Figure 1. Nevada DOT IMO 3 Snow Plow
Notice

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

The U.S. Government is not endorsing any manufacturers, products, or services cited herein and any trade name that may appear in the work has been included only because it is essential to the contents of the work.
**4. Title and Subtitle**
NEVADA INTEGRATED MOBILE OBSERVATIONS 3 PROJECT

**5. Report Date**
January 29, 2018

**7. Author(s)**
Jeffrey LaCombe, Eric Wang, Rodney D. Schilling

**9. Performing Organization Name and Address**
University of Nevada Reno  
Nevada Department of Transportation

**12. Sponsoring Agency Name and Address**
Nevada Department of Transportation  
1263 S. Stewart St.  
Carson City, NV 89712

**16. Abstract**
This phase of the project was undertaken as an effort to leverage what was learned in Phases 1 and 2 to create a more modular architecture with a hybrid communications platform for a sustainable system that incorporates Dedicated Short-Range Communications (DSRC) data telemetry (common to other Connected Vehicles projects). Then to deploy these capabilities in the everyday use of the road weather maintenance personnel at NDOT by working with the National Center of Atmospheric Research (NCAR) to implement, test and accept the Pikalert System for an Enhanced Maintenance Decision Support System (EMDSS) while improving data communication standards and access to the TMC personnel for a more weather responsive traveler information system incorporating selected features of NCAR's Motorist Advisory Warning (MAW) system, both of which ingest mobile road weather data.

**17. Keywords**
DSRC, NCAR, EMDSS, MAW, Pikalert System, data communication standards, weather responsive

**18. Distribution Statement**
No restrictions. This document is available to the public through the Nevada Department of Transportation.
Acknowledgements

The Nevada Department of Transportation would like to express their gratitude to the University of Nevada Reno (UNR), US Department of Transportation Federal Highway Transportation (FHWA), Michigan Department of Transportation (MDOT), Minnesota Department of Transportation (MnDOT), National Center of Atmospheric Research (NCAR), Booz Allen Hamilton, and all the dedicated staff of the Nevada Department of Transportation for their guidance, encouragement and support in carrying out this project.

**US Department of Transportation Federal Highway Transportation:** Paul Pisano, Gabe Guevara, Roemer Alfelor, Ray Murphy.

**Nevada Department of Transportation:** Denise M. Inda, Rodney D. Schilling, James Whalen, Israel Lopez, Mark Aragon, Gary Molnar, Mylinh Lidder, Amber Angel.

**University of Nevada, Reno:** Eric Wang, Jeffrey LaCombe, Pablo Rivera, Joao Paulo Braz, Marek Hajek, William Nagel, Amanda Nelson, Kyle Gorsiski, Cody Zampella, Destiny Phan, Tamzin Atkins.

**National Center of Atmospheric Research:** Gerry Wiener, Amanda Anderson.

**Booz Allen Hamilton:** Brenda Boyce.
## Table of Contents

### Executive Summary ................................................................. 1
- Purpose ...................................................................................... 1
- Stakeholders .............................................................................. 2

### Introduction and Background ........................................................................ 3
- Vision and Goals for IMO at NDOT ......................................................... 4

### Project History and Scope ........................................................................ 5
- Summary of the NIMO Project Phases ............................................................. 5
  - Phase 1 of NIMO Project .............................................................................. 6
  - Phase 2 of NIMO Project .............................................................................. 7
  - Phase 3 of NIMO Project .............................................................................. 8

### Current (NIMO Phase 3) Implementation .................................................. 9
- System Overview .............................................................................. 9
  - NIMO System-Overview ......................................................................... 9
  - In Vehicle Components-Overview ............................................................... 12
  - Road Side Components-Overview ............................................................... 14
- NIMO DSRC FCC & FAA Site Registration Procedure – Overview Error! Bookmark not defined.
- Back-End Components-Overview ................................................................ 17

### NIMO System Hardware .................................................................. 18
- On Board Unit Hardware (OBU) .............................................................. 18
  - Technologic CPU .................................................................................. 19
  - Arada DSRC OBU .................................................................................... 20
  - Cradlepoint Cellular Modem ................................................................. 21
  - Comet Weather Sensor .......................................................................... 21
  - Adam Data Acquisition & Counter ......................................................... 22
  - RoadWatch Road Surface Temperature Sensor ........................................ 22
- Sensor Selection and Placement ............................................................ 23
  - On Board Unit (OBU) and Sensor Data Connections ................................ 23
  - Plow On Board Hardware and Sensor Placement ...................................... 24
# Table of Contents

- Roadside Unit Hardware ...................................................................................................................... 26
- Other Hardware Considerations ......................................................................................................... 26
- NIMO System Software .......................................................................................................................... 26
  - OBU Software ....................................................................................................................................... 26
  - RSU Software ....................................................................................................................................... 27
  - Backend IMO Server ............................................................................................................................ 27
  - NDEX ..................................................................................................................................................... 27
- NIMO System Networking ........................................................................................................................... 27
- NIMO System Costs .................................................................................................................................... 28
  - Vehicle Equipment Costs ..................................................................................................................... 28
  - RSU Equipment Costs .......................................................................................................................... 28
  - Backend Equipment Costs ................................................................................................................... 29
  - Recurring Costs (data plans, etc.) ....................................................................................................... 29
  - Development Costs (University and NDOT) ....................................................................................... 29
- NIMO System Performance Evaluation ............................................................................................................ 30
  - Functionality ................................................................................................................................................. 30
    - Overview of Functionality ..................................................................................................................... 30
    - Maintenance and Support .................................................................................................................... 30
    - Retrograde Functionality Enhancements ............................................................................................ 30
  - Data ............................................................................................................................................................... 31
    - DSRC Data Telemetry .......................................................................................................................... 31
    - Cellular Data Telemetry ........................................................................................................................ 31
    - Multimodal Data Telemetry Features .................................................................................................... 31
    - Weather Data ........................................................................................................................................ 31
    - Vehicle and Spreader Data .................................................................................................................. 31
  - Impact ........................................................................................................................................................... 32
- Applications ............................................................................................................................................... 33
  - NDEX ................................................................................................................................................................... 33
  - Weather Forecasts .............................................................................................................................................. 33
  - Automated Vehicle Location - AVL .................................................................................................................... 34
  - Enhanced Maintenance Decision Support System - EMDSS ......................................................................... 34
  - Material Management System - MMS ............................................................................................................... 36
  - Secondary (Peripheral) Applications ................................................................................................................. 37
    - Traction (Friction) Measurement (research phase) ................................................................................... 37
    - Salinity (Melting Point) Measurement (research phase) ........................................................................... 38
- Lessons Learned ............................................................................................................................................. 39
## Table of Contents

- Lessons Learned - Organizational
  - Organizational Champion and Buy-In
  - Personnel & Expertise
  - Fleet Selection and Management

- Lessons Learned - Hardware
  - General NIMO System Overview
    - COTS vs. Custom
    - Normal wear and tear
    - Hardware "stability" over installation lifetime (evolving technology)
  - Vehicle Installation
  - Smart Spreader Controllers
  - Multimodal Telemetry
  - Proprietary Technologies & Custom Electronics
  - Vendor Selection and Product Support
  - Assessment of System Functionality
  - DSRC Hardware
  - Hardware Package Size

- Lessons Learned - Software and Data
  - CAN bus Data Sources
  - Software Version Control
  - Remote Software Updates
  - Computing Platforms
  - Networking Infrastructure
  - IPv6
  - DSRC Message Formats
  - Cellular Plans expensive and coverage limited by provider
  - Use of Data Standards

- Lessons Learned - Applications

- Lessons Learned - Other
  - Costs
    - Hardware Costs
    - Software Costs
    - Other Costs
    - Licensing
# Sustainability

# Scalability

# Data Sharing

## Moving Forward

- NDOT Commitment to Weather Savvy Roads (WSR) Road Weather Management Program
- Develop an NDOT Connected Vehicle/Autonomous Vehicle Implementation Plan
- Develop and incorporate IMO into a Statewide NDOT TSMO Plan
- Develop a Statewide TSMO Plan
- Identify Performance Measures for TSMO
- Include TSMO Components in Long-Range Planning
- Develop an Outreach Approach for TSMO
- Develop a TSMO Review for all new Projects
- Create a Repository for NDOT TSMO Information
- TSMO Staff Development
- Review TSMO Program and Define Next Steps
- NDOT TSMO Asset Management & Performance Measures Business Plan

## Appendix A. Acronyms

## Appendix B. NIMO Phase 3 Configuration Manual v2.0

### Outline

- Preconfiguration
- Configuration Procedures
- Vehicle IP Configuration Data
- Configuration of DSRC Geofencing
- Accessing IMO Data
- Login and Password Summary for System Configurations
List of Tables

Table 1. Vehicle Equipment Cost Breakdown ..................................................................................28
Table 2. Vehicle IP Configuration Data IMO#1 ........................................................................................... 89
Table 3. Vehicle IP Configuration Data IMO#2 ........................................................................................... 90
Table 4. Vehicle IP Configuration Data IMO#3 ........................................................................................... 91
Table 5. Vehicle IP Configuration Data IMO#4 ........................................................................................... 92
Table 6. Vehicle IP Configuration Data IMO#5 ........................................................................................... 93
Table 7. Vehicle IP Configuration Data IMO#6 ........................................................................................... 94
Table 8. Vehicle IP Configuration Data IMO#7 ........................................................................................... 95
Table 9. Vehicle IP Configuration Data IMO#8 ........................................................................................... 96
Table 10. Vehicle IP Configuration Data IMO#9 ......................................................................................... 97
Table 11. Vehicle IP Configuration Data IMO#10 ....................................................................................... 98
Table 12. Vehicle IP Configuration Data IMO#11 ....................................................................................... 99
Table 13. Vehicle IP Configuration Data IMO#12 ..................................................................................... 100
Table 14. Vehicle IP Configuration Data IMO#13 ..................................................................................... 101
Table 15. Vehicle IP Configuration Data IMO#14 ..................................................................................... 102
Table 16. Vehicle IP Configuration Data IMO#15 ..................................................................................... 103
Table 17. Vehicle IP Configuration Data IMO#16 ..................................................................................... 104

List of Figures

Figure 1. Nevada DOT IMO 3 Snow Plow ................................................................................... Cover Page
Figure 2. Moving vehicles connected with other vehicles and with stationary infrastructure ..................... 10
Figure 3. NIMO Pilot Test Corridor .............................................................................................................. 11
Figure 4. Vehicle Data Transmission .......................................................................................................... 12
Figure 5. Schematic layout of the system components located in each vehicle............................................. 13
Figure 6. In-vehicle hardware backplane .................................................................................................... 14
Figure 7. Installation of Road Side Unit (RSU) ........................................................................................... 15
Figure 8. Schematic depicting the NIMO Concept of Operations and the data flow paths of the NIMO program .................................................................................................................................................. 18
Figure 9. Schematic of electrical and data connections between NIMO components on the system backplane ............................................................................................................................................. 19
Figure 10. Technologic Linux CPU .............................................................................................................. 20
Figure 11. Arada Locomate Classic On-Board Unit (OBU) ........................................................................... 21
Figure 12. Cradlepoint cellular modem & GPS ........................................................................................... 21
Figure 13. Comet Ethernet weather sensor ................................................................................................ 22
Figure 14. ADAM Ethernet enabled I/O module ........................................................................................... 22
Figure 15. RoadWatch SS road surface temperature sensor ........................................................................... 23
Figure 16. NIMO On Board Units (OBU) and Sensor Data Connections ................................................... 24
Figure 17. NIMO Plow On Board Hardware and Sensor Placement .......................................................... 25
Figure 18. Arada Locomate Road Side Unit (RSU) DSRC radio and a typical site installation ................. 26
Figure 19. New high-impact technologies require addressing all three “legs of the stool” ...................... 32
Figure 20. NV-ROADS 511 Road Weather Sensor locations ................................................................. 34
Figure 21. Example user interface screens for the Pikalert EMDSS application for Nevada ............... 35
Figure 22. Heat map of where a patrolling snow plow deployed sand/salt mixture ............................ 36
Figure 23. NIMO heat map .................................................................................................................... 37
Figure 24. Raman spectra of water ........................................................................................................ 38
Figure 25. NIMO OBU dimensions ...................................................................................................... 42
Figure 26. Comet Physical Mounting .................................................................................................... 63
Figure 27. Comet Physical Connections ............................................................................................... 64
Figure 28. Comet Factory-Default Process .......................................................................................... 64
Figure 29. Technologic Internal Ribbon Cable Connections ............................................................... 73
Figure 30. Technologic motherboard .................................................................................................. 75
Executive Summary

The U.S. Department of Transportation (USDOT) Federal Highway Administration (FHWA) and Research and Innovative Technology Administration (RITA) have been jointly working to promote safety, mobility and productivity on the nation’s surface transportation system by advancing road weather research.

In this project, the USDOT FHWA Road Weather Management Program (RWMP) works to demonstrate how weather, road condition, and related vehicle data may be collected, transmitted, processed, and used for decision making as part of the broader Intelligent Transportation System (ITS) Connected Vehicles (CV) program\(^1\).

Ultimately, decision makers will have the benefit of decision support tools that have access to data provided by millions of vehicles through the IMO, Connected Vehicle\(^2\), and other related programs. The use of existing fleet infrastructures and wireless communication technology allows this prototype project to help determine the procedures and processes required to integrate weather, road condition, and vehicle status data messages into existing programs.

This project also expands the capabilities of the National Center of Atmospheric Research (NCAR) Vehicle Data Translator (VDT), which incorporates vehicle based measurements of the road and surrounding atmosphere with other, more traditional weather data sources, to create road and atmospheric hazard products for variety of user, including NDOT. Additionally, this project integrates mobile weather and road condition data into the National Oceanic and Atmospheric Administration’s MADIS System and the Weather Research Data Environment System. These collect weather and road condition data from mobile and stationary sensors across the U.S. and parts of Canada, and then make the data available over the Internet with text and graphics based retrieval and visualization systems.

Purpose

The purpose of this project was to advance the effective operations and management of transportation systems using Integrated Mobile Observations (IMO) methods and tools in Nevada.

---

\(^1\) The CV Program is a multimodal initiative to enable wireless communications among vehicles, the infrastructure, and passengers’ personal communications devices. It will enhance Americans’ safety, mobility and quality of life, while helping to reduce the environmental impact of surface transportation.
The Nevada Department of Transportation (NDOT), in coloration with the University of Nevada, Reno (UNR), NCAR, Michigan Department of Transportation (MDOT), and Minnesota Department of Transportation (MnDOT), participated with the FHWA in this Phase 3 pilot project.

Stakeholders

**NDOT:**

The Nevada Department of Transportation, is the key stakeholder, and is responsible for representing the interest of the users of Nevada’s roads and highways. It is envisioned that the products of this project will work to address multiple adverse weather and maintenance needs within the state. The key individuals at NDOT are the Maintenance and Asset Management personnel, District Maintenance personnel, Equipment personnel, Traffic Operations personnel, District Regional Operation Center personnel. Rodney D. Schilling served as the NDOT Project Manager and represented others within NDOT for the purposes of interfacing with the IMO project.

**NCAR:**

The National Center of Atmospheric Research (NCAR) worked with the team over the development of this project to conduct research, develop, and deploy the Pikalert system to incorporate connected vehicle observations and other advanced road weather forecasting technologies. This operational prototype system is currently running for Nevada and has promise in helping NDOT better adapt to changing hazardous road weather conditions. The key individuals at NCAR are Gerry Wiener and Amanda Anderson.

**FHWA:**

The Federal Highway Administration (FHWA) is a stakeholder in that the methods and tools developed as part of this effort will continue to the broader knowledge base of the Intelligent Transportation System (ITS) Connected Vehicle (CV) program. As a stakeholder, FHWA furthers its aim by developing intelligent vehicle systems and capabilities that have potential for further development beyond the scope of this project, and working towards broader implementation in Nevada and the broader nation. The key individual stakeholder at FHWA is Gabriel Guevara, who will represent others within FHWA (and subcontractors) for the purposes of interfacing with the IMO project.
Introduction and Background

The U.S. Department of Transportation's Intelligent Transportation Systems Joint Program Office (USDOT-ITS) has been working to promote safety, mobility and productivity on the nation's surface transportation system through Connected Vehicles. One facet of this effort seeks to achieve these goals by advancing road weather research. The USDOT FHWA's Road Weather Management Program (RWMP) has coordinated and sponsored multiple states' Department of Transportation to develop capabilities and demonstrate how weather, road condition, and related vehicle data may be collected, transmitted, processed, and used for winter road maintenance decision making. This has been done as part of the Integrated Mobile Observations (IMO) project, under the Connected Vehicles (CV) initiative. The vision underlying this effort is that eventually, road maintenance decision-makers will have the benefit of decision support tools that leverage road condition and weather-related data gathered by (potentially) millions of vehicles while driving. Effective use of this large volume of data will be facilitated through the lessons learned by the IMO and other Connected Vehicle efforts.

The Integrated Mobile Observations (IMO) project is a collaboration led by the Federal Highway Administration, and involves participants from three departments of transportation (Nevada, Minnesota, and Michigan) as well as the National Center for Atmospheric Research (NCAR). In Nevada, the IMO project is denoted NIMO (Nevada-IMO).

The Nevada Department of Transportation (NDOT) became engaged in the IMO project (Phase 1) in early 2011 after spending several years prior to this working on developing radio-based Automatic Vehicle Location (AVL) capabilities that would function in the rural areas of Nevada that did not have cellular voice or data coverage (a significant challenge in Nevada). For the Nevada Department of Transportation, a key over-arching motivation was to lay the foundation for a future implementation of a Maintenance Decision Support System (MDSS) and a Material Management System (MMS), which would leverage mobile data—i.e., incorporate weather and road information not only from fixed locations (such as RWIS), but also from mobile measurements obtained from NDOT vehicles. Mobile data was envisioned to help NDOT better characterize the significant geographical variations in weather and road conditions in Nevada and facilitate the optimization and effectiveness of deploying resources (staffing and usage of materials) during winter weather events. An MDSS system consists of both software and hardware and is typically provided through a commercial vendor. A typical MDSS system uses (at the least) a Global Positioning System (GPS) based Automated Vehicle Location (AVL). More sophisticated implementations of AVL (the focus here) add a Mobile Data Computer (MDC) that monitors various mobile sensors. All this data (location and sensors) must be telemetered to the back-end where users receive and interpret the information, and act upon it. At the system's core is software which uses weather forecasting models to predict storm events and, when combined with a road weather model, can forecast changing roadway conditions on routes that are selected by the DOT. The MDSS software makes recommendations of treatments to supervisors or snow plow drivers.

Today, the most common implementations use cellular data telemetry to feed data from the field back to the MDSS and downstream users of this information. Even in 2017, many of the roads maintained by NDOT do not have cellular coverage. Communication with maintenance personnel (snow plow drivers, for example) is generally conducted using a state-wide 800 MHz Enhanced Digital Access
Communication System (EDACS) truncated radio system that is shared with other private, state, and municipal users. This system has become antiquated in 2017 and is planned for replacement, but at present is still in use across NV where no other communications modes exist. Phase 1 of the NIMO project identified the key limitation of this EDACS system—namely that it is mainly intended for voice use, and its data capability is limited to an extremely low bandwidth (i.e., use for road weather purposes is limited to a few hundred bytes of data every 5 minutes (per vehicle) to avoid interfering with higher priority users of the system).

Phase 2 of the NIMO project built on the experiences gained through Phase 1, but with exploration of the use of new sensors, in-vehicle data processing capability, and adding cellular data telemetry capability. The architectural differences of the system used in NIMO Phase 2 gave our system enhanced capabilities with richness and frequency of transmitted observations, with remote configuration of systems out in the field, and with remote debugging and software updates. However, with these expanded capabilities came complexity and a greater demand on project personnel.

Phase 3 was undertaken as an effort to leverage what was learned in Phases 1 and 2 to create a sustainable system for the NIMO project that leveraged Dedicated Short-Range Communications\(^2,3\) (DSRC) data telemetry (common to other Connected Vehicles projects), and employed these capabilities in every-day use by road maintenance personnel at NDOT.

**Vision and Goals for IMO at NDOT**

**Vision:** The vision of the Nevada Integrated Mobile Observations program (NIMO), at the Nevada Department of Transportation, is to leverage information and technology resources to improve the safety, reliability, mobility, and productivity of Nevada’s road system during adverse weather events.

To examine the goals that follow from this vision, it should be noted that the nationally-broad IMO project can be looked at from different points of view for each of the participating organizations. The FHWA’s goals are tightly connected to the visions of the Connected Vehicle and Road Weather Management Programs, whereas the individual participating states each embarked on the project starting with different capabilities, resources, expertise, and needs. For the Nevada Department of Transportation, a key overarching motivation was to lay the foundation for a future implementation of a Maintenance Decision Support System (MDSS) and a Material Management System (MMS), which would help NDOT optimize the effectiveness of deploying resources (staffing and usage of materials) during winter weather events. To achieve the IMO vision in Nevada, the following goals have been set.

\(^2\) DSRC is summarized well at [https://www.its.dot.gov/factsheets/dsrc_factsheet.htm](https://www.its.dot.gov/factsheets/dsrc_factsheet.htm)

\(^3\) A DSRC fact sheet can be found at: [https://www.its.dot.gov/factsheets/pdf/JPO-034_DSRC.pdf](https://www.its.dot.gov/factsheets/pdf/JPO-034_DSRC.pdf)
Goals:

1. NDOT will develop and use a Maintenance Decision Support System (MDSS) that facilitates the planning and management of maintenance activities associated with winter weather events. The MDSS will leverage real-time information from NDOT sources (RWIS and mobile data) as well as more general information from weather data and forecast providers to generate quality road weather forecasts and make reliable road treatment recommendations.

