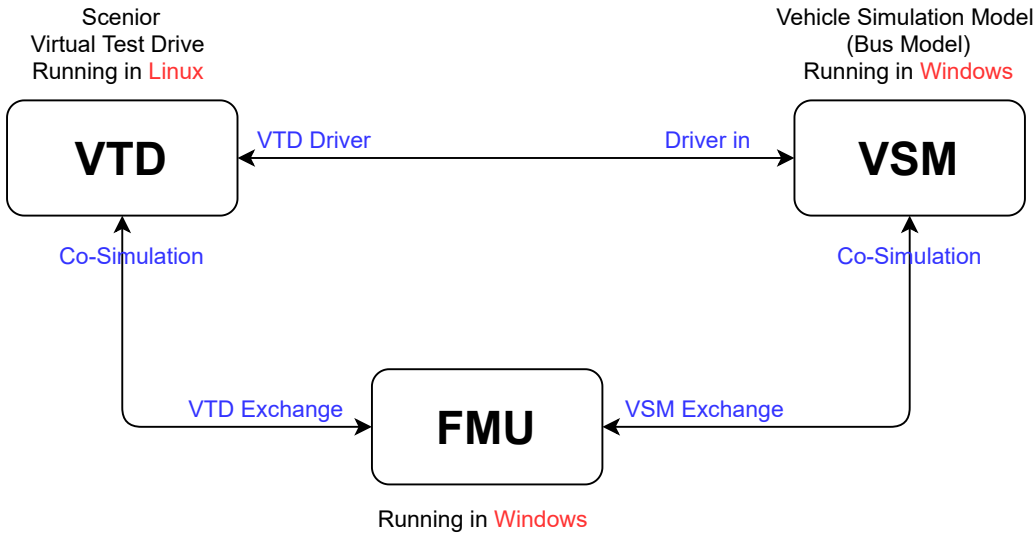
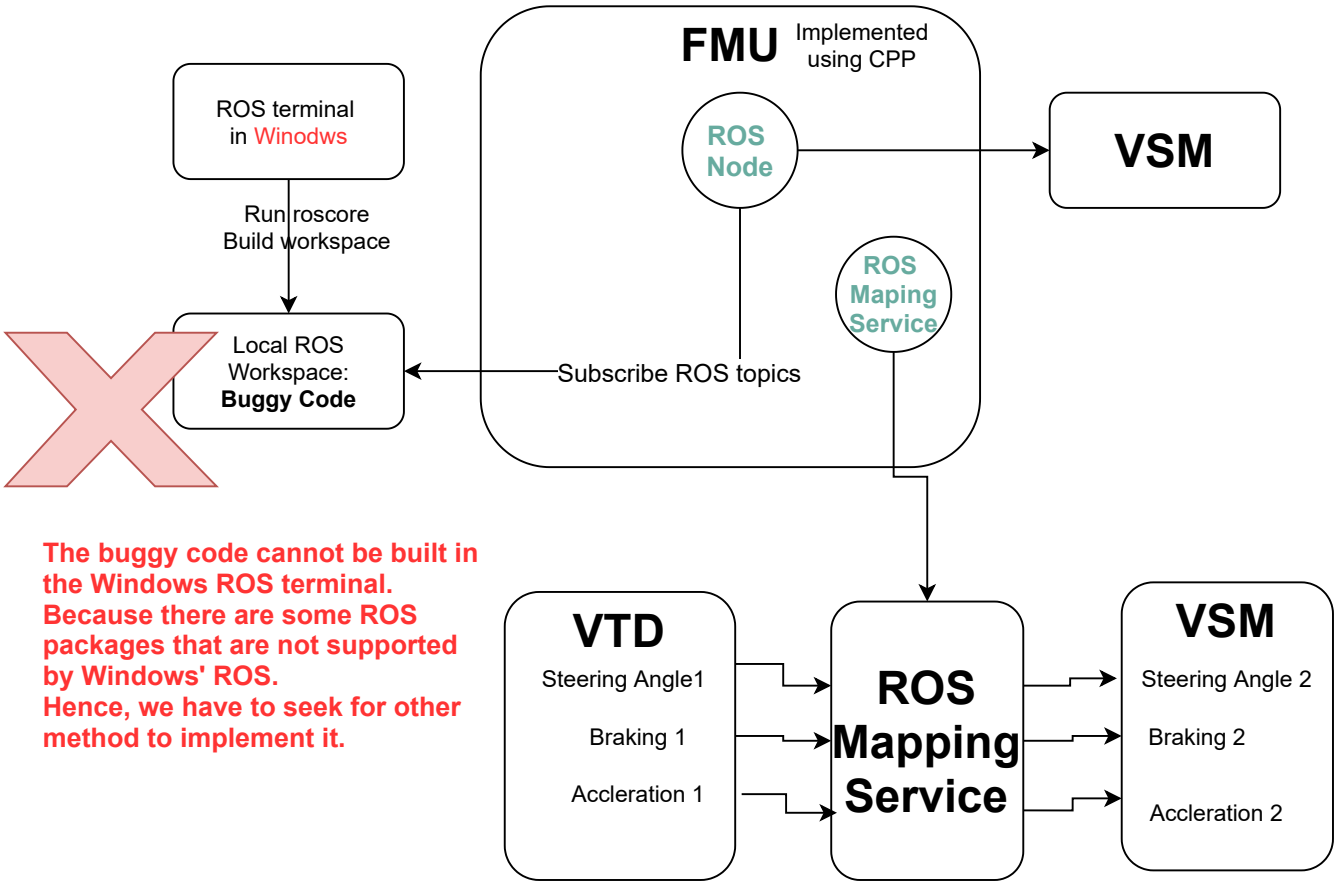
Inital Project Structure

