with the DataProbe and Surface Patrol spreader device. But coordination issues resulted even after sending them the IOGear device to show the device worked. Despite these challenges, a UM Electrical Engineering graduate student was able to analyze and determine the root causes of the problem to be undocumented restrictions within the DataProbe software along with the improper setup of the Bluetooth serial adapters. Identification of these hardware and software problems allowed for a new integration of a new adapter into the DataProbe system.
- Costly monthly phone charges created budget issues. Each cell phone cost \$37.00 for monthly data charges on 60 phones for a total monthly cost of \$2,220. Some monthly savings were achieved by putting the phones in the WMTs on “vacation” during the summer months.
- Because of the UM relationship with Verizon, the project was able to purchase 60 extra phones for \$1.00 each to use as backups in case the phones we had in service failed. During the project, seven phones were replaced, primarily because of batteries that overheated. The extra phones are an important issue as phone models can quickly go out of production or change dramatically from year to year. Having this occur in the middle of a project can be an expensive additional cost if the DataProbe software needs to be re-written to support a different type of phone.
- Pictures taken from most of the WMTs were initially taken upside down because of the way the phones were installed (from the headliner). A software fix in DataProbe allowed the photos to turn the photos right side up, but there were two cases where the phones were not installed upside down. In this case the DataProbe fix was turning photos that were correct, upside down.
- Some phones considered for the project only allowed photos to be taken by a touchscreen instead of an exterior button. This would have made it more difficult for drivers to take photos. The phones used in the project did have a button on the side of the phone. It was also possible to re-program the phone to use one of the exterior buttons for taking photos.
- The interface between the CAN bus and the Android-based phone running DataProbe is a Bluetooth OBD key. Our experience with these devices found that there are times when the link between the OBD key and the phone breaks and the phone does not register any data from the CAN bus. Usually the driver can reset the process by taking out and then replacing the OBD key into its socket. At other times the OBD key can be swapped between vehicles and become functional again. Due to a lack of automatic fault detection, drivers must be aware that the blue light on the OBD key is blinking on a

regular basis in order for it to transfer data. Drivers can also look at the phone display to see that the phone is not receiving data because there is no “C” in the section that displays the type of data the phone is receiving. The OBD key is also not functioning if the blue light is on all the time. This means that the device is frozen. Training drivers about this situation is an important part of their IMO training, as well as need to contact technical support when this situation occurs.

- Securing the phone mounts to the dashboard or overhead headliner via screws or epoxy is necessary to allow the phones to maintain stability, especially if the phone is taken in and out of the vehicle during very hot temperatures. Cold temperatures do not seem to create performance issues despite very cold temperatures during the winter of 2013-2014 (+10°F to -10°F for several weeks).
- Wiring the phone directly to the accessory fuse rather than relying on the driver to plug and unplug the charger from the cigarette lighter socket, eliminated the need to glue the charger plug into the phone mount. This power input was very prone to loosen during operation, causing the phone to lose charge during operation because it is not connected to a charging outlet.
- Some of the vehicles in the project are not driven every day, and sometimes the phone loses its charge because it has not been recharged by vehicle usage. When the vehicle eventually is started, it takes time for the phone to be recharged enough to be re-started and run DataProbe. The software must be able to turn itself on when the phone is sufficiently charged and connect via Bluetooth to the OBD key and Surface Patrol links. At times the drivers took the phone inside to attach it to a phone charger to recharge it. In whatever situation the phone is placed, it must adjust and open and close DataProbe appropriately during vehicle operation.
- The Surface Patrol devices: road and air temperature, and humidity sensors have been a continuous challenge during the project. Early in the project, there were issues with the use of a different Bluetooth device to send data from the Surface Patrol spreader box to the phone. Later, the challenge was receiving data from the humidity sensor that seemed to have issues working in the harsh WMT environment and cleaning protocol. Vaisala (Surface Patrol supplier) provided replacement units for failed units during the project. Vaisala also provided good direction as to mounting and installation on the vehicle to ensure device performance. The humidity sensors cannot withstand the pressure washing that is common for WMTs cleaning at the end of the shift.