2. NDOT will implement a Materials Management System (MMS) that uses sensing and data management technologies to accurately track the use of materials (such as sand and deicer materials) and to use this information both during weather events (as an input to the MDSS) as well as afterwards to assess performance.

3. NDOT will use the IMO resources for Performance Measurement, where the data sources will facilitate the quantitative and qualitative assessment of safety, reliability, mobility, and productivity of Nevada's roadways.

Project History and Scope

Summary of the IMO Project Phases

The Nevada Integrated Mobile Observations (NIMO) project began in 2011 with Phase 1. The core aim of the NIMO program at that time was to demonstrate data telemetry capability from vehicles across all areas of NV, using the Interstate-80 corridor as the test bed. This 400-mile corridor extends east-west across NV from CA to UT, and consists predominantly of rural areas with no cellular coverage. Because data telemetry and mobile observations are central to the IMO project, the large rural areas of NV necessitated use of non-cellular data connectivity. This was achieved using an EDACS radio network operated by the State of Nevada. Nevada's EDACS network is shared by a multitude of agencies (Highway Patrol, Department of Public Safety, Division of Forestry, Nevada Energy, Washoe County, Emergency announcements, etc.). Due to its importance to communications across the state, access to the EDACS system is carefully controlled and bandwidth-limited. With NIMO, it was necessary to place a low cap on bandwidth usage. This system was developed and installed into 20 NDOT vehicles. The vehicles participating in the project transmitted approximately 500 bytes (0.5 Kb) every 5 minutes. Since the outset of the NIMO project, cellular service has become more available (particularly in the areas surrounding more urban areas such as Reno, as well as smaller rural towns. This has changed how NDOT is approaching the data telemetry in the NIMO program. Presently, NDOT is working to implement a new statewide communications scheme that may vary from region to region (even if only during a transition phase), requiring an MDSS system to handle multiple modes of telemetry depending on the vehicle’s location.

Starting in 2013, Phase 2 of the NIMO system was developed and installed into approximately 25 NDOT vehicles. Phase 2 was almost a complete redesign of the in-vehicle hardware used in Phase 1. Phase 2 focused on vehicles within the Reno-Carson vicinity with cellular service. Most of the hardware and software for the system was replaced with different devices and applications to the use cellular communications instead of EDACS. The new choice of communications system brought with it orders-of-magnitude better bandwidth availability, and a bi-directional data link that provided the ability to remotely conduct software updates. In addition, because the cellular link is layered with TCP/IP (Transmission Control Protocol/Internet Protocol) and HTTP (Hypertext Transfer Protocol), little custom code was needed to provide retransmission of lost data, error-checking, and resumption of interrupted transmissions. On the other hand, the cellular network has much more limited coverage in the state of...
Nevada compared with the ubiquitous EDACS network, mostly being available in corridors found along major highways and in urban areas.

In early 2014, Phase 3, NDOT developed a prototype version of the NIMO system, which combined multiple modes of radio connectivity to maximize the strengths provided by each (EDACS and cellular) while minimizing their weaknesses. The network modes that were tested within this prototype include cellular, EDACS, and Wi-Fi. This multi-modal telemetry system was conceived to also work with Vehicular Satellite (using the Iridium network) or other radio-based systems such as P25\(^4\) or DSRC (Dedicated Short-Range Communications), which would eventually become the focus of Phase 3 of the NIMO project. The multi-modal approach to data telemetry provides coverage, even in rural areas, while fully utilizing the high-bandwidth cellular network found along interstate highway corridors and DSRC within urban areas such as Reno and Carson City.

**Phase 1 of NIMO Project**

Interstate 80 is a major east-west interstate corridor through Nevada, and is a major economic freight and traveler corridor which can better service the public through improved and coordinated maintenance and traveler information services. The I-80 Coalition, including California, Nevada, Utah, and Wyoming, exists to 1) establish institutional structure for coordinating operations on I-80 in the western states; 2) aggregate weather conditions information from multiple sources; 3) identify traffic data collection capabilities and share information with other agencies; 4) establish existing capabilities and near-term enhancements to identify specific continuity issues; and 5) research innovative practices from other areas of the country facing similar challenges.

The Nevada Department of Transportation (NDOT), in collaboration with the University of Nevada, Reno (UNR), the National Center for Atmospheric Research (NCAR), and the FHWA Road Weather Management Program (RWMP) demonstrated the IMO project with the main project goal of developing a prototype IMO system that within Nevada, addressed two of NDOT’s critical needs:

1. **Improve system-wide performance monitoring and measurement methods and tools.** The Nevada IMO (NIMO) project used on-vehicle instrumentation to dynamically gather near-real-time localized weather, road condition, material usage, and vehicle-related data that would be used to gauge road conditions and better inform safety processes (such as chain controls) by providing this data to supervisors and operators through mediums such as *Clarus*.

2. **Improve Freeway/Highway Maintenance Operations and Equipment Maintenance.** Near-real-time weather, vehicle, and road condition data would be used by NDOT personnel to improve

---

\(^4\) P25 is a suite of standards for digital mobile radio communications designed for use by public safety organizations in North America. P25 radios are a direct replacement for analog UHF radios but add the ability to transfer data as well as voice, allowing for a more natural implementation of encryption or messaging.
highway maintenance operations both in terms of increased level of service and cost savings through applications such as Maintenance Decision Support System (MDSS) and a Maintenance Management System (MMS).

In Phase 1 of the NIMO project, 20 vehicles were instrumented, including 11 Peterbilt snowplow trucks (model year 2007 or 2009) and 9 Ford light duty vehicles (LDV, model years 2001-2008 F250 and F450) at sites along the I-80 corridor across northern Nevada. During winter, these trucks operate regularly along the I-80 corridor focusing around Reno and Elko, as well as adjoining roads within approximately 70-90 miles from I-80. In summer, NDOT uses the same plow trucks, converted for summer maintenance activities, as well as use the light duty vehicles in the Elko and Reno areas.

Phase 1 focused on the objectives of researching and developing fundamental capabilities in Nevada for Automated Vehicle Location (AVL) and weather data acquisition and telemetry. Key challenges surrounded the need to develop all the hardware and software and to telemeter data in near-real-time using the statewide Enhanced Digital Access Communication System (EDACS) radio network that permits NDOT to communicate with snow plows and maintenance vehicles across the state. Phase 1 involved 1) developing and prototyping the hardware and software installed in the vehicles, 2) Installing systems into 20 vehicles, 3) Studying different sensor hardware options in the field, 3) Exploring the use of the Open Sky radio network as a potential future replacement for the EDACS radio network, 4) exploring the use of vehicle CAN bus\(^5\) data, and 5) Telemetering data from the field vehicles through the NDOT system and through the internet to downstream users at FHWA and NCAR (for testing purposes).

**Phase 2 of NIMO Project**

Phase 2 of the NIMO project continued the collaboration with two main project goals, each with their own set of narrower objectives. These goals were 1) refining and expanding data gathering and processing systems, and 2) using the resulting data products in applications identified as most relevant in Nevada. More detailed objectives (proposed at the outset of the project) are listed below:

1. Expand the size of the instrumented vehicle fleet to provide more data to end-users.
   a) Reduce system hardware costs
   b) Simplify hardware and software upgrade process
   c) Simplify processes for handling data products
   d) Refine the data set to the most essential information
   e) Fill in coverage areas to include “gaps” in coverage from Phase I of the IMO project.
   f) Extend coverage area to include the more populated Reno-Carson corridor and surrounding areas.

---

g) Include passenger vehicles in the fleet, in addition to existing snow plows and light duty trucks.

2. **Develop and implement prototype applications that improve maintenance operations and equipment maintenance.** Based upon NDOT’s needs and resources, the six potential applications identified by the FHWA were ranked in following prioritized order:

   a) Information for Maintenance and Fleet Management Systems.
   b) Enhanced Maintenance Decision Support System
   c) Information for Freight Carriers.
   d) Information and Routing Support for Emergency Responders.
   e) Motorist Advisories and Warnings.
   f) Variable Speed Limits for Weather-Responsive Traffic Management

NDOT’s primary interest was to enhance its Maintenance Management System (MMS) and the NDOT fleet (equipment) maintenance capabilities. As the level of integration of in-vehicle equipment (instruments and telemetry capability) rises, NDOT also planned to investigate the implementation of a broader Maintenance Decision Support System (MDSS) that leverages the equipment and experiences gained in the present project.

Phase 2 of the IMO project included a task being led by NCAR of developing and testing an enhanced **MDSS** option (EMDSS) for DOTs that made effective use of data from mobile observations (not only fixed RWIS and other sparsely-located weather stations). UNR and NDOT worked together to build on the Phase 1 experiences and focus on gathering data from a more concentrated region (the Reno-Carson-Tahoe area), to provide a “critical density” of data for use in NCAR's EMDSS prototype.

A related application that would contribute to effectively using an MDSS system is a Material Management System (MMS), which was also a peripheral goal to the activities in Phase 2 of the project. At the time, while there were records being kept on how much de-icing and anti-icing materials were being used by snowplows as they drive their routes, there was no accurate information on where along the route material was being used. Without having a good understanding of how segments of a road are being treated, there was no effective way of determining where MDSS benefits or losses were being experienced. Developing this capability was a secondary goal of Phase 2 of the NIMO project.

**Phase 3 of NIMO Project**

Phase 3 of the NIMO project was tasked with taking the final step towards establishing an effective and sustainable IMO program. A second major objective was to fully incorporate, test, and evaluate the use of Dedicated Short-Range Communications (DSRC) data telemetry as part of the IMO system. For this, NDOT established a 30-mile DSRC test corridor along Routes 395 and 580 connecting Reno and Carson City. The third major objective was to fully incorporate multimodal data telemetry, with all test vehicles equipped with both cellular and DSRC data telemetry capability, and the ability to switch to the most effective mode depending on the vehicle’s location. This permitted inclusion of numerous “offshoots” (away from the DSRC equipped road segments) along mountain pass roads that extend towards Lake Tahoe over Mt. Rose (SR 431) and Spooner Summit (SR 50) which see more vigorous winter weather.

The goal of a sustainable IMO program guided many aspects of Phase 3 and led to a much closer collaboration between researchers at the University of Nevada, Reno and NDOT personnel. UNR worked
with NDOT and other contractors to bring key IMO system elements “in house” at NDOT, including hosting of data servers, upkeep of the databases and data streams, and maintenance of field-deployed hardware. This necessitated training and full involvement of NDOT IT\(^6\) and other professional staff for the duration of the pilot test, who were critical to these activities.

**Current (NIMO Phase 3) Implementation**

The hardware and software developed and deployed in Phase 3 of the NIMO project was envisioned from the outset to have a high “technology readiness” level that could be sustained and maintained (as well as used) by NDOT personnel, with minimal involvement of the research team at UNR (i.e., Phase 3’s emphasis on deployment contrasted with Phases 1 & 2 of the IMO project, which were more focused on research and development). The Phase 3 system is described as follows.

**System Overview**

**NIMO System - Overview**

Phase 3 of the NIMO program establishes a corridor for a pilot test of a multimodal connected vehicle system, nominally situated along I 580/US 395, and connecting Reno and Carson City, where a system of DSRC road side units (RSUs) are installed periodically over the 30-mile corridor.

---

\(^6\) Nevada Department of Transportation’s Information Technology (IT) section of Traffic Operations Division is a dedicated group providing quality IT support for Traffic Operations.
DSRC as depicted in Figure 2 is a short range, two-way communications system that permits rapid connections and high bandwidth data exchanges, and is well-suited for data communications involving moving vehicles. The communications range is typically less than 1km, necessitating a network of communications nodes (towers, antennas) along the road to maintain The Department of Transportation’s Intelligent Transportation Systems (ITS) Joint Program Office (JPO) states that “The U.S. DOT’s commitment to DSRC for active safety communications contributes to safer driving. Vehicle safety applications that use vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communications need secure, wireless interface dependability in extreme weather conditions, and short time delays; all of which are facilitated by DSRC.” The NIMO program’s use of DSRC falls under the V2I (Vehicle-to-Infrastructure) category, with weather-related data formatted as a generic Wave Short Message Protocol (WSMP) packet, which best accommodates our data and the multimodal telemetry scheme in use by NIMO (i.e., cellular/DSRC switching).

The NIMO test corridor depicted in Figure 3 also includes non-DSRC road segments (using cellular data links) along mountain pass roads that extend toward Lake Tahoe over Mt. Rose Highway (SR 431) around Lake Tahoe (US 28) and connecting back to Carson City over Spooner Summit (US 50) which see more vigorous winter weather. The NIMO DSRC corridor provides ideal opportunities to tie the connected vehicle system into the existing ITS system in this area, which includes Road Weather Information Sites (RWIS), Highway Advisory Radio (HAR), Dynamic Message Signs (DMS), weather-based high-profile vehicles prohibitions and a weather-based variable speed limits.

Several dozen NIMO-instrumented vehicles were previously operated along or near this corridor using earlier “developmental” versions of the NIMO hardware from Phases 1 & 2 of the project. The hardware in a subset of these vehicles was upgraded with the further-evolved NIMO system that was designed to be robust enough for full commercial use and upkeep (i.e., not an R&D-focused system but rather a system that uses off-the-shelf components to every extent possible).

Figure 4 illustrates how the NIMO system works to take measurements at the vehicle and telemeter it to users in near-real-time. As instrumented vehicles travel along the test corridor, NIMO data will be telemetered using the best available mode, prioritized as 1) DSRC, 2) Cellular. Future adaptations of the NIMO system can readily accommodate other telemetry modes such as OpenSky, P25, satellite, or the Enhanced Digital Access Communications System (EDACS), recognizing that each mode has varying polling rates. Telemetered data is routed through the NDOT Data Exchange (NDEX) warehouse utilizing the Traffic Management Data Dictionary (TMDD) protocol, for internal use in applications, and for
integration into other IMO applications (such as those developed by NCAR and the WxDE). If neither DSRC nor cellular connectivity are available, the data will be stored on board the vehicle and relayed once communications are re-established by the vehicle moving into a valid coverage area (store and forward).

Figure 3. NIMO Pilot Test Corridor. Red denotes 32 miles of road segments served by 18 DSRC Road Side Units (RSUs) with varying coverage density. Blue denotes 54 miles of road segments with cellular coverage (no DSRC).
Figure 4. Vehicle Data Transmission. Data transmitted is either DSRC or cellular, depending on which mode is available (prioritizing DSRC when both are available). The back-end receiving station will archive the data and forward it to downstream applications and users.

In Vehicle Components - Overview

The IMO equipment mounted within the vehicles (such as a snow plow) serves the functions of:

1. Sensing data such as
   a. Location data (GPS)
   b. Weather parameters (such as temperature)
   c. Vehicle parameters (such as spreader rate)
2. Temporarily caching these data, formatting it, and preparing for transmission to NDOT
3. Data Transmission: i.e., selecting a telemetry mode, sending the data, and confirming successful transmission (when possible)
   a. Cellular Modem
   b. Dedicated Short Range Communications (DSRC)

To perform these tasks, the in-vehicle system is depicted schematically in Figure 5. The in-vehicle system uses a central CPU to perform oversight of system behavior and manage the associated peripheral equipment, including sensors and radio/modems. Communications between the CPU, the sensors, and the communications devices is mostly mediated through Ethernet interconnections between the devices, with the exceptions of a single serial (RS232) device and digital inputs (managed through an Ethernet-enabled data acquisition device).
Figure 5. Schematic layout of the system components located in each vehicle.

The in-vehicle hardware components are powered by the vehicle’s 12V power system, which is provided to the NIMO Hardware Backplane (see description below) through a single connection, and is then redistributed to the individual NIMO system components from a fuse-governed power block located directly on the backplane.

The IMO hardware backplane depicted in Figure 6 is the main physical component of the installation, and is typically located within the passenger compartment of the vehicle, such as behind the passenger seat, or on the back, interior wall of the cabin. The backplane is protected by a transparent cover to protect it from casual damage, but with two of the sides open to permit cooling and ventilation as well as easier access to cables and connections for things like remote sensor heads and antennas. In addition to the backplane, each vehicle has several additional installation points for the sensors and antennas. These require several cabin pass-through points for the associated wires and cables, and for the components located outside the cabin. These require suitable protection from the elements (rain, sun, heat, etc.).

Nominally, the hardware within the vehicles is powered on automatically when the vehicle is turned on, taking a few minutes for the equipment to “boot up” and begin normal functioning.

Most basic physical maintenance of the vehicle hardware (and software) can be performed through local access to the hardware backplane, or to the other devices of concern (sensors, cables, etc.). Maintenance of the software or firmware of the various devices can often be performed remotely through
the wireless network gateway of the Cradlepoint cellular modem, provided the vehicle is turned on. These remote activities can also be performed locally using a laptop and Ethernet cable connection.

Figure 6. In-vehicle hardware backplane. The backplane reveals the individual components and connections (not showing off-board antennas and sensors).

Road Side Components - Overview

The DSRC Pilot Test Corridor extends between Reno and Carson along I-580/US-395, with 18 Road Side Units (RSUs) installed like Figure 7 for both clusters of contiguous coverage road segments as well as more sparsely separated road segments with RSUs that can be used as “drive-by hotspots” for data dumps. RSUs are connected to the back end of the system through either direct fiber connections (most common) as well as cellular modems. These links permit the transfer of telemetry data from the DSRC system (vehicle-to-roadside) back to NDOT. The essential job of the RSU is to convert Wave Short Message Protocol (WSMP) packaged DSRC data into TCP/IP (the same format as data arriving over the cellular link), and to forward this on to the NIMO server. Depending on where the vehicle is (which it determines using GPS and geo-fencing rules), the on-board system will determine whether to use DSRC or cellular to send the current data (Figure 4). If a transmission is attempted and is unsuccessful, that data is stored until transmission becomes possible (i.e., it will re-try).
The DSRC system is “short range” per the FCC, which in the NIMO implementation, means transmission ranges are limited to approximately 1km (with line-of-sight), although radio power level can be varied to manage the signal propagation.

The process of selecting effective locations for the Road Side Unit installations generally should consider a range of issues:

1. Site location relative to neighboring RSU sites (to provide desired density of coverage along the road segment).
2. Line of sight from the RSU antennas along the road segment (factoring in height of the antennas, terrain, bridges and overpasses, signage, structures, and curves in the road).
3. Availability of existing mounting infrastructure (poles, etc.). Mounting on existing infrastructure can lead to significant installation cost savings.
4. Availability of roadside electrical power and backhaul network connectivity (such as fiber lines). When not available, data backhaul can be arranged using cellular backhaul.
5. RSUs should be configured to use the minimum necessary RF power for the task.
6. Omnidirectional or directional antennas can be used on the RSUs to tune the signal propagation direction and distance.
7. Each RSU installation site needs to be reviewed and licensed\(^7\) by Federal Communications Commission (FCC). This can take time on the order of 9 months, which should be factored into project planning timelines.

**NIMO DSRC FCC & FAA Site Registration Procedure – Overview**

The following procedure is required for adding DSRC radios to new and existing ITS poles:

1. Verify whether an Federal Aviation Administration (FAA) study is required. This can be done quickly by going to TOWAIR\(^8\), and entering the geographical coordinates, site elevation, and structure height. Do not assume that an airport is not near the ITS pole location. Allow 30 days with all processes and resources to complete.
2. If the TOWAIR determination indicates a structure registration is required, then a study request must be submitted to the FAA. This study takes up to 45 days to complete. Also, the FAA may require additional lighting and/or markings on the pole. Allow 90 days with all processes and resources to complete.
3. Upon completion of the FAA study and approval, an FCC Form 854 (application for antenna structure registration) must be submitted; this will take up to 45 days for approval. Allow 90 days with all processes and resources to complete.
4. Follow up with the FCC requirement for local public notification and amend the current structure registration application – allow 30 days.
5. Upon completion of the amended structure registration, RSU site registration can be completed online with FCC approval within 30 days.

**Local Notice:**

A. You must provide local notice of your proposed tower through publication in a local newspaper or other appropriate means, such as by following the local zoning public notice process. The local notice must contain the following information:
   - B. The Form 854 File Number;
   - C. The geographical location, structure type, height, and anticipated lighting for the proposed new or modified structure;
   - D. A statement that interested persons may review the application\(^9\) and entering the Form 854 file Number;
   - E. A statement that interested persons may raise environmental concerns about the proposed structure filing a Request for Environmental Review with the FCC;

---

\(^8\) See TOWAIR resources following link at [http://wireless2.fcc.gov/ULsApp/AsrSearch/towairSearch.jsp](http://wireless2.fcc.gov/ULsApp/AsrSearch/towairSearch.jsp)

\(^9\) See the FCC resources link at [www.fcc.gov/asr/applications](http://www.fcc.gov/asr/applications)
A. A statement that the Federal Communications Commission strongly encourages interested parties to file Requests for Environmental Review online, with the instructions for making such filings\textsuperscript{10} to be found; and

B. The mailing address for interested parties that would prefer to file a Request for Environmental Review by paper copy; FCC Requests for Environmental Review, Attn: Ramon Williams, 445 12th Street SW, Washington, DC 20554.

C. You may provide local notice under both this process and the Commission’s procedures implementing Section 106 of the National Historic Preservation Act (NHPA) through a single publication, provided the notice explains the different procedures for public participation under each of these processes.

D. You will be required to report the local notice date in Part 2 of your Form 854 filing. If local notice is not provided before the requested national notice date, you must amend your Form 854 submission to request a new national notice date.

