Clear Roads

CR 21-06: Calculating Plow Cycle Times from AVL Data

Technical Memorandum: Online Tool Framework

Objective

As outlined in the technical scope of this project, one of the activities in Task 5 is to develop a functional tool framework for Calculating Plow/Treatment Cycle Times from Automatic Vehicle Location (AVL) Data. AVL systems are widely deployed on winter maintenance fleets. The objective of this document is to outline a comprehensive framework for developing a dashboard/tool to measure and analyze plow/treatment cycle times. This framework aims to facilitate efficient data utilization and accessibility across the organization by integrating various data sources and formats.

This document includes an analysis of data requirements, a database design outlining data conversion processes, Application Programming Interface (API) connections, and a storage options plan for archiving historical data. This document outlines recommendations for a suitable dashboard platform and hosting option, use cases and workflows for the tool, a visualization design diagram, and a roadmap for development and testing stages. The dashboard/tool is intended to provide a user-friendly interface for visualizing plow/treatment cycle time data, enabling effective decision-making and operational planning. By accomplishing these goals, the dashboard/tool framework will provide a valuable resource for monitoring and optimizing plow operations, enabling data-driven decision-making, and enhancing overall operational efficiency.

It should be noted that cycle time refers to the estimated duration required to service a site, service area, or route in a single cycle. Numerous factors, including Level of Service (LOS) and production rate(s), influence cycle time. Additionally, a case study is provided to model treatment of a route with three segments by trucks equipped with AVL.

Requirements

This tool is supported by several components or subsystems shown in the high-level architecture (HLA) of Figure 1. The HLA representation is useful for understanding the functional components of an overall system and the relationships between them, including components for data ingestion and data processing. This is a critical step in identifying software components and connections that must be developed to support this tool. The architecture flows from left to right, beginning with data inputs on the left, which flow into data cleaning and formatting modules, followed by data processing, Artificial Intelligence (AI)/Machine Learning (ML), and output modules. The function of these modules is described in Table 1. Note that there are required paths (solid lines) and optional paths to enhance the tool (dashed lines) within Figure 1.

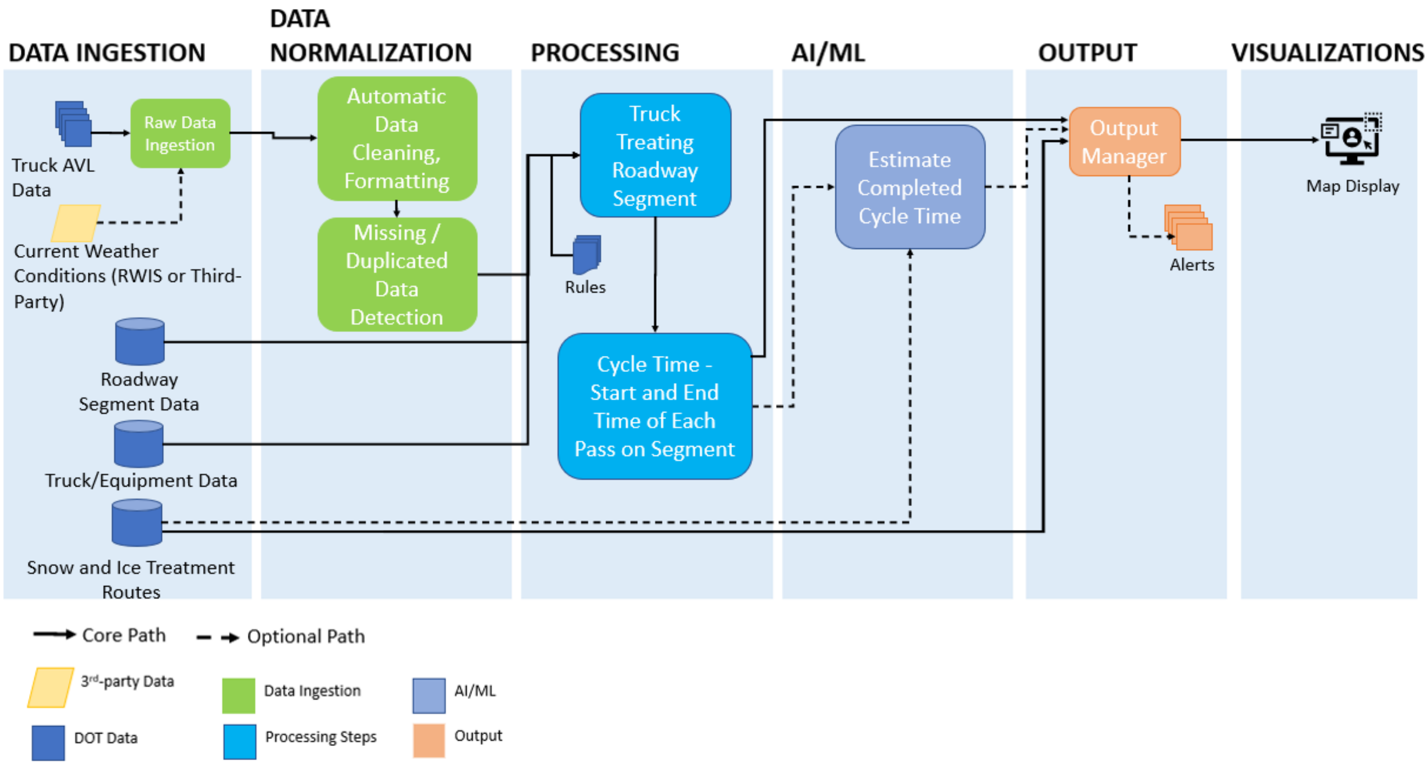


Figure 1: High-level Architecture of Cycle Time Tool

Table 1: Cycle Time Tool Module Descriptions

Module Name	Module Type	Description
(AI/ML) Estimate Completed Cycle Time	AI/ML Module	An AI/ML model trained to estimate the plow/treatment cycle time based on historical data ideally fused with weather data.
Raw Data Ingestion	Data Ingestion Module	Device-specific data handler
Automatic Data Cleaning, Formatting	Data Normalization Module	Clean and format data
Missing/Duplicated Data Detection	Data Normalization Module	Detect duplicated or missing data
Rules - Action Threshold Comparison	Processing Module	Compare input data to threshold values to determine if a response is needed
Cycle Time - Start and End Time of Each Passing Segments	Processing Module	Calculate cycle time ratio based on results of previous module while collecting time stamps, noting when cycle is complete (ratio is greater than or equal to one)
Truck Treating Roadway Segment	Processing Module	Pull proper treatment width based on the truck's treatment widths and compare to road segment width

Module Name	Module Type	Description
Output Manager	Output Module	Manage output type and content
Map Display	Visualizations	Plow/Treatment Cycle Time Tool/Dashboard

Data Ingestion and Normalization

This tool harnesses a combination of live and stored data to provide comprehensive insights. Live data encompasses real-time information sourced directly from the winter maintenance fleet's AVL system, allowing for immediate monitoring and analysis. Stored data refers to information stored in existing databases, comprising critical details like road segment specifics, truck profiles, and route data. Additionally, this tool offers the flexibility to integrate optional sources for enhanced accuracy and coverage such as weather data from environmental sensors or 3rd party weather services.

Stored data are assumed to have already been through a data cleaning, formatting, and validation process, so these data sources may bypass those steps. Therefore, these data types are treated differently in the HLA. Live data are assumed to be raw and are therefore ingested using source-specific ingestion modules, which then pass the data to automatic cleaning, formatting, and missing/duplicated detection processes within the tool development process. If the AVL data are already clean, this step might not be required. Missing and duplicated data are defined by a set of rules that may be customized based on the data source. Examples of missing or duplicated data for winter maintenance operations can range from missing route start times, duplicated route ids, or inconsistent route geometry. Rules can be implemented to set flags when these instances occur in a dataset to check for accuracy and quality.

The refined data is subsequently routed to the processing modules. While clean data can also be stored in an agency storage database, this step is omitted from the Cycle Time tool's HLA diagram for the sake of clarity.

Processing

Data are combined and analyzed within a process step that detects the maintenance activity of a truck on a winter operations route. To achieve this, basic rules and associations are employed. These rules consist of predefined criteria and conditions that serve as the foundation for decision-making within the system. For instance, a basic rule might specify that if a vehicle's GPS coordinates fall within the geographic boundaries of a designated winter maintenance route during a specific time frame, it is considered to be on that route.

Associations, on the other hand, establish connections between different data points or attributes. For example, associating GPS coordinates with a specific maintenance route allows the system to link a vehicle's location with a known route, enabling tracking and analysis.

