Low-Comotovation: System Design Document

Michael Ghaben

Contents

0.1 Versioning & Authorship

Version 0.1
Low-Comotovation ©
Software Design Specification: Low-Comotovation
Status: Preliminary Release: Software Design Review

0.2 References

During the development of this document, IEEE 1016 was utilized.

0.3 Purpose

This document will specify the architecture and design of the Low-Comotovation train system. It shall discuss the structural and design and considerations of the train system and the accompanying subsystems of the train system. It shall also detail design considerations in vital subsystems.

0.4 Stakeholders & Concerns

The stakeholders of this document are anticipated to be the following:

- Future Design Teams: Future design teams are anticiapted to utilize this document to guide their usage of the track controller system
- Pittsburgh Rail Company: The rail company utlizing the Software Design Specification (SDS) to guide the development of physical systems associated with the software

Future design teams associated with the continued development beneift from increased documentation of the original system by allowing for more efficient software design procedures in future revision by potentially unrelated developers.
The benefits to the Pittsburgh Rail Company from a detailed software design specification are twofold. First, a detailed SDD provides developers of railway hardware the information required to produced a paired system. Second, a documented SDD allows the Pittsburgh Rail Company to evaluate the designs ability to meet specifications for vitality.

1 Introduction

To ensure safe, predictable, and reliable operation of the system, there are three primary considerations:

1. *Vitality:* Vitality of a system within this document referse to a safety-critical system.

2. *Testability:* Any system implemented must be easily tested to ensure reliability

3. *Modularity:* Any system designed must reuse code wherever possible

2 System Design Use Cases

In this section, we detail the use cases of each subsystem. The use case of each subsystem is accompanied by brief descriptions of the use cases.

2.1 Track Model

In this subsection, the use cases of the train model are provided.

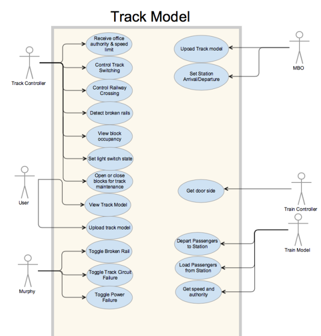

Figure 1: Track model use case diagram

2.2 Track Controller

In this subsection, the use cases of the track controller are provided.

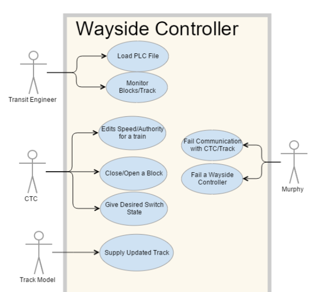

Figure 2: Track controller use case diagram

2.3 Train Model

In this subsection, the use cases of the train model are provided.

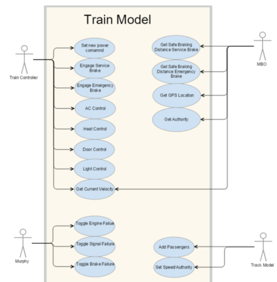

Figure 3: Train model use case diagram

2.4 Train Controller

In this subsection, the use cases of the train controller are provided.

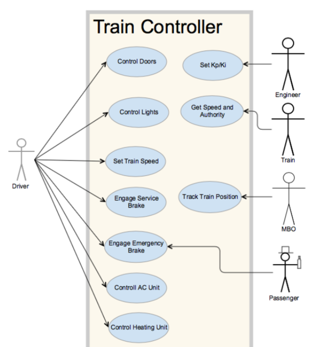

Figure 4: Train controller use case diagram

2.5 Moving Block Overlay

In this subsection, the use cases of the Moving Block Overlay (MBO) are provided.

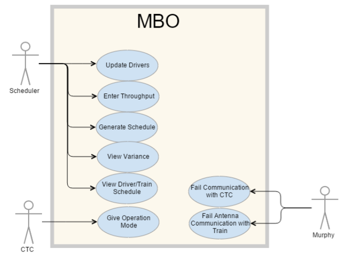

Figure 5: Train controller use case diagram

Table 1: Update

Actors	Scheduler
Description	The Scheduler is able to update the list of drivers. This will change whether or not a driver is able to be scheduled.
Data	filename
Stimulus	Click drivers button
Response	Loops through a CSV file to add all the drivers to the list of drivers. When adding a driver, a driver object will be created with the entered properties. This object will then be added to the Driver Schedule where it can be accessed as part of the list.
Comments	There will be a default file so that it can be saved between sessions.