**Back End Components – Overview**

The vehicle mounted, and road side components accumulate the mobile data and relay it to the NIMO server (back end). This dedicated server is behind the NDOT firewall and receives incoming data via TCP/IP for both cellular and DSRC telemetry paths. The server decodes the data and ingests it into a JSON compatible database, where the data is archived and made available to client users on request. For example, a common request will come from the NDEX system, which will pull the up-to-date data from the NIMO server in the form of a comma delimited text file (.CSV) that contains all the data received during the current (or any specified) calendar day. The NDEX serves as the system through which NDOT personnel (including road maintenance crews) interact directly with the NIMO data. Additional users of the .CSV formatted data files include FHWA’s Pikalert System and the FHWA’s Weather Data Environment (WxDE). The architecture of the data flow is schematically depicted in Figure 8.

\textsuperscript{10} See the FCC resources link at \url{www.fcc.gov/asr/environmentalrequest}
Figure 8. Schematic depicting the NIMO Concept of Operations and the data flow paths of the NIMO program.

NIMO System Hardware

The components used in the NIMO system were selected based on the experience gained from the earlier Phases 1 and 2 of the NIMO project. Heavy use of Commercial Off-The-Shelf (COTS) hardware components was made to facilitate future expansion and the long-term stewardship of the resources. Custom software was used to mesh together these components, which was developed by the university partners (UNR) as well as consulting software contractors.

On Board Unit Hardware (OBU)

The in vehicle NIMO hardware is based around a physical backplane shown in Figure 6, with connectivity between the individual components on the backplane depicted schematically in Figure 9. Beyond the backplane, there are a set of external antennas (GPS, cellular, DSRC) and external sensors (weather sensors, spreader, wipers). Summaries of the basic features and functionality of the system components are presented below.
Figure 9. Schematic of electrical and data connections between NIMO components on the system backplane.

*Technologic CPU*

The CPU shown in Figure 10 serves as the main processing device for the NIMO in vehicle system. It is an industrial ruggedized Linux device with multiple Ethernet, USB, and serial ports for connection to external devices. It serves to shepherd and manage the other NIMO devices in the vehicle through custom software. Information Technology (IT) packages the sensor data and pushes it to the Arada radio when DSRC telemetry is being used, or sends using TCP/IP via the Cradlepoint when cellular telemetry is being used.

The Technologic CPU and the NIMO system management software running on it, represent the most complex and customized element of the NIMO system, and its configuration is somewhat complex. Simplification of its configuration for each vehicle is one of the current objectives of the NIMO project.
The first of the data telemetry modes used by the NIMO system is DSRC. This is mediated through the Arada Locomate Classic radio shown in Figure 11. This DSRC radio was selected for the NIMO project for its cost, its utility for in-vehicle applications, the multiple available Ethernet ports, and the support and assistance offered by the manufacturer for customization of the device for the NIMO tasks. It also incorporates an embedded Linux system, which the NIMO project team had initially intended to use for general processing purposes. However, this general processing functionality was eventually moved to the Technologic CPU because of the greater configurability (libraries, compilers, etc.) it afforded. The Locomate Classic does include additional features such as GPS which are not used in the current NIMO implementation, but are available for potential future use. The Locomate is (in principle) a fully capably Linux based embedded computer system that has the potential to play a larger role in the NIMO hardware than it is currently used for (such as replacing the Technologic CPU). The OBU DSRC radio receives packaged data from the Technologic CPU and re-transmits over DSRC (using WSMP) to the Road Side Units (RSUs). It also manages confirmations and acknowledgements (which are in turn relayed back to the Technologic for multimodal functionality).

Configuration and use of the Locomate DSRC radio does require custom programming, although the manufacturer provides examples and tools to facilitate this. The configuration is specific to each vehicle (unique host name and network settings). The NIMO project also found it necessary to use a specific firmware version (for proper functionality) that had to be loaded manually.

---

11 The Lear Corporation announces that they acquired Arada Systems on November 30, 2015 see Lear Acquires Automotive Technology Company, Arada Systems
Cradlepoint Cellular Modem

The Cradlepoint cellular modem shown in Figure 12 serves to transmit and receive data along the second of our data telemetry modes (3G/4G cellular). The Cradlepoint is used as the NIMO system’s GPS for geolocation, with an external antenna attached for this. This device also features 802.11ac Wi-Fi capability, although the NIMO project does not use this feature at present.

Configuration of the Cradlepoint is done using a web-browser-based application, and management of these devices is readily incorporated into NDOT’s broader “fleet” of similar cellular modems used for managing other services such as roadside messaging system boards (etc.).

Comet Weather Sensor

The Comet T7511 shown in Figure 13 is a remote thermometer, hygrometer, and barometer with an Ethernet interface. The display/body is mounted on the NIMO backplane while its remote head is mounted on the exterior mirror of the vehicle to measure the atmospheric conditions. The Ethernet interface makes this device easy to integrate into the rest of the NIMO system.

Configuration of the Comet weather sensor is performed using a simple web-browser-based application.
Source: UNR, January 19, 2018

Figure 13. Comet Ethernet weather sensor.

ADAM Data Acquisition & Center

The ADAM-6051 shown in Figure 14 is an Ethernet enabled digital I/O device with an integrated 2-channel counter. This device is used to interface to the “customized” sensor elements of the NIMO system. These include 1) snow plow spreader material conveyor belt speed (with speed sensor integrated into the conveyor belt motor); 2) windshield wiper speed (signal measured using a Hall Effect sensor); and a material detector (capacitive proximity sensor for detecting sand/salt mixture on conveyor belt). Configuration of the ADAM module is performed using a simple web-browser-based application.

Source: UNR, January 19, 2018

Figure 14. ADAM Ethernet enabled I/O module.

RoadWatch Road Surface Temperature Sensor

The RoadWatch SS shown in Figure 15 serves to measure the temperature of the ambient air, as well as the road surface temperature. The sensor head is mounted outside the vehicle, pointing downwards towards the road, and the indicated temperature is displayed in the cabin for use by the driver. The RoadWatch sensors are standard (pre-existing) equipment on all NDOT snow plows. Interfacing these to the NIMO hardware required only an RS-232 adapter (sold separately) and connection to the Technologic CPU via a serial port.

Configuration of the RoadWatch SS is not generally necessary.
Sensor Selection and Placement

The in vehicle NIMO hardware sensor location and placement are based around the physical backplane (Figure 6) of the OBU and the connectivity between the individual components of each weather sensor located on board each snow plow as shown in Figure 17. The external antennas (GPS, cellular, DSRC) and external sensors (weather sensors, spreader, wipers) locations are depicted below.

On Board Unit (OBU) and Sensor Data Connections

The OBU shown in Figure 16 depicts the connectivity between each component of the OBU and the weather sensors.
Figure 16. NIMO On Board Units (OBU) and Sensor Data Connections.

**Plow On Board Hardware and Sensor Placement**

The Plow On Board Hardware placement shown in Figure 17 depicts the placement of the external antennas (GPS, cellular, DSRC) and external sensors (weather sensors, spreader, wipers) locations.
Figure 17. NiMO Plow On Board Hardware and Sensor Placement.

Source: NDOT, January 26, 2018
**Road Side Unit (RSU) Hardware**

DSRC communications are relayed from the vehicle to a Road Side Unit (RSU). The NIMO project has 18 such RSU sites, with their locations depicted in Figure 2. The Arada Locomate installation shown in Figure 18 was selected as the NIMO RSU radio primarily to complement the Arada OBU radios (discussed above), although it is not necessary to use OBUs and RSUs from the same DSRC product manufacturer. In addition to the radio, a Power Over Ethernet (POE) injector is needed, as well as cabling between the nearby service box and the mounted device. Typically, the NIMO installations were (intentionally) made near pre-existing infrastructure to support this (power and network access).

As discussed above for the OBUs, the RSU Locomate is (in principle) a fully capable Linux-based embedded computer system that requires configuration programming. The RSU’s software differs from the OBU in its use of the onboard GPS in the radio for timestamping of the packet transmission steps. Configuration and use of the Locomate DSRC radio does require custom programming, although the manufacturer provides examples and tools to facilitate starting this.

**Other Hardware Considerations**

One of the key considerations for the NIMO project’s design was the fleet of snow plow vehicles in use by NDOT. Most plows in service did not have AVL capability or spreader systems with CAN bus data interfaces that could be integrated into the NIMO system. This made it necessary to retrofit some capabilities into the vehicles as part of the NIMO system design. It is expected that future additions to the NDOT fleet will have these features already available, and it will then be necessary to integrate these more “data capable” vehicles into the NIMO system. The flexible nature of the NIMO hardware should accommodate this with a minimum of engineering redesign (software) of the on-board systems. The back-end elements of the system (NIMO server and NDEX) can also be readily updated to accommodate any new data parameters that become available.

**NIMO System Software**

**OBU Software**

The functionality of the in-vehicle system’s software is to read the sensors, perform data formatting (unit conversions, etc.), package the data into java script notation (JSON) database format, and perform a multimodal transmission using geo-fencing rules and messaging acknowledgements. These data packages are either pushed to the OBU, or over cellular 3G/4G using TCP/IP over the Cradlepoint cellular modem.
**RSU Software**

The RSU’s receive data packets via wave short message protocol (WSMP) from the vehicles. The RSU then relays packets as database transactions via TCP/IP to the NIMO server. It does this using the same exact database transactions that would be done over cellular, which simplifies the receiving functionality of the NIMO server.

**Backend IMO Server**

The IMO server at the “back end” runs under Windows on a virtualized machine at NDOT. This database server receives the pushed TCP/IP data from the vehicle for both the DSRC and cellular telemetry modes. It stores all data locally using a JSON compatible data protocol (CouchDB).

The IMO server machine also runs two web applications that manage database access. The first serves the updated daily .csv file upon request (this is the main functionality of the server). The second application serves maps of the data (instead of a .csv of the data) which is useful especially for simple system checks and debugging. Thus, the two applications serve up different “views” of the data.

**NDEX**

The NDEX\(^{12}\) is a set of resources for NDOT data distribution and sharing. It runs on a server that provides services to various users of data. The NDEX interfaces with other services or applications that can request data within the NDEX service, and can publish their data to NDEX services. The NDEX incorporates data archiving, security, and reporting components. Example uses of the NDEX include Connected Vehicles (the NIMO project), traffic management, emergency management, maintenance and construction management, and numerous other current and future applications or systems. In the NIMO system, the NDEX pulls data from the NIMO server database as a .CSV file (using standardized National Transportation Communications for ITS Protocol (NTCIP)\(^{13}\) 1204, Society of Automotive Engineers (SAE)\(^{14}\) J2735, and Nation Marine Electronics Association (NMEA)\(^{15}\) 0183 data formats) containing all the current day’s data (this file is updated by the NIMO server when the request is received). The NDEX

---

\(^{12}\) A summary of the NDEX can be found at: [http://www.westernstatesforum.org/Documents/2015/presentations/Nevada_Lopez_FINAL_NevadaDataXchange.pdf](http://www.westernstatesforum.org/Documents/2015/presentations/Nevada_Lopez_FINAL_NevadaDataXchange.pdf)

\(^{13}\) See link for National Transportation Communications for ITS Protocol at [https://www.ntcip.org/](https://www.ntcip.org/)


\(^{15}\) See link for National Marine Electronic Association at [https://www.nmea.org/](https://www.nmea.org/)
periodically request this .csv file, and upon receipt, stores the data in the NDEX database system. Once in the system, the data becomes available to NDEX subscribers (such as maintenance crew supervisors) in a variety of customizable formats.

**NIMO System Networking**

The NIMO program uses the (public) 3G/4G cellular network, DSRC, and the private NDOT network behind a firewall. At present, telemetry legs over the public cellular network are managed through a Virtual Private Network (VPN) for security. The DSRC leg (using WSMP) is not currently encrypted or otherwise secured, but this capability is available with the DSRC radios. Once the data reaches either an RSU (for DSRC telemetry) or the NIMO Server, it is then behind the NDOT firewall, and secured. The public part of the NDOT network is used for external access to the NDEX database from external clients such as the WxDE and Pikalert System (etc.). These requests are controlled through wrappers that ensure only read-only requests are acted on (for security from users outside of the firewall).

**NIMO System Costs**

**Vehicle Equipment Costs**

*Table 1: Vehicle Equipment Cost Breakdown*

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main Processor</td>
<td>Technologic Systems Linux Computer</td>
<td>$400</td>
</tr>
<tr>
<td>Weather Sensor – Air</td>
<td>Comet T7511 Ethernet</td>
<td>$600</td>
</tr>
<tr>
<td>Weather Sensor –Surface</td>
<td>RoadWatch</td>
<td>$750</td>
</tr>
<tr>
<td>DSRC Radio</td>
<td>Arada Locomate</td>
<td>$1350</td>
</tr>
<tr>
<td>Cellular Modem</td>
<td>Cradlepoint IBR1100</td>
<td>$800</td>
</tr>
<tr>
<td>Data Acquisition</td>
<td>ADAM 6051 Ethernet DIO/Counter</td>
<td>$200</td>
</tr>
<tr>
<td>Misc. Components</td>
<td>Hall effect sensor, material sensor, cabling, backplane fabrication</td>
<td>$550</td>
</tr>
</tbody>
</table>

**TOTAL (Vehicle Hardware):** $4,500

In addition to the hardware component costs, the in-vehicle equipment requires the time of qualified personnel to assemble the components onto the backplane, and to mount the equipment into the vehicle. These tasks will take up to 3-5 days per vehicle, depending on the complexity of the installation into a specific vehicle type and the experience of the installing personnel.
**RSU Equipment Costs**

DSRC hardware costs for the Road Side Unit costs are on the order of $3150 per site for the Arada Locomate Commando radios. A significant portion of the costs for each RSU are in the installation and any associated infrastructure costs. For the NIMO project, contractor costs for the RSU installations averaged $5,500 per site, and required approximately 2 days each for installation.

**Backend Equipment Costs**

Back-end equipment costs for the NIMO project are minimal. The NIMO server is run as a virtualized machine on existing NDOT resources. The cost for the servers is absorbed in the normal operating and maintenance annual budget.

**Recurring Costs (data plans, etc.)**

The NIMO project makes use of cellular 3G/4G data service, which does have costs that are included as part of a wider-scale NDOT cellular data plan. NIMO data use is estimated at $180 per vehicle per year.

**Development Costs (University and NDOT)**

Estimating the costs for the development of the NIMO project is complex and challenging, but an approximate estimate totals to over $1.5M over 6 years. Key aspects of the development costs to consider include:

- Phase 1 research was conducted on AVL and MDSS for NDOT by the University of Nevada, Reno. This research directly led into the formal NIMO project, and was funded at the level of $213,792.00 over a period of 1 year.
- Phases 2 costs were principally for the University of Nevada, Reno’s activity. This was funded at $312,445.00 over 3 years. During this approximate period, separately funded activities that were partially related to the IMO project were also funded at approximately $300k.
- Phase 3 costs supported both the University of Nevada, Reno and NDOT (directly). The UNR portion of the budget was funded at approximately $389,000.00 over 2 years, and the NDOT portion at 236,000.00 over 2 years. During this period, UNR was separately funded for activities that were partially related to the IMO project at approximately $200k.
- Over the course of the NIMO development, there have been numerous NDOT personnel involved, especially during Phase 3. These costs are not well-characterized, but it can be expected that there will be costs associated for personnel working with the various programs associated to the NIMO project including but not limited to the hardware installation personnel, Maintenance and Asset Management Division personnel, Equipment Division personnel, Traffic Operations personnel, District personnel, as well as IT professionals.

During the NIMO development process, there was extensive collaboration with partners at the states’ DOT in both Minnesota and Michigan as well as NCAR and the FHWA. These took the form of approximately 3 meetings per year where ideas were generated, lessons learned where shared, and
plans coordinated for development activities. It is difficult to quantify the dollar value of this collaboration, but it demonstrates the importance of leveraging the knowledge and experiences of other professionals associated with IMO projects in other states. This is envisioned to extend leveraging the services of commercial service providers as the IMO “marketplace” develops in the future. This collaboration and leveraging of knowledge will assist the department making future recommendations to reduce costs associated over the three phases of the NIMO deployments.

With the department embarking upon the Transportation Systems Management and Operations (TSMO) program which focuses on actively managing the multimodal transportation network to deliver an improved safe and mobile transportation system

**NIMO System Performance evaluation**

**Functionality**

*Overview of Functionality*

The NIMO Phase 3 system was designed to leverage the experiences gained during Phases 1 and 2. The system made extensive use of COTS hardware devices, leaving the rest of the system’s functionality largely to the custom software elements. The NIMO system Phase 3 is well-established along the Reno to Carson City pilot corridor, with approximately 10 vehicles instrumented for DSRC use, and with plans to expand the number of NIMO-equipped vehicles to other areas with cellular coverage. In the longer term, the vision is to further expand NIMO vehicles into areas with data telemetry coverage using future connectivity modes (P-25, Wi-Fi, etc.), and the NIMO system is ready for these further evolutions.

**Maintenance and Support**

An important feature for the NIMO system is the ability for updates to be made and for problems to be identified and debugged. The mobile nature of the system’s hardware can make these actions complicated, as they are often in the field, and are always based at NDOT sites distributed across the state. We have found that it is occasionally necessary to make a “house call” visit to a vehicle in the field to make a physical diagnosis or correction (disconnected cables, etc.). However, most aspects of the in-vehicle hardware can be monitored or diagnosed remotely, provided 1) the vehicle power is on, and 2) a basic level of network connectivity is available. In these cases, which can usually be arranged with the help of NDOT personnel at the remote location to turn on the vehicle for a period, the NIMO team can remotely log into the hardware components (which are mostly Ethernet devices), and make a diagnosis. Once identified, if the problem cannot be resolved remotely, at least a more refined diagnosis is in-hand to expedite any in-field service that is needed.

In addition to the mobile hardware, the NIMO project also uses stationary field devices (RSUs), which also can often be interacted with through remote network login. When this is not sufficient to resolve problems, an on-site visit may be necessary. It is usually sufficient to access the road-side connection box, although because the physical RSU is pole-mounted, it is conceivable that a bucket-truck may be necessary for some maintenance actions (although this has been rare, once the system was initially installed).
Retrograde Functionality Enhancements

Phases 1 and 2 of the NIMO project included functionalities that are no longer supported by the NIMO project. The first of these is the use of EDACS data telemetry in the rural areas of Nevada. As discussed earlier, the EDACS system, while still operating in Nevada, is on the verge of being phased out and replaced by a new system. The strategic decision was made by the NIMO team to deprecate the EDACS functionality, as its short remaining lifespan, its complexity, and its significantly lower bandwidth capability were considered too much of a cost for inclusion of this telemetry mode in Phase 3 of the IMO project.

The second functionality that is no longer supported in Phase 3 of the NIMO project is the monitoring of OBD/CAN bus sources. While this was effectively demonstrated during the earlier phases of the NIMO project, it was found that a large fraction of the snow plow fleet still uses plows that are not equipped with CAN bus capability (J1939), and those that do, have significant vehicle-to-vehicle variations in which data parameters are available. The core of an OBD/CAN bus data monitoring capability is available, but until NDOT’s fleet becomes more uniform and modern (which is happening gradually), these data sources require more engineering effort to monitor as part of the NIMO program.

Data

DSRC Data Telemetry

The DSRC telemetry mode was demonstrated to work very well once properly configured. The vendor used for the DSRC radios was Arada. DSRC links were observed to be established quickly and reliably (as vehicles moved through communications range) and the DSRC standard was found to be flexible enough to accommodate several different approaches to sending the information. NIMO made effective use of the custom WSMP messages that are not part of the standard Basic Safety Message format, although this option is available. Installation of the DSRC infrastructure (RSUs and backhaul) made effective use of NDOT personnel already engaged in these aspects of IT operations.

Cellular Data Telemetry

The cellular 3G/4G telemetry mode is the “workhorse” mode for transmitting data in the NIMO project, as it has much more broad coverage than DSRC, but also with a sufficiently higher bandwidth. Cellular telemetry does, however, involve a monthly data charge through the cellular service provider (whereas, DSRC does not). Within our test area, coverage is strong, and telemetry was observed to be quick and reliable—even during winter weather events.

Multimodal Data Telemetry Features

The NIMO system effectively uses a multimodal telemetry system that selects the best telemetry mode (cellular or DSRC), depending on the vehicle’s location. Challenges were found in establishing reliable geofencing rules, especially as this is related to handling data packet receipt confirmations and (as needed) re-transmission as the vehicle moves in and out of telemetry mode zones. This is more complex when the vehicles are frequently moving through different telemetry mode zones.

Weather Data

The NIMO system monitors and reports key weather data elements including the air temperature, surface temperature, air pressure, and humidity. NIMO also monitors the vehicle’s windshield wipers, which is
used as a proxy for a rain sensor. These data are considered to have the highest relevance for refining road weather forecasts.

**Vehicle and Spreader Data**

To provide information for material management decisions, Phase 3 of the NIMO project developed and deployed sensors for monitoring when the spreader was being used and material was being put on a given road segment. These new features of the NIMO system were demonstrated to function during the winter of 2017.

**Impact**

The impact of the NIMO project at NDOT can be measured anecdotally several ways, and has progressed stepwise over the course of the program. Perhaps the most direct impact has been the achievement of a “critical mass” of experience and investment. During Phase 3 of the project, NDOT made notable increases in the investment level and involvement of personnel (especially from IT), which had the side-effect of drawing more serious interest from other areas of NDOT (maintenance, equipment, etc.). An *esprit de corps* surrounding NIMO has begun to develop, although the experiences of our DOT peers in Minnesota and Michigan have prepared NDOT to work on earning “buy-in” from across the organization, as this will (ultimately) be a key factor in achieving a high impact level in NDOT operations.

To have a high impact in organizational operations, it is first necessary to have a system with sufficiently high functionality. This functionality also needs to show evident “promise” to the envisioned users. One of the challenges that the NIMO program has been confronted with has been the need to simultaneously address the “three legs of the stool” as illustrated in Figure 19 and the need for high impact. Much of the engineering effort in the NIMO program has focused on the center leg, i.e., developing the system’s functionality so that it works as intended and as advertised to the prospective users. Where the NIMO project found more challenges was on the “human legs” of the stool (i.e., in 1) alerting users to the potential value of the applications that NIMO will enable (such as MDSS, MMS, etc.), and 2) providing sufficient training so that users are able to realize the promised value).