The robustness of this processing step plays a vital role in enhancing the efficiency and accuracy of the tool/dashboard. By harnessing these fundamental rules and associations, the system can reliably discern the presence of trucks and ascertain treatment widths, providing invaluable insights for optimizing winter maintenance operations.

Once a truck's treatment is verified, its data are sent to the next processing module: the Cycle Time – Start and End Time of Each Pass on Segment. This is the module that will take time stamps for each pass and calculate the cycle time ratio. Once the cycle time ratio equals or exceeds 1, the cycle is considered complete, and the next loop will begin to accumulate the cycle time ratio till it equals or exceeds 1.

Once complete the module could go to the optional AI/ML modules or directly to the Output Manager.

AI/ML

An AI module may be developed for this tool. The module will use advanced techniques to estimate predictive cycle time based on historical data and/or conditions.

Output and Visualizations

This tool will output a spatial dashboard or user interface that will allow users to review cycle time results. The tool can allow routes to be reviewed as a whole or drill down to individual segments within a route.

Database Design

The Extract, Transform, Load (ETL) process, in the context of dashboard development for calculating plow/treatment cycle times, is a crucial methodology for efficiently gathering, transforming, and loading the necessary data into the dashboard system (Figure 2). This process ensures that the data essential for accurate cycle time calculations are effectively extracted from relevant sources, undergo appropriate transformations, and are loaded into the dashboard for visualization and analysis. To calculate plow/treatment cycle times, the ETL process encompasses the following key steps. First, data are extracted from diverse sources, including AVL systems, weather data providers, and other relevant databases. This extraction phase involves meticulous identification and retrieval of the required data elements, such as plow/treatment locations, timestamps, weather conditions, and any additional parameters pertinent to accurate cycle time calculations.

Subsequently, the extracted data undergo meticulous transformations to prepare it for cycle time calculations. These transformations may encompass data format conversions, thorough data cleansing and validation procedures, necessary calculations to derive cycle time metrics, data aggregation at desired intervals (e.g., minutes), and the application of any essential business rules or adjustments to ensure the accuracy of cycle time measurements. Finally, the transformed data are loaded into the dashboard system, where they are structured and organized within the dashboard's data model for seamless visualization and analysis. This loading process ensures that the transformed data are readily available within the dashboard, enabling real-time or historical cycle time calculations and providing users with comprehensive visualization capabilities.

The ETL process plays a pivotal role in the creation of a plow/treatment cycle time dashboard by enabling the research team to harness and integrate relevant data from disparate sources. For the proof-of-concept tool developed within this project, the team embarked on utilizing AVL data obtained from a Skyhawk system to model plow cycle times based on historical routes. The ETL process served as a foundational framework to transform raw data into a usable format for analysis and visualization.

The initial step of the ETL process involved extracting the AVL data from the Skyhawk system. This included gathering information such as vehicle locations, timestamps, and other pertinent data points associated with plowing/treating activities if possible. However, the extracted data required significant cleanup and refinement to ensure their quality and consistency. For instance, the coordinate values provided in the raw data needed to be split into latitude and longitude points for precise geospatial analysis. A similar process will apply to other AVL datasets, and the user will have to review the format of the data and determine the proper processing needed to get the data in a workable format for this tool. Also, the additional sensor data, such as material spreading system or plow position, will allow the tool to be more accurate. Without these data sets, the cycle time may still be calculated but more assumptions will be necessary. For example, if a truck only provides location information without specifying treatment activity, users will need to infer the treatment width used by the truck while operating on the road during an event. In cases where a truck supplies plow and material with AVL data, the system will employ conditional statements to establish if treatment is taking place and, if confirmed, identify the correct treatment width based on information retrieved from the equipment dataset.

Extraction from APIs	• Identify the APIs you need to retrieve data from. This could include RESTful APIs, SOAP APIs, GraphQL APIs, or other web services. Obtain the necessary API credentials (e.g., API keys, tokens) to access the data. Use API client libraries or HTTP requests (GET, POST, etc.) to extract data from the APIs. Many programming languages have libraries or frameworks that simplify working with APIs, such as requests in Python or libraries for specific APIs.
Data Transformation	• Once you've extracted data from the API, you may need to transform it to make it suitable for your target system.
Loading the Data	• After transformation, load the data into your target destination. This could be a database (e.g., SQL database, NoSQL database) or a data warehouse (e.g., Amazon Redshift, Google BigQuery). Ensure that the loading process is robust, supports incremental loads if needed, and handles any data consistency or integrity issues.
Automation and Scheduling	• Consider automating the ETL process by setting up scheduled jobs or triggers to periodically fetch and transform data from the APIs. This ensures that data are always up-to-date. Use ETL orchestration tools or job schedulers to manage and monitor the execution of ETL workflows.
Error Handling and Logging	• Implement error handling mechanisms to deal with issues that may arise during data extraction, transformation, or loading. Log errors and exceptions for troubleshooting and auditing purposes.
Security and Rate Limiting	• Ensure that the ETL process complies with security best practices. Protect API credentials and data during transmission and storage. Be aware of API rate limits and implement mechanisms to respect them to avoid being blocked or throttled by the API provider.
Monitoring and Alerts	• Set up monitoring for your ETL processes to track the performance, and configure alerts for any anomalies or failures.
Scalability and Performance	• Design ETL process to be scalable, especially if dealing with large volumes of data. Consider parallel processing or distributed ETL solutions as needed.
Documentation	• Document your ETL processes thoroughly, including API endpoints, data schemas, transformation rules, and any scheduled jobs or configurations. This documentation will be valuable for maintenance and troubleshooting.
Testing and Validation	• Rigorously test the ETL pipelines, including edge cases and error scenarios, to ensure data accuracy and reliability.

Figure 2. Extract, Transform, and Load Overview

The next crucial phase involved transforming the extracted data into a standardized and structured format that could be processed and analyzed. The research team undertook various data transformation operations, including data cleansing, normalization, and enrichment. By following these steps, any inconsistencies, errors, or missing values in the dataset were addressed, thus enhancing the accuracy and reliability of the subsequent analysis.

Furthermore, the research team delineated the necessary layers for the cycle time dashboard. These layers encompassed essential components such as road segment data, winter maintenance routes, and equipment information. Each layer provided distinct insights and contributed to a comprehensive understanding of the plow/treatment cycle process. The ETL process facilitated the extraction of relevant data for each layer, enabling that only the required information was included in the subsequent analysis and visualization stages.

The choice of ETL option is dependent on agency's specific requirements, constraints, and preferences for data integration. Factors like data volume, complexity, budget, and the need for real-time processing will influence the decision. Additionally, factors like data security, data governance, and compliance requirements should be considered when choosing an ETL solution. The project team opted for an ETL tool option that allows for data integration with visualization capabilities. While Table 2 outlines different ETL options, it implicitly conveys valuable insights regarding their respective strengths and limitations. For instance, custom ETL scripts offer unparalleled flexibility but can be time-consuming and necessitate specialized development expertise. ETL tools, on the other hand, provide a user-friendly graphical interface but may entail a learning curve and potentially higher costs depending on the tool selected.

Table 2: ETL Options

ETL Options	Description	Pros	Cons
Custom ETL Scripts/Code	Writing custom scripts or code in programming languages like Python, Java, or Ruby to perform ETL tasks. This approach offers maximum flexibility but can be time-consuming and requires development expertise.	Maximum flexibility, can be tailored to specific needs.	Time-consuming, requires development expertise.
ETL Tools	There are many ETL tools available, both open-source and commercial, that provide a graphical interface for designing ETL workflows.	Provides graphical interface, may have pre-built functionalities.	May have a learning curve, may be costlier depending on the tool.
Cloud-Based ETL Services	Many cloud providers offer ETL services that are fully managed and scalable.	Fully managed, scalable, often integrated with cloud platforms	Might involve ongoing costs, dependency on the cloud provider.
Open-Source ETL Frameworks	Build ETL processes using open-source frameworks and libraries. These frameworks provide flexibility and can be customized as needed.	Flexible, customizable, often community supported.	Required expertise, potential integration challenges.
Data Integration Platforms	Some platforms offer comprehensive data integration capabilities, including ETL, data transformation, and data quality.	Comprehensive solution, covers ETL, transformation, and quality,	Potentially higher cost, may be complex to implement.