Table 2: Enter Throughput

Actors	Scheduler
Description	The Scheduler enters the number of trains they would like to be on the track at a certain point in time.
Data	number of trains
Stimulus	Click submit button
Response	The number of trains is entered by the scheduler. This is used to generate both the train and driver schedules for both MBO and FB modes.
Comments	

Table 3: View Train/Driver Schedule

Actors	Scheduler, CTC
Description	Scheduler can see a list of all trains, as well as their station arrival times. Scheduler can see a list of all current drivers, as well as their corresponding break times and current train.
Data	train ID, arrival times, driver name, ID, break times
Stimulus	Updates triggered by clock
Response	Two tables will be displayed, one for the train schedule, and one for the driver schedule. The train schedule will list IDs as the rows and station names as the columns. Each cell will contain the time that train will arrive at that station. The driver schedule will what train they are on at what times. It will also show whenever they start and stop work and when they are on breaks.
Comments	The table that is displayed will automatically update itself when triggered by the clock.

Table 4: View Variance

Actors	Scheduler
Description	Scheduler can see a list of all trains, as well as their corresponding speed and current position. The suggested speed and authority will be displayed as well as the variance between the two.
Data	train ID, speed, suggested/actual position/authority, variance
Stimulus	Updates triggered by clock
Response	In Fixed Block mode the current block will have to be kept track of based on past block occupancy. In MBO mode the position can be gotten through GPS.
Comments	In Fixed Block mode the position is denoted as the current block. In MBO mode the position is denoted as the current block and the distance into that block.

Table 5: Generate Schedules

Actors	Scheduler
Description	When required a schedule will be generated based on the input data. This will then be displayed for the scheduler/CTC. It is used to dispatch trains and calculate a path for a train.
Data	number of trains, track data
Stimulus	On launch, change in number of drivers, clock triggered
Response	A schedule will be generated for trains and drivers. It will have to take into account the mode of operation (MBO or FB), speed limits, track occupancy, drivers break times, and other variables.
Comments	Can only happen in automatic mode - schedule will be either fixed block or MBO depending on dispatcher's selection of mode.

Table 6: Give Operation Mode

Actors	CTC
Description	The CTC sends the mode of operation whenever it is changed.
Data	mode
Stimulus	CTC changes the mode.
Response	The mode is updated in the MovingBlockOverlay class. Any shutdown procedures to switch between modes are performed.
Comments	The default mode will be manual.

Table 7: Fail Communication with CTC

Actors	Murphy
Description	Murphy breaks communication between CTC and MBO.
Data	communication failure
Stimulus	CTC clicks Fail Communication with MBO button.
Response	Since scheduling will be unavailable without communication with the MBO, the CTC will be forced into manual mode and let the dispatcher know with a message.
Comments	

Table 8: Fail Communication with Train

Actors	Murphy
Description	Murphy breaks communication between Train and wayside.
Data	communication failure
Stimulus	Click Fail Communication with Train button.
Response	The MBO can no longer receive the GPS position of individual trains and is therefore unable to safely operate in MBO mode. So a transition must be made to either Fixed Block mode or to manual mode.
Comments	

2.6 Centralized Train Control

In this subsection, the use cases of the Centralized Train Control (CTC) are provided.

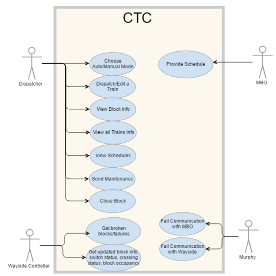

Figure 6: Centralized Train Control use case

3 Class Diagrams

In this section, we detail the use class diagrams for each subsystem..

3.1 Track Model

3.2 Track Controller

3.3 Train Model

3.4 Train Controlled

3.5 Moving Block Overlay

3.6 Centralized Train Controller

4 Sequence Diagrams

In this section, we detail the sequence diagrams each subsystem.