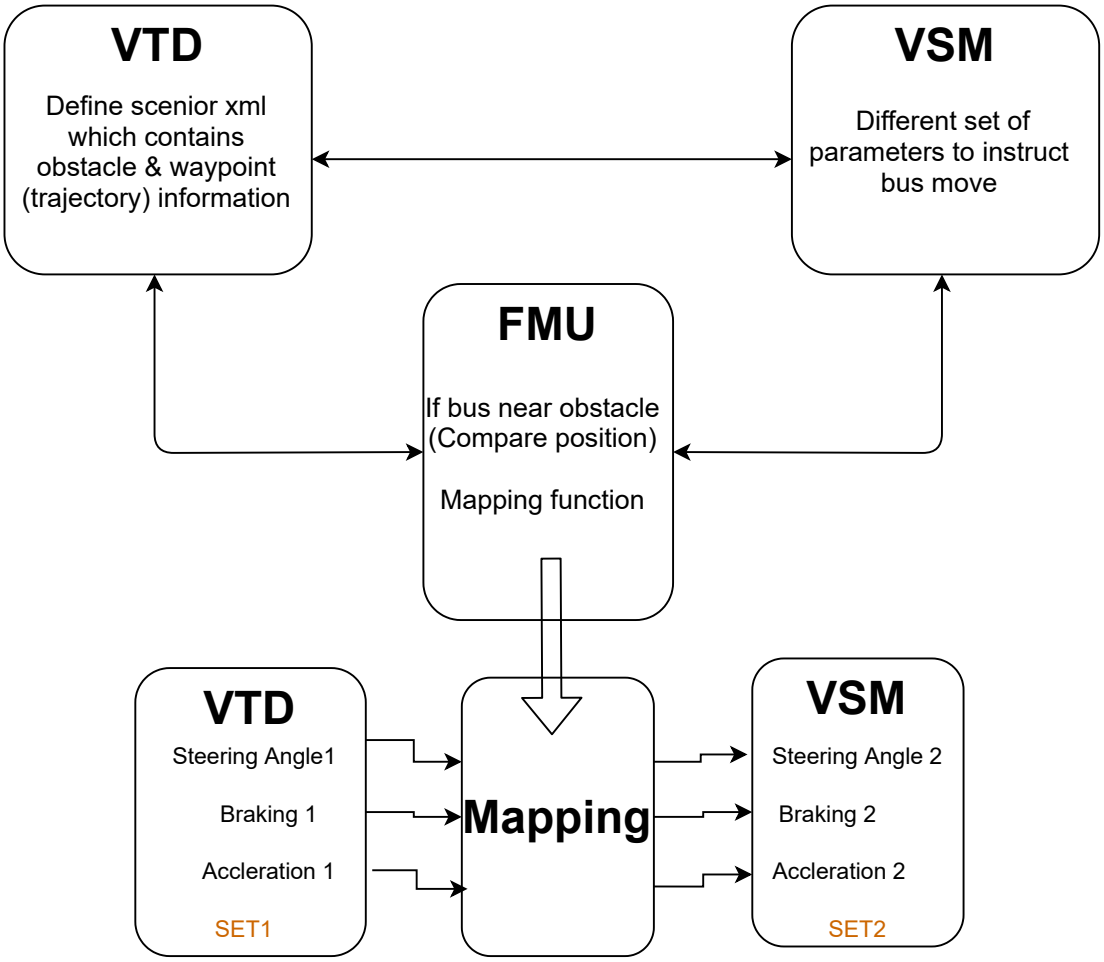


ROS & Mapping FMU



The buggy code cannot be built in the Windows ROS terminal. Because there are some ROS packages that are not supported by Windows' ROS. Hence, we have to seek for other method to implement it.

New Project Design



At the start, the bus should follow the trajectory to move at a predefined **steering angle1, braking1 and acceleration1**. We assume the bus will always localize itself (known waypoints).

Meanwhile, the obstacle information is also defined in the scenario (fixed position). Once the bus detects there is an obstacle in front (bus's position near the obstacle's position), it will trigger the Mapping function in FMU.

Through the Mapping function, the new set of parameters (**steering angle2, braking2 and acceleration2)** will be generated. The bus will move at this set for a period of time. Then the bus will return to the previous SET.

Open questions:

1. What are suitable values for mapping function parameters?
   SET1:
   SET2:

2. Previously we implement it based on ROS. Since we need to use ROS buggy code, we should take out ROS for simplicity.
   How to implement mapping functions without ROS?
   Traning needed

3. Assume the bus always localizes itself, how we derive the bus position.
   Ideally, VTD should have two ports to give the bus's position and obstacle's position. Then we can just compare two positions to trigger the FMU Mapping function.
   How to compare positions (two waypoints information)? What is the format of position data?
   Talk to AVL

4. What is the ideal time period for bus moving at SET2?