ETL Options	Description	Pros	Cons
Data Integration as a Service	There are SaaS solutions that provide ETL and data integration capabilities, often with pre-built connectors to popular data sources.	SaaS model, pre-built connectors for data sources.	Subscription based-cost, dependency on service-provider.
Serverless ETL	ETL processes using serverless computing platforms, which can be cost-effective and scalable.	Cost-effective, scalable, event-driven model.	May require adaptation to serverless paradigm, potential integration challenges.

Cloud-based ETL services offer scalability and full management but may involve ongoing expenses and a degree of dependency on the chosen cloud provider. Open-source ETL frameworks present a flexible and customizable solution yet may require expertise and potentially encounter integration challenges. Data integration platforms offer a comprehensive solution, encompassing ETL, transformation, and data quality, but may involve higher costs and complexity in implementation.

Data integration as a service follows a SaaS model, often with pre-built connectors, but is subscription-based and reliant on the service provider. Serverless ETL processes are cost-effective and scalable yet may require adaptation to a serverless paradigm and face potential integration challenges. Recognizing these implicit pros and cons is paramount in making an informed decision tailored to the specific needs and constraints of the agency.

In summary, the ETL process served as the foundation for the creation of a cycle time dashboard by extracting AVL data, transforming it into a standardized format, and loading the pertinent information for analysis. The ETL process included data cleanup, including splitting coordinate values, and facilitated the extraction of data layers crucial for understanding road segments, snow and ice routes, and equipment information.

The following sections aim to provide the Clear Road's Project Committee with the necessary variables to construct a dashboard/tool capable of determining plow/treatment cycle times. Furthermore, they present information regarding desired variables that could potentially enhance the functionality of the tool. In conjunction with the variables, the following sections provide an outline of the logical and methodological framework for the tool, ensuring accurate determination of cycle times. This cycle time tool will empower agencies to make more informed decisions about their fleet during active winter events and facilitate post-event performance evaluation.

Roadway Segment Data Geodatabase

The roadway segment database is an ESRI (Environmental Systems Research Institute) Geodatabase consisting of an LRS (Linear Reference System) polyline feature class. LRS features can locate points or lines along a route. More specifically, they contain x, y, and m values, where m is a value that represents the distance from the beginning of the line or a segment along the route. The polyline feature class will be segmented by turnaround points, cross-over points, or intersections along winter maintenance routes. Segments will also be attributed with route ID to calculate the total plow/treatment cycle time for a given pass.

The following table represents the Roadway Segment polyline feature class database architecture to satisfy calculation of plow/treatment cycle times. Notice that the table does not include a source since this dataset will be pre-populated with the required data within the attributes. The table includes a column for suggested data type that must be selected when creating each attribute field. The available data types include Short (integer), Long (integer), Float (or single precision floating-point numbers), Double (or double-precision floating-point numbers), and Text. When selecting data types for each attribute consider the data needs such as whole numbers (Short or Long) and fractional numbers (Float and Double).

Table 3: Roadway Segment Geodatabase Sample

Attribute Field	Purpose	Data Type
Route_ID	Represents a given route or pass.	Text
Segment_ID	Represents an individual segment.	Text or Long
Segment_Lanes	The number of lanes of a given segment which will be used in the cycle time calculation.	Short
Segment_Width	The width of a given segment that will be used in the cycle time calculation.	Float
Segment_Shoulder	(Optional) The width could be provided if shoulder exists. It is recommended that the agency add this dimension to the Segment_Width attribution field.	Float
Segment_Median	(Optional) The width could be provided if median exists. It is recommended that the agency add this dimension to the Segment_Width attribution field.	Float
From_DFO	DFO (Distance from Origin). Contains values for each segment from the DFO based on the Linear Reference System.	Double
To_DFO	DFO (Distance from Origin). Contains values for each segment to the DFO based on the Linear Reference System.	Double
Start_Latitude	First vertices representing the beginning of the segment. Lat/Long can potentially be used for determining when a vehicle begins or ends a given segment.	Double
Start_Longitude	First vertices representing the beginning of the segment. Lat/Long can potentially be used for determining when a vehicle begins or ends a given segment.	Double

Attribute Field	Purpose	Data Type
End_Latitude	Last vertices of the ending of the segment. Lat/Long can potentially be used for determining when a vehicle begins or ends a given segment.	Double
End_Longitude	Last vertices representing the ending of the segment. Lat/Long can potentially be used for determining when a vehicle begins or ends a given segment.	Double
Len_Miles	This attribute is added to store the length in miles of each line feature.	Double

Truck/Equipment Data Geodatabase

The Truck/Equipment Data will contain the truck ID and treatment width. Additional fields based on treatment type may be developed. Treatment width may be different based on material settings and plow positions. If AVL has these data points, additional fields for specific settings can be developed for the process to pull when specific conditions are met.

Table 4: Truck/Equipment Data Geodatabase

Variable	Purpose	Data Type
Truck Identification Number	To determine the capacity of the truck while treating a road	Text
Plow and/or Treatment Width	Based on the truck ID, the tool will determine how much of the roadway width was treated per pass. Each truck has a different treatment width based on size of plow(s) and chemical treatment spreading abilities.	Decimal Number

Snow and Ice Treatment Routes Geodatabase

The Snow and Ice Treatment Routes Geodatabase will include a polyline feature class like the one found in the Roadway Segment Geodatabase. This polyline feature class will be used for the overall cycle time calculation of a route (not a segmented route). The schema of this dataset will be smaller than that of the Roadway Segment but will still include Route_ID, DFO fields, and Len_Miles as variables. The attributes included, in conjunction with those of the previously mentioned Geodatabase, allow for the calculation of the total amount of time to clear a route. This will allow for filtering/drilling down functions on an interactive tool.

Overall, the Snow and Ice Treatment Routes Geodatabase is a valuable dataset for managing winter weather conditions and ensuring the safety and mobility of the public. By providing detailed and up-to-date information about treatment routes, plow cycles, and weather conditions, the dashboard/tool enables more effective and transparent decision-making and improves the overall efficiency and effectiveness of winter maintenance operations.

Table 5: Snow and Ice Treatment Routes Geodatabase

Attribute Field	Purpose	Data Type
Route_ID	Represents a given route or pass.	Text
From_DFO	DFO (Distance from Origin). Contains values for each segment from the DFO based on the Linear Reference System.	Double
To_DFO	DFO (Distance from Origin). Contains values for each segment to the DFO based on the Linear Reference System.	Double
Start_Latitude	First vertices representing the beginning of the segment. Lat/Long can potentially be used for determining when a vehicle begins or ends a given segment.	Double
Start_Longitude	First vertices representing the beginning of the segment. Lat/Long can potentially be used for determining when a vehicle begins or ends a given segment.	Double
End_Latitude	Last vertices of the ending of the segment. Lat/Long can potentially be used for determining when a vehicle begins or ends a given segment.	Double
End_Longitude	Last vertices representing the ending of the segment. Lat/Long can potentially be used for determining when a vehicle begins or ends a given segment.	Double
Len_Miles	This attribute is added to store the length in miles of each line feature.	Double

AVL Data

The AVL data is an Esri Geodatabase containing a point feature class that will store data populated by the AVL data services, spatially joined data from the Roadway Segment Geodatabase.

There are several different types of AVL systems used by transportation agencies to monitor vehicle locations and operational status of equipment for winter maintenance operations. An AVL system provides automatic vehicle location tracking for dispatchers and maintenance supervisors. In addition, when integrated with vehicle health monitoring systems, AVL systems can provide vehicle maintenance technicians with valuable information about vehicle diagnosis. AVL systems can also be integrated with existing vehicle components used for snowplow operations such as spreader controllers and plow positions to provide reports to maintenance supervisors on material and plow usage applied by snowplow operators.

ESRI offers two platforms for AVL systems: Geo Event and Velocity. Both can incorporate real time data through API's but are potentially costly for DOT's. The advantage is that they are equipped with processes that are formatted for a point feature class in a geodatabase.

Another, potentially more affordable, option is to create a custom system that incorporates Microsoft Azure and AGOL (ArcGIS Online). The real-time data could be processed within the Azure database and then appended to the cycle time process.

To compute Cycle Times, spatial joins at the given intervals received by an AVL system would be conducted between the Truck/Equipment point feature class with the Roadway Segment polyline feature class.