NDOT envisions that the NIMO project will contribute to several main impacts to its operations, both of which will benefit from a basic level of AVL capability.

The first targeted impact is the deployment of an MDSS (Maintenance Decision Support System) for winter operations at NDOT. Although an MDSS does not strictly require the mobile data features of the NIMO system, mobile weather data has the potential to greatly impact road weather forecasting, and is envisioned to be particularly important in the “microclimates” of Nevada, which make weather forecasting more difficult.
The second targeted impact is the implementation of an MMS (Material Management System) for better tracking of where materials (sand, salt, etc.) are deployed, and what the impact was. As more snow plows are instrumented with the NIMO system, this will become more feasible and provide higher resolution information than is currently available to road maintenance crews.

Future impacts beyond these two (MDSS and MMS) are envisioned to also be likely. These may include a greater informing and engagement of motorists through providing travel alerts, recommendations, and (eventually) crowd-sourcing of travel weather related information.

Applications

NDEX

The Nevada Data Exchange (discussed earlier) is a set of resources for NDOT data distribution and sharing. It incorporates data archiving, security, and reporting components. For the NIMO system, the NDEX pulls data from the NIMO server database containing all the current day’s data and stores the data in the NDEX database system. Once in the system, the data becomes available to NDEX subscribers (such as maintenance crew supervisors) in a variety of customizable formats. The NDEX (with NIMO functionality still under development) will be the primary channel for internal (NDOT) and external (i.e., WxDE, NCAR, National Oceanic and Atmospheric Administration’s MADIS system, and others) users of the NIMO data.

Weather Forecasts

NDOT uses the local National Weather Service (NWS) offices to provide road weather forecasts used by maintenance supervisors (and crews) for all weather event maintenance planning and operations, both in advance of and during an event. It has been hypothesized that the forecast accuracy in Nevada suffers from the terrain features and the generally widely-dispersed deployment of Road Weather Information System (RWIS) road-side weather sensors. At present, there are approximately 85 strategically positioned RWIS sites positioned along the 5,400 miles of road that NDOT is responsible for maintaining. These RWIS data sources, combined with NWS data provided are used to forecast road weather over the 110,000 square miles of the state of Nevada. On average, the RWIS stations are situated with one per every 64 miles of road featured on Nevada DOT’s 511 home page shown in Figure 20, which illustrates that vehicle-mounted mobile weather sensors can provide the valuable addition of

16 NDOT’s maintenance responsibilities are summarized at: https://www.nevadadot.com/doing-business/about-ndot

17 See Nevada DOT’s 511 home page at http://www.nvroads.com/511-home
higher resolution weather and road conditions (at more locations) during the winter weather events and any other time the vehicles are on the road.

Automated Vehicle Location – AVL

At the heart of the NIMO project is GPS-based Automatic Vehicle Location (AVL), whose geolocation capability is central to most of the other applications. The NIMO AVL adds a Mobile Data Computer (MDC) that monitors various mobile sensors. All this data (location and sensors) is telemetered to the back-end where users receive and interpret the information, and act upon it. The NIMO AVL core encompasses these features, and is designed for future expandability, as needs and resources evolve.

Enhanced Maintenance Decision Support System – EMDSS

For the Nevada Department of Transportation, a key over-arching motivation was to lay the foundation for a future implementation of a Maintenance Decision Support System (MDSS) and a Material Management System (MMS), which would leverage mobile data (i.e., incorporate weather and road information not only from fixed locations (such as RWIS), but also from mobile measurements obtained from NDOT vehicles). Mobile data was envisioned to help NDOT better characterize the significant geographical variations in weather and road conditions in Nevada and facilitate the optimization and effectiveness of deploying...
resources (staffing and usage of materials) during winter weather events. An MDSS system consists of both software and hardware and is typically provided through a commercial vendor. A typical MDSS system uses (at the least) a GPS-based Automatic Vehicle Location (AVL). At the system’s core is software which uses weather forecasting models to predict storm events and, when combined with a road
weather model, can forecast changing roadway conditions on routes that are selected by the DOT. The MDSS software makes recommendations of treatments to supervisors or snow plow drivers. During the NIMO Phase 3 project, NDOT used the Pikalert EMDSS application, made available for this pilot study by

Figure 21. Example user interface screens for the Pikalert EMDSS application for Nevada. Information is presented in map format, with additional details available upon drill-down by the user.
the National Center for Atmospheric Research (NCAR)\textsuperscript{18} under contract to the FHWA’s Road Weather Management Program. Typical user displays are shown in Figure 21. The EMDSS for DOT’s makes effective use of data from mobile observations (not only fixed RWIS and other sparsely-located weather stations).

Material Management System – MMS

Through the NIMO project, NDOT is working to make improvements to its tracking of material usage. For this, a Material Management System (MMS) is being developed. Vehicle location information, coupled with data on the use of the snow plow’s spreader, permits tracking of how specific segments of a road are being treated with material (sand, salt, etc.) during winter maintenance operations shown in Figure 22. In turn, this information will permit managers and engineers determine where EMDSS benefits or losses are being experienced, and to use this to optimize maintenance operations. The MMS outputs (material type, timing, amount dispensed, etc.) would serve as the backbone for the EMDSS on how well the DOT is performing to the level of service on routes as an important step in gathering the data and getting the most out of the system. Automation of NDOT’s MMS for material usage is recommended. If snow plows can record and transmit where and when salt/sand is being used along their routes (using a spread rate sensor), an even more accurate understanding of material usage can be had. NDOT would than use this information, as well as other measurements, to quantitatively represent how well they are treating the roads. Formulating a performance index can either be done entirely in the department, or other state’s performance indices can be referenced.

Secondary (Peripheral) Applications

The federally-organized IMO initiative has focused attention on the general set of needs for effective winter road maintenance. As part of this, new ideas and opportunities frequently emerge, which can lead

\textsuperscript{18} The Pikalert application is summarized by NCAR at: https://ral.ucar.edu/solutions/products/pikalert\%c2\%ae
to development of new technologies in support of road weather management. The NIMO project involved two such peripheral projects, both currently in the concept development research phase.

**Traction (Friction) Measurement (research phase)**

The primary detailed objective of the IMO project is to make forecasts of inclimate road conditions (i.e., the road’s traction) using a variety of predictive metrics (weather data, most commonly). These data are generally used as proxies for how much traction the road surface will provide. There are a variety of approaches to directly (or indirectly) measuring road traction, each having its own set of strengths and weaknesses. Most of the available technologies manifest as proprietary approaches that are built into the vehicles themselves. Any information concerning road traction, although it may be present on the vehicle’s CAN bus, is not publicly available, and follows no broad standards from manufacturer to manufacturer. Furthermore, the information that the vehicles can generate is not generally correlated with the prevailing weather conditions (i.e., if a vehicle’s wheels are slipping, it may not be known if this is because of the driving style, or the road surface conditions (snow, ice, rain, dry)). It is of value to develop correlations between weather conditions and the road conditions, as the weather conditions can (at some level) be forecast, permitting planning of road maintenance activities in advance of an event.

The NIMO project undertook the side-project of working to use vehicle acceleration as a proxy for the force between the tires and the road, and to use this with wheel speed sensors to determine the road condition. The benefit of this approach is that acceleration sensors are inexpensive and simple to incorporate in a vehicle (they are already present in many forms in modern vehicles). The concept behind the NIMO approach is to simultaneously measure the wheel rotation speed for both the driving and driven axles of the vehicle, any differences between these is an indication of wheel slippage. When this "slip ratio" exceeds a certain magnitude, it is a sign that either the road is slippery, the driving is more aggressive, or some combination of these. The NIMO approach to estimating the level of traction is still in the research phase, but has shown promising results in being able to quantify the amount of slippage and correlate this with road surface condition (dry, wet, snow/ice). When coupled with the rest of the NIMO system, this can generate a live wheel traction report in map form as shown in Figure 23, which can alert road maintenance decision makers to potential problems with slippery roads.

![Figure 23. NIMO heat map. The heat map depicts vehicle traction along specific mountain pass road segments during a winter weather event.](source: UNR, January 27, 2018)
Salinity (Melting Point) Measurement (research phase)

One parameter that is of interest to winter road managers is the temperature at which any moisture on the road surface will freeze (or melt). Nominally, this is 0 ºC, but when additional chemicals are present, such as residual amounts of salt on the road, this melting/freezing temperature is shifted (this is the concept underlying the use of salt or other chemicals on roads). Excessive use of these chemicals wastes resources (and money), and can have a detrimental impact on the environment surrounding the road. It is useful to know whether a road segment has already been sufficiently treated with chemicals, and (by extension) it is for road crews to know when the snow or ice on a road will start to freeze or melt as the weather conditions evolve.

Melting point sensors are commercially available at present, although they are not yet commonly deployed, and at present, most commercially available technologies are permanently embedded into the road surface. There is, however, interest in developing mobile sensors for detecting the salinity of water on the road, which (through thermodynamics) permits accurate estimates of the melting or freezing point (i.e., the water on the road, with dissolved salt present, constitutes a saline solution, and a measurement of the salinity will directly lead to an estimate of the melting/freezing temperature).

The NIMO project’s approach to the problem of measuring salinity from a mobile platform involves the use of Raman spectroscopy. This technique uses a monochromatic light source (laser) to excite the molecules of the saline solution. The excited molecules, in turn, re-emit the light at slightly shifted wavelengths, where the shifts are characteristic of the chemical bonding that is present in the solution shown in Figure 24. Therefore, the Raman spectra that are produced this way act as a “chemical fingerprint” of the amount of salt that is present in the solution. This technique is non-contact in nature, and has been demonstrated in the literature to be viable over the measurement distances that would be needed for a mobile sensor platform (vehicle mounted).

Although this work is still under way, the NIMO team has demonstrated this methodology in the laboratory, and has used it to characterize numerous saline solutions using chemicals commonly employed in winter road maintenance (i.e., NaCl, MgCl₂, CaCl₂, KAc (potassium acetate), and CMA (calcium magnesium acetate)). At present, the key remaining challenges are in development of a
ruggedized vehicle-mountable system that works at the desired standoff distances and under typical ambient light conditions. Viable approaches to each of these remaining challenges have been addressed in the existing literature, and are considered “engineering challenges” and implementation tasks.

Lessons Learned

Lessons Learned – Organizational

Organizational Champion and Buy-In

The NIMO project was, at its core, a moderate-scale research and development project, which is not traditionally at the core of the mission of most state DOTs. It was a collaboration between NDOT, the University of Nevada, Reno, and several other out-of-state organizations (notably, the FHWA RWMP) working with both applied and research focused missions. To most effectively build meaningful impact of the NIMO project, it is necessary for a key person or small group to build all three legs of the “impact stool” (Figure 19) (i.e., someone must be responsible for selling the “value” of the project (buy-in), it must deliver on its technical promises, and users must be fully trained on how to maximize its benefits). It is important to note that in such a collaborative endeavor, no one collaborator is well situated to deliver on all three aspects, which places importance on the project champion. This person can be considered a project lead or project engineer, whose job is to ensure that everyone working on the “three legs of the impact stool” are doing so with maximum effectiveness.

Personal & Expertise

Beyond the organizational “champion” (discussed above), it is essential to have participation from personnel with a range of expertise engaged in the project. The first skill group is the IT staff. The NIMO project was extensively interconnected with elements that fell under the purview of networking, communications, and data handling at NDOT, and the involvement of this group was essential to project success. Second, it was necessary to consult with and train vehicle maintenance staff (and management) on the installation process for the in-vehicle NIMO hardware. This group is also responsible for some aspects of the continuing maintenance of the equipment. Third, it will be necessary to involve electronics technicians (or similar) on the continuing maintenance of the in-vehicle equipment, as once initially installed, most of the envisioned upkeep and repairs will be related to the sensors and computer-related equipment. Fourth, the road maintenance crews (and supervisors) should be kept “in the loop” and aware of the general aspects of how the system works, and the equipment that they can find in their vehicles. This group essentially constitutes the key end-user, and therefore will need to be trained on the NIMO applications and solicited for feedback on future enhancements or any problems.

Fleet Selection and Management

The vehicles that were selected for inclusion in the NIMO fleet consisted primarily of snow plows because of their expected deployment before and during winter weather events, and their winter-specific equipment (spreaders and sensors). In addition to the snow plows, a Freeway Service Patrol van was also included in the fleet because of its frequent use along the pilot test corridor, regardless of the weather conditions.
Fleet management in an organization that is geographically distributed like NDOT can be complex. Managing and supporting the hardware in the vehicles will become increasingly time consuming as the fleet size grows. The initial installation is somewhat time-consuming (over 3 day's) and must therefore be arranged to take place at a convenient time for the vehicle operators (such as when it is being called in for other service activities). Any difficulties or inefficiencies in arranging access to the vehicle fleet can lead directly to significant installation timeline impacts, especially during the winter-weather months of the year.

Lessons Learned – Hardware

General NIMO System – Overview

COTS vs. custom
The NIMO project, over its three phases, worked towards the use of a modular design that makes use of industrial-grade COTS components whenever possible. Although a custom designed system (NIMO Phase 2) could be less expensive in terms of initial costs, the long-term costs of a modular COTS system will be lower, as it is less necessary to maintain personnel with the necessary skillsets to design, build, and operate the hardware and software elements of the design. It should be noted, though, that even the COTS-based system will involve a significant amount of custom software development at first, although once the system is designed and in place, these custom work costs will be significantly reduced to maintenance and upgrades.

Normal wear and tear
Phases 1 and 2 of the NIMO project experimented with a wide range of sensor and electronic hardware components of a range of costs, complexities, and build quality. Normal “wear and tear” in the environment of a snow plow is understandably demanding, and the equipment used in the NIMO Phase 3 system is justifiably rugged and industrial in nature. Consumer grade components (such as cell phones) should not be used for this (and a range of other) reasons.

Hardware “stability” over installation lifetime (evolving technology)
Consumer-grade products, in addition to being less ruggedized than comparable industrial-grade products, also tend to have a shorter product life, after which manufacturer support wanes and replacements become difficult to procure (witness the rapid product cycles in the cell phone industry).

Vehicle Installation
Presently, the NIMO hardware package installation takes 3-5 days per vehicle, which has associated costs in terms of technician time, and vehicle out-of-service time. A smaller hardware package, external mounting, or some other solution that simplifies the installation process is desirable.

Smart Spreader Controllers
Once NDOT has reached the point when its snow plow fleet is sufficiently updated with modern “smart” spreader controllers, the NIMO system may be updated to integrate with these devices to more fully track and report spreader and plow blade use as well as other pertinent vehicle and equipment information.
**Multi-Modal Telemetry**

The deployment of an IMO-type system will likely require a multi-modal capability in most real-world circumstances. DSRC systems, while of great interest in more urban areas with high traffic volumes, are inherently a short-range system, requiring numerous RSU installation sites. This makes it impractical for deployment over large distances (such as in rural areas), unless a widely-spaced network is used with vehicles saving data for periodic download to the backhaul when the vehicle passes an RSU. It is not presently known if this makes for a viable operation mode for DOT operations. As DSRC systems become installed into more areas, it makes sense to take advantage of this mode of data telemetry, provided the vehicle also retains the ability to use other modes of telemetry (such as cellular) when not within DSRC range. Cellular-based systems are presently more omnipresent than DSRC, but even cellular systems can become unavailable in rural areas, requiring the use of still other modes of telemetry. This multimodal feature is essential in Nevada, as the only truly omnipresent mode of data telemetry is extremely low bandwidth, and being phased out (EDACS).

**Proprietary Technologies & Custom Electronics**

Specialized proprietary technologies or systems can be very effective, provided they are fully supported. This is most viable when a commercial vendor is standing behind it. Without this, proprietary technologies and custom components should be minimized or avoided unless the state DOT has sufficient in-house expertise and resources (and priorities) to take responsibility for these elements of the system.

**Vendor Selection and Product Support**

During the development stages, the quality and effectiveness of a vendor’s technical support is critical. Some technologies which are more specialized (OBD, CAN bus, DSRC) are especially subject to the need for technical support. At the earliest stages, developers should apply due diligence to researching the vendors, including soliciting input from other customers and extensive conversations with vendor representatives.

**Assessment of System Functionality**

Mechanisms and procedures for determining whether the NIMO system is transmitting correctly or whether the vehicle is simply not being driven (or not being driven frequently) is a need that would facilitate system upkeep and monitoring.

**DSRC Hardware**

At present, DSRC is still a technology in its infancy, with numerous hardware manufacturers continuously entering the market and developing new products. The number of commercial support providers for DSRC does not yet appear to be significant. It is hopeful that as the technology matures and gains market penetration, a base of knowledge will grow in the community of users that facilitate new users to navigate needs such as licensing, system design (RSU placement), selection and use of appropriate message formats, installation challenges, and facilitate sharing of DSRC data in and out of the immediate environment of users.
**Hardware Package Size**

The current NIMO design, primarily due to its modular COTS nature, has the drawback in that the assembled hardware components make up a large (OBU dimensions are approximately 24” L x 18” W, shown in Figure 25) package that introduces challenges for mounting in a vehicle such as a snow plow’s cabin. Alternatives to mounting the hardware inside the (climate controlled) environment of the cabin can be considered, including external vehicle mounting in weather-proof enclosures.

![NIMO OBU Dimensions](image)

*Source: NDOT, January 27, 2018

Figure 25. NIMO OBU dimensions.*

**Lessons Learned – Software and Data**

**Lessons Learned – CAN bus Data Sources**

The NIMO project made use of standard vehicle On-Board Diagnostics (OBD) data (SAE J1979) as well as the more industrial-equipment-based SAE J1939 data. Of these, the OBD data is more consistently available on vehicles already in service. However, both light duty vehicles and snow plows have CAN bus systems that make extensive use of proprietary information that is not generally available without the use of expert consultants or direct collaboration with the vehicle manufacturers. The NIMO project made use of both resources to good effect. However, for long term implementation, where the vehicle makes, models, and years are expected to vary over time, the use of CAN bus data was deemed to be of
insufficient value for the effort required. In short, each different make/model/year vehicle would need to be configured differently for the NIMO system, which is a significant amount of effort. If a standardized dataset can be developed across all manufacturers that includes weather-related data (windshield wiper status, traction control events, atmospheric conditions, etc.), this would be a major benefit to IMO-type technologies. Without this, though, it is recommended that CAN bus data not be used, in favor of other sensors and data sources.

**Software Version Control**

It is valuable to implement careful version control to ensure the stability of the field-deployed systems. This is also useful for maintaining functional test systems for “bench testing” and debugging if/when problems are discovered.

**Remote Software Updates**

It is very valuable (almost necessary) to incorporate an autonomous or semi-autonomous capability to remotely push software updates to OBUs and RSUs. Without this capability, it becomes necessary to make physical trips to widely-distributed vehicles and field installations to perform all maintenance and updates.

**Computing Platforms**

The three phases of the NIMO project used three different in-vehicle central CPU devices. All three hardware systems operated using the robust Linux operating system, which is readily used on embedded microcomputers. In both Phases 2 and 3, secondary microcontrollers were used that were based on Arduino (Phase 2) and Linux (Phase 3). Familiarity with Linux-based embedded systems are a useful skillset for managing the system, and is essential for initial system development. If not available within the DOT, consultants with these skills can be enlisted for this role.

At the back-end of the system, the NIMO project used a physical server system for the NIMO Server during its development stages, and then shifted this server to a virtualized windows machine at NDOT once the system became fully functional. The virtualized system simplifies the management for NDOT’s IT group.

**Networking Infrastructure**

It is important to involve IT personnel with networking expertise, as the telemetry paths all eventually work through the NDOT network (the DSRC path uses the NDOT network more extensively, though, with incorporation of the roadside fiber links). During initial setup, the NIMO team experienced numerous networking challenges related to component configuration, addressing, and firewalls. These took time to navigate and NDOT IT personnel were essential for this task.

**IPv6**

Not all NIMO component devices support IPv6, which is what the selected DSRC hardware requires for most IP. This requirement was circumvented, however, with the use of WSMP messaging. It may also be possible to use IPv4 over IPv6 tunneling to address this issue.
**DSRC Message Formats**

The NIMO system uses WSMP messaging through DSRC, but this must be converted to TCP/IP once the messages reach the backhaul (fiber or cellular from the RSU to the NIMO server). It would be of value to further investigate DSRC message type protocols to decide on a better approach to a multimodal telemetry system (Basic Safety Messages, etc.).

**Cellular Plans expensive and Coverage limited by provider**

Cellular data plans involve costs that can (with an appropriate data plan and qualifying data modem devices) share the data usage across many devices in the organization. This type of cellular plan is much more economical than the typical flat fee per device per month plan. An additional consideration for selecting cellular service is that in the rural areas of Nevada, different service providers provide different coverage areas, which should be considered for best area coverage.

**Use of Data Standards**

It is of significant value to (whenever possible) use existing modern data standards for the storing, transmitting, or sharing of IMO-related data. For the NIMO project, this has included SAE J2735 (DSRC), NTCIP 1204 (Environmental Sensor Station Interface Standard), NMEA 0183 (Instrumentation), and Traffic Management Data Dictionary\(^\text{19}\) (TMDD). The use of standards like these will often greatly facilitate later tasks that were not initially envisioned at the outset.

**Lessons Learned – Applications**

The key lesson learned regarding the implementation of applications again derives back to the “three-legged stool of impact” depicted earlier in Figure 19 (i.e., to value the available applications and effectively use them, it is essential to have a sustainability plan moving forward while meeting the department’s mission, vision and goals. Applications such as EMDSS or MMS is a key benefit of the Road Weather Management program. Making the applications known to potential users and the necessary training are tasks best managed by professionals familiar with the end users (such as supervisors and managers, or other DOT personnel or contractors with training responsibilities).

