LTE-V2X Test Data Overview

There are two groups of data available for public download.

- **Moving Vehicle Data** – transmission and receipts for the two moving vehicles under test
- **Raw Test Data** – all test data, including data for the congesting devices

Moving Vehicle Data:

- Line-by-Line transmission and receipt data for the two moving vehicles under test is organized by Scenario 1, 2, 3, and 4 (as shown below).
- The test parameters for these data sets can be mapped using the ***USDOT LTE V2X Test Parameters*** spreadsheet. Each tab of the spreadsheet corresponds to a specific scenario. The test number is included in the individual CSV file names. Each test number has up to 5 trials. More detailed instructions are provided on the next slide.

LARGE-SCALE CONTROLLED FIELD DATA

The following datasets provide transmission and receipt data for the two moving vehicles under test. Each test case can be mapped to the Field Test Parameters (linked above). The distance between the moving vehicles is also included in these data files.

- Scenario 1 Moving Vehicle Data
- Scenario 2 Moving Vehicle Data
- Scenario 3 Moving Vehicle Data
- Scenario 4 Moving Vehicle Data

Moving Vehicle Data

Reading the Moving Vehicle CSV Files

- Each device under test is transmitting (**txdevice**) and receiving (**rxdevice**)
- Each test case (***test_num***) consists of approximately five trial runs (***trial_num***).
- Each transmission and receipt are captured in time: (**tx_time**) and (**rx_time**)
  - When rx_time = 0, the packet was missed
- **Veh1** corresponds to the txdevice ID listed first in the file name
- **Veh2** corresponds to the txdevice ID listed second in the file name
- The lat and lon columns are used to calculate the distance between the two moving vehicles (**veh1_veh2_distance(m))**

txdevice	rxdevice	test_type	test_num	trial_num	tx_time	rx_time	lat	lon	veh1_veh2_distance(m)
XX_XX_XX_0d_69_d3	XX_XX_XX_ba_e_tcpdump	bsm	14	1	1629823713	0	40.1976	-75.1457	295.9757623
XX_XX_XX_ba_e_tcpdump	XX_XX_XX_0d_69_d3	bsm	14	1	1629823713	0	40.19965	-75.1479	295.9757623

Accessing the Raw Data on AWS

Raw Data on Amazon Web Services (AWS)

- The **RAW DATA** link on the bottom of the V2X Analysis page provides a link to an Amazon Web Services page where the raw test data is stored for public access.

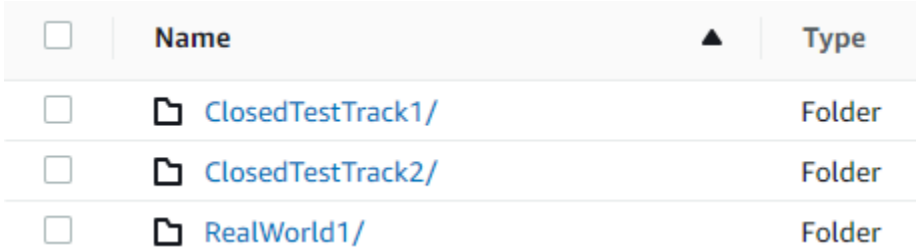
Instructions for Accessing the Raw Data in AWS

- After clicking the RAW DATA link, you will be sent to the AWS Sign-in page
  - If you don't already have an account, click on "Create a new AWS account"
  - Input your email address and create an AWS account name and click "Verify email address"
    - Retrieve verification code from email and enter it on the site
    - Create a password
  - Enter your Contact Information
    - This information is required by AWS; it will not be collected or saved by U.S. DOT
  - When it asks for Billing Information, **leave these fields blank**. You do not need to provide credit card information. An email will be sent to you from AWS that will allow you to enter the site without completing this step (usually ~15 minutes)

AWS Raw Data Organization

AWS Data

- The raw data on AWS is organized into the following folders:



ClosedTestTrack1 = Testing Completed at Willow Grove

ClosedTestTrack2 = Testing Completed at Turner Fairbank Research Center

RealWorld1 = Testing Completed in Tampa, Florida

Congestion and Interference Data

- The majority of the test data can be found in the *ClosedTestTrack1* folder, in the *250DeviceTest* sub-folder
  - The 250DeviceTest sub-folder is organized by Scenario 1, 2, 3, 4
    - The test parameters for these data sets can be mapped using the ***USDOT LTE V2X Test Parameters*** spreadsheet. Each tab of the spreadsheet corresponds to a specific scenario. The test number is included in the individual CSV file names. Each test number has up to 5 trials

AWS Raw Data Organization

AWS Data Files

- After choosing one of the top folders (i.e. *ClosedTestTrack1*) and one of the sub-folders (i.e. *250DeviceTest*), there are two folders for acme and bsm data. The majority of data is under "bsm" data.