Table 6: AVL Data

Variable	Purpose	Format
Latitude	North-South location of truck. To determine when truck is on a route	Decimal Degree
Longitude	East-West location of truck. To determine when truck is on a route	Decimal Degree
Date and Timestamp	To timestamp when a truck starts and ends on a route, the duration will be required for cycle time calculation	MM/DD/YYYY HH:MM ZZZZ
Treatment Data (Optional)	If available, this will provide more accurate data of when treatment is occurring. Spreader On/Off, Plow activity, spinner speed, and other setting options that contribute to a change in a truck's treatment width are another option that may be included depending on the AVL data available.	Binary (1,0)
Truck Identification Number	To determine the capacity of the truck while treating a road	Text

Weather Data – Optional Enhancement

It may be ideal to integrate weather information and/or index for severity. This will allow for the users to determine the impact of the weather on cycle time. The weather information data will be similar to data presented in the table below. The date and time will link to treatment/cycle time activity and the segment or route data will link the proper weather data to the proper service area.

Table 7: Weather Data

Variable	Purpose	Format
Latitude	North-South location of weather station.	Decimal Degree
Longitude	East-West location of weather station.	Decimal Degree
Segment_ID/Route_ID	Link weather data to segments or Route	Text
Date and Timestamp	To timestamp for weather conditions	MM/DD/YYYY HH:MM ZZZZ
Weather Data – precipitation, wind and temperature (Optional)	To determine when an event has started. Also, may be used for cycle time vs storm severity analyses.	Decimal Number

Storage Options

Selecting the most suitable data storage option depends on several factors. Here are some key considerations to help select the right data storage option:

Data Volume and Size: Begin by carefully evaluating the magnitude of data that requires storage. It's imperative to recognize that various storage solutions cater to different scales. While some are proficient in handling small to medium-sized datasets, others outperforms when it comes to managing large-scale data. For instance, AVL data tends to possess a larger volume compared to equipment or route data.

Data Structure and Format: Delve into a detailed analysis of the data's underlying structure and format. This step is pivotal in selecting a storage option that seamlessly accommodates the data's specific format, ensuring optimal efficiency.

Data Access Patterns: Gain a comprehensive understanding of how the data will be accessed. Different storage systems are intricately designed and optimized for distinct access patterns. This knowledge is foundational in making an informed choice.

Scalability Requirements: Anticipate future growth and consider scalability needs. The selected data storage solution should possess the capacity to gracefully accommodate expansion over time. Cloud-based storage solutions, in particular, often present a straightforward path to scalability.

Performance Requirements: Gauge the performance demands of the application or analytics. Certain databases are finely tuned for swift, low-latency, high-throughput transactions, while others shine in handling intricate analytical queries.

Consistency and ACID (Atomicity, Consistency, Isolation, Durability) Compliance: Determine whether stringent data consistency and robust transaction support are essential. In such cases, databases that provide ACID compliance should be prioritized. Conversely, NoSQL databases often emphasize flexibility and scalability over absolute consistency.

Data Security and Compliance: Ensure that the chosen data storage solution aligns seamlessly with security and regulatory compliance requirements. This encompasses critical aspects such as encryption, access controls, and compliance with standards like GDPR, HIPAA, or other pertinent regulations.

Budget: Prudently factor in budget considerations. Certain data storage solutions may entail substantial upfront costs, ongoing operational expenses, or cloud storage charges. It is imperative to evaluate the total cost of ownership (TCO) in a comprehensive manner.

Integration with Existing Systems: If there are existing systems or applications in place, carefully consider how well the chosen data storage option integrates with them. Compatibility plays a pivotal role in ensuring smooth data transfer and system architecture.

Data Lifecycle Management: Devise a meticulous plan for data retention, archiving, and purging. Some storage solutions come equipped with built-in features that facilitate effective data lifecycle management.

Backup and Disaster Recovery: Establish a robust strategy for safeguarding data through backup and disaster recovery measures. Many cloud-based storage options offer integrated features for these critical functions.

Community and Support: Give due consideration to the availability of a supportive community or reliable support channels for the chosen data storage technology. This resource can prove invaluable for troubleshooting and staying ahead of best practices. It's a cornerstone of a sustainable and effective data storage strategy.

Common data storage options include:

- Relational Databases: Suitable for structured data and transactions. Examples include MySQL, PostgreSQL, Oracle, and SQL Server.
- NoSQL Databases: Offer flexibility for semi-structured and unstructured data. Types include document-oriented (MongoDB), key-value (Redis), column-family (Cassandra), and graph databases (Neo4j).
- Data Warehouses: Designed for analytical queries and aggregations. Examples include Amazon Redshift, Google BigQuery, and Snowflake.
- Cloud Storage: Provides scalable, cost-effective storage solutions like Amazon S3, Google Cloud Storage, and Azure Blob Storage.
- File Systems: For storing unstructured data like files and documents. Examples include NFS, CIFS, and distributed file systems like HDFS.
- In-Memory Databases: Offer extremely fast read and write access by storing data in RAM. Examples include Redis and Apache Ignite.
- Object Stores: Great for storing large volumes of unstructured data. Examples include Amazon S3, Google Cloud Storage, and Azure Blob Storage.

Ultimately, the data storage option decision depends on specific requirements. It's often a good idea to consult with data architects and engineers who can help tailor the choice to the agency’s needs. For a comprehensive Cycle Time dashboard tool encompassing Road Segment, Routes, Truck/Equipment, AVL, and an optional Weather dataset, it is recommended to have a combination of storage options to best accommodate the diverse nature of the data:

1. **Relational Database (e.g., PostgreSQL or MySQL):** This would serve as the backbone of the storage system. A relational database is excellent for structured data and transactions, making it suitable for Road Segment, Routes, Truck/Equipment, and AVL data. The relational model helps maintain relationships between different entities, providing a structured foundation.
2. **NoSQL Database (e.g., MongoDB):** Given the optional Weather dataset, which may contain semi-structured or unstructured data, incorporating a NoSQL database can provide the necessary flexibility. MongoDB, for instance, excels in handling diverse data types and can seamlessly integrate with structured data.
3. **Cloud-Based Storage (e.g., Amazon S3):** Utilizing cloud storage options like Amazon S3 for storing large binary objects such as images, videos, or any other media associated with the Cycle Time dashboard can be advantageous. It provides scalable and cost-effective storage solutions for unstructured data.
4. **In-Memory Database (e.g., Redis):** Consider integrating an in-memory database for caching frequently accessed data or for scenarios requiring extremely fast read and write access. This can enhance the real-time performance of the dashboard, especially when dealing with AVL data.
5. **Data Warehousing (e.g., Amazon Redshift):** If a need for complex analytical queries on large volumes of data is anticipated, a data warehouse solution like Amazon Redshift could be beneficial. It's designed for high-performance analytics and aggregations.

Regardless of the chosen storage options, implementing a robust backup and disaster recovery strategy is essential to safeguard against data loss or system failures. This could involve automated backup routines, versioning, and regular testing of recovery procedures. It is a good practice to verify that all chosen storage options meet the security and compliance requirements pertinent to the data being stored. This includes encryption, access controls, and adherence to any industry-specific or governmental regulations. By implementing this multi-tiered storage approach, agencies will be able to leverage the strengths of each solution to effectively handle the diverse datasets in the Cycle Time dashboard tool. It provides a scalable and flexible foundation to support current needs and future expansion.

Dashboard Platform and Hosting

The follow is a summary of the platform and hosting options for this tool.

Table 8: Platform and Hosting Options

Hosting Platform	Data Visualization	Mapping & Spatial Analysis	Data Source Types	Integration Capabilities	Collaboration & Sharing	Cost	System Requirements
Power BI	✓	✗	Various (CSV, Excel, SQL)	Good integration with Microsoft products, REST APIs, SDKs	Collaboration features, sharing with Power BI Pro license	Varies based on licensing plan (e.g., Power BI Pro, Premium)	Internet connectivity, compatible web browser
Tableau	✓	✗	Various (files, databases, cloud)	Connectors, APIs, third-party integrations	Collaboration features with Tableau Creator license	Varies based on licensing plan (e.g., Creator, Explorer)	Internet connectivity, compatible web browser
ESRI	✓	✓	Geospatial data formats, GIS databases	Integration with ArcGIS Online, ArcGIS Enterprise	Collaboration features with ArcGIS Online	Varies based on licensing plan (e.g., ArcGIS Online, Enterprise)	Internet connectivity, compatible web browser
QlikView	✓	✗	Various (files, databases, cloud)	APIs, connectors, and custom integrations	Collaboration features, sharing capabilities	Varies based on licensing plan	Internet connectivity, compatible web browser
Google Data Studio	✓	✗	Various (Google Sheets, BigQuery, etc.)	Integration with Google products, connectors	Collaboration features, sharing capabilities	Free for basic usage, additional costs for advanced features	Internet connectivity, compatible web browser

Agencies can select any platform that will allow the methodology to be programmed and provide the performance measures and outputs in the desired visual format. Based on the resources the project team had available, the team opted to use ESRI as the hosting platform for the demo dashboard/tool designed to calculate plow/treatment cycle times. While ESRI may not offer the same level of data visualization and interactive dashboarding capabilities as Power BI or Tableau, it excels in mapping and spatial analysis functionalities, which are critical for winter maintenance scenarios. The ability to integrate with ESRI's ArcGIS Online and ArcGIS Enterprise provides extensive geospatial data support, allowing for accurate representation of plow routes and assessment of cycle times. Additionally, ESRI offers collaboration features through ArcGIS Online, enabling the research team to work together efficiently on analyzing and improving plow cycle times. While ESRI's licensing costs may vary based on the selected plan, the investment in ESRI's platform is justified by its robust GIS capabilities, making it an excellent choice for winter maintenance analysis and planning.