\(^{19}\) TMDD authored by ITE: [www.ite.org/standards/tmdd/](http://www.ite.org/standards/tmdd/)
Lessons Learned – Other

Costs

Hardware Costs

Most individual hardware components used in the NIMO system are of low to moderate cost (see Table 1). Exceptions can be data-capable radios (EDACS, DSRC). Some data devices require service plans (cellular) whose costs can rapidly accumulate over time, dominating the overall costs of the technology. Hardware installation costs may also need to be factored in, which can be estimated to be approximately 1-2 days of a technician’s time.

Software Costs

Software development costs for the initial standup of an IMO system are the most significant component of the overall system’s cost. The use of an expert commercial service provider may lower these costs (but introduce other costs).

Beyond the initial development costs, additional expenses associated with software or service licenses such as MDSS or MMS commercial product costs should be factored into any planning.

Other Costs

Cellular data plan expenses, as discussed above, should be considered a significant component of the overall system costs.

Licensing

When filing for FCC licensing, don’t assume that you are not within a Federal Aviation corridor. Be sure to follow the process checking with the FAA and gather all pertinent structural information necessary to fill out the appropriate forms. Allow an approximately nine-months’ time to complete the process.

Sustainability

Long-term sustainability of the NIMO program requires a range of considerations be addressed. These include:

- Documentation of the hardware, software, installation processes, and overall system design
- Planning for regular maintenance, testing validation, and service of the system
- The test corridor should evolve into a permanent service corridor, and expanded as needed
- Programmatic funding for the NIMO activities should be arranged and justified
- Training of personnel should be an ongoing activity so that all involved personnel are familiar with the salient aspects of the NIMO system
- Vendor Support should be developed and secured as the system matures. This includes involvement of weather data providers, MDSS product providers, etc.
Scalability

Phase 3 of the NIMO system has been designed for flexible scalability with linear or sub-linear scaling. Adding additional vehicles will increase costs and complexity approximately proportionally for each vehicle, until the system grows to a point where dedicated personnel become necessary for the NIMO system. A parallel case is seen in Minnesota’s DOT, which has dedicated personnel for their IMO program, but also oversees many hundreds of IMO-equipped vehicles in their fleet.

Other costs, such as data bandwidth and storage scale well with growth of the system. Additional DSRC RSU sites may scale more economically as well when the system is expanded in larger steps (groups of sites). In cases where a roadside fiber networking pathway is not available, RSU expansion will add to the cellular data backhaul (at additional data cost).

Data sharing

Data sharing is a feature that has been incorporated into the NIMO program from the outset. The use of accepted data standards has facilitated this. While some data may be considered proprietary, the majority of NIMO data is shared with various users with the aim of maximizing value and utility for the constituencies of NDOT.

Moving Forward

NDOT Commitment to Weather Savvy Roads (WSR) Road Weather Management Program

- NDOT will focus on getting more consistent actionable messages regarding road and weather conditions out to the public (which may be facilitated by IMO technologies).
- NDOT would like to host a workshop or peer-exchange with the NDOT PIO’s, NWS PIO’s, and other state experts that have a strong formalized program
- Continued development with the Pikalert system to support NDOT’s yearly maintenance road weather operations

Develop an NDOT Connected Vehicle/Autonomous Vehicle Implementation Plan

- Planning level document
- Snow Plow initiative
Develop and incorporate IMO into a Statewide NDOT Transportation System Management and Operations (TSMO) Plan

**Develop a Statewide TSMO Plan**

Develop a multi-year statewide TSMO Plan and related process improvements to include TSMO strategies at all levels of NDOT.

**Identify and engage a team of TSMO champions**

Establish an NDOT TSMO Core Team (TCT) and identify Senior NDOT Leader (SNL) advocates for NDOT’s statewide TSMO Plan.

**Identify Performance Measures for TSMO**

Develop goals for the TSMO program and identify relevant performance measures that can evaluate both the current state of the TSMO program and how TSMO measures can help in achieving NDOT’s goals. Identify the methods and timeframes for collecting the performance measure data.

**Include TSMO Components in Long-Range Planning**

Include TSMO components in the LRTP as an approach to improve traffic flow instead of and/or as integral components of capital capacity projects.

**Develop an Outreach Approach for TSMO**

Develop an approach to facilitate and encourage outreach to internal and external stakeholders regarding TSMO.

**Develop a TSMO Review for all new Projects**

Develop a TSMO review as part of the development of plans and projects where all new plans and projects are to be reviewed to determine if TSMO could be used to achieve the project objectives and/or if TSMO components should be included in the plans or projects.

**Create a Repository for NDOT TSMO Information**

Create and maintain a repository of TSMO information that is available and accessible to staff to ensure NDOT keeps us with the state of the practice.

**TSMO Staff Development**

Identify staff with TSMO skills and further their TSMO knowledge through training and other staff development.
Review TSMO Program and Define Next Steps

Advance NDOT’s maturity in all areas regarding TSMO by reviewing and assessing successes and failures of the TSMO CMM implementation pan.

NDOT TSMO Asset Management & Performance Measures Business Plan

Further develop ITS asset management practices including programmatic level performance measures that best support the TSMO Implementation Plan and MAP-21, aligns with NDOT’s Transportation Asset Management Plan (TAMP), maintenance management system and long-range planning; and meets the needs of all NDOT districts.
<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AVL</td>
<td>Automated Vehicle Location</td>
</tr>
<tr>
<td>CAN bus</td>
<td>Controller Area Network bus</td>
</tr>
<tr>
<td>COTS</td>
<td>Commercial Off-The-Shelf</td>
</tr>
<tr>
<td>CPU</td>
<td>Central Processing Unit</td>
</tr>
<tr>
<td>CV</td>
<td>Connected Vehicles</td>
</tr>
<tr>
<td>DMS</td>
<td>Dynamic Message Sign</td>
</tr>
<tr>
<td>DSRC</td>
<td>Dedicated Short Range Communication</td>
</tr>
<tr>
<td>EDACS</td>
<td>Enhanced Digital Access Communication System</td>
</tr>
<tr>
<td>EMDSS</td>
<td>Enhanced Maintenance Decision Support System</td>
</tr>
<tr>
<td>FAA</td>
<td>Federal Aviation Administration</td>
</tr>
<tr>
<td>FCC</td>
<td>Federal Communications Commission</td>
</tr>
<tr>
<td>FHWA</td>
<td>Federal Highway Administration</td>
</tr>
<tr>
<td>GPS</td>
<td>Global Positioning System</td>
</tr>
<tr>
<td>HAR</td>
<td>Highway Advisory Radio</td>
</tr>
<tr>
<td>HTTP</td>
<td>Hypertext Transfer Protocol</td>
</tr>
<tr>
<td>IMO</td>
<td>Integrated Mobile Observations</td>
</tr>
<tr>
<td>IP</td>
<td>Internet Protocol</td>
</tr>
<tr>
<td>ITS</td>
<td>Intelligent Transportation System</td>
</tr>
<tr>
<td>JPO</td>
<td>Joint Program Office</td>
</tr>
<tr>
<td>LDV</td>
<td>Light Duty Vehicle</td>
</tr>
<tr>
<td>MADIS</td>
<td>Meteorological Assimilation Data Ingest System</td>
</tr>
<tr>
<td>Acronym</td>
<td>Full Form</td>
</tr>
<tr>
<td>---------</td>
<td>------------------------------------</td>
</tr>
<tr>
<td>MAW</td>
<td>Motorist Advisory Warning</td>
</tr>
<tr>
<td>MDC</td>
<td>Mobile Data Computer</td>
</tr>
<tr>
<td>MDOT</td>
<td>Michigan Department of Transportation</td>
</tr>
<tr>
<td>MDSS</td>
<td>Maintenance Decision Support System</td>
</tr>
<tr>
<td>mESS</td>
<td>Mobile Environmental Sensor Station</td>
</tr>
<tr>
<td>MMS</td>
<td>Material Management System</td>
</tr>
<tr>
<td>MnDOT</td>
<td>Minnesota Department of Transportation</td>
</tr>
<tr>
<td>NCAR</td>
<td>National Center of Atmospheric Research</td>
</tr>
<tr>
<td>NDEX</td>
<td>Nevada Data Exchange</td>
</tr>
<tr>
<td>NDOT</td>
<td>Nevada Department of Transportation</td>
</tr>
<tr>
<td>NIMO</td>
<td>Nevada Integrated Mobile Observation</td>
</tr>
<tr>
<td>NMEA</td>
<td>National Marine Electronics Association</td>
</tr>
<tr>
<td>NTCIP</td>
<td>National Transportation Communications for ITS Protocol</td>
</tr>
<tr>
<td>OBD</td>
<td>On-Board Diagnostics</td>
</tr>
<tr>
<td>OBU</td>
<td>On-Board Unit</td>
</tr>
<tr>
<td>RITA</td>
<td>Research and Innovative Technology Administration</td>
</tr>
<tr>
<td>RSU</td>
<td>Road Side Unit</td>
</tr>
<tr>
<td>RWIS</td>
<td>Road Weather Information System</td>
</tr>
<tr>
<td>RWMP</td>
<td>Road Weather Management Program</td>
</tr>
<tr>
<td>SAE</td>
<td>Society of Automotive Engineers</td>
</tr>
<tr>
<td>TCP</td>
<td>Transmission Control Protocol</td>
</tr>
<tr>
<td>TMDD</td>
<td>Traffic Management Data Dictionary</td>
</tr>
<tr>
<td>UNR</td>
<td>University of Nevada Reno</td>
</tr>
<tr>
<td>USDOT</td>
<td>United States Department of Transportation</td>
</tr>
<tr>
<td>V2I</td>
<td>Vehicle-to-Infrastructure</td>
</tr>
<tr>
<td>Acronym</td>
<td>Definition</td>
</tr>
<tr>
<td>---------</td>
<td>--------------------------------</td>
</tr>
<tr>
<td>V2V</td>
<td>Vehicle-to-Vehicle</td>
</tr>
<tr>
<td>VDT</td>
<td>Vehicle Data Translator</td>
</tr>
<tr>
<td>VPN</td>
<td>Virtual Private Network</td>
</tr>
<tr>
<td>WSMP</td>
<td>Wave Short Message Protocol</td>
</tr>
</tbody>
</table>
Appendix B. NIMO Phase 3 Configuration Manual v2.0

1. Outline

1.1. Vehicle OBU Configuration

1. Prepare configuration materials including:
   a. Installation computer (Configuration Host) with Live USB stick drive loaded with bootable Linux operating system
   b. Configuration data (IP address assignments for vehicle being installed)
   c. IMO Backplane plate and mounting hardware (screws, nuts, etc.), and all components, parts, to be configured (including cables)
   d. USB drive for file transfer

2. Mount individual components to the IMO backplane and attach power cabling, data cabling, and off-backplane devices (antennas, sensors, etc.)

3. **ADAM DI/O/Counter**: Configure the ADAM over an Ethernet cable using the manufacturer’s Windows utility (Section 3.2).

4. **Comet Weather Sensor**: Configure using Web-based utility (Section 3.3).

5. **Arada OBU**: Configure the Arada via Ethernet cable using the provided instructions (Section 3.4) and interactive script on the Configuration Host (Linux LiveUSB) computer. This will require a firmware update and reboot for the Arada. Follow detailed instructions.

6. **Technologic CPU**: Configure the Technologic CPU over an Ethernet cable using the provided instructions (Section 3.5) and interactive script on the Configuration Host (Linux LiveUSB) computer. This process will require manual transfer of files using a USB stick between the Configuration Host and the Technologic CPU. Follow detailed instructions.

7. **Cradlepoint cell modem/GPS**: This is configured by NDOT personnel. This step mainly consists of confirming GPS functionality and communications between the Cradlepoint and the Technologic CPU (Section 3.6).

8. **RoadWatch surface temperature sensor**: RoadWatch sensors are typically pre-configured by NDOT personnel, and are already installed into most NDOT maintenance vehicles. This step involves adding a RS-232 breakout box so that the Technologic can receive the data, and testing receipt of this data by the Technologic. (Section 3.7)

1.2. DSRC RSU Configuration

1. **Arada RSU**: Arada RSU configuration is nearly identical to the OBU radios. The main difference is in the configuration file that is used for setup. Configure the Arada RSU via Ethernet cable using the provided instructions (Section 3.8, with references to Section 3.4) and interactive script on the Configuration Host (Linux LiveUSB) computer. This process will require a firmware update and reboot for the Arada. Follow detailed instructions.
2. Pre-Configuration Information

2.1 Preparing Equipment/Software for Configuration

1. Host Computer: Many of the steps for configuring new IMO hardware will require the use of a Linux-based host computer with the following characteristics. The need for Linux is based in the use of configuration scripts that are used to expedite much of the setup. For the (single) step of configuring the ADAM device, you will need a Windows-based system.
   a. An Ethernet port on the host computer will be required for most operations through ping and ssh utilities commonly available through Windows/Mac/Linux.
   b. A Serial port on the host computer will be required for initial configuration of the Technologic CPU (typically for first time boot-up only).

2. A computer with the Windows operating system, with an Ethernet port and the .Net framework installed, will also be needed to configure the ADAM hardware (i.e., the setup utility will require this).

2.2 Gather Configuration Information

Network configuration settings (IP addresses, etc.) are shown in Section 4. (check NDOT records if this list is out of date). You will need the set of IP addresses for the vehicle you will be installing into. As the NIMO fleet grows, it will be necessary to refer to separate organizing documents (spreadsheets, etc.) for the Subnet address assignments for each vehicle.

2.3 Obtain IP Address Set for New Vehicle

In addition to Section 4 of this document, the assigned IP addresses for all the devices in the new vehicle will be specified in the master .csv file, which can be read manually using these steps:

1. On the Linux Configuration Host, once booted, locate an icon labeled sdb2 near the bottom of the desktop screen. This is the internal drive that contains the configuration data (including the IP addresses).
2. Single click on this icon. This will mount the sdb2 device and you should see a file manager window open showing this device’s directory.
3. Open a console window by single clicking on the desktop icon (near top). Within this console, you can locate the IP address configuration file at the following:
   \[ cd /mnt/sdb2/installation/ \]
   The configuration file (csv) can be examined using: less IMO_D2_Host_IP_Plan.csv

   It can be edited from the Configuration Host using: nano DSRC_Pilot_v2.csv

2.4 Prepare Configuration Host Computer

Most of the devices used by the NIMO project will require configuration for them to work with the NIMO system. For this purpose, a pre-configured system has been provided on a bootable USB flash drive (USB “thumb drive”). When inserted into a computer (laptop is
recommended), and the computer is rebooted, it will load the Linux operating system (with
desktop) from this USB drive, and you will have access to the installation programs and
configuration information. Note that some computers will need to be booted using special
key combinations for it to boot from the USB drive rather than the internal hard disk. This will
vary from system to system.

2.4.1 OPTIONAL - Duplicating the LiveUSB Linux drive (USB Stick)

You can copy the LiveUSB stick using any utility that will duplicate the entire drive, with all
its volumes intact. It is important that whichever method you use, it must maintain the
USB stick as a bootable device.

The process for doing this using Linux (on a Linux system with suitable RAM and disk
space) is as follows (using the Linux command line).

2.4.1.1 Create Master Copy (if you don’t have one already)

1. After the host system is booted and running, insert the original
LiveUSB stick (the one to be duplicated). The disk should have two
volumes: a 3.2GB volume, and a 13GB volume.

2. Examine the output, looking for mention of the device name for the
Newly inserted USB device.

dmesg
or,
dmesg | grep “sd”

3. Depending on your distribution of Linux, you may also see the USB
device appear on the desktop. The LiveUSB should show TWO
devices on the desktop, such as: /dev/sdb1 and /dev/sdb2. (or
/mnt/sdb1 and /mnt/sdb2). These instructions will assume that the
mounting point is under /dev.

4. Because we want to copy the whole USB device, we will drop the “1”
and “2” from the end, and use /dev/sdb for the path of what we are
copying. This will duplicate everything on the USB stick, (not just an
individual partition like sdb1 or sdb2).

5. Move (change directory) into a directory on the host system where you
can create a 16 GB disk image

cd /home/user/Documents

6. Create a disk image of the liveUSB stick, using the path you found
above (you may be prompted for a password, so execute this with a
suitably high level of security privileges (super user, etc.).)

sudo dd if=/dev/sdb of=IMO_ConfigHost_USB.iso bs=4096

7. This may take a while (15-30 mins). Once it is done, you will have a
16 GB disk image file (.iso) in the directory where you executed the
command. This ISO disk image file can be treated as a “master copy”
of the liveUSB disk image, and can be copied and backed up as any
2.4.1.2 Create Copies of the Master File (USB stick)

1. To copy (duplicate) this disk image onto a new USB stick, you first need to insert the new USB stick into the computer and (as above) determine what device it appears as. For example, it may appear as /dev/sdd. Make note of this.

2. Copy the disk image onto the new USB stick similarly to the above step, but with different input files and output files:

   ```
   sudo dd if=IMO_ConfigHost_USB.iso of=/dev/sdd bs=4096
   ```

3. This may take a while (15-30 minutes), and while running, your system may not show obvious signs that it is still copying the image (only that the prompt has not returned yet). Once you think it may be done (prompt returns), you can try ejecting the USB stick. If it protests that it is still in use, then just let it finish (don’t force eject).

4. Once done, you should be able to re-insert the new (duplicate) liveUSB stick and see that it now comes up with two volumes (sdc1 and sdc2 for example). You are done duplicating it, so eject it, and you should be able to boot directly off this liveUSB stick.

2.4.2 Running LiveUSB on the Configuration Host Computer

Windows-type computers (i.e., non-Mac) can usually be configured to boot from a USB drive by properly adjusting the BIOS settings so that the USB drive is checked for a valid operating system before the internal hard drive is. Within the BIOS settings (or similar pre-boot menu), set the Boot Option Priorities (or something of similar name) for this.

Some laptop manufacturers may also have special key combinations for booting directly to the USB drive (like Mac). For example, on some Dell laptops, hit F12 once you see the Dell logo, and this will go to a “one-time boot” menu.

Once you begin the boot process, if it pauses to ask, you should select the “Puppy” Linux installation, and the bootup will begin, ending in a Linux Desktop. You may see a bunch of “first time boot” stuff that you can skip over.

Once it is booted, you will be logged into default root account.

2.4.3 Notes on adding/removing files on the Configuration Host computer

- Device sdb1 is NOT persistent and shouldn’t be touched. You can destroy the LiveUSB if you are not careful.

- Device sdb2 is persistent. I.e., changing files here will be permanent. If you want to save files, this is where to do it. The whole of sdb2 is persistent.
3. Configuration Procedures

3.1 Overview

This description pertains to configuring new IMO hardware, and will mostly be needed for new vehicle installations, although some parts of this section will relate to adding new RSU sites.

3.2 ADAM

3.2.1 ADAM Physical Connections

- 6” Ethernet cable connected to WAN port on Cradlepoint (Right most port on Cradlepoint).
- 12 V Power connection (to backplane power block)
- Wiper Sensor signal line to DI 8, Ground to Vehicle Ground and Vehicle +12V
- Spreader Speed Sensor signal line to DI 9, Ground to Vehicle Ground and Vehicle +12V
- Material Sensor signal line to DI 10, Ground to Vehicle Ground and Vehicle +12V
- Jumper between +Vs and the Iso Gnd ports.

3.2.2 Prepare Configuration Computer for ADAM Configuration

The configuration utility for the ADAM is a windows-based program available from the manufacturer (download or supplied CD).

1. On the Windows configuration computer: Through Network Settings (or similar), set:
   a. IP Address: 10.0.0.2
   b. Subnet: 255.255.0.0
2. Install Advantech Adam/Apax.NET Utility.
3. Run the utility
4. Look on the left-hand hand side for the Ethernet section and select Ethernet
5. Run Tools>Search Device and you should see a new sub-selection under Ethernet (on LHS) titled Others.
6. Under Others, select the IP address and you should see the port settings. This should be the current IP address for any ADAM devices that were discovered. If the ADAM is new, it will likely find the address of 10.0.0.1, with a subnet like 255.255.0.0.
7. From this panel, under the Network tab, configure the ADAM to have the desired (new) IP address settings for the vehicle backplane’s subnet. For example, for IMO_11, set the ADAM to:
a. ADAM IP address: < >
b. Subnet: < >
c. Default gateway: < > (IP address of the Cradlepoint)

Select the Static option under ”configured”. Press Apply Change button. Once you do this, it will ask you for the ADAM password, which is eight zero characters (00000000).

Once this is done, the configuration computer will no longer be talking to the ADAM, as it will no longer be on the (newly configured) subnet that the ADAM is set up for. So, let’s reset the windows configuration computer’s IP address again...

8. On the Windows configuration computer: Through Network Settings (or similar), set:
   a. IP Address: < > (RESERVED address on IMO_11)
   b. Subnet: < >

3.2.3 Start ADAM Web Configuration Interface

1. Re-run the ADAM configuration program if it is not running. Select Setup\Refresh Serial and Ethernet to update the program’s view of the system.

2. Repeat the search for the ADAM using Tools\Search for Modules to get the ADAM to appear in the LHS of the display. You may need to select Ethernet-Others on the LHS to get the refresh to take place.

3. At this point, you should see the current IP configuration of the ADAM (which should agree with what was set earlier. Except now, because it is found under the same subnet, it will no longer appear under Others. It will be sorted under the IP address of the host computer (but with its own IP).

4. Select the device’s IP address and the main window will show many tabs with information for the ADAM’s configuration. Select the 6051 that appears under the IP address. It will ask for the password, so enter it (00000000). This will reveal the devices channels.

