Work Zone Data Collection Documentation Updates

Technical Documentation

Table of Contents

Contents

1 Message Generation

Message generation for this proof of concept toolset is done in the cloud, using an Azure Function Application. This function Is triggered by the upload of a path data file to a specific blob storage container. Before any message generation begins, the associated configuration file is downloaded (found by Work Zone ID naming convention, path data file: path-data—WZID.csv and configuration file: config—WZID.json)

For more information on the messages and file types (Path Data, RSM, WZDx), please read the V2X POC Interface Control Document in the Work Zone Data Collection Toolset repository, located here.

Sample files are located here.

3.1 RSM Message Generation

The first step in message generation is to generate the RSM message. This is done even in cases where the RSM message is not used, such as when the accuracy of the path data is too low (< 2m).

Note: This RSM message generation is based off of the CAMP tool but has been significantly modified. Documentation from the CAMP tool team of the RSM message spec is located here.

3.1.1 Path point compression

Collected path data can be quite dense, especially with long straight roads using a 10Hz GPS. It is possible to reduce the number of path points (often significantly) without losing very much path information.

This application utilizes the algorithm described in Design Method One of SAE J2945/1, Path History Design. One minor modification was made to include mapped feature locations in the compressed path data. These features include lane closure start/end, end of lane closure/opening tapers and the start/end of worker presence.

In the same step, lane center lines are generated from the compressed path data. The path data is taken to be the center line of the driven lane (specified in configuration file). The lane width (also from configuration file) is used to extrapolate the location of the other lane center lines, perpendicular to the instantaneous bearing of the vehicle.

3.1.2 Message Segmentation

Very long work zones may be split into multiple RSM messages. Under following conditions, multiple RSM message segments must be generated:

1. When number of nodes generated per lane exceeds 63 (2..63)

2. When expected UPER encoded message size exceeds 1100 octets (PDU Upper limit is 1500) + space for header, security and other stuff for about 400 octets

3.1.3 Common Container

The RSM common container contains basic information about the road event/work zone including the dates, times of day that worker is active, type of work (Cause Codes), and mapped nodes of the approach region (compressed path before the reference point, broken down by lane).

A common container dictionary object is populated for each of the message segments.

3.1.4 RSZ Container

The RSM RSZ container contains all of the mapped nodes of the work zone (from compressed path data). Information in the RSZ container includes speed limits, presence of workers, node locations, lane closures and lane tapers.

An RSZ container dictionary object is populated for each of the message segments.

3.1.5 UPER (Binary) Conversion

Each complete message segment is converted from XML to UPER format. A java-based ASN.1 encoder is used to make the conversion. To reduce the complexity of the Azure Function Application, a self-contained linux java installation is included with deployed files and is accessed directly to run the conversion. If the conversion fails, the Azure Function throws an error but continues on to WZDx message generation.

3.2 WZDx Message Translation/Generation

After the RSM messages are generated, a custom RSM to WZDx translator is utilized to generate a single WZDx message.

This translator takes RSM messages and other information (from the configuration file) and returns a WZDx message. The translation is relatively simple, as the messages have similar content. The most

complicated step is to transpose and aggregate the RSM's lane-based information into a single path with status information for each lane, with a new GeoJSON feature collection for each status change.

GUIDs are generated for each of the IDs of the message.

WZDx messages are validated using Schema provided by the USDOT JPO WZDx repository.