Development and Testing Stages

To create an effective and user-friendly tool/dashboard, the development and testing process will be divided into three key stages. The first stage, Wireframe/Mockup, focuses on visualizing the tool/dashboard's layout and presenting the data and methodology in a visual format. This stage serves as an opportunity for the committee to review and provide feedback on the proposed design. Additionally, the final tool platform and hosting options will be determined during this stage, ensuring optimal performance and accessibility.

Following the approval of the Wireframe/Mockup, the development process advances to the Alpha Development stage. The development team will work diligently to incorporate the feedback received during the previous stage, ensuring that the tool/dashboard meets all the specified requirements. The committee's input and guidance will be instrumental in refining the functionality and usability of the tool/dashboard. Once all the comments and issues from the Alpha phase have been addressed, the development process enters the final stage: Beta Development. In this phase, the tool/dashboard application is finalized, and accompanying documentation for administration and user support is prepared. Rigorous user testing will also be conducted during this stage. The purpose of this testing is to verify that the tool/dashboard not only meets the agency's user requirements but also performs optimally in real-world scenarios. This thorough testing ensures that the final tool/dashboard is ready for deployment and effectively supports the needs of the intended users.

Wireframe / Mockup Stage

The Cycle Time Dashboard (Figure 3) boasts a thoughtfully crafted layout, optimized for intuitive navigation. The tool was carefully curated with a design and visualization scheme that balances aesthetics with functionality. The dashboard embraces a dark color scheme, providing a sleek backdrop that places emphasis on the informative elements. The map adopts a light gray palette, offering a subdued canvas for data representation. This unobtrusive background allows routes to be symbolized with distinct colors, ensuring clear differentiation and easy identification. Crucially, color is strategically employed to enhance user experience. The Legend, Map, and Bar Graphs are the focal points for color utilization. The Legend serves as a visual guide, employing colors to represent Facilities, Routes, Roads, and other key elements. This aids users in quick interpretation and reference. Its design prioritizes user-friendly functionality, ensuring that users can effortlessly access and interpret critical information.

The dashboard's layout strategically positions key elements such as the Selector, Legend, and Interactive Map for easy identification and interaction. The intuitive placement of these components guides users through the initial selection process, streamlining the analysis of plow/treatment cycle data. The inclusion of an interactive map at the center of the dashboard allows for dynamic exploration and selection of specific areas of interest. This feature empowers users to focus on precise road segments, facilitating a deeper understanding of cycle times and performance metrics. Incorporating informative metric sections on the right-hand side ensures that users have immediate access to essential data summaries. These sections employ clear and concise visualizations, including bar graphs and detailed segment information, to convey insights at a glance.

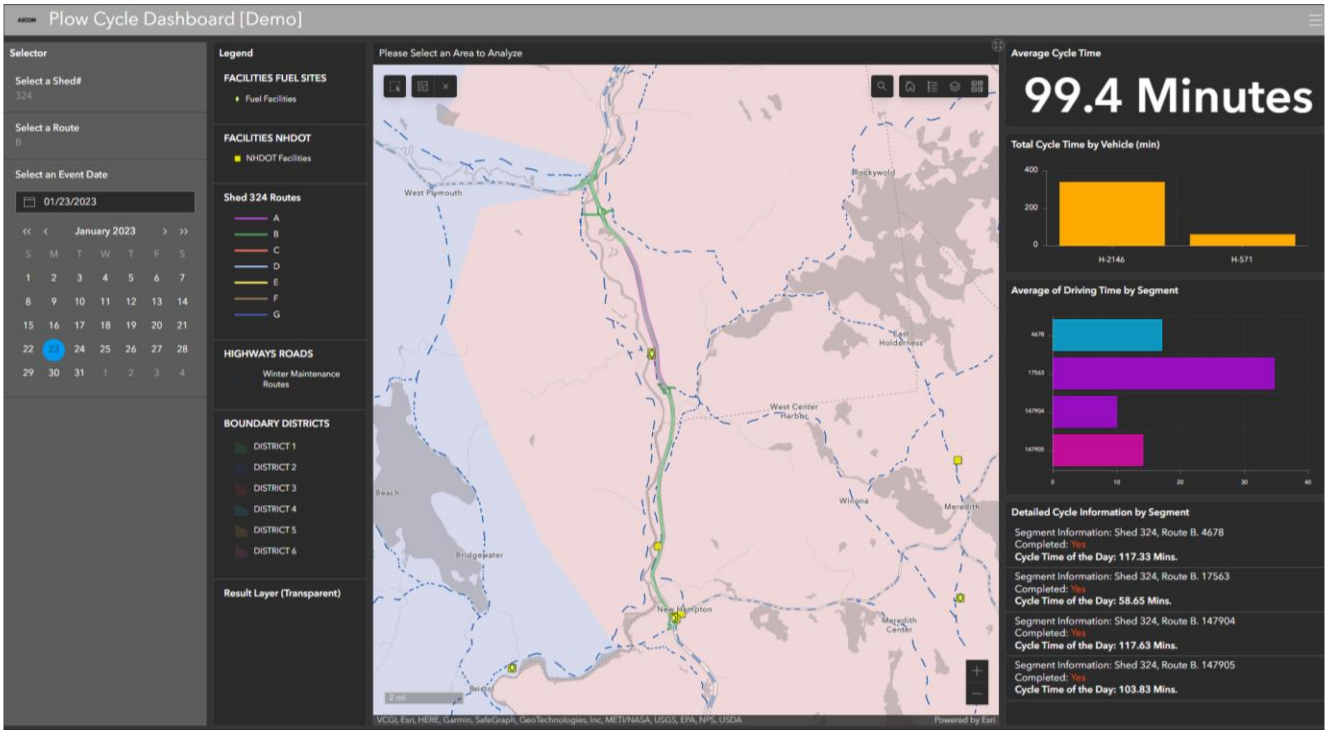


Figure 3. Plow Cycle Times Dashboard Interface

Overall, the Cycle Time Dashboard's layout design prioritizes accessibility, usability, and data clarity. This strategic arrangement of components enables users to efficiently extract valuable insights for informed decision-making in winter maintenance planning.

Alpha and Beta Development Stage

The Cycle Time Dashboard was developed to be a tool for users to analyze plow/treatment cycle data during winter maintenance events. This intuitive dashboard provides precise insights into critical metrics, including average cycle times, vehicle performance, and segment-specific information.

Throughout the development and testing phases, meticulous attention was devoted to optimizing user experience and functionality. The interface guides users through a seamless process of selecting sheds, routes, and event dates, ensuring a smooth analysis. The inclusion of an interactive map and detailed metric sections elevates the tool's utility for users.

Following extensive testing and validation, the dashboard's accuracy and reliability have been tested for quality. It excels in providing actionable data to assist in winter maintenance planning, enabling more efficient and effective operations. The Cycle Time Dashboard exemplifies the potential of GIS technology in refining winter maintenance strategies and stands as a valuable asset for users in the field.

Conclusion

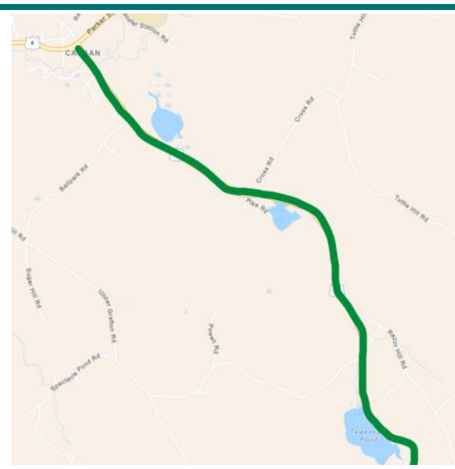
The goal of this framework is to provide the requires, steps, and considerations for the development of a plow/treatment cycle tool. AVL and other required data of each agency may be unique; however, this framework should provide enough information to develop a plow/treatment cycle tool within the agency's environment, as long as the base requirements are met. The cycle time calculation methodology with a case study is presented in Appendix A, and a proof-of-concept tool was developed for the Clear Road's committee.

