Joder Un tren empezará siempre en una estación, creándose una relación entre estación y tren, a continuación, si puede moverse a la dirección que quiere y no tiene que esperarse, debido a que la vía este ocupada, empezará su camino tardando dos minutos en llegar. Una vez allí esperará un minuto a que se suban/bajen pasajeros. En el caso de que el tren lleve una dirección y la estación sea el final de la línea, este cambiará su dirección y dará la vuelta. Por otro lado, los trenes pueden averiarse, se pondrá out of order y se le sacará de la vía para que no la ocupe y puedan pasar los demás trenes. Además, cuando queramos también podemos repararlos y podrán volver a moverse.

Aquí mostramos algunos puntos importantes a tener en cuenta del diagrama que podrían facilitar su comprensión:

* Explicar que las líneas son independientes de los trenes, por lo que no tiene sentido hacer composiciones entre líneas y trenes. De hecho, podemos tener 0 trenes en una línea en algunos casos, y los trenes pueden cambiar de línea, aunque en su funcionamiento siempre siguen una.
* Líneas y segmento son agregación porque un segmento puede pertenecer a varias líneas
* Una estación se conecta a otra mediante dos vías, las cuales forman un segmento. De esta forma, como no hay estaciones aisladas, una estación siempre tiene dos vías y un segmento. Siempre habrá un número par de vías en una estación.
* No tiene sentido una línea sin segmento ni un segmento que no pertenezca a una línea.
* Un segmento puede pertenecer a más de una línea, y una línea está formada por segmentos.
* En un momento dado un tren está en una estación o en una vía.
* Debido a las multiplicidades, dos trenes no van a estar en la misma vía

Aunque en algunas fotos de nuestros diagramas aparezca un atributo llamado “aforo” en estación, pero es irrelevante y en el código .use se ha eliminado, ya que no tenía utilidad.

En nuestro diagrama de clases tenemos varias clases que modelan nuestro problema, para empezar tenemos una clase tren que representa los trenes con sus distintos atributos necesarios para implementar un correcto funcionamiento y ciertas operaciones que modelaran el comportamiento del tren como son move, stop, ir a estación, esperar en estación, reparar o averiar. Por otro lado tenemos una clase reloj, con su clase abstracta, estas son para modelar el tiempo y el paso del tiempo.

Para representar la red de metro tenemos una clase estación que se referirá a las estaciones o desde un punto de vista mas matemático nuestros nodos, cada par de estaciones se relacionará por dos vías, una para cada sentido. Este par de vías compondrán el segmento, ya que un segmento puede pertenecer a varias líneas o a una sola. Y por último como ya hemos mencionado tenemos una clase línea que modelará las líneas de metro. Cabe mencionar que cada tren pertenece a una sola línea o a ninguna, ya que en la vida real obviamente cada línea tiene sus trenes, ya que estos no pueden saltar de línea a línea.