Connected Vehicle DSRC Message Parser User Guide

1. Introduction

The Connected Vehicle (CV) DSRC Message Parser (DMP) is a tool designed to make it easier to work with network packet capture (PCAP) files containing dedicated short-range communication (DSRC) data from CV applications. To facilitate this end, the DMP converts PCAP files containing binary encoded DSRC messages such as Basic Safety Messages (BSM), Signal Phase and Timing (SPaT) messages, and Geographic Intersection Description (GID/MAP) messages, into a more human (and machine) readable format known as comma separate values (CSV). These CSV files can be opened by many programs such as Microsoft Excel or Microsoft Notepad as they are simply text files that adhere to the CSV specified format.

2. Document Scope

The purpose of this document is to provide instruction for end users of the DMP software on how to install, run, configure, and operate the DMP software. It is aimed at non-developer users who are familiar with the Windows 7 operating system but not necessarily familiar with how Java programs work or how the command line works.

3. Installation

3.1. Dependencies

The DMP depends on the Java Runtime Environment (JRE) version 1.8 or higher. The JRE is available from Oracle's website at http://www.oracle.com/technetwork/java/index.html. Find the version of the JRE appropriate for your operating system (DMP is designed for Windows 7 64 bit) and download and install it. All the other dependencies of the DMP are bundled inside the application's Java Archive (JAR) file.

3.2. Application Installation

The DMP can be installed in any folder that the user has permission to write to. The application is installed by opening the distribution zip file and extracting the JAR into the desired folder.

4. Starting the Application

The DMP software package is distributed as an executable JAR file. If your system is configured to associate JAR files with the Java Runtime Environment this application can be started simply by double clicking the JAR file. If this association was not created by the JRE installer it is necessary to run the DMP

from the command line by manually invoking the JRE. To begin this process open the Windows CMD prompt by pressing Win + R, typing “cmd.exe” and pressing enter, as shown in Figure 1.

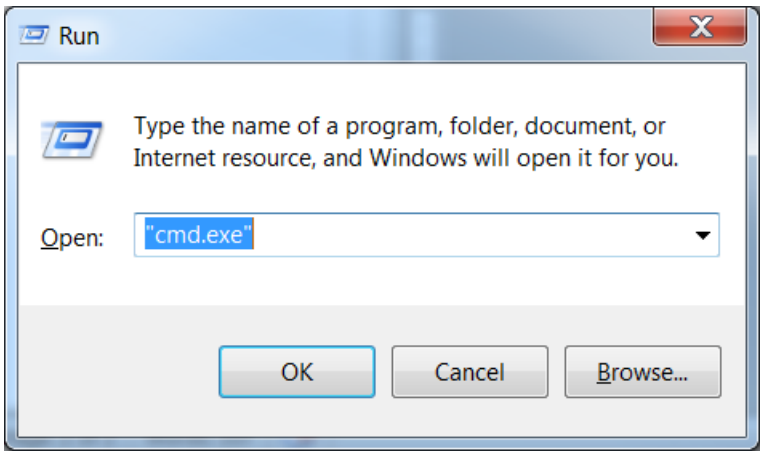

Figure 1 – Starting the command prompt

You should see a window similar to the following Figure 2:

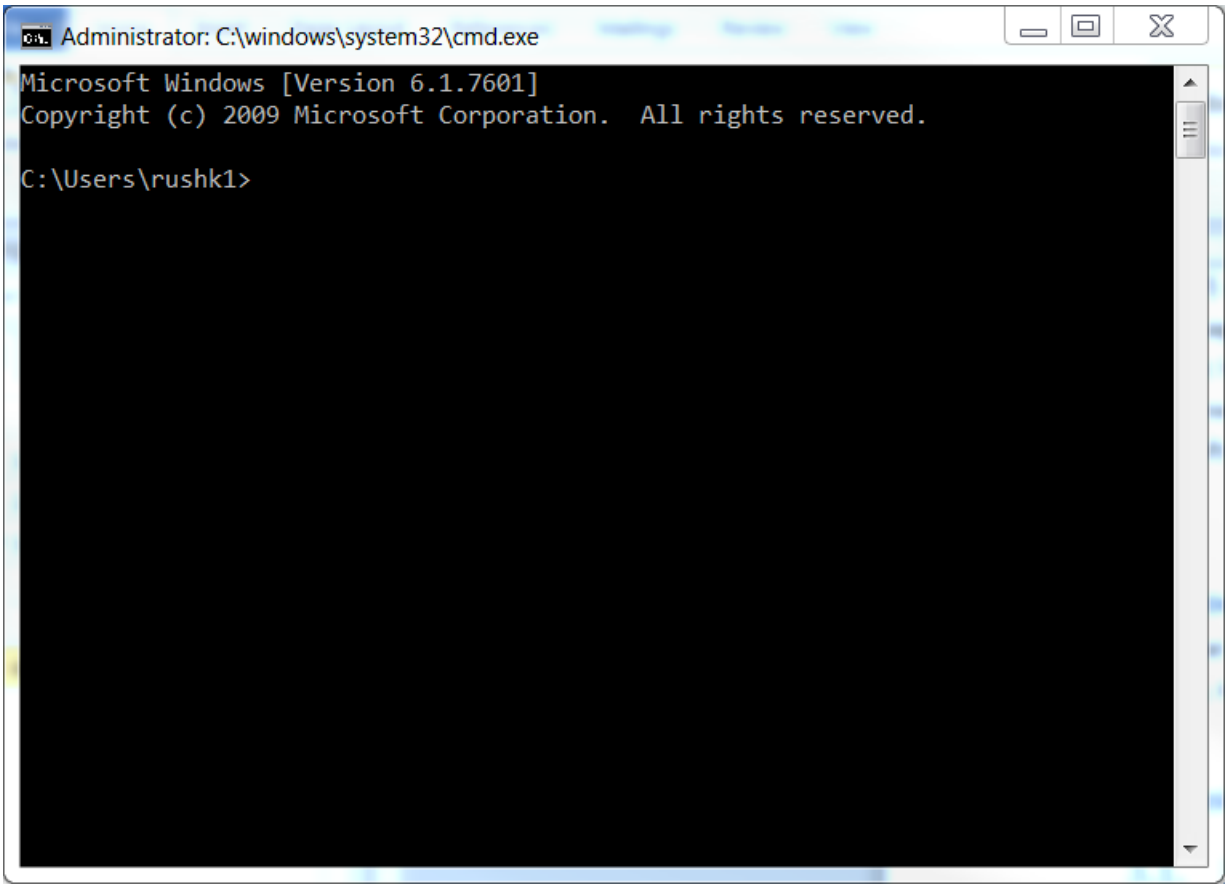

Figure 2 – The command prompt window

Next, type “cd <absolute/path/to/folder/containing/dataparser.jar>” and press enter, where <absolute/path/to/folder/containing/dataparser.jar> is the Windows path to the location you extracted the JAR file to such as “C:\Users\<username>\Desktop” or “C:\Program Files\dataparser”. Finally, type “java –jar dataparser.jar” and press enter.

5. Configuring the Application

Upon starting the application the screen shown in Figure 3 will be visible:

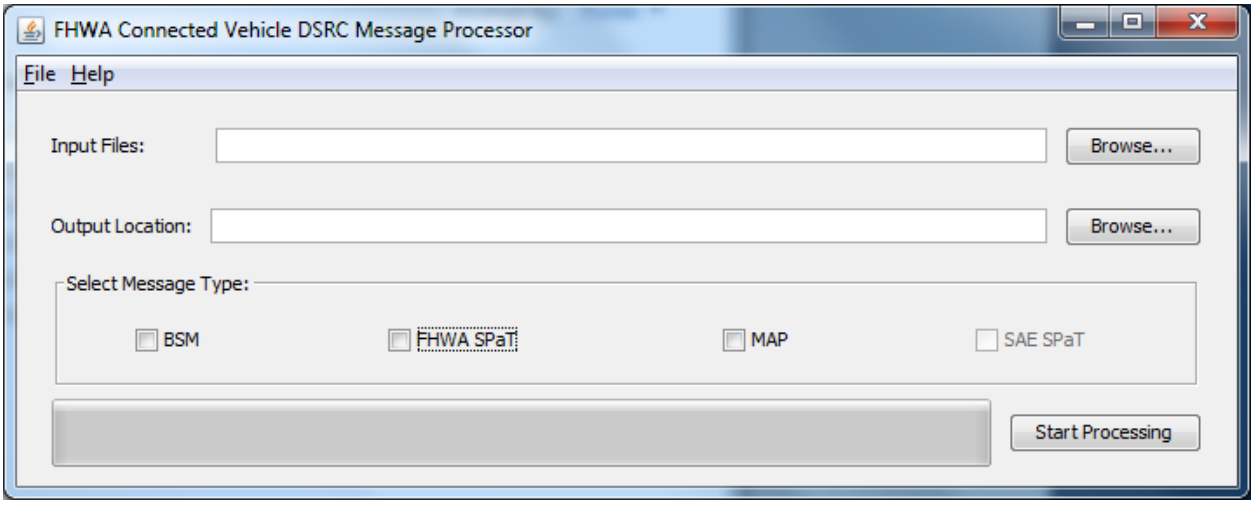

Figure 3 – Application main window

From this screen you must configure the parameters of the processing step before it can proceed. First, select an input folder. This folder should be where the PCAP files you want to process are stored. This folder can be specified either by typing the full path to the folder in the text box or by clicking the “Browse…” button and making your selection through that window, as in Figure 4.

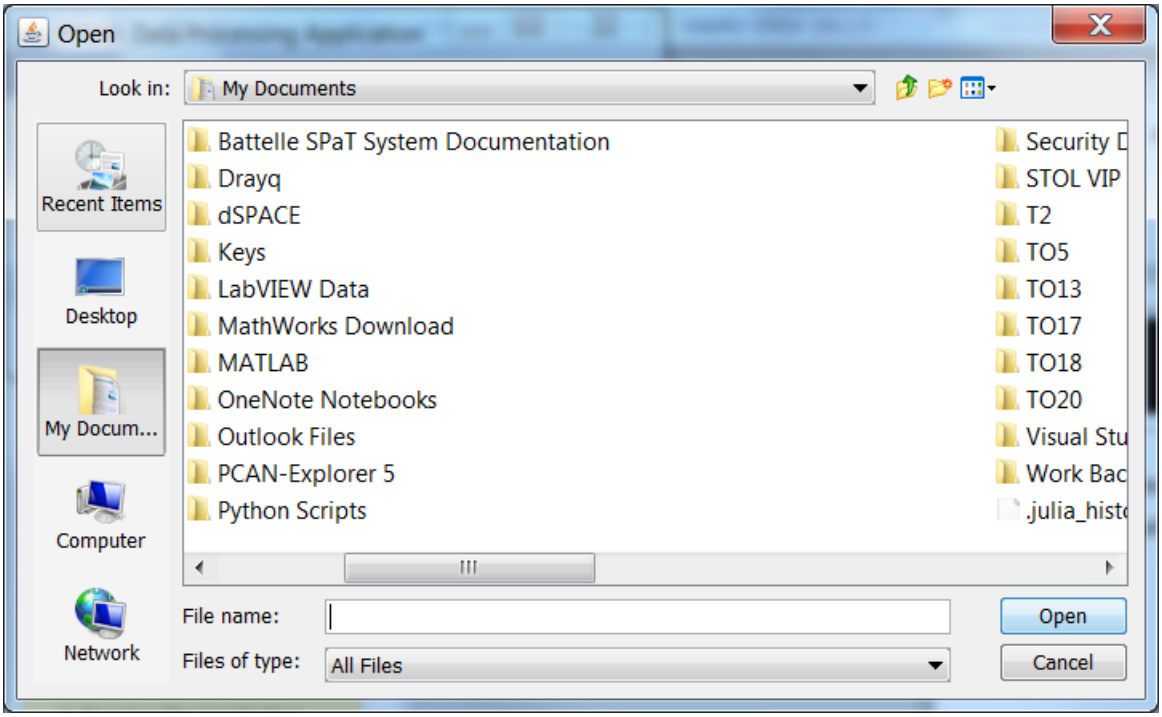

Figure 4 – Browsing for input directory

Next, you must select an output folder. This is the folder in which the files containing the converted PCAP data will be created. If the folder you wish to store the converted files in does not exist it must be created prior to beginning the processing step. Like the input folder, this configuration parameter may be set by typing the full path into the text box or by choosing a folder with the “Browse…” button.