Software

- Because of the numerous changes needed in the DataProbe program, it became difficult, given MDOT and UMTRI resources, to keep all the vehicles in the fleet updated with the most recent version of DataProbe. This led to numerous differences in file structure which affected how the weather analysts worked with the collected data. A remote update version of DataProbe was later developed that checks with the UMTRI server to see if there is an updated version available. If one is available, the phone downloads, decodes, and begins the download process locally. This new process allows for better control of the DataProbe system, including managing file structures and bug fixes.

- Having an external vendor outside the U.S. re-writing the DataProbe code was very time consuming because of the number of iterations needed in the rewrite along with back and forth communication and testing delays. It is important to have a programmer on staff full time during development in order to write and test code.
- The CAN message codes needed to be tested and programmed into DataProbe for every variation of car or truck in the study, which was very time-consuming. This process continues every year as new vehicles enter the fleet and new CAN codes are needed.
- Automating the photo taking process during any ABS or Traction Control events, took many months to implement because of the many different CAN codes of different vehicle model years and models.
- The auto focus on the smartphone cameras sometimes generated blurry photos, so a re-designed version of DataProbe automatically was designed to take three photos whenever a photo was triggered by the user, from the portal, or by an ABS or traction control event. Also, DataProbe was programmed so the camera did not use the flash that created an unwanted reflection off the windshield.
- Designing the DataProbe program to turn on automatically whenever the vehicle is turned on and to close down properly when the vehicle is turned off was a continuous challenge. The program had a tendency not to turn on during startup, and the driver would have to manually start the program. Some drivers were more diligent about this than others and all the WMT drivers were trained not to interact with the phone, so there are many miles of data that were probably not received because the phone was not turning on and off properly.
- Early in the project, the files received and sent to the weather analysts took a very long time from the time of creation to receipt by the analysts. Some files stayed on phones and were not sent at all. A re-design of DataProbe made sure files did not stay on the phone but instead were uploaded in a timely manner such that the time from creation (five minutes) to UMTRI to the weather analysts totaled only seven to nine minutes on average.
- Some early problems with the DataProbe application such as turning off while the program uploaded files, loss of the GPS signal while driving, loss of phone charge, poor connectivity between the OBD key and DataProbe, and poor opening and closing of DataProbe with occasional application shutdown during operation were clearly identified during our survey of drivers and solved through a thorough re-write of the DataProbe code.
- Early in the project, drivers reported that their phones were trying to pair with other Bluetooth devices in the area. The DataProbe software was re-written to only pair with the specific devices in their vehicles, and once paired, the system would remember the pairings and not try to pair with other local devices.
- Cell phone suppliers can change operating system versions, hardware, and drivers, which may mean a revision of DataProbe may be necessary to work correctly with the new system. When the provider “pushes” a new operating system update to all of its phones, the driver may not have a choice about accepting or declining the update. When this occurs, problems with the phones will surface quickly and project programmers must be ready to address the problems by providing a new version of DataProbe to the entire fleet

of vehicles quickly. This process includes testing new versions before “pushing” them to the smartphones. Designing a remote update program made this whole process manageable.