3.2.4 Configuration of the ADAM’s channels

Select the (+) sign on the LHS near the 6051 and it will expand the dropdown list to show the available channels on the ADAM. We are using Channels 8-10.

Select Channel 8 (Windshield Wiper).

Change the DI mode to Counter. Press Apply Mode, then OK.

Select the Invert Signal box and press Apply to All.
Ensure that “Keep last value...” and “Enable digital filter” are unchecked. Counter value should be set to 0 times.

Select the “6051” on the LHS and you should go back to the overview display. Now all the DI 8 indicators should be un-selected, indicating that the signal is logic zero.

Select Channel 9 (Spreader).

Change the DI mode to Frequency. Press Apply Mode, then OK.

The other settings should disappear.

Select Channel 10 (Material sensor).

Change the DI mode to DI. Press Apply Mode, then OK (note that it may already be in DI mode).

The other settings should remain as the defaults.

3.2.5 Confirm Functionality of ADAM

Confirm ADAM is reading channels properly: Attach all three of the ADAM input channels and monitor the outputs using the windows configuration utility. The current value of each channel is shown at the bottom of the display if you select each of the DI-8, DI-9, DI-10 (etc.). The value will update live to confirm the ADAM is functioning.

Confirm data is being logged properly: Full functioning of this device is best confirmed after the Technologic CPU has been configured (Section 3.5.6).

Change user name and password: Once ADAM functionality is confirmed, replace default password and user name.

3.3 Comet

The Comet weather sensor is configured using the host computer through the Ethernet port, using a common web browser. No special software is needed.

3.3.1 Comet Physical Mounting

The Comet display unit (main unit) is mounted to the backplane with the sensor head cable facing the edge of the backplane.

The sensor head is mounted on the exterior of the vehicle, inside secondary enclosure that is attached to the passenger side mirror’s vertical support tube as shown in Figure 26. The secondary enclosure should have numerous small holes drilled through the sides and bottom to permit air to flow into the enclosure (especially when the vehicle is moving), and any moisture to drain out the bottom.

Source: UNR, January 26, 2018

Figure 26. Comet Physical Mounting.
3.3.2 Comet Physical Connections

Once the Comet has been configured through a direct connection to the host computer, it can be detached, and interfaced into the normal IMO backplane connections as shown in Figure 27. This will include:

1. 12 V Power connection (to backplane power block)

2. 18” Ethernet cable from Comet to the Cradlepoint’s LAN 1 port.

3.3.3 OPTIONAL - Resetting the Comet to Factory Defaults

A brand-new Comet should not require this step. If the Comet has previously been configured with a non-default IP address, this procedure can be used to restore it to factory settings including network parameters (IP address, Subnet mask, etc.). Settings relating to measurement are not restored by factory defaults. For factory-defaults follow these steps listed per the Figure 28:

1. disconnect the power supply
2. unscrew upper cover of the device case
3. close the jumper inside device
4. press button inside device and power on device at same time
5. keep the button pressed for 10 secs
6. re-open the jumper inside device
7. close the device

3.3.4 Prepare the Ethernet Connection

Connect the host computer directly to the Comet using an Ethernet cable. The host computer should be configured with an IP address that will be on the same subnet as the device. For an “out-of-the-box” new Comet (or one set to factory defaults), this will work:

Set the Configuration Host (laptop) IP address:

```
ifconfig eth0 192.168.1.1 up
ifconfig eth0 netmask 255.255.0.0
```
3.3.5 Start Comet Web Configuration Interface

You will need to know the Comet's current (or default) IP address, which is typically (out of the box) 192.168.1.213 or .211 or .212. You can check which one it is using by:

`ping 192.168.1.213` (or .211 or .212)

Once you find the current IP address, enter it into the address bar of a web browser (as the URL), and it should load up the web configuration utility directly in the browser window.

If you have trouble at this stage (such as can't ping it or connect via the web configuration utility), consider a hard reset of the Comet, to force it back to its factory default IP configuration (below).

3.3.6 Set Comet IP Address

Within the web configuration utility, click the Settings button, then select Network. From here, configure it (example for vehicle IMO #2):

- Obtain an IP address automatically: Off
- IP address: `< >`
- Default Gateway: `< >` (IP address of the Cradlepoint in this vehicle)
- DNS server IP: 0.0.0.0
- Standard subnet mask: unchecked, and then use `< >`
- Periodic restart interval: off
- Then Apply Settings

You can now exit the browser and restart the Comet. It will be using the new IP address now.

3.3.7 Set Comet Measurement Units

NOTE: changing the units (if required) involves temporarily closing a jumper inside the Comet while using the web configuration utility.

1. disconnect the power supply
2. unscrew upper cover of the device case
3. close the jumper inside device
4. Power on the device
5. Within the web configuration utility (see above), click the Settings button, then select Measuring. From here, select the units for the below.

   **Temperature:** °C
   **Computed Unit:** Dew Point
Atmospheric Pressure Unit: kPa

6. Power off the device

7. re-open the jumper inside device

8. close the device

9. Power on the device (for normal operation). You can also use the web utility to confirm the desired units are being used.

3.3.8 Confirm Functionality of Comet

1. Configure the Configuration Host test laptop to work on the IMO backplane subnet. Do this by setting the laptop’s IP address to the last available address in the subnet range for the vehicle (labeled RESERVED). For example, for vehicle IMO_11, use < > for the laptop’s IP address.

2. Confirm Comet is reading channels properly: From a web browser window on the laptop, enter the IP address of the Comet. For example, for IMO_11, use < >. This should open the same web configuration utility as above. The weather data should be updating live.

3. Confirm data is being logged properly: Full functioning of this device is best confirmed after the Technologic CPU has been configured (Section 3.5.6).

3.4 Arada DSRC On Board Unit (OBU)

The Arada setup process is (preferably) handled through a semi-automated script run from the NIMO Configuration Host computer (i.e., the system run from the bootable USB stick).

3.4.1 Arada DSRC Physical Mounting

The Arada is mounted onto the backplane with the antenna ports facing the outside edge of the backplane to permit antenna cable access.

3.4.2 Arada DSRC Physical Connections

The connections to the Arada are as follows:

1. 12 V Power connection (to backplane power block)

2. 2 antenna cables from the 5-in-1 antenna with RP-SMA connectors connected to the Arada’s DSRC ANT 1 and DSRC ANT 2 ports.

3. 7” Ethernet cable from the Arada’s WAN port to the Technologic’s ETH-A port.

4. Approximately 60” Ethernet cable attached to the Arada’s LAN1 port, and left neatly coiled and tucked away, but easily accessible. This will be used for initial setup, as well as for field-servicing of the IMO components using a laptop without removal of the plow’s seat (etc.).
3.4.3 Host Computer Configuration

You need to use the IMO configuration system as the host computer for this step (not just any laptop) because we will need to copy files from the configuration host to the Arada.

3.4.4 OPTIONAL - Resetting the Arada to factory defaults

**OPTIONAL:** If the Arada is not a new (out of the box) device, it may have already had its IP address changed. You can try pinging it from the Configuration Host computer at the default address of 192.168.0.40.

```
root@HostComputer:~# ifconfig eth0 192.168.0.1
root@HostComputer:~# ifconfig eth0 netmask 255.255.255.0
```

Note 1: This puts the Configuration Host computer onto the same subnet as the Arada.

Note 2: Depending on what computer (laptop) you are using to run the Linux USB stick, the laptop’s Ethernet port may be configured as `eth0` or `eth1` (etc.). You may need to replace this in the above command.

Note 3: The cursor prompt may differ on your system. The important part is usually the command itself (shown in blue)

```
root@HostComputer:~# ifconfig eth0 (confirm above address was set)
root@HostComputer:~# ping 192.168.0.40 (or .41 etc.)
```

If you don’t see a response, then you can restore the Arada’s factory configuration as follows:

1. Power up the Arada. It has finished booting when the top and bottom LEDs are solid on.

2. Press and hold the small reset button on the Arada (with a small screwdriver etc.) until you see the bottom green LED (power) flash rapidly (5-10 seconds), then release the button.

3. This should reset the IP address to the factory default and reboot the Arada.

4. The reboot of the Arada has completed when the top and bottom LEDs are solid on (not flashing). Confirm that it is using the default IP address by pinging it (as described above).

3.4.5 Setting up the Configuration Host computer

Connect the Configuration Host computer’s Ethernet port (such as `eth0`) to LAN1 of the Arada.

“Out of the box”, the Arada will typically have an IP address of **192.168.0.40 (or .41)**. We will usually need to start with this factory default IP address and then change it to the IMO desired address. If necessary, see the above section on Resetting the Arada to factory defaults and do that first.
Appendix B. NIMO Phase 3 Configuration Manual v2.0

root@HostComputer:~# ifconfig eth0 192.168.0.1
root@HostComputer:~# ifconfig eth0 netmask 255.255.255.0

This puts the Configuration Host computer onto the same subnet as the Arada’s default IP address.

Confirm the new IP address setting using:
root@HostComputer:~# ifconfig eth0

Confirm the connection to the Arada using:
root@HostComputer:~# ping 192.168.0.40 (or .41 etc.)

3.4.6 Acquire an Up-To-Date Copy of the IP Configuration Plan

Before you proceed with configuring the Arada, ensure that the host computer has an up-to-date copy of the IMO IP configuration plan .csv file and that the file version you are using includes the up-to-date vehicle information for the configuration you are doing. See the above Section 2.1.2 for details. Summarizing, the configuration .csv file should be up-to-date with all the vehicles and current IP address plan. This file is found at:

/mnt/sdb2/installation/ IMO_D2_Host_IP_Plan.csv

3.4.7 Run the OBU Configuration Setup Shell Script

Note: The easiest way to configure the Arada is by running the below script. In the event that this does not work, it is also possible to use the Command Line Interface on the Arada, following the script source as a guide. This is for advanced users only, though.

On the Linux Configuration Host, locate an icon labeled sdb2 near the bottom of the desktop screen. This is the internal drive that contains the installation files.

Single click on this icon. This will mount the sdb2 device.

In the console window, change into the Configuration Host computer’s IMO installation directory:

cd /mnt/sdb2/installation/

Run the OBU setup shell script (which should be in this directory), following the prompts

./obu_install.sh

The shell script will first ask you for the vehicle number. Type it in (for example, for vehicle 11)

Site#: 11

Note that you may see a message saying that the Host is not in the trusted hosts file, and asks if you want to continue. Answer YES (y).
You will then see a report of the configuration information that it found for the Arada OBU. Something along the lines of the below. Just ensure nothing looks like clear signs that the data was not found properly in the .csv file (note that these IP addresses below are not typical of the IMO vehicles and are used here only as an example).

Selected site: mESS 11 OBU
id=11; gw=192.168.1.201; nm=255.255.255.248; ip=192.168.1.205; host=D2-mESS-11-DSRC; d=mESS 11 OBU

Press ctrl-c to abort or <enter> to continue

It is important to confirm the IP address and netmask are present and correct. If there is a problem reading the configuration file (such as an incorrectly formatted file), one symptom is that there will not be any IP addresses showing (see gw=, nm=, ip= portions of above output).

After pinging the Arada, the script will require that you enter the default password, which (initially) will simply be password. The script dialog will tell you this.

Typing Shortcut:

You are going to have to type the password a lot (gets tedious). A shortcut is to put the password into the clipboard, and then paste from there.

1. In a separate window on the Configuration Host (a console window works fine), type (literally) the password (password). Then use the mouse to select it, and then copy it to the clipboard using CTRL+Insert.
2. Now, when you go back to the Arada installation script (running in another console window), you can paste the password from the clipboard by pressing SHFT+Insert.

Proceed through the script, pasting the password (which is password at this point) when requested. This password should be requested maybe a dozen (or so) times, and there will be instances where it asks you to continue (press enter).

The script will be re-setting a variety of settings in the firmware of the Arada, including the IP address and the admin password, which will take effect once the new configuration files have been uploaded, the firmware is updated, and the Arada is restarted.

The script will arrive at a point where the new firmware needs to be loaded, which requires a bit more user actions (not just pasting the password). The dialog text will change a bit at this point. Look for something like:

The next several steps will require manual entry of commands by the operator.

1. Below, enter the password one more time.

2. At prompt, enter the Command Line Interface (CLI) by typing “cli” and <enter>.
3. Once in the cli, the cursor prompt will change. Run the command “apphalt”.

4. Then type : exit <enter>" to leave the cli

5. Then, from the prompt, run the below (long) command

   cd /tmp; nohup /usr/local/bin/firmware-upgrade-file WAVE_LOCOM

   Following the on-screen instructions, enter the password (password) again

   root@192.168.0.40’s password: password

   The script should now be showing a command line, and you should first run
   the command line interface (cli) and then (from within the cli environment) type
   'apphalt':

   root@Arada01F9F0 /root]# cli

   Arada01F9F0# apphalt

   You should then see a few messages of the things being terminated. When it
   stops, exit from the cli environment using 'exit', which will bring you back to the
   root prompt.

   Arada01F9F0# exit

   Use  CTRL+Insert to copy the long command text to the clipboard and then
   SHFT+Insert to paste it onto the command line. **NOTE: THIS WILL REBOOT
   THE ARADA AND UPDATE ITS FIRMWARE, WHICH TAKES A WHILE.**

   root@Arada01F9F0 /root]# cd /tmp; nohup /usr/local/bin/firmware-upgrade-
   file WAVE_LOCOMATE-200_1.90.0.13_detroit_firmware.tar > /tmp/a.out

   **Wait** until the bottom LED on the Arada stops flashing and a while more for the top
   and bottom LEDs to be solid ON. This signals that the reboot is complete.

   Note that in your console window, you may get a message like: **“ssh: Exited: Error
   connecting: Network is unreachable”**. This just means that the Ethernet link to
   the Arada has gone down with the reboot. Once the bottom LED on the Arada
   stops flashing, the firmware has been updated and the Arada has rebooted.

   Once the Arada fully re-boots, in a **separate** terminal window (not the one that is
   running the setup script) on the host computer, change the host’s IP address to
   something on the same subnet as the Arada will be once it reboots. It is
   recommended that you use the reserved IP address (last in the range assigned
   for the vehicle) for the new Configuration Host IP address. For working on the
   system for IMO_11, this would be

   root@HostComputer:~# ifconfig eth0 <               >

   root@HostComputer:~# ifconfig eth0 netmask <               >

   root@HostComputer:~# ifconfig eth0  (review changes to confirm)
Now the configuration host will be on the same subnet as the Arada (which if we are setting up IMO_11, is now using (after it reboots) the IP address < >

Now, back in the window with the setup script running, go ahead and “press enter to continue”. The dialog will tell you that the new Arada password has been copied to the clipboard for your convenience. It also gives you some suggested commands for checking the setup.

To paste the password from the clipboard, use Shift+Insert.

root@192.168.1.205’s password: < > [or, paste from clipboard with SHFT+Insert]

At this point, you are done with the configuration and can test it.

1. You should still be ssh’d into the Arada. In case it is already running, stop the transmitter script:
/var/NIMOv3/stop.sh

2. Test the start script:
/var/NIMOv3/start.sh
Normally, start.sh will run on boot of the Arada. But you should test it now as well. Once run, you should see output on the terminal at the same time as you are looking at the prompt. It will look messy. The normal way of stopping something is with Ctrl-C, but that doesn’t work because the script is running in the background. To stop the script, you need to type /var/NIMOv3/stop.sh <return> while disregarding what you see on the screen because what you type gets merged with the running script’s output. But when you then hit <return> it should stop the background script.

3. Perform a clean shutdown of the Arada
poweroff

4. After this clean shutdown is done (this once), the Arada becomes more robust for handling harsh shutdowns. The clean shutdown should be done every time you change something in the /var partition as a matter of good practice.

5. If you wish to later log into the Arada, you can do it using ssh from a host on the same subnet (i.e., set the host IP up as described above). For example, to log into IMO_11:
ssh root@< > password: < >

3.4.8 Confirm Functionality of Arada

Confirming functionality of the Arada DSRC radio is more involved than most devices on the OBU backplane, as it required that the vehicle be within DSRC communications distance from an RSU.

1. **Confirm Arada is configured properly:** Successful ssh into the Arada (see above) provides evidence of the correct IP configuration for this device.
2. **Confirm data is being sent properly over Arada radio:** Full functioning of the radio is best confirmed by examining vehicle data received at the back end, after the vehicle has entered a DSRC geo-fence zone (i.e., has been near an RSU).

### 3.5 Technologic CPU (Linux Box)

#### 3.5.1 Prepare Host Configuration System

##### 3.5.1.1 Technologic – Additional Components Needed

**microSD Card:** The Technologic Linux Box will require a microSD memory card (4GB minimum) for the operating system (permanently installed). This can be either purchased from Technologic (**highly preferred**) at the time of purchase of the motherboard, or you can make your own using the instructions and downloads from Technologic.

If you did not purchase the microSD card from the same supplier as the CPU box (Technologic), you need to prepare your own microSD card, follow these steps.

1. Procure a microSD card with at least 4 GB.

**USB Memory Stick (drive):** The Technologic configuration process will also temporarily require a spare USB stick (4 GB should be large enough) for performing file transfers during the initial setup process. Formatting the USB drive as FAT32 works fine.

**Null Modem Serial Cable:** The configuration process will require serial communications from the Configuration Host (laptop) to the Technologic box. If the laptop has a serial port, you will need a **DB9F-to-DB9F Null Modem cable** (or equivalent serial cable with null modem adapter and appropriate genders). If there is no serial port on your Configuration Host computer, you will also need to use a **USB-to-Serial adapter** along with a null modem cable of appropriate genders.

##### 3.5.1.2 Interfacing to the Technologic during Configuration

Out of the box, the Technologic is configured to be interfaced with through **COM1**. We will work to switch this over to an Ethernet port early in the configuration process (more convenient). The Technologic’s data cabling should be as follows:

- Technologic **COM1** to Configuration Host (Linux laptop) serial port (or USB via a suitable adapter)
- Technologic **COM2** to Roadwatch sensor serial cable
- Technologic **COM3** to TBD (planned for in-cabin lights from spreader)
- Technologic **ETH-A** to Cradlepoint **LAN2**
- Technologic **ETH-B** to the Arada **WAN**.
5.
If you purchased the Technologic to come with the pre-loaded bootable microSD card, then just insert it into its designated spot on the motherboard along the back edge.

3.5.2 Technologic CPU Physical Connections
The physical connections to the Technologic CPU (Linux Box) are as follows:
1. 12 V Power connection (to backplane power block)
2. 15' Serial cable (DB-9 connectors) from Roadwatch RS232 Adapter Box to the Technologic's front panel **COM2** port. See note below on legacy systems and com ports.
3. 7" Ethernet cable from the Arada’s **WAN** port to the Technologic’s **ETH-A** port.
4. 12" Ethernet cable from the Cradlepoint’s **LAN 2** port to the Technologic’s **ETH-B** port.

**Note:** Some of the legacy systems (earlier IMO vehicles) have the Roadwatch attached to COM3. It is possible that there was an internal cabling difference inside the Technologic that could explain this (i.e., it was labeled COM3 on the outside of the Technologic box, but was internally configured as COM2).

3.5.3 Technologic Internal Ribbon Cable Connections
The internal serial port ribbon cable needs to be installed from COM2 on the main board to COM2 on the daughter board behind the front panel of the Technologic shown in Figure 29.

![Figure 29. Technologic Internal Ribbon Cable Connections.](source: UNR, January 26, 2018)

3.5.4 **OPTIONAL** – Re-Configuring a pre-existing Technologic Linux system after resetting it to factory defaults.
If you are using a brand-new Technologic (and microSD) that has never been configured before, then you can skip this section and proceed to Section 3.5.5 (Technologic Configuration). Otherwise, follow the below to reset the Technologic to factory defaults.

**Resetting the Technologic to factory defaults – OPTIONAL**

1. Note that the Technologic manufacturer’s website has a wiki page that describes the initial steps for virgin power-up. A brand-new Technologic CPU will require many things to be done to it. If it is NOT brand new, you can reset it to factory defaults with the below steps.

2. You will be interacting with the Technologic CPU from the Configuration Host (laptop) using the Null Modem serial cable. Attach the cable from COM1 of the Technologic to the Configuration Host computer. You may need to use a USB-to-Serial converter to interface to the host. The technologic should NOT be powered on yet.

3. Check to see what the Configuration Host has for serial ports

   `dmesg | grep tty`

   If the Configuration Host computer has a built-in serial port, then most likely, it will have `ttyS0` available (which you will access as `/dev/ttyS0`), which is the default. Note that if you are using a USB-serial adapter, you may see a different port such as `/dev/ttyUSB0`. You can Google minicom for instructions.

4. Open a new terminal window on the Configuration Host, and from the command line, run:

   `minicom 115200`

   This will start the program using `/dev/ttyS0` with the default 115200 baud N811 settings. But because the Technologic is not powered on yet, the program will simply start (open a window) and wait.

5. Follow these steps to reset the Technologic to Factory Default settings:

   1. With power disconnected, remove jumper 3 (JP 3) from the Technologic motherboard shown in Figure 30.

---

2. Remove the SD card (found along back edge of the motherboard).

3. Boot it up to the serial port attached to COM1 of the Technologic box (monitoring from the Configuration Host laptop with minicom again). Plug the Technologic power in and monitor the minicom terminal window for boot messages. The initial boot messages will start appearing quickly. The complete boot may take a minute or so, ending with a login prompt. If you are resetting a system that has already been set up before, use the current system login (i.e., not the factory default password). The default account is root.

   ts7250-4ce699 login: root

   It may not ask for a password, but if it does, the default is Ha3mUMsh. You can also find the default root password in the text file on the Configuration Host laptop:
   /mnt/sdb2/installation/TECHNOLOGIC_INSTALL/tl_pass.txt

4. These instructions will set the software switch following the Technologic Wiki under section 2.3 (busybox). Set soft jumper 1 to revert to initramfs. Do this from the prompt after booting by entering:

   tshwctl --removejp=1

   poweroff

   Note that this will reboot it, and not actually power off. So, let it gracefully shut down (watch status in minicom window) and when it starts to reboot, manually remove power from the Technologic.