Finally, you must select the message types you would like to process in each file. Click the checkbox next to each message type you would like to process to select it. At least one message type must be selected to begin the processing step. The screenshot in Figure 5 shows an example of what the window looks like when it is fully configured.

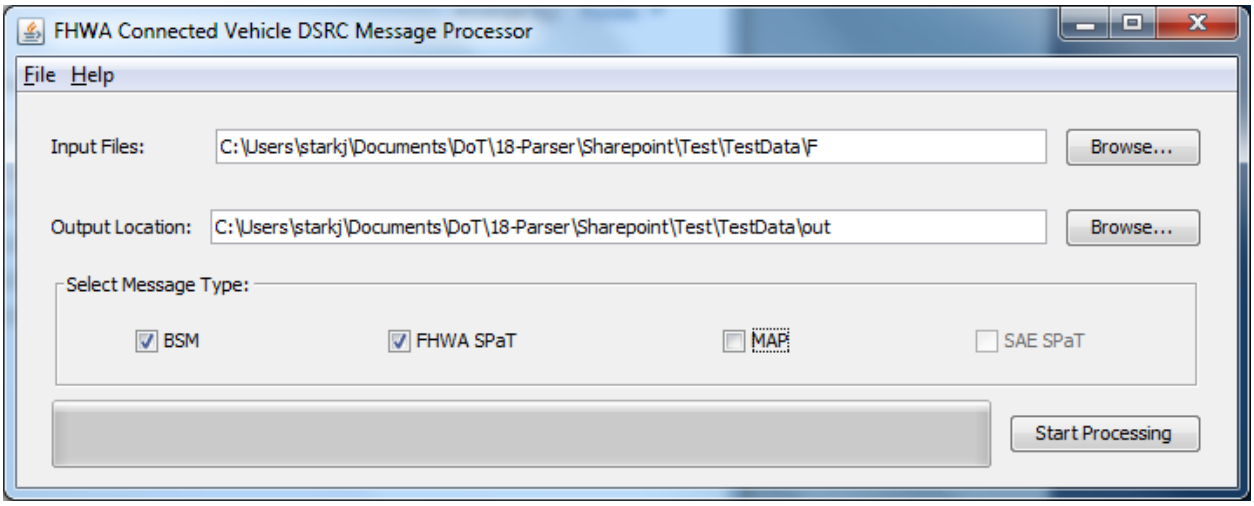

Figure 5 – Ready to start processing

6. Operating the Application

Once the application is configured the user can press the “Start Processing” button to begin the processing of input files. The application will read the files in the input folder and convert the names of the input files into the names of the output files. In the event a filename conflict occurs between an already existing (or pending to be created) file and an output file name the application will pop up a dialog prompting the user for a strategy to use to resolve the conflict, like the one shown in Figure 6. Clicking the “Rename” button will attach a numbered suffix to the end of the output file’s name or increment it if one already exists. Clicking the “Overwrite” button will cause the new file to replace the old file, losing any data contained in the old file. Clicking the “Skip” button will cause the output file to not be written, losing any data that might have been converted.