Appendix A – Methodology and Case Study

Cycle Time Calculation Methodology

Using the variables outlined above, the following is the methodology and logic of the tool.

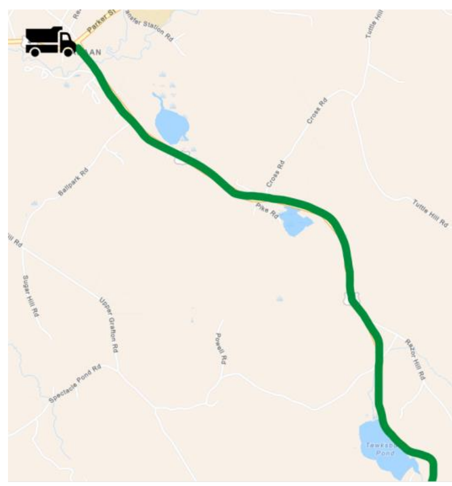
Route A – **Snow and Ice Route**



Step One: Identify that a truck has started on a segment of road via **latitude/longitude. Optional treatment data** - check if **spreader/sprayer is ON,** if TRUE, the following process that will occur when:

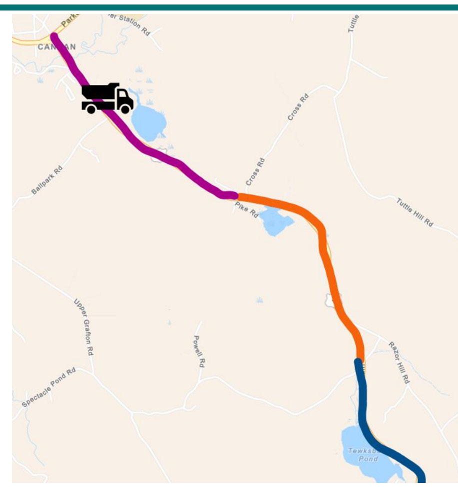
- Start **Date and Timestamp, t_{si}** for the Cycle
- Start **Date and Timestamp, t_{sij}** for the individual pass
- Based on **truck's ID and optional treatment data** - query **truck's plow and treatment width/Capacity, C_T**

where, *i* = road segment and *j* = individual pass along the road segment.



Step Two: Compare C_T to the road segment lane count or width, w_i to get a cycle ratio.

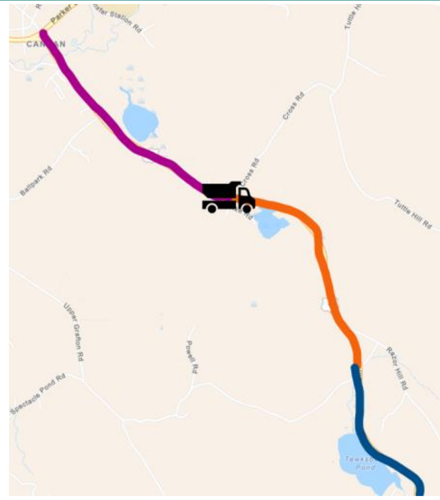
$$Cycle\ Ratio, \theta_j = \frac{C_T}{w_i}$$



Step Three: Calculate duration to treat an individual pass along the segment.

- End **Date and Timestamp,** $\boldsymbol{t_{eij}}$

$$\boldsymbol{Duration, D_j = t_{eij} - t_{sij}}$$

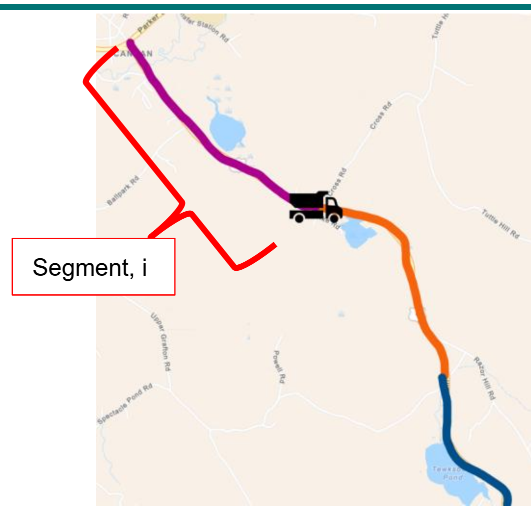


Step Four: Calculate Results

$$\boldsymbol{Total\ Duration\ to\ Clear\ Segment, T_i = \sum_{\theta_j}^{\theta_j => 1} D_j}$$

When $\boldsymbol{\sum \theta_j = 1}$; End **Date and Timestamp,** $\boldsymbol{t_{ei}}$ for Cycle

$$\boldsymbol{Cycle\ Time_i = t_{ei} - t_{si}}$$

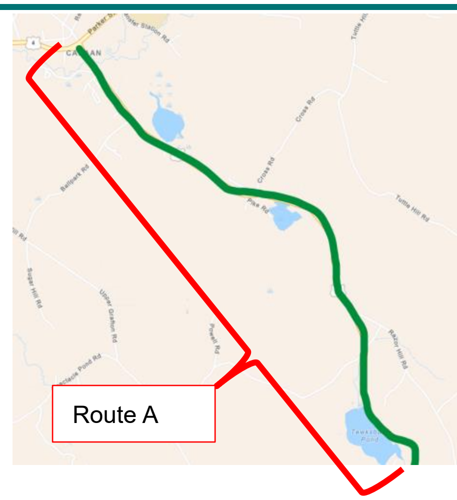


Step Five: Drill up to review multiple segment (total route) Cycle Time

$$\boldsymbol{Total\ Time\ to\ Clear\ Segment, T_{selected} = \sum T_i}$$

For Cycle Time

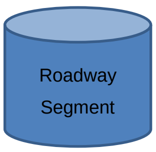
$$\boldsymbol{Cycle\ Time = Latest(t_{si}) - Earliest(t_{ei})}$$

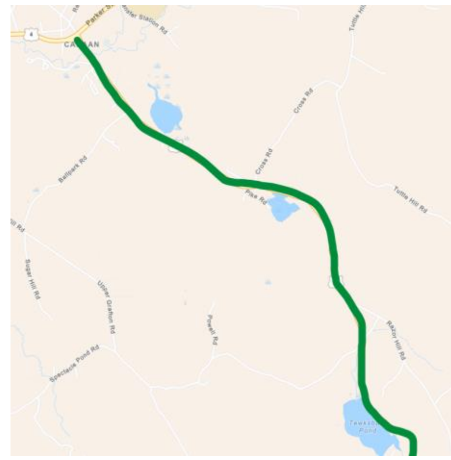


Case Study

Scenario: Trucks equipped with AVL are out treating the agency's routes. Route A has three segments.

Route A – Snow and Ice Route

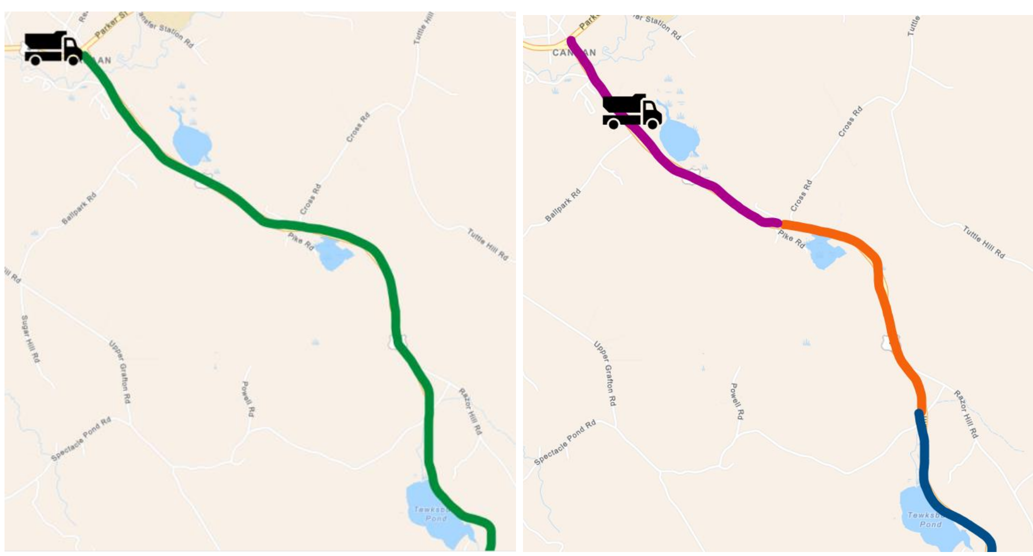




Roadway Segment Database

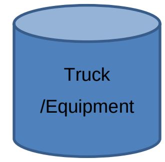
Road Segment (i)	Route	Width (ft)
XXX1	A	24
XXX2	A	24
XXX3	A	24
XXX4	B	36
… n^{th}	…n^{th}	…n^{th}

<u>Step 1</u> – Truck ID 1234 enters Road Segment XXX1 on 12/20/2023 at 2:30 AM. Truck 1234 was not plowing, salting only with spinner at 2.