Communication

Users/Drivers

- Coordination with the weather analysts throughout the country was conducted early in the process to determine data retrieval needs. Data was either pushed or pulled, depending on the receiver’s data collection needs. Towards the end of the project, the flow of data from IMO vehicles became more consistent, so focusing on the accuracy of the data became a higher priority.
- Driver engagement was a key issue for this project since there was no way of knowing for sure when a vehicle was in service. After three months of data collection, it was decided to survey participating drivers to see what issues they were facing with the devices in their vehicles. Survey responses came from 35 of 38 drivers. The survey alerted UMTRI researchers to the severe problems drivers had with vehicles acting erratically. It also helped to sort out what priorities were needed for re-designing the DataProbe program, specifically issues related to slow turn off while the program uploaded files, loss of the GPS signal while driving, loss of phone charge, poor connectivity between the OBD key and DataProbe, and poor opening and closing of DataProbe with occasional closing during operation.
- Driver engagement continued after the survey with a group phone call to all the drivers, reminding them of the importance of the project. MDOT also instituted a weekly report sent to all drivers showing each driver the number of miles driven reported by DataProbe as well as the number of photos taken. Each week drivers were sent another round of graphs showing their mileage and photos for that week and a cumulative number of miles and photos from previous weeks.
- At the same time that MDOT instituted the weekly reports to drivers it also invited lead IMO supervisors from the Lansing, Metro, University, and Southwest regions to the weekly calls. These leaders worked directly with the drivers in their regions who drove IMO vehicles, reminding them of how the technology worked, installing updated versions of DataProbe, answering questions about the technology, listening to and reporting problems certain drivers were having with the technology, and acting as test beds for new versions of DataProbe in development. Communication and training with the drivers has proved to be invaluable towards providing support for IMO data collection, vehicle/system operations, maintenance, and device functionality.
- Besides MDOT supervisors working with drivers, UMTRI established relationships with drivers by letting them know when problems occurred with their systems. UMTRI called, emailed, and setup meetings helping to mitigate operating issues. This process supported the work of the MDOT IMO supervisors and emphasized the important work the drivers provided to the project.
- MDOT instituted weekly meetings to discuss major issues and review/report progress. The MDOT weekly graphs sent to drivers that displayed miles driven and photos taken,

were generated from UMTRI's vehicle activity report that was emailed to each team member (MDOT IMO managers and supervisors, UMTRI researchers, weather analysts, FHWA, supporting consultants, and technology providers). This report included quality checking analyses of certain technologies such as Surface Patrol data. Team members also received a detailed report that served as the basis for each week's discussion of the major issues facing the team for the coming week. Each issue was ranked in order of importance and included the problem description, the status, the risk, the mitigation plan, the expected completion date, the project goals affected, and the number of vehicles affected. As the project matured and stabilized, the focus moved solely from implementation to a combination of implementation and data quality checking.

Continuous Improvements

The nature of the IMO 2.0 project is that something can always be improved. Using consumer hardware and software in potential harsh environments or for purposes it was not originally designed to support is fraught with possible problems from hardware malfunctions to software challenges that can only be overcome through continual testing, updating, and collaboration and coordination with users/drivers. Though the use of the DataProbe program within an Android-based smartphone showed itself flexible and capable of many very complex processes, work continues to improve both the hardware and software. The following are some examples of the continuous IMO project improvements:

- Re-designing the web portal in order to use it to provide a true count of the vehicles that are in service at any given time. The inability to know if data was received from the DataProbe program proved to be a challenge faced throughout the course of the project. The portal has the potential to provide at least a point in time assessment (a snapshot) of what data should be received, though it must be accurate.
- The portal re-design might also include a visualization of photos taken by the phones or from the portal. This would provide the portal user with a near instant view of what type of weather conditions the driver is facing, allowing the portal user to take more or fewer photos depending on the situation.
- The portal re-design may also offer an opportunity to include a map that plots an individual or all vehicles in service at a particular point in time.
- Because the WMT drivers do not have time to take photos due to the multiple functions they must manage inside the cabin, a re-design of the DataProbe WMT version to take photos automatically every 10 or 15 minutes while the vehicle is moving above a certain speed is being considered. Given the assumption that a WMT would not be on the road unless there is reason for it to be there, meaning there is bad weather and/or snow to be plowed, this would be a valued tool for this application.
- Photos taken by DataProbe can be interrogated and coded to provide quick support for weather analysis. Photos move through the DataProbe system faster than data files, and their geo coding also provide a micro level view of a geographic point where weather may be changing quickly. A photo interrogation program would provide instant coding

of the weather or road conditions in the photo and transmit the coding and the photo to weather analysts or traffic management specialists.