5. With the Technologic powered off, replace the SD card and Jumper 3.

6. Turn the power back on, it should do a "minimal boot" that takes only a couple of seconds. The very fast boot is the sign that the factory reset has been properly done. After this booting, your prompt will be a # sign. At this point, you will be logged into a very limited version of Linux. Type:

   exit

   This will initiate a full reboot, which (this time) will end in a normal login prompt where you can log into the root account:

   ts7250-4ce699 login: root

   Password: Ha3mUMsh
7. If this is not a new installation, there will be numerous files in the /root
directory. You should delete them, or else they will cause problems upon
later startup attempts (it can go into an infinite loop, trying to gather IMO
data).

```
mount -o remount,rw /  # remount the drive as read/write
rm -Rf *  # erase normal files in the current directory, /root
reboot  # type this at the prompt for a soft reboot
```

8. After this reboot, it will again end up at the # prompt. As earlier, proceed
with:

```
exit  # This will initiate a reboot, which (this time) will end in a
normal login prompt where you can log into the root account:
```

```
ts7250-4ce699 login: root
Password: Ha3mUMsh
```

9. Now, you can safely go ahead and shut it down gracefully

```
poweroff
```

Wait for boot to start, then remove power manually.

10. The system is now fully reset to factory defaults and you can proceed with
the normal setup process (below).

3.5.5 Technologic Configuration

At this juncture, the configuration process is assuming that you are starting with a
Technologic in "factory fresh" or "factory default" state. If this is not the case, see
above instructions on resetting the Technologic.

- The Arada should be configured FIRST. Don’t proceed if that wasn’t done
  yet.
- Do not (yet) power up the Technologic.

3.5.5.1 Setting up your Host Computer (Laptop)

1. You will be interacting with the Technologic CPU from the
Configuration Host using the Null Modem serial cable. Attach the
cable from COM1 of the Technologic to the Configuration Host
computer. You may need to use a USB-to-Serial converter to interface
to the host. The Technologic should NOT be powered on yet.

2. Check to see what the Configuration Host has for serial ports

```
dmesg | grep tty
```

If the Configuration Host computer has a built-in serial port, then
most likely, it will have ttyS0 available (which you will access as
/dev/ttyS0), which is the default. Note that if you are using a
USB-serial adapter, you may see a different port such as
/dev/ttyUSB0. You can Google minicom for instructions.
3. Open a new terminal window on the Configuration Host, and from the command line, run:

   `minicom 115200`

   This will start the program using `/dev/ttyS0` with the default 115200 baud N81I settings. But because the Technologic is not powered on yet, the program will simply start by opening a new window and waiting.

4. Note which vehicle unit number you are working with (in this example, we are using 11)

5. From the `TECHNOLOGIC_INSTALL` directory on the Configuration Host computer, locate the `README` file and open it to begin the install process (see details below). **Follow the README file instructions:** Before running the installation script, you will need to:

   ```
   cd /mnt/sdb2/installation/TECHNOLOGIC_INSTALL
   less README
   ```

6. Prepare USB drive (thumb drive) for transferring files:

   a. Locate a USB flash drive that can be used for porting files to the new Technologic. Network transfers don’t work (yet), so you will need a USB drive (or thumb drive). Formatting the USB drive as FAT32 works fine. Other file systems may work fine too.

   b. **Insert the fresh USB drive (NOT the LiveUSB drive that you are running Linux from)** into the Configuration Host laptop, you should see sdc1 appear on the desktop. **Click on it to mount this new USB drive (it may take a few seconds for it to mount and open the window).**

7. On the **Configuration Host** computer, in the console window, set environment variables (right at the command prompt). For vehicle IMO_11, and assuming the transfer USB drive is sdc1, and the serial port is ttyS0 type:

   ```
   UNIT_NO=11   (this will vary with vehicle)
   EXT_DRV_MP=/mnt/sdc1/
   MINICOM_PORT=/dev/ttyS0
   TL_IP=<      > (this will vary with vehicle)
   ```
You can (optionally) confirm any of these using (for example)
```
  echo $UNIT_NO
```

8. On the Configuration Host computer, 
   (/mnt/sdb2/installation/TECHNOLOGIC_INSTALL), run the script 
   `gen_tl_cfgs.sh`. For vehicle 11, this would be: 
   ```
   cd /mnt/sdb2/installation/TECHNOLOGIC_INSTALL
   ./gen_tl_cfgs.sh $UNIT_NO
   ```

9. It will let you know what settings it will change (confirm) such as the 
   various IP address info. Confirm the values (such as IP addresses) 
   are correct for the vehicle you are using.

10. The script will also generate several configuration files within the 
    `configs` subdirectory that will be transferred (via USB drive) onto the 
    Technologic. The files will be placed into the directory you are running 
    the script from (`configs`). Note that the `configs` directory will have the 
    files for EACH of the NIMO vehicles. The files (human readable and 
    editable) are:

    a. Network Configuration Info: (example 
       `tl11__etc_network_interfaces`)

    b. Master Configuration file for IMO application on the 
       Technologic: (example: `tl11_imo_configs.py`). This 
       includes all the various configuration information for the 
       devices on the backplane, geofencing definitions for RSUs, 
       etc.

    c. Local Startup Script for the Technologic bootup: (example: 
       `tl11_local.sh`)

11. Insert a temporary USB drive into the Configuration Host computer, 
    and ensure it is mounted (it should appear as `sdc1`)

12. Copy over the TECHNOLOGIC_INSTALL directory onto the USB 
    drive. This is easiest to do using copy (CTRL+Insert) and paste  
    (SHFT+Insert) of this long command from the README file (it's a 
    long, compound command).

    ```
    cp –av ‘./TECHNOLOGIC_INSTALL’ ‘${EXT_DRV_MP}/’ 
    for f in _etc_network_interfaces imo_configs.py local.sh; do 
        cp –v “configs/tl${UNIT_NO}.$f” “${EXT_DRV_MP}/TECHNOLOGIC_INSTALL/${f}” 
    done
    ```

    Then press <enter>
13. Unmount the USB from the Configuration Host (again, copy and paste the command from the README).

\[ \text{umount } \text{EXT_DRV_MP} \]

14. The USB drive should now be ready and unmounted. You can go ahead and unplug it from the Configuration Host. You will use this to transfer files to the new Technologic.

15. Next, you will be interacting with the Technologic CPU from the Configuration Host using \text{minicom} through the \text{null modem serial cable}. Refer to earlier steps on attaching this. The Technologic should NOT be powered on yet.

16. \textbf{Power up the Technologic} and monitor the \text{minicom} window for boot messages. The initial messages will start quickly. Observe for problems. You are good to go if the boot took only a second or so and you see a # prompt in the serial terminal. If you DON’T get this fast, minimal boot that leads to the # prompt, then the system needs to be reset to factory defaults. See the above section for details on how to do this.

17. Once you are at the # prompt, \textbf{resume following the README instructions} with the instructions to take place on the TL (Technologic)

a. \textbf{Plug in the USB drive} into one of the USB ports on the Technologic. AVOID BOOTING WITH THE USB DRIVE INSERTED (it may try to boot off the USB drive).

b. \texttt{dmesg | tail} to determine what the thumb drive’s device name is. For example, it will probably come up as \texttt{sda}. (the following assumes this device name).

c. Mount the USB drive using the command/path in the README file.

\[ \text{mount } -rt \text{ vfat } /dev/sda1 /mnt/root/mnt/usbdev \]

d. Continue the README instructions to re-mount the root directory as read/write.

\[ \text{mount } -o \text{ remount, rw } /mnt/root/ \]

e. Continue README instructions to copy the Technologic install scripts to your local directory on the TL.

\[ \text{cp } /mnt/root/mnt/usbdev/TECHNOLOGIC_INSTALL/install[12].sh /mnt/root/root/ \]
f. Continue README instructions to make the install program executable (chmod).

   chmod +x /mnt/root/install[12].sh

g. Continue README instructions to execute /root/install1.sh

   /mnt/root/root/install1.sh

h. Follow the on-screen instructions by typing in reboot. It will take about a minute to reboot to the full Linux kernel.

   reboot

i. After reboot, when it comes up, log in using:

   login credentials: root, Ha3mUMsh

j. Before proceeding with install2.sh, re-ensure that the drive is read/write:

   mount -o remount,rw /

   (do this again)

k. Continue README instructions on the Technologic to execute /root/install2.sh

   /root/install2.sh

l. It will install a bunch of stuff and arrive at a prompt asking for a new UNIX password. Enter Ha3mUMsh (twice). This should be shown on the screen one line above the prompt (you can copy paste if you want).

m. Once the password is updated, it will do a few more steps and then prompt you to enter reboot. Do this.

n. After this second reboot, the Technologic will attempt to find the Cradlepoint, but it may not (yet) have the proper IP address for this. So, it will go into an infinite loop that you cannot interrupt through the serial terminal. We can side-step this by accessing the Technologic (after the re-boot) using the Ethernet (via ssh). The README file should say something to this effect. Either way (whether it finds the Cradlepoint) we will proceed via ssh.

o. Set network settings on the Configuration Host (for example for IMO_11) using:

   ifconfig eth0 < >

   ifconfig eth0 netmask < >

   ifconfig eth0 (confirm the settings)
p. Confirm you can ping the Technologic from the Configuration Host:

```
  ping <           >
```

q. `ssh` into the TL using the password presented above, and you should get a prompt.

```
  ssh root@<           > (password is Ha3mUMsh)
```

Note that if the IP is not “new” to the host system, it may give you a host key error. You can remove the Technologic’s IP address from the known_hosts file manually, and this should permit you to try again with the ssh (select “y” to continue anyway when it asks).

r. You can now quit the `minicom` session (Ctrl-A followed by z will bring up a menu. Select x and confirm with “yes”). You can now unplug the serial cable from the TL and Configuration Host.

s. Back in the ssh window, verify that the application is running on the TL. The simplest way to do this is to look at the log for the data collection python script:

```
  tail -f /tmp/collect.py.log
```

You should see things like GPS sentences, messages about telemetry, etc. If GPS does not have signal, you will only see the instrument data (until it acquires signal). **You should NOT see things like “error” or “exception”, or the output simply stopping.**

Note: occasionally, if a Cradlepoint is not configured correctly, the GPS port will not be configured. If this happens, collect.py will attempt to read the GPS once, and then abort running (i.e., when there is no network connection). In this case, collect.py.log will have just one entry, and the last lines will say “No CP NMEA socket to read. Nothing to do. Closing”. This problem with the GPS port configuration can be confirmed by using netcat (see Cradlepoint section).

t. `ps –A` should show a list of processes. You should see `collect.py` as one of the running processes. This is the main data collection looping routine.

u. `./stop.sh` will stop the script (if you want)

v. `./start.sh` will start it (if you want)
w. exit from the ssh session to the Technologic (from the console window on the Configuration Host).

x. Remove the power from the Technologic, and then remove the temporary USB stick that was used to transfer the files between the Configuration Host and the Technologic.

3.5.6 Confirming Functionality of Technologic

3.5.6.1 Logging into Technologic via ssh:

Set network settings on the Configuration Host (for example for IMO_11) using:

```bash
ifconfig eth0 < >
ifconfig eth0 netmask < >
```

Confirm you can ping the Technologic from the Configuration Host:

```bash
ping < >
```

ssh into the TL using the password, and you should get a system prompt.

```bash
ssh root@< > (password is Ha3mUMsh)
```

Once you have logged into the Technologic, you can make configuration changes and monitor the data logging process.

3.5.6.2 Monitoring Live Data Stream on the Technologic CPU

Verify that the main logging application is running and collecting data by looking at the log file for the data collection python script:

```bash
tail -f /tmp/collect.py.log
```

The tail command will give you a continuously-updating view of the end of the log file (i.e., the most recent data). You should see things like GPS sentences, messages about telemetry, etc. If GPS does not have signal, you will only see the instrument data (until it acquires signal). You should NOT see things like “error” or “exception”, or the output simply stopping.

Note that if some sensors or devices were not present when the data collection (logging) program started, those devices may not appear in the stream even after you plug them in. (i.e., you may need to restart the Technologic so that it recognizes newly-attached devices.)

3.5.6.3 Examine/Edit Vehicle Configuration Data on the Technologic CPU
You can examine (or manually edit using \texttt{nano}) the system configuration for the vehicle. This includes the geofencing configuration.

\texttt{less /root/src/imo_configs.py}

Maps\textsuperscript{21} of daily OBU activity

3.6 Cradlepoint

The Cradlepoint is normally configured by NDOT’s IT staff, using web-based configuration software. The key parameters that need to be set are:

1. IP Address
2. Gateway configuration setup (the Cradlepoint will act as the gateway). Not sure what this entails, such as subnet, mask, etc.
3. Tunneling protocol
4. Physical port settings for LAN 1, LAN 2, and WAN ports
5. GPS settings

3.6.1 Cradlepoint Physical Mounting

The Cradlepoint module is mounted on the backplane, with the antenna connectors facing the edge to permit antenna cable access.

3.6.2 Cradlepoint Physical Connections

The Cradlepoint is mounted directly onto the IMO Backplane with the following connections:

1. 12 V Power connection (to backplane power block)
2. 2 antenna cables with SMA connectors connected to the Cradlepoint’s two 3G/4G antenna ports.
3. 1 antenna cable with an SMA connector attached to the Cradlepoint’s GPS port.
4. 18” Ethernet cable from the Cradlepoint’s \texttt{LAN 1} port to the Comet.
5. 12” Ethernet cable from the Cradlepoint’s \texttt{LAN 2} port to the Technologic’s \texttt{ETH-B} port.
6. 7” Ethernet cable from the Cradlepoint’s \texttt{WAN} port to the ADAM.

3.6.3 Confirm Functionality of Cradlepoint Cellular Modem

\textsuperscript{21} See NIMO Maps at \url{http://mess.its.nv.gov/maps4/}
1. Power up the Cradlepoint and give it time to fully boot up.

2. Ensure that the cellular and GPS antennas are attached (the cellular connectivity test should work without GPS signal though). The lights on the Cradlepoint should indicate that the network link is established (solid green LED light).

3. Configure the Configuration Host test laptop to work on the IMO backplane subnet. Do this by setting the laptop’s IP address to the last available address in the subnet range for the vehicle (labeled RESERVED). For example, for vehicle IMO_11, use <xxxx> for the laptop’s IP address.

4. **Confirm Cradlepoint GPS is receiving NMEA sentences:** From the Configuration Host (or at least behind the NDOT firewall), using the netcat application, type (for example, using the Cradlepoint IP address for vehicle IMO_11)

   ```
   nc < >
   ```

and monitor the responses. This means “listen to port 1234 on the Cradlepoint (IP address) for the GPS stream”. If you have a GPS signal, you should see NMEA sentences once every 10 seconds. This confirms connectivity to the Cradlepoint’s GPS from somewhere on the vehicle subnet.

5. **Confirm GPS data is being logged properly:** Full functioning of the GPS feature of this device is best confirmed after the Technologic CPU has been configured and the log file is updating with GPS data (Section 3.5.6).

6. **Confirm data is being sent properly over Cradlepoint Cellular Modem:** Full functioning of the cell modem is best confirmed by examining vehicle data received at the back end, when the vehicle is clearly NOT in a DSRC geo-fence zone. Note that the NDOT District 2 yard is at the outer limits of a DSRC zone, and with the current geofencing rules, the radio may try to use DSRC, but may fail if signal is poor. It won’t even try to use cellular from such a location. A driving test (in and out of DSRC zone) is the best way to check.

3.7 Roadwatch

For existing vehicles, it is likely that they already will have a Roadwatch installed in the vehicle. We will, however, need to add the RS232 Adapter Box (sold separately by the Roadwatch folks) to be able to read the data into the IMO hardware. The RS232 Adapter Box will be installed behind the dashboard (attached to the Roadwatch), with a new serial cable running back to the IMO hardware. Configuration of this functionality is not needed. The IMO code will automatically configure the serial port to operate with the Roadwatch (baud, etc.) and use the required commands to read the data (etc.). **No additional extra configuration is typically necessary (beyond the physical interfacing).**

3.7.1 Roadwatch Physical Mounting

The Roadwatch sensor head should be mounted in accordance with the manufacturer’s recommendations. (i.e., on the exterior of the vehicle, with the sensor pointing downwards to the ground, and in an area that is not too vulnerable.
to accumulation of dirt and ice. NDOT commonly mounts the sensor head on the driver side mirror. An in-vehicle display is also mounted in or under the dashboard so that the driver can see the air and surface temperatures in real-time.

The RS232 Adapter Box is mounted behind the dashboard (with the rest of the Roadwatch cabling), and the serial cable is typically routed back to the IMO hardware under or along the flooring of the vehicle interior.

3.7.2 Roadwatch Physical Connections

Once the Comet has been configured through a direct connection to the host computer, it can be detached, and interfaced into the normal IMO backplane connections. This will include:

1. 12 V Power connection (to backplane power block)
2. The Roadwatch sensor head connects to the RS23 Adapter Box, which in turn connects to Technologic.
3. 15’ Serial cable (DB-9M to DB-9F) from Roadwatch RS232 Adapter Box to the Technologic’s COM2 port. Note that this port assignment is contingent on the correct ribbon cable configuration on the inside of the Technologic CPU box. Alternatively (if there is a problem with the internal cabling) you can try to RW on the port for COM3. However, the software is looking at COM2. If COM3 works, it is probably because the ports have their internal ribbon cables swapped. This has happened in the past.

3.7.3 Test Roadwatch Functionality

1. Configure the Configuration Host test laptop to work on the IMO backplane subnet. Do this by setting the laptop's IP address to the last available address in the subnet range for the vehicle (labeled RESERVED). For example, for vehicle IMO_11, use < > for the laptop’s IP address.
2. **Confirm data is being logged properly**: Full functioning of this device is best confirmed after the Technologic CPU has been configured (Section 3.5.6).

3.8 Adding Additional (or Changing) DSRC Road Side Units (RSUs)

3.8.1 Arada DSRC Physical Mounting

The Arada will be mounted to a roadside utility pole (or similar). These (below) steps are for initial setup only (before going into the field).

3.8.2 Arada RSU DSRC Physical Connections

The connections to the Arada RSU are as follows:

1. 12 V Power connection should be made using Power Over Ethernet (POE) adapter
2. 2 antennas directly connected to the Arada’s antenna ports.
3. Ethernet cable from the Arada’s WAN1 port to the POE adapter, and then another Ethernet cable from the POE to the Configuration Host’s Ethernet port.
3.8.3 Configuring a new RSU

The Arada setup process is (preferably) handled through a semi-automated script run from the NIMO Configuration Host computer. This process is nearly identical (with very minor differences) to the one used for the OBU radios (see Section 3.4).

See Section 5 for discussion of configuring the DSRC geofencing rules.

3.8.4 Host Computer Configuration

3.8.4.1 Resetting the Arada to factory defaults

See Section 3.4.4 (earlier) for restoring the Arada radios to factory default IP address.

3.8.4.2 Setting up the Configuration Host computer

See earlier section on OBU Arada setup for more details. Summary: If the Arada radio is working with the factory default IP address, set up the configuration host’s IP using:

```bash
root@HostComputer:~# ifconfig eth0 192.168.0.1
root@HostComputer:~# ifconfig eth0 netmask 255.255.255.0
root@HostComputer:~# ping 192.168.0.40 (or .41 etc.)
```

3.8.5 Acquire an Up-To-Date Copy of the IP Configuration Plan

Before you proceed with configuring the Arada RSU, ensure that the Configuration Host computer has an up-to-date copy of the IMO project’s RSU IP configuration plan .csv file and that the file version you are using includes the up-to-date RSU site information for the configuration you are doing. The expected path/location for this is

```
/mnt/sdb2/installation/DSRC_Pilot_v2.csv
```

The RSU IP configuration file (csv) can be examined using

```
cd /mnt/sdb2/installation
less DSRC_Pilot_v2.csv
```

It can be edited from the Configuration Host using: `nano DSRC_Pilot_v2.csv`

3.8.6 Run the OBU Configuration Setup Shell Script

Note: The easiest way to configure the Arada is by running the below script. If this does not work, it is also possible to use the Command Line Interface on the Arada, following the script source as a guide. Also see earlier OBU instructions in Section 3.4.7.

Change into the host computer’s IMO installation directory:
cd /mnt/sdb2/installation/

Run the OBU setup shell script (which should be in this directory), following the prompts

```
./rsu_install.sh
```

The shell script will first ask you for the site number. Type it in (for example, 8)

```
Site#: 8
```

Note that you may see a message saying that the Host is not in the trusted hosts file, and asks if you want to continue. Answer YES (y).

You will then see a report of the configuration information that it found for the Arada OBU. Something along the lines of the below. Just ensure nothing looks like clear signs that the data was not found properly in the .csv file (note that these IP addresses below are not typical of the IMO vehicles and are used here only as an example).

```
Site#: 8
id=8; gw=< >; nm=< >; ip=< >; host=D2-I580WashoeValley-DSRC; d=I580/WASHOE_VALLEY
```

Confirm that the IP addresses appear correctly. If they do not appear at all, there is probably a formatting problem with the configuration file on the host (/mnt/sdb2/installation/ DSRC_Pilot_v2.csv).

After pinging the Arada, the script will require that you enter the default password, which (initially) will simply be password. Use the clipboard copy (CTRL+Insert) and paste (SHFT+Insert) to paste the password and progress through the script. Again, see earlier OBU instructions in Section 3.4.7.