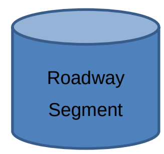
Truck / Equipment Database (C_T)

Truck ID	C_{T1} Width (ft) Plow = ON	C_{T2} Width (ft) Plow = OFF Salt = ON Spreader = 1	C_{T3} Width (ft) Plow = OFF Salt = ON Spreader = 2	C_{T4} Width (ft) Plow = OFF Salt = ON Spreader = 3
1234	12	12	14	18
2234	11	12	14	18
3334	12	12	14	18
... nth	...nth	...nth	...nth	...nth



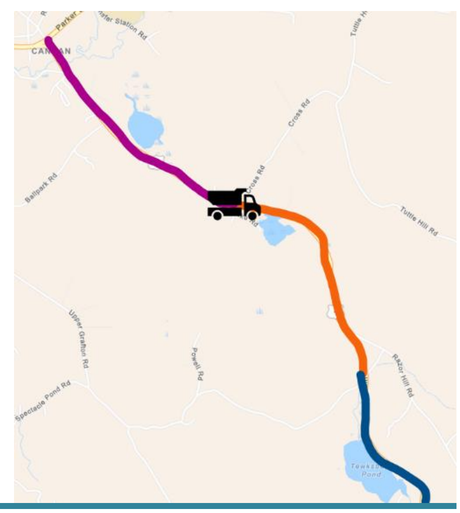
Roadway Segment Database

Road Segment (i)	Route	Width (ft)
XXX1	A	24
XXX2	A	24
XXX3	A	24
XXX4	B	36
... nth	...nth	...nth



Start Date and Timestamp, $t_{start\ on\ Segment\ xxx1}$ for the Cycle = **12/20/2023 at 2:30 AM**

Start Date and Timestamp, $t_{start\ on\ Segment\ xxx1\ for\ first\ past}$ for the individual pass = **12/20/2023 at 2:30 AM**



$$Cycle\ Ratio, \theta_{xxx1} = \frac{C_T}{w_i} = \frac{14}{24} = 0.58$$

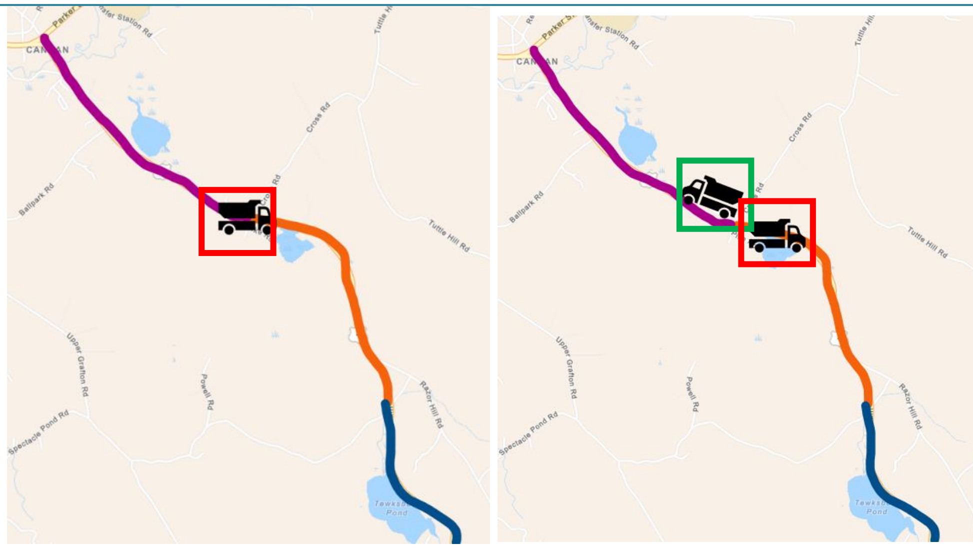
$$Check\ if\ \sum \theta_{xxx1} => 1 \quad \sum \theta_{xxx1} = 0.58\ therefore\ FALSE$$

End Date and Timestamp, $t_{End\ on\ Segment\ xxx1\ for\ first\ past}$ for the individual pass = **12/20/2023 at 2:55 AM**

$$Duration, D_{Segment\ xxx1\ for\ first\ past} = 2{:}55\ AM - 2{:}30\ AM = 0.25\ hours$$

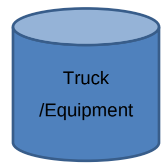
<u>Step 2</u> – Truck ID 1234 enters Road Segment XXX2 on 12/20/2023 at 2:55AM. Truck 1234 was not plowing, salting only with spinner at 2.

Truck ID 2234 enters Road Segment XXX1 on 12/20/2023 at 2:55AM, was not plowing, salting only with spinner at 3.



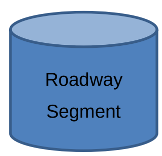
Truck / Equipment Database (C_T)

Truck ID	C_{T1} Width (ft) Plow = ON	C_{T2} Width (ft) Plow = OFF Salt = ON Spreader = 1	C_{T3} Width (ft) Plow = OFF Salt = ON Spreader = 2	C_{T4} Width (ft) Plow = OFF Salt = ON Spreader = 3
1234	12	12	14	18
2234	11	12	14	18
3334	12	12	14	18
... n^{th}	...n^{th}	...n^{th}	...n^{th}	...n^{th}



Roadway Segment Database

Road Segment (i)	Route	Width (ft)
XXX1	A	24
XXX2	A	24
XXX3	A	24
XXX4	B	36
... n^{th}	...n^{th}	...n^{th}



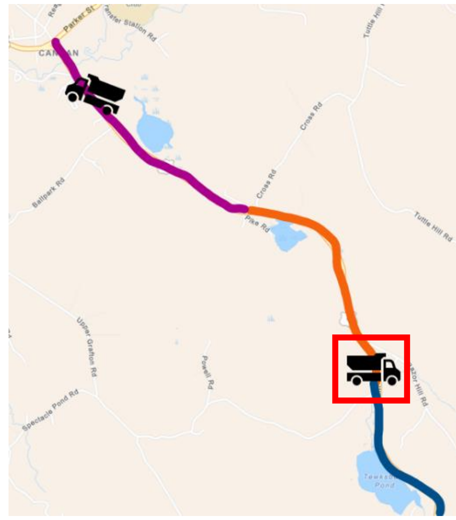
Start Date and Timestamp, $t_{start\ on\ Segment\ xxx2}$ for the Cycle = **12/20/2023 at 2:55 AM**

Start Date and Timestamp, $\boldsymbol{t_{start\ on\ Segment\ xxx2\ for\ first\ past}}$ for the individual pass = **12/20/2023 at 2:55 AM**

End Date and Timestamp, $\boldsymbol{t_{End\ on\ Segment\ xxx2\ for\ first\ past}}$ for the individual pass = **12/20/2023 at 3:05 AM**

$$\boldsymbol{Cycle\ Ratio, \theta_{xxx2\ first\ pass} = \frac{C_T}{w_i} = \frac{14}{24} = 0.58}$$

$$\boldsymbol{Duration, D_{Segment\ xxx2\ for\ first\ past} = 3{:}05\ AM - 2{:}55\ AM = 0.16\ hours}$$

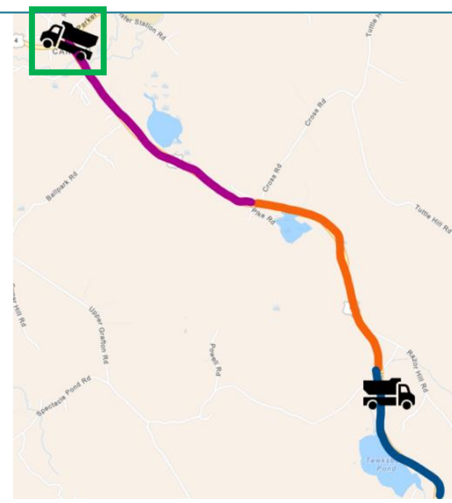


Start Date and Timestamp, $\boldsymbol{t_{start\ on\ Segment\ xxx1}}$ for the Cycle = **12/20/2023 at 2:30 AM**