- DataProbe will be programmed to always open and close properly during vehicle startup and shut down. This issue has been an intermittent problem throughout the project, depending on what version of DataProbe was in use. This can be seen as part of the component “hardening” process where a component and its software are designed never to fail over the life of the vehicle. In the auto industry, there are specific development and testing processes used to reach this goal, but even these sophisticated processes cannot anticipate all the potential situations a vehicle or component might face during its lifespan. The DataProbe program seems to be in its infancy in terms of understanding how it needs to react in all situations. The goal is to identify each of the situations and adjust the software to respond appropriately.

Conclusions

Despite some substantial obstacles during the implementation of the newest version of the DataProbe data acquisition system, the project was able to meet most of its expectations including instrumenting a 60 vehicle fleet (despite dropping 20 vehicles and instrumenting 20 new vehicles in the middle of the project); collecting all available data and photos from the sensors on each vehicle (including 196,204 valid data files that translate into nearly 400,000 miles traveled, as well as 44,594 photos); sending accurate and timely data to six weather analyst groups throughout the U.S.; and designing a web portal to track vehicles in service, take photos remotely, and send text messages to drivers as needed.

However, there are numerous challenges and improvements needed to make the DataProbe system more robust and reliable. There continues to be breaks in the links between DataProbe and the OBD key and the Surface Patrol devices, and the DataProbe program continues to be erratic during startup and shutdown, including times when the phone has lost its charge due to non-use. Relations with key CAN data support engineers at the auto companies will continue to be a necessary requirement for future DataProbe development as new vehicle models enter the fleet. The web portal also needs a re-design to better connect to vehicles in service and a viewer of photos taken remotely will be of value. Finally, the \$2,200 monthly phone charge for a 60 vehicle fleet may prove more costly than expected for some public entities.

But the DataProbe data acquisition platform continues to evolve and improve over time. It continues to offer micro level weather observations in a timely manner that are difficult to duplicate. The challenge will be to devise offerings that make good use of the data collected by working more closely with weather analysts to discuss the possibilities of leveraging the DataProbe program for better micro level weather reporting, possibly using coded photos taken at regular intervals to support this type of weather reporting.

Recommendations

The world of micro level weather data is upon us. Weather analysts are challenged to meet the requirements of micro level weather reporting. The old weather models based on the broad swath

of radar cannot compete with micro level weather reporting. As phones become enabled to gather and report micro level weather data, the crowd sourcing model can provide dramatic effects. Imagine one's phone receiving a notification that the road two miles ahead is experiencing icy conditions, and that the notification is correct down to the mile marker on the highway. One does not usually need a notification to see a large storm ahead on the highway, but how about a notification of flooding or hydroplaning conditions three miles ahead and again it is correct down to a mile marker? These are crowd sourcing notifications but they are not based on people sending messages but vehicles sending messages based on how the vehicle responds and what a vehicle sees in the environment. They could be based on vehicle data, sensor data, or video data, but the messages are coming.

Micro level reporting is already used by the major personal navigation devices and phone applications for reporting upcoming traffic on a route entered by the driver. These devices use data (positioning, speed) collected by phones in the area (anonymously), along with predictive models for traffic in the area and construction updates, and "fuse" them into traffic notices. Our recent research into this capability showed the challenges all the major providers face in reporting dynamically changing processes like traffic. The IMO 2.0 project shows that reporting weather has many of the same challenges.

One cannot review a project such as the IMO project without thinking that the technology used is in its infancy. One can make a similar statement about other connected vehicle projects. These technologies are not ready to automatically steer or brake a vehicle. One may also question whether the safety warnings they are capable of providing are accurate enough for daily driving without many false positives. But what these projects show is that the technology is moving in the right direction. For the IMO project, any move to the most ubiquitous technology such as the smartphone is a move in the right direction. It is just a matter of working out the kinks in the system (and there are a number of them) before the data from the vehicles that drove on the road ahead of your vehicle only a short time before provide important information for your experience on the same road.