After the firmware is updated and it reboots, change the Configuration Host’s IP address and netmask. Once this is done, you can ssh into the RSU to confirm connectivity (etc.). For example, for RSU_8 this would be:

```
ssh root@< > password: < >
```

3.8.7 Confirm the RSU Configuration on the Arada

Log into the Arada

```
ssh root@< > password: < >
```

You can see if the receiver script is running:

```
ps –A
```

And look for /var/NIM0v3receiver_ack to be listed as running.
You can also look at:

```
    echo show configuration | cli; uname -a
```

This will type to the screen all the changes that were made. You should confirm a few things:

1. Confirm that `application6` is configured to be `/var/NIMOv3/receiver_ack` and that it is enabled.

2. Confirm that the Arada Locomate is running the `.13_detroit` version of the Linux build, which is necessary for the Arada to be set up with (of note for future installations that may come “out of the box” with later versions.

You should confirm that the application for the RSU is ready to run on bootup.

```
    cd /var/NIMOv3/
    ls -la
```

You should see `start.sh*` in the file list. This starts the main RSU code upon boot of the Arada. You can also run it manually by typing `/start.sh` at the prompt. This will (in turn) run some other stuff like `receiver_ack`. The script (start.sh) will respond by displaying the process IDs (for example `1053 1036`).

At this point, you are done with the Arada RSU configuration. You can terminate the ssh session from the host and can `poweroff`, or just physically remove power to the Arada.
4. Vehicle IP Configuration Data

See Table 2 (Vehicle IP Configuration Data IMO #1).

4. Table 2. Vehicle IP Configuration Data IMO#1

<table>
<thead>
<tr>
<th>IMO Vehicle Number:</th>
<th>IMO #1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vehicle Plate Number:</td>
<td>2275</td>
</tr>
<tr>
<td>Vehicle Type:</td>
<td>Plow</td>
</tr>
<tr>
<td>Location:</td>
<td>Carson City</td>
</tr>
<tr>
<td>Subnet:</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Mask:</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Subnet Size:</td>
<td>6</td>
</tr>
<tr>
<td>Host Range:</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Broadcast:</td>
<td>&lt; &gt;</td>
</tr>
</tbody>
</table>

IMO Host Device

<table>
<thead>
<tr>
<th>Host Device</th>
<th>Host Device IP Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cradlepoint (Gateway)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Comet (Weather Sensor)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>TechnoLogic (Linux CPU)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Adam (Generic Data I/O)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Arada DSRC On Board Unit (OBU)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>RESERVED (use for laptop access)</td>
<td>&lt; &gt;</td>
</tr>
</tbody>
</table>
See Table 3 (Vehicle IP Configuration Data IMO #2).

4. Table 3. Vehicle IP Configuration Data IMO#2

<table>
<thead>
<tr>
<th>IMO Vehicle Number:</th>
<th>IMO #2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vehicle Plate Number:</td>
<td>TBD</td>
</tr>
<tr>
<td>Vehicle Type:</td>
<td>FSP Van</td>
</tr>
<tr>
<td>Location:</td>
<td>Reno</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Subnet:</th>
<th>&lt; &gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mask:</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Subnet Size:</td>
<td>6</td>
</tr>
<tr>
<td>Host Range:</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Broadcast:</td>
<td>&lt; &gt;</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>IMO Host Device</th>
<th>Host Device IP Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cradlepoint (Gateway)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Comet (Weather Sensor)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>TechnoLogic (Linux CPU)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Unused/ADAM</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Arada DSRC On Board Unit (OBU)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>RESERVED (use for laptop access)</td>
<td>&lt; &gt;</td>
</tr>
</tbody>
</table>
See Table 4 (Vehicle IP Configuration Data IMO #3).

### 4. Table 4. Vehicle IP Configuration Data IMO#3

<table>
<thead>
<tr>
<th>IMO Vehicle Number:</th>
<th>IMO #3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vehicle Plate Number:</td>
<td>3200</td>
</tr>
<tr>
<td>Vehicle Type:</td>
<td>Plow</td>
</tr>
<tr>
<td>Location:</td>
<td>Fernley</td>
</tr>
</tbody>
</table>

| Subnet: < > | Subnet Size: 6 |
| Mask: < > | Host Range: < > |
| Broadcast: < > |

<table>
<thead>
<tr>
<th>IMO Host Device</th>
<th>Host Device IP Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cradlepoint (Gateway)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Comet (Weather Sensor)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>TechnoLogic (Linux CPU)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Adam (Generic Data I/O)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Arada DSRC On Board Unit (OBU)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>RESERVED (use for laptop access)</td>
<td>&lt; &gt;</td>
</tr>
</tbody>
</table>
See Table 5 (Vehicle IP Configuration Data IMO #4).

### 4. Table 5. Vehicle IP Configuration Data IMO#4

<table>
<thead>
<tr>
<th>IMO Vehicle Number:</th>
<th>IMO #4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vehicle Plate Number:</td>
<td>2274</td>
</tr>
<tr>
<td>Vehicle Type:</td>
<td>Plow</td>
</tr>
<tr>
<td>Location:</td>
<td>Reno</td>
</tr>
<tr>
<td>Subnet:</td>
<td><code>&lt; &gt;</code></td>
</tr>
<tr>
<td>Mask:</td>
<td><code>&lt; &gt;</code></td>
</tr>
<tr>
<td>Subnet Size:</td>
<td>6</td>
</tr>
<tr>
<td>Host Range:</td>
<td><code>&lt; &gt;</code></td>
</tr>
<tr>
<td>Broadcast:</td>
<td><code>&lt; &gt;</code></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>IMO Host Device</th>
<th>Host Device IP Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cradlepoint (Gateway)</td>
<td><code>&lt; &gt;</code></td>
</tr>
<tr>
<td>Comet (Weather Sensor)</td>
<td><code>&lt; &gt;</code></td>
</tr>
<tr>
<td>TechnoLogic (Linux CPU)</td>
<td><code>&lt; &gt;</code></td>
</tr>
<tr>
<td>Adam (Generic Data I/O)</td>
<td><code>&lt; &gt;</code></td>
</tr>
<tr>
<td>Arada DSRC On Board Unit (OBU)</td>
<td><code>&lt; &gt;</code></td>
</tr>
<tr>
<td>RESERVED (use for laptop access)</td>
<td><code>&lt; &gt;</code></td>
</tr>
</tbody>
</table>
See Table 6 (Vehicle IP Configuration Data IMO #5).

### 4. Table 6. Vehicle IP Configuration Data IMO#5

<table>
<thead>
<tr>
<th>IMO Vehicle Number:</th>
<th>IMO #5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vehicle Plate Number:</td>
<td>0267</td>
</tr>
<tr>
<td>Vehicle Type:</td>
<td>Plow</td>
</tr>
<tr>
<td>Location:</td>
<td>Mt. Rose</td>
</tr>
<tr>
<td>Subnet:</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Mask:</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Subnet Size:</td>
<td>6</td>
</tr>
<tr>
<td>Host Range:</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Broadcast:</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>IMO Host Device</td>
<td>Host Device IP Address</td>
</tr>
<tr>
<td>Cradlepoint (Gateway)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Comet (Weather Sensor)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>TechnoLogic (Linux CPU)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Adam (Generic Data I/O)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Arada DSRC On Board Unit (OBU)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>RESERVED (use for laptop access)</td>
<td>&lt; &gt;</td>
</tr>
</tbody>
</table>
See Table 7 (Vehicle IP Configuration Data IMO #6).

### 4. Table 7. Vehicle IP Configuration Data IMO#6

<table>
<thead>
<tr>
<th>IMO Vehicle Number:</th>
<th>IMO #6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vehicle Plate Number:</td>
<td>2272</td>
</tr>
<tr>
<td>Vehicle Type:</td>
<td>Plow</td>
</tr>
<tr>
<td>Location:</td>
<td>Reno</td>
</tr>
</tbody>
</table>

| Subnet: | < | > |
| Mask: | < | > |
| Subnet Size: | 6 |
| Host Range: | < | > |
| Broadcast: | < | > |

<table>
<thead>
<tr>
<th>IMO Host Device</th>
<th>Host Device IP Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cradlepoint (Gateway)</td>
<td>&lt;</td>
</tr>
<tr>
<td>Comet (Weather Sensor)</td>
<td>&lt;</td>
</tr>
<tr>
<td>TechnoLogic (Linux CPU)</td>
<td>&lt;</td>
</tr>
<tr>
<td>Adam (Generic Data I/O)</td>
<td>&lt;</td>
</tr>
<tr>
<td>Arada DSRC On Board Unit (OBU)</td>
<td>&lt;</td>
</tr>
<tr>
<td>RESERVED (use for laptop access)</td>
<td>&lt;</td>
</tr>
</tbody>
</table>
See Table 8 (Vehicle IP Configuration Data IMO #7).

### 4. Table 8. Vehicle IP Configuration Data IMO#7

<table>
<thead>
<tr>
<th>IMO Vehicle Number:</th>
<th>IMO #7</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vehicle Plate Number:</td>
<td>0763</td>
</tr>
<tr>
<td>Vehicle Type:</td>
<td>Plow</td>
</tr>
<tr>
<td>Location:</td>
<td>XXXX</td>
</tr>
<tr>
<td>Subnet:</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Mask:</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Subnet Size:</td>
<td>6</td>
</tr>
<tr>
<td>Host Range:</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Broadcast:</td>
<td>&lt; &gt;</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>IMO Host Device</th>
<th>Host Device IP Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cradlepoint (Gateway)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Comet (Weather Sensor)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>TechnoLogic (Linux CPU)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Adam (Generic Data I/O)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Arada DSRC On Board Unit (OBU)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>RESERVED (use for laptop access)</td>
<td>&lt; &gt;</td>
</tr>
</tbody>
</table>
See Table 9 (Vehicle IP Configuration Data IMO #8).

### 4. Table 9. Vehicle IP Configuration Data IMO#8

<table>
<thead>
<tr>
<th>IMO Vehicle Number:</th>
<th>IMO #8</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vehicle Plate Number:</td>
<td>0323</td>
</tr>
<tr>
<td>Vehicle Type:</td>
<td>Plow</td>
</tr>
<tr>
<td>Location:</td>
<td>Fernley</td>
</tr>
</tbody>
</table>

| Subnet: < > | Subnet Size: 6 |
| Mask: < > | Host Range: < > |
|           | Broadcast: < > |

<table>
<thead>
<tr>
<th>IMO Host Device</th>
<th>Host Device IP Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cradlepoint (Gateway)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Comet (Weather Sensor)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>TechnoLogic (Linux CPU)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Adam (Generic Data I/O)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Arada DSRC On Board Unit (OBU)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>RESERVED (use for laptop access)</td>
<td>&lt; &gt;</td>
</tr>
</tbody>
</table>
See Table 10 (Vehicle IP Configuration Data IMO #9).

### 4. Table 10. Vehicle IP Configuration Data IMO#9

<table>
<thead>
<tr>
<th>IMO Vehicle Number:</th>
<th>IMO #9</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vehicle Plate Number:</td>
<td>3319</td>
</tr>
<tr>
<td>Vehicle Type:</td>
<td>Plow</td>
</tr>
<tr>
<td>Location:</td>
<td>Reno</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Subnet: &lt; &gt;</th>
<th>Subnet Size: 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mask: &lt; &gt;</td>
<td>Host Range: &lt; &gt;</td>
</tr>
<tr>
<td></td>
<td>Broadcast: &lt; &gt;</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>IMO Host Device</th>
<th>Host Device IP Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cradlepoint (Gateway)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Comet (Weather Sensor)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>TechnoLogic (Linux CPU)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Adam (Generic Data I/O)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Arada DSRC On Board Unit (OBU)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>RESERVED (use for laptop access)</td>
<td>&lt; &gt;</td>
</tr>
</tbody>
</table>
See Table 11 (Vehicle IP Configuration Data IMO #10).

### Table 11. Vehicle IP Configuration Data IMO#10

<table>
<thead>
<tr>
<th>IMO Vehicle Number:</th>
<th>IMO #10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vehicle Plate Number:</td>
<td>1915</td>
</tr>
<tr>
<td>Vehicle Type:</td>
<td>Plow</td>
</tr>
<tr>
<td>Location:</td>
<td>Carson</td>
</tr>
<tr>
<td>Subnet:</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Mask:</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Subnet Size:</td>
<td>6</td>
</tr>
<tr>
<td>Host Range:</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Broadcast:</td>
<td>&lt; &gt;</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>IMO Host Device</th>
<th>Host Device IP Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cradlepoint (Gateway)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Comet (Weather Sensor)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>TechnoLogic (Linux CPU)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Adam (Generic Data I/O)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Arada DSRC On Board Unit (OBU)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>RESERVED (use for laptop access)</td>
<td>&lt; &gt;</td>
</tr>
</tbody>
</table>
See Table 12 (Vehicle IP Configuration Data IMO #11).

4. Table 12. Vehicle IP Configuration Data IMO#11

<table>
<thead>
<tr>
<th>IMO Vehicle Number:</th>
<th>IMO #11</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vehicle Plate Number:</td>
<td>TBD</td>
</tr>
<tr>
<td>Vehicle Type:</td>
<td>TBD</td>
</tr>
<tr>
<td>Location:</td>
<td>TBD</td>
</tr>
<tr>
<td>Subnet:</td>
<td>&lt;</td>
</tr>
<tr>
<td>Mask:</td>
<td>&gt;</td>
</tr>
<tr>
<td>Subnet Size:</td>
<td>6</td>
</tr>
<tr>
<td>Host Range:</td>
<td>&lt;</td>
</tr>
<tr>
<td>Broadcast:</td>
<td>&gt;</td>
</tr>
<tr>
<td>IMO Host Device</td>
<td>Host Device IP Address</td>
</tr>
<tr>
<td>Cradlepoint (Gateway)</td>
<td>&lt;  &gt;</td>
</tr>
<tr>
<td>Comet (Weather Sensor)</td>
<td>&lt;  &gt;</td>
</tr>
<tr>
<td>TechnoLogic (Linux CPU)</td>
<td>&lt;  &gt;</td>
</tr>
<tr>
<td>Adam (Generic Data I/O)</td>
<td>&lt;  &gt;</td>
</tr>
<tr>
<td>Arada DSRC On Board Unit (OBU)</td>
<td>&lt;  &gt;</td>
</tr>
<tr>
<td>RESERVED (use for laptop access)</td>
<td>&lt;  &gt;</td>
</tr>
</tbody>
</table>
See Table 13 (Vehicle IP Configuration Data IMO #12).

### 4. Table 13. Vehicle IP Configuration Data IMO#12

<table>
<thead>
<tr>
<th>IMO Vehicle Number:</th>
<th>IMO #12</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vehicle Plate Number:</td>
<td>TBD</td>
</tr>
<tr>
<td>Vehicle Type:</td>
<td>TBD</td>
</tr>
<tr>
<td>Location:</td>
<td>TBD</td>
</tr>
<tr>
<td>Subnet:</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Mask:</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Subnet Size:</td>
<td>6</td>
</tr>
<tr>
<td>Host Range:</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Broadcast:</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>IMO Host Device</td>
<td>Host Device IP Address</td>
</tr>
<tr>
<td>Cradlepoint (Gateway)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Comet (Weather Sensor)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>TechnoLogic (Linux CPU)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Adam (Generic Data I/O)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Arada DSRC On Board Unit (OBU)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>RESERVED (use for laptop access)</td>
<td>&lt; &gt;</td>
</tr>
</tbody>
</table>
See Table 14 (Vehicle IP Configuration Data IMO #13).

### 4. Table 14. Vehicle IP Configuration Data IMO#13

<table>
<thead>
<tr>
<th>IMO Vehicle Number:</th>
<th>IMO #13</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vehicle Plate Number:</td>
<td>TBD</td>
</tr>
<tr>
<td>Vehicle Type:</td>
<td>TBD</td>
</tr>
<tr>
<td>Location:</td>
<td>TBD</td>
</tr>
<tr>
<td>Subnet:</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Mask:</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Subnet Size:</td>
<td>6</td>
</tr>
<tr>
<td>Host Range:</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Broadcast:</td>
<td>&lt; &gt;</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>IMO Host Device</th>
<th>Host Device IP Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cradlepoint (Gateway)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Comet (Weather Sensor)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>TechnoLogic (Linux CPU)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Adam (Generic Data I/O)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>Arada DSRC On Board Unit (OBU)</td>
<td>&lt; &gt;</td>
</tr>
<tr>
<td>RESERVED (use for laptop access)</td>
<td>&lt; &gt;</td>
</tr>
</tbody>
</table>
See Table 15 (Vehicle IP Configuration Data IMO #14).

### 4. Table 15. Vehicle IP Configuration Data IMO#14

<table>
<thead>
<tr>
<th>IMO Vehicle Number</th>
<th>Vehicle Plate Number</th>
<th>Vehicle Type</th>
<th>Location</th>
<th>Subnet</th>
<th>Mask</th>
<th>Subnet Size</th>
<th>Host Range</th>
<th>Broadcast</th>
</tr>
</thead>
<tbody>
<tr>
<td>IMO #14</td>
<td>TBD</td>
<td>TBD</td>
<td>TBD</td>
<td>&lt;</td>
<td>&gt;</td>
<td>6</td>
<td>&lt;</td>
<td>&lt;</td>
</tr>
<tr>
<td>IMO Host Device</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cradlepoint (Gateway)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Comet (Weather Sensor)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TechnoLogic (Linux CPU)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Adam (Generic Data I/O)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Arada DSRC On Board Unit (OBU)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RESERVED (use for laptop access)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
See Table 16 (Vehicle IP Configuration Data IMO #15).

### 4. Table 16. Vehicle IP Configuration Data IMO#15

<table>
<thead>
<tr>
<th>IMO Vehicle Number: IMO #15</th>
<th>Vehicle Plate Number: TBD</th>
<th>Vehicle Type: TBD</th>
<th>Location: TBD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Subnet: <code>&lt;                  &gt;</code></td>
<td>Subnet Size: 6</td>
<td>Host Range: <code>&lt;     &gt;</code></td>
<td></td>
</tr>
<tr>
<td>Mask: <code>&lt;                    &gt;</code></td>
<td></td>
<td>Broadcast: <code>&lt;      &gt;</code></td>
<td></td>
</tr>
<tr>
<td>IMO Host Device</td>
<td>Host Device IP Address</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cradlepoint (Gateway)</td>
<td><code>&lt;                        &gt;</code></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Comet (Weather Sensor)</td>
<td><code>&lt;                        &gt;</code></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TechnoLogic (Linux CPU)</td>
<td><code>&lt;                        &gt;</code></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Adam (Generic Data I/O)</td>
<td><code>&lt;                        &gt;</code></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Arada DSRC On Board Unit (OBU)</td>
<td><code>&lt;                  &gt;</code></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RESERVED (use for laptop access)</td>
<td><code>&lt;             &gt;</code></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
See Table 17 (Vehicle IP Configuration Data IMO #16).

4. Table 17. Vehicle IP Configuration Data IMO#16

<table>
<thead>
<tr>
<th>IMO Vehicle Number:</th>
<th>IMO #16</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vehicle Plate Number:</td>
<td>TBD</td>
</tr>
<tr>
<td>Vehicle Type:</td>
<td>TBD</td>
</tr>
<tr>
<td>Location:</td>
<td>TBD</td>
</tr>
</tbody>
</table>

| Subnet:             | < >             | Subnet Size: 6 |
| Mask:               | < >             | Host Range: < > |
|                     |                 | Broadcast: < > |

IMO Host Device | Host Device IP Address |
----------------|------------------------|
Cradlepoint (Gateway) | < > |
Comet (Weather Sensor) | < > |
TechnoLogic (Linux CPU) | < > |
Adam (Generic Data I/O) | < > |
Arada DSRC On Board Unit (OBU) | < > |
RESERVED (use for laptop access) | < > |

5. Configuration of DSRC Geofencing

Each vehicle is configured with a geofencing configuration data file that it refers to when deciding whether to transmit data using DSRC or cellular radios. This configuration data file is found on the OBU’s Technologic CPU in the file:

```
/root/src/imo_configs.py
```

This file can be manually edited, but these edits will only apply to that one vehicle’s behavior. To make changes to the default geofencing rules that will be applied to all vehicles that are subsequently configured, it will be necessary to edit the master file located on the installation LiveUSB drive (i.e., the Configuration Host). This way, all vehicles set up using this LiveUSB will use the new configuration file. Edit the file using the Linux `nano` editor application from the command line.

```
nano /mnt/sdb2/installation/TECHNOLOGIC_INSTALL/imo_configs.py
```

Edits should be made to maintain the existing data file formatting. Key information that will be needed include the RSU site’s

- RSU Site Name
Appendix B. NIMO Phase 3 Configuration Manual v2.0

- Latitude
- Longitude
- Geofence distance (range, in meters) from the RSU

6. Accessing IMO Data

The raw mESS data (back-end) can be retrieved using the below URL\(^\text{22}\). This is currently publicly accessible from outside the NDOT firewall. The last portion of the URL designates the date for the data being sought. This file will include ALL the vehicles that reported on the provided date. The first line of the file will be the data value headings.

7. Login and Password Summary for System Components

See Table 18 (Login/User ID Password for System Components).

5. Table 18. Login/User ID Password for System Components

<table>
<thead>
<tr>
<th>Device</th>
<th>Login/User ID</th>
<th>Password</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arada OBUs and RSUs</td>
<td>root</td>
<td>nTi?emk1</td>
</tr>
<tr>
<td>Technologic CPU</td>
<td>root</td>
<td>Ha3mUMsh</td>
</tr>
</tbody>
</table>

\(^{22}\) See mESS data [http://mess.its.nv.gov/NCAR/Transmitted/?d=2017-04-10](http://mess.its.nv.gov/NCAR/Transmitted/?d=2017-04-10)