Start Date and Timestamp, $\boldsymbol{t_{start\ on\ Segment\ xxx1\ for\ Second\ past}}$ for the individual pass = **12/20/2023 at 2:55 AM**

End Date and Timestamp, $\boldsymbol{t_{End\ on\ Segment\ xxx1\ for\ second\ past}}$ for the individual pass = **12/20/2023 at 3:15 AM**

$$\boldsymbol{Duration, D_{Segment\ xxx1\ for\ 2nd\ pass} = 3{:}15\ AM - 2{:}55\ AM = 0.33\ hours}$$



$$\boldsymbol{Cycle\ Ratio, \theta_{xxx1} = \frac{C_T}{w_i} = \frac{14}{24} = 0.58}$$

Check if $\sum \theta_{xxx1} => 1$ $\sum \theta_{xxx1} = 0.58 + 0.58$ therefore TRUE

Reset θ_{xxx1} And Conduct Total Calculations

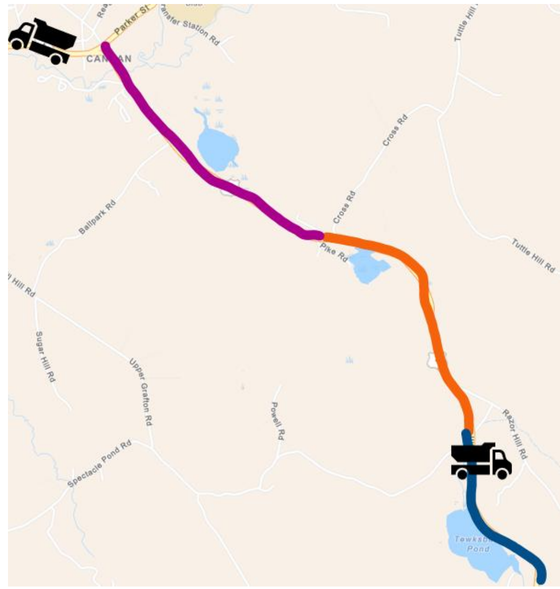
$$\boldsymbol{Total\ Duration\ to\ Clear\ Segment, T_{xxx1} = \sum_{\theta_j}^{\theta_j => 1} 0.33 + 0.25 = 0.58\ hrs}$$

$$\boldsymbol{Cycle\ Time_{for\ segment\ xxx1} = 3{:}15\ AM - 2{:}30\ AM = 45\ mins = .75\ hours}$$

Notes: still need cycle time for Route A (not just segments)

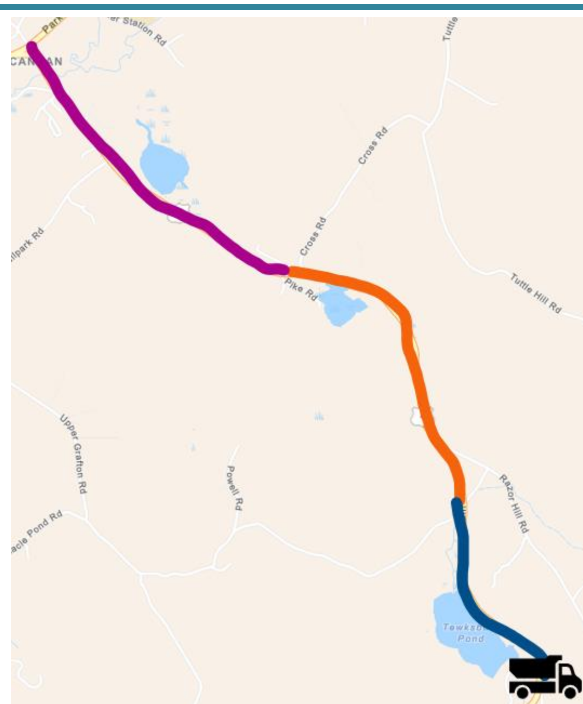
<u>Step 3</u> – Truck ID 1234 enters Road Segment XXX3 on 12/20/2023 at 3:15 AM. Truck 1234 was not plowing, salting only with spinner at 3.

Calculations continue as shown above with spinner 3 data



Truck ID 1234 ended first pass and started second pass on Road Segment XXX3 on 12/20/2023 at 3:25 AM. Truck 1234 was not plowing, salting only with spinner at 3.

Calculations continue as shown above with spinner 3 data



Truck ID 1234 completed second pass on Road Segment XXX3 on 12/20/2023 at 3: 40 AM. And started second pass on Road Segment XXX2.

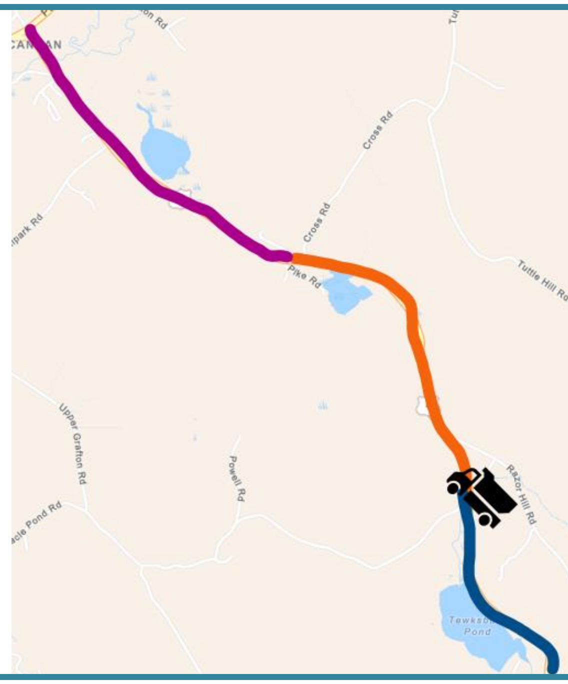
$$\textit{Total Duration to Clear Segment}, T_{xxx3}$$

$$= \sum_{\theta_j}^{\theta_j => 1} 0.16 + 0.25 = 0.42\ hrs$$

$$\textit{Cycle Time}_{for\ segment\ xxx3}$$

$$= 3{:}40\ AM - 3{:}15\ AM$$

$$= 0.42\ hrs = 25\ mins$$



Truck ID 1234 completed second pass on Road Segment XXX2 on 12/20/2023 at 3: 55 AM. Truck 1234 was not plowing, salting only with spinner at 3.

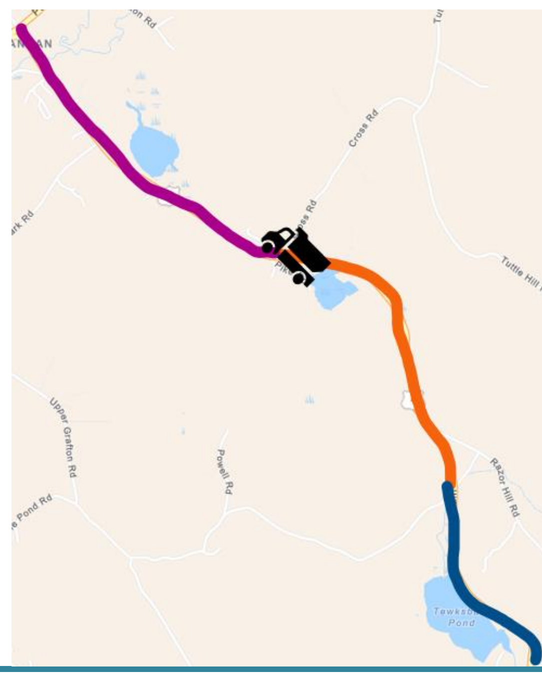
$$\textit{Total Time to Clear Segment}, T_{xxx2}$$

$$= \sum_{\theta_j}^{\theta_j => 1} 0.2 + 0.25 = 0.45\ hrs$$

$$\textit{Cycle Time}_{for\ segment\ xxx2}$$

$$= 3{:}55\ AM - 2{:}55\ AM = 1\ hrs$$

$$= 60\ mins$$



<u>Step 3</u> – Route Cycle Time

For Cycle Time of ROUTE A

$$\textit{Cycle Time} = \textit{Latest}(t_{si}) - \textit{Earliest}(t_{ei})$$

$$\textit{Cycle Time} = 3{:}55\ AM - 2{:}30\ AM = 1.42\ hr = 85\ mins$$