But can a centralized asset management system, such as those found in state departments of transportation play well with a crowd sourcing model based on consumer vehicle input? For example, expanding DataProbe to the entire MDOT fleet of 2,500 vehicles is a very small number compared to the number of miles or roads in the state. As we know from experience, not all these vehicles are on the road at the same time. So, finding more nodes to support the large transportation infrastructure may need to include a consumer component that provides useful information to the consumer in exchange for access to some of its vehicle data that is gathered via smartphone. It is a complicated model when compared to the institutional model used for the IMO project, but there needs to be more nodes on the roads in order to truly have thorough micro level weather reporting. The other option is the MDOT model (using fleet owned vehicles) that can only provide micro level weather reporting in areas where their vehicles are travelling at that moment. Again, it is about moving in the right direction. If the MDOT fleet shows its value in micro level weather reporting, can the consumer piece be far behind once consumers see the value?

Of course, moving from a 60 vehicle fleet to a 2500 vehicle fleet (or even one that is heavily focused in only one area in order to create more nodes/crowd sourcing on the roads) is a costly endeavor. The UM server costs would also increase dramatically, as would communication backhaul and the programming of the CAN codes from all the vehicles.

There may be a breakthrough on the manufacturer CAN code conundrum, where automakers do not focus on the CAN codes for older vehicles as they focus on their new models. One manufacturer is creating a collaborative Web-based environment where CAN codes are available after signing off on some liability issues. The key is that the collaborative environment is based on a group of users who discuss lessons learned and share expertise on these very complicated codes. If other manufacturers eventually sign on, then maybe the manufacturer CAN code challenge will be alleviated. Again, this is a program in its infancy, but heading in the right direction based on a collaborative environment.

Analyzing photos is also a potential area of expansion, as mentioned earlier in this report. In this scenario, codes for weather or road conditions are created based on what is in the photo taken by the smartphone and reported to the weather analysts to support micro level weather information. The geo-coding on a photo will place the photo within meters on a road. It may be able to do it faster than a data file that needs to be collected and analyzed along with other supporting data.

Also, the current OBD key process is cumbersome and slow. Researchers are investigating a new technology that may be an OBD key replacement that is more powerful and reliable.

Of course there are the necessary improvements that need to be made to the current system to make it more useful including:

- making the web portal less static and more active by showing photos taken remotely or photos noted as important by the drivers who took the photos, as well as maps showing the location of the vehicles in service
- designing the web portal application to run on a smartphone, allowing administration easy access to the fleet
- designing a database system to more easily manage and analyze the data collected
- designing DataProbe to have a 24/7 program that notifies central administration if the battery is about to die or if it is not receiving data from a source it is expecting
 - This system represents an enterprise wide central management model that monitors all the nodes of the enterprise
 - These are more vehicle to infrastructure and infrastructure to vehicle models that affect mobility rather than the vehicle to vehicle connectivity that focuses on safety and emissions

All of these technologies are part of an advanced IMO technology path that can be part of the process that will make micro level weather reporting a reality. The technology is moving in the right direction. It is just a matter of time before it reaches a freeway near you.