4.1 Track Model

4.2 Track Controller

4.3 Train Model

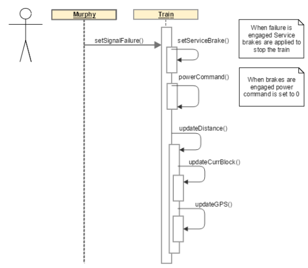

Figure 7: Train controller use case diagram

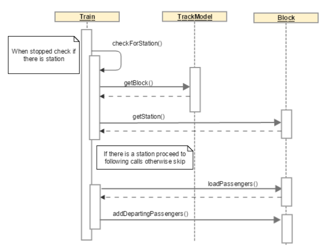

Figure 8: Train controller use case diagram

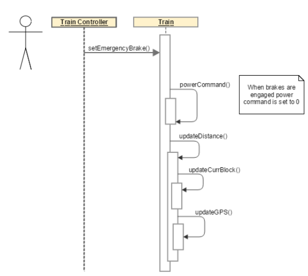

Figure 9: Train controller use case diagram

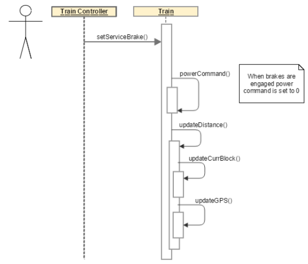

Figure 10: Train controller use case diagram

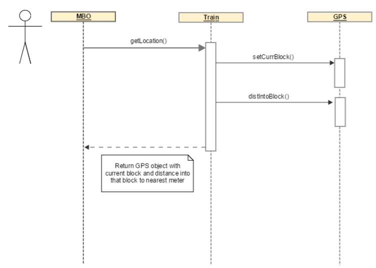

Figure 11: Train controller use case diagram

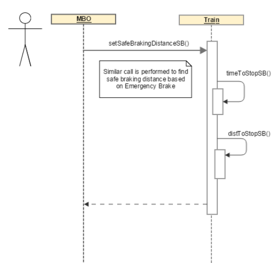

Figure 12: Train controller use case diagram

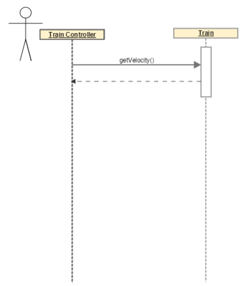

Figure 13: Train controller use case diagram

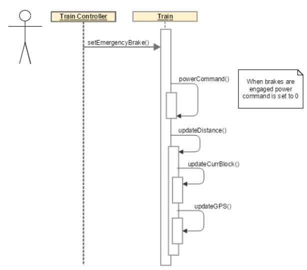

Figure 14: Train controller use case diagram

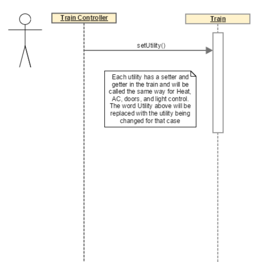

Figure 15: Train controller use case diagram

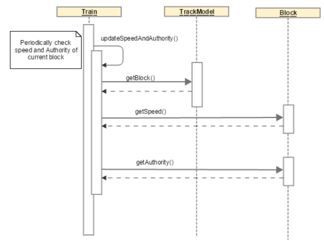

Figure 16: Train controller use case diagram

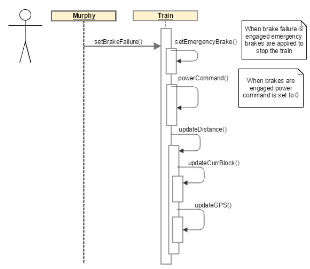

Figure 17: Train controller use case diagram

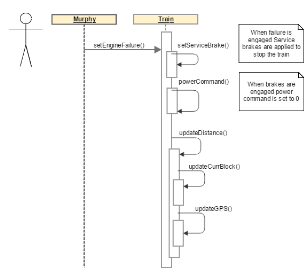

Figure 18: Train controller use case diagram

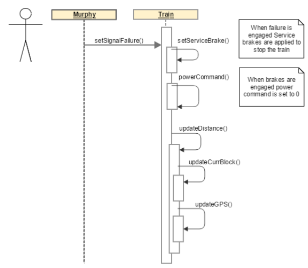

Figure 19: Train controller use case diagram

4.4 Train Controller

4.5 Moving Block Overlay

4.6 Centralized Train Controller