Acme = Data for devices running the Acme application (a small number of RSUs)

BSM = Data for all of the OBUs under test

Reading the CSV Files

- Each file contains the transmissions and receipts for each device under test (txdevice/rxdevice) for a particular test number and trial (as printed in the file name).
- The **tx_time** and **rx_time** represent each packet sent and received (or not, if rx_time = 0)
- **Latency** and **Inter-Packet Gap (IPG)** statistics are shown for each line item
- Metrics, such as **Packet Error Rate (PER)** and **Consecutive Packets Missed**, are averaged across all trial runs per test number, so the resulting number is only printed on one line per test number.

AWS Raw Data Download

To download multiple data files at once, you need to follow the directions below to set up AWS CLI (Command Line Interface):

- Download and run the AWS CLI MSI installer **for windows (64-bit)**:

  https://awscli.amazonaws.com/AWSCLIV2.msi

  *Follow the instructions on the following webpage to install AWS CLI on **other Operating Systems (macOS, Linux)**:*

  https://docs.aws.amazon.com/cli/latest/userguide/getting-started-install.html

- To confirm the installation, open the Start menu, search for cmd to open a command prompt window, and at the command prompt use the “aws --version” command.

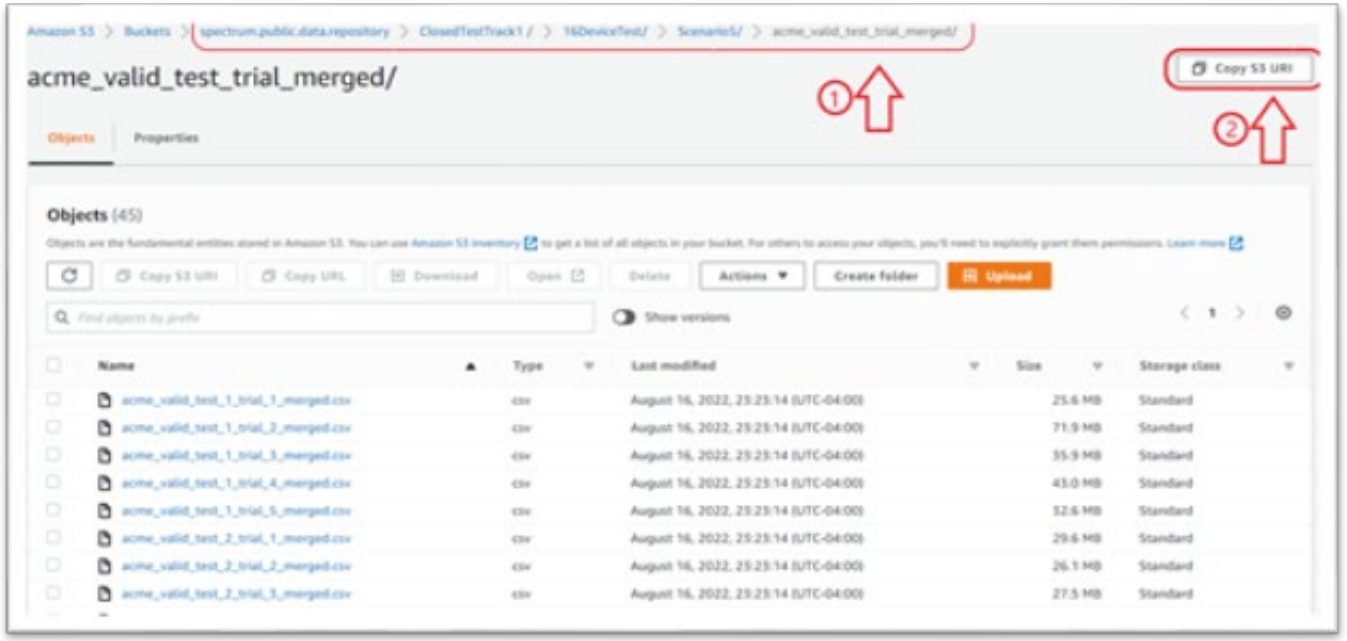
```
C:\> aws --version
aws-cli/2.4.5 Python/3.8.8 Windows/10 exe/AMD64 prompt/off
```

AWS Raw Data Download

Once AWS CLI MSI is installed, follow these instructions to download multiple files or folders:

- Use the link provided on the USDOT webpage, to navigate to the desired file/folder location on AWS S3.

  The following screenshot shows the file location for all ACME tests, conducted for 16-device spectrum testing, under Scenario 5:

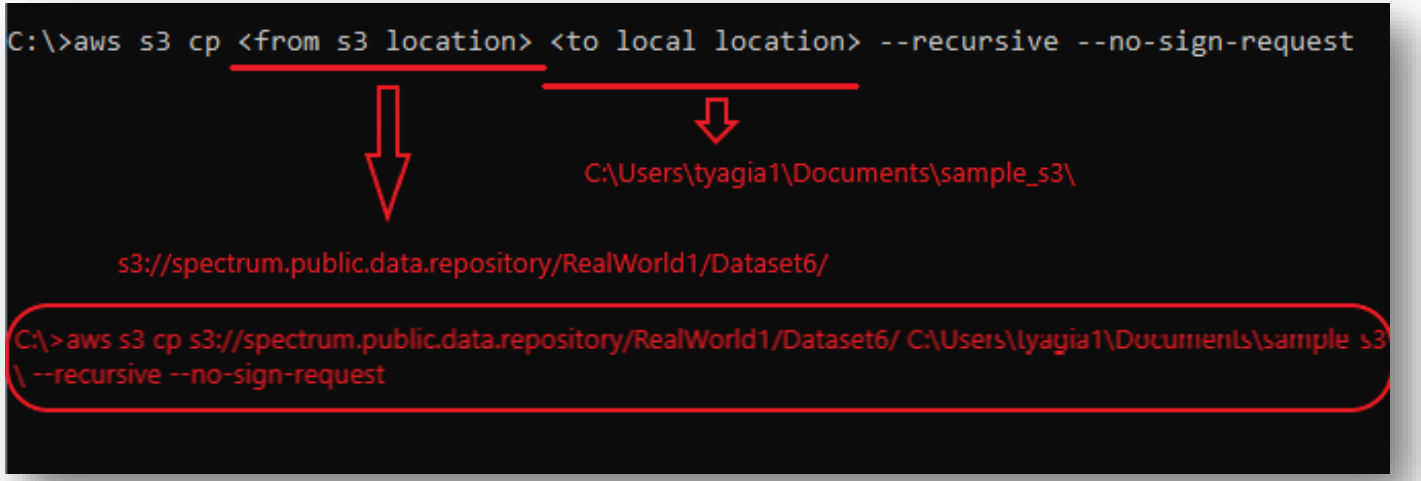


AWS Raw Data Download

- Once you have navigated to the desired location (as shown by label “1” in screenshot above), from where you need to download all available files and folders, click on the “Copy S3 URI” button on the top-right corner of the AWS Console page. It’s shown and labeled as “2” on the screenshot above.

- The general command to download S3 data to your local machine is:
  C:\> aws s3 cp <from s3 location> <to local location> --recursive --no-sign-request

- Replace the <from s3 location> with the S3 URI copied in step 2
- Replace the <to local location> with the directory path on your computer where you want to download the data to.

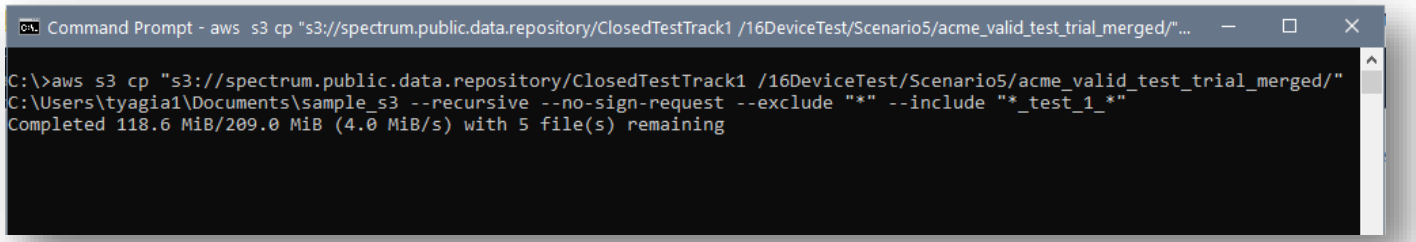


AWS Raw Data Download

- For more specific control over which files to download, please use the “exclude” and “include” command-line parameters, as shown in the following example.

- Consider all the files shown in screenshot 1, but we are only interested in the files related to “test case 1”. To download the subset of files, for test case 1, please use the command as:

C:\>aws s3 cp <from s3 location><to local location> --recursive --no-sign-request --exclude "" --include "*_test_1_*"*



⇨ Please note the order in which “exclude” and “include” parameters are passed to the command.

⇨ For more information on these parameters and their usage, please refer https://docs.aws.amazon.com/cli/latest/reference/s3/#use-of-exclude-and-include-filters

Performance Metrics

Performance Metrics used to inform analysis (*helps answer questions on previous slide*):

- Packet Error Rate (Tx (transmitted packet) → Rx (received packet))
  - We calculate PER using two main methods:
    1) (Tx – Rx) / Tx, per second
    2) (Tx – Rx) / Tx, per 50-meter bin
- Consecutive Missed Packets
  - Number of times two consecutive packets are missed
- Channel Busy Ratio
- Inter-packet Gap (IPG)
  - Time between two successive received packets
- Transmit Time Interval (TTI)
  - Time between two successive sent packets

Performance Metrics

TTI *– Transmit device perspective*

- TTI provides an insight into how often the application was able to broadcast periodically.
- TTI is affected by the congestion control mechanism added on the application layer.
- TTI tend to be in clusters of distinct means indicating congestion control mechanism taking over when enabled.

IPG *– Receive device perspective*

- IPG is a result of channel congestion state and the application transport and control
- IPG is measured with successfully decoded APDU.