Appendix

1. DataProbe File Naming Conventions

IMPORTANT FILE NAMING CONVENTIONS		
CHANGES SINCE THE LAST VERSION OF THE FILE NAMING CONVENTIONS LIST ARE HIGHLIGHTED IN YELLOW		
DATA FILES		
1) All non-WMTs will eventually use the full VIN as the first digits of the file name. (As of 3/6/14, 34 of 40 non-WMTs use this convention) 2) All WMTs will eventually use the full 10 digit phone number as the first digits of the file name. (As of 3/6/14, 19 of 20 WMTs use this convention) 3) 10001 at the beginning of the file name means the phone cannot link to CAN data nor the phone number. It can be treated as missing data. 4) Winter Maintenance Trucks (WMTs) (snowplows) do not send CAN data (They are identified by the "w" at the end of their DataProbe version number) 5) Some non-WMTs also do not send CAN data when there is a connectivity problem between the phone and the CAN data device (OBD Key). These file names will begin with the 10 digit phone number instead of the full VIN		
Data Files (CSV) with CAN data	DataProbe Version (v3.7.1 or higher)	
	First 17 digits	Full 17 digit VIN
	Digit 18 = Underscore	
	Digits 19-26	month/day/year (12262013)
	Digit 27=Underscore	
	Digits 28-34	DataProbe Version (v3.7.1d)
Data Files with no CAN data (CSV)	DataProbe Version (v3.7.1 or higher)	
	First 10 digits	Full 10 digit phone number
	Digit 11 = Underscore	
	Digits 12-19	month/day/year (12262013)
	Digit 20=Underscore	
	Digits 21-27	DataProbe Version (v3.6.7 to v3.7.0)
PHOTO FILES		
1) All non-WMTs will eventually use the full VIN as the first digits of the file name. (As of 3/6/14, 34 of 40 non-WMTs use this convention) 2) All WMTs will eventually use the full 10 digit phone number as the first digits of the file name. (As of 3/6/14, 19 of 20 WMTs use this convention) 3) 10001 at the beginning of the file name means the phone cannot link to CAN data nor the phone number. It can be treated as missing data.		
Photo files from vehicles also sending CAN data	DataProbe Version (v3.7.1 or higher)	
	First 17 digits	Full 17 digit VIN
	Digit 18 = Underscore	
	Digits 19-26	month/day/year (12262013)
	Digit 27=Underscore	
	Digits 28-34	DataProbe Version (v3.7.1d)
Photo files from vehicles not connecting or receiving CAN data via	DataProbe Version (v3.7.1 or higher)	
	First 10 digits	Full 10 digit phone number
	Digit 11 = Underscore	
	Digits 12-19	month/day/year (12262013)
	Digit 20=Underscore	
	Digits 21-27	DataProbe Version (v3.6.7 to v3.7.0)
Photo files for Winter Maintenance Trucks (WMTs)	DataProbe Version (v3.7.1 or higher)	
	First 10 digits	Full 10 digit phone number
	Digit 11 = Underscore	
	Digits 12-19	month/day/year (12262013)
	Digit 20=Underscore	
	Digits 21-27	DataProbe Version (v3.6.7 to v3.7.0)