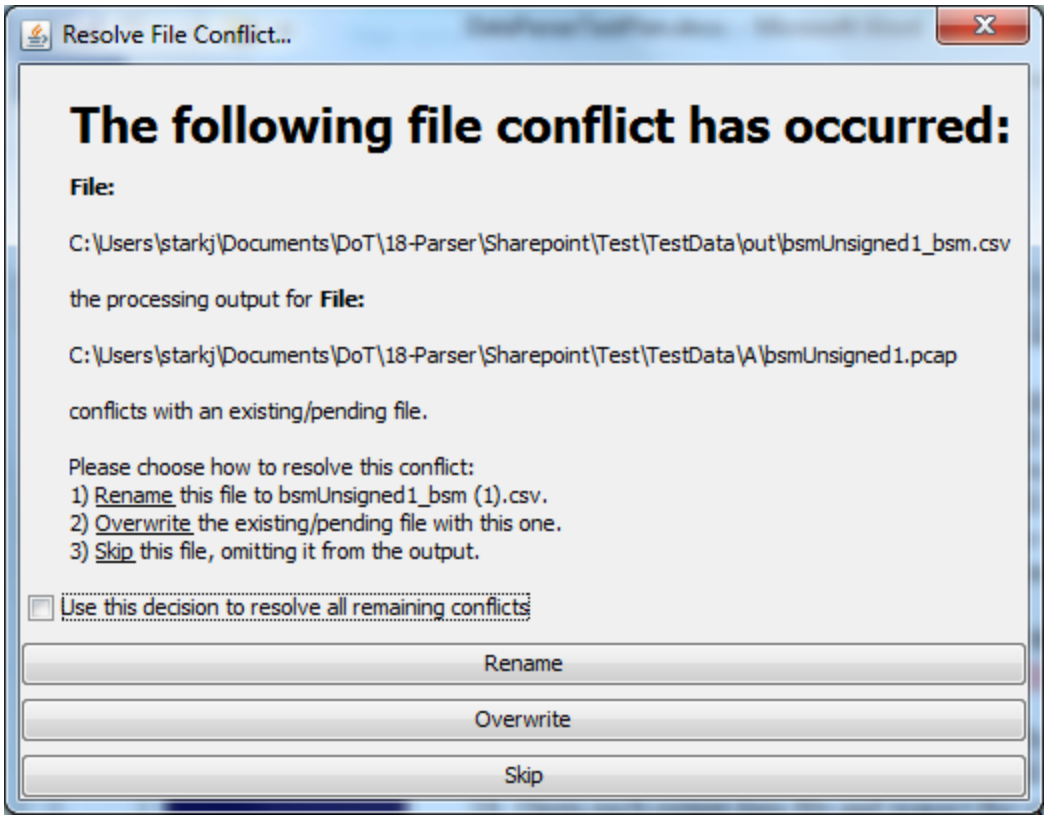

Figure 6 – Handling a filename conflict

Once all conflicts have been resolved the application will display information about the processing tasks it has generated. Clicking “OK” will cause the DMP to proceed with the processing step and make changes to the files in the output folder. Clicking “Cancel” will bring the application back to the configuration step.

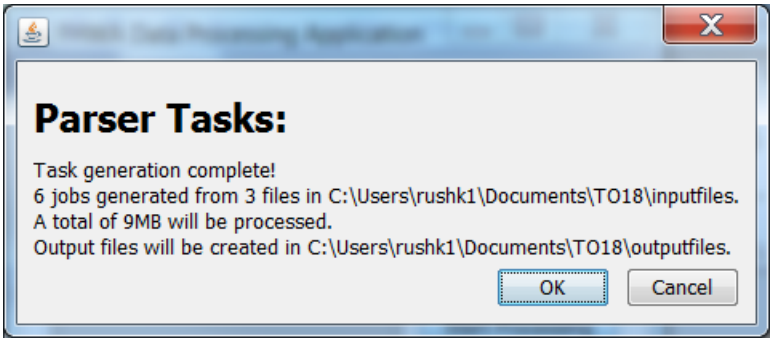

Figure 7 – Tasks ready to run

During processing the application will display its progress via the progress bar located at the bottom of the main window. This progress is measured in terms of bytes processed out of total bytes to be processed. Once the processing is complete the following dialog will pop up with information about the completed task, as shown in Figure 8.

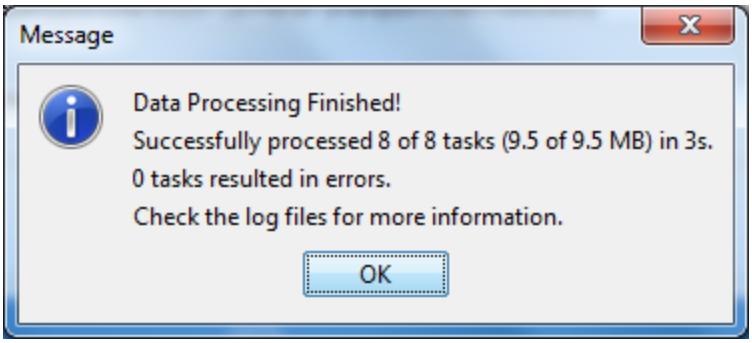

Figure 8 – Processing complete

After the application has completed its processing it will return to the main window with the same configuration state it had to begin processing. From here you may edit the configuration and run again or simply exit the application.

Logs for the application are stored in the “logs” folder in the same folder as “dataparser.jar”. These logs may be opened with any text editor such as notepad or even with Microsoft Excel. The log files will provide much more detailed information about the application’s behavior.