2. DataProbe Data Dictionary

UMTRI/MDOT IMO 2.0 Data Dictionary 3/6/14					
CHANGES SINCE THE LAST VERSION OF THE DATA DICTIONARY ARE HIGHLIGHTED IN YELLOW					
IMO Data Dictionary					
Data Source	Data Field	Description (Number of Vehicles Providing Data)	Data Type	Format	Primary Alias (v3.6.5 or later)
Header Info	Size (Phone Data)	Size of file (60)	Unsigned Integer	Number of bytes	Size:
	File name (Phone Data)	Name of file (60)	String	File name string without extension	File Name:
	VIN (CAN Data)	Vehicle Identification Number (40)	String	Raw VIN	VIN:
	Date (Phone Data)	Date of creation of the file (60)	Date String	Month/Day/Year (2/26/2014)	Date:
	Air Temp (CAN Data)	Air Temperature from CAN (40)	Signed Decimal	Degrees Fahrenheit	Air Temp:
	Barometer (CAN Data)	Barometric Pressure from CAN (40)	Unsigned Decimal	KiloPascal	Barometer:
	Coolant Temperature (CAN Data)	Temperature of Coolant from CAN (40)	Signed Decimal	Degrees Fahrenheit	Coolant Temp:
	Odometer (CAN Data)	Odometer Reading from CAN (40)	Unsigned Decimal	Miles	Odometer:
Phone Data	Current Time	Current local time on the phone (60)	Time String	hh:mm:ss, 24 hour standard	TIME
	Latitude Position	Vehicle latitude coordinate (60)	Signed Decimal	Latitude decimal degrees	LAT
	Longitude Position	Vehicle longitude coordinate (60)	Signed Decimal	Longitude decimal degrees	LONG
	Altitude	Vehicle altitude above sea level (60)	Signed Decimal	Altitude above sea level in feet formatted with two decimal places. Actual value may be within +/- 10-15ft	ALT
	# of GPS satellites visible	Number of currently visible GPS satellites. For use in determining GPS validity (60)	Unsigned Integer	Number of satellites available	SAT
	Compass Heading	Current vehicle bearing derived from GPS (60)	Unsigned Decimal	Heading in decimal degrees	HEAD_G
	GPS Speed Measurement	Current vehicle speed derived from GPS (60)	Unsigned Decimal	Speed in MPH formatted as X.d, XX.d, or XXX.d, fixed resolution at 0.1MPH	SPD-G
OBD CAN Data	Tachometer (RPM)	The vehicles engine speed reported over OBD (40)	Unsigned Integer	Direct RPM value	RPM
	Speedometer	The vehicles road speed reported over OBD (40)	Unsigned Decimal	Speed in MPH formatted as X.d, XX.d, or XXX.d, fixed resolution at 0.1MPH	SPD-C
	Throttle	Current engine throttle percentage (40)	Unsigned Decimal	Percentage of full throttle being applied formatted as X.d, XX.d, or XXX.d, fixed resolution at 0.1%	ACC
	Brakes	Vehicle braking flag (40)	Boolean Flag	0: Inactive, 1: Active	BRK
	Antilock Braking System	Antilock braking system activity flag (40)	Boolean Flag	0: Inactive, 1: Active	ABS
	Electronic Stability Control	Electronic stability control activity flag (40)	Boolean Flag	0: Inactive, 1: Active	ESP
	Traction Control Engine Event	Traction control engine event flag (40)	Boolean Flag	0: Inactive, 1: Active	TCSE
	Traction Control Braking Event	Traction control braking event flag (40)	Boolean Flag	0: Inactive, 1: Active	TCSE
Surface Patrol Data	Surface Temperature	Road surface temperature from Surface Patrol sensors (20)	Signed Decimal	Road surface temperature in °F, X.d, XX.d, or XXX.d, fixed resolution at 0.1°F	STMP
	Dew Point	Computed dew point from Surface Patrol system (20)	Signed Decimal	Dew point temperature in °F, X.d, XX.d, or XXX.d, fixed resolution at 0.1°F	DPNT
	Ambient Air Temperature	Ambient air temperature from Surface Patrol (20)	Signed Decimal	Ambient air temperature in °F, X.d, XX.d, or XXX.d, fixed resolution at 0.1°F	TMP-SP
	Humidity	Air relative humidity measured by Surface Patrol sensors (20)	Unsigned Decimal	Relative humidity percentag formatted as X.d, XX.d, or XXX.d, fixed resolution at 0.1%	HUMD
Accelerometer Data	Accelerometer X-axis	Phone accelerometer X-axis data (60)	Signed Decimal	Signed Decimal acceleration value (m/s ²)	X1, X2, ..., X100
	Accelerometer Y-axis	Phone accelerometer Y-axis data (60)	Signed Decimal	Signed Decimal acceleration value (m/s ²)	Y1, Y2, ..., Y100
	Accelerometer Z-axis	Phone accelerometer Z-axis data (60)	Signed Decimal	Decimal acceleration value (m/s ²)	Z1, Z2, ..., Z100
Checksum	Data Checksum	CRC Checksum for data transfer validation (60)	Unsigned Integer	Integer: CRC-32 bit for non-accelerometer data fields	CRC