Document title: Intersection Traffic Supervision System of Systems

Version	Status	Date
0.5	draft	2020-10-02

Author	Contact
Zoltan Micskei	E-mail, Phone

Intersection Traffic Supervision System of Systems (ITS-SoS)

Authors: Márton Elekes (BME), Zoltán Micskei (BME)

Abstract

This document defines the Intersection Traffic Supervision System of Systems (ITS-SoS). ITS-SoS is an Internet of Things (IoT)-based system, that supervises traffic at railroad and motor-road intersections. ITS-SoS is composed of various sensors that can detect the rail and motor vehicle traffic and a central supervisory system that monitors traffic flow and can send notifications to vehicles if necessary. The sensors can be placed in highly customizable setups, and the functionality of the central supervisory system depends on the available information from the current setup. The dynamic configuration and orchestration of the systems in ITS-SoS are provided by the Arrowhead Framework.

NOTE: Although this System of Systems was inspired by real use cases, this is a fictional specification created for educational and training purposes. The document is deliberately simplified and contains errors.

Table of contents

1. System of Systems Overview

The main objective of the Intersection Traffic Supervision System of Systems (ITS-SoS) is to monitor traffic flow in intersections. The ITS-SoS is specifically designed for intersections of railroad and motor-road with limited safety equipment (e.g. crossing barriers). An example intersection is depicted on Figure 1.

Figure 1. Example intersection (Photo by Michael Morse from Pexels)

ITS-SoS includes various types of sensors for detecting and reporting road vehicles and railways. ITS-SoS contains a central supervisor that sends notifications to vehicles whether they can proceed through the intersection based on the information received from other systems inside the SoS. ITS-SoS is based on the Arrowhead Framework[1].

Systems. ITS-SoS contains the systems depicted on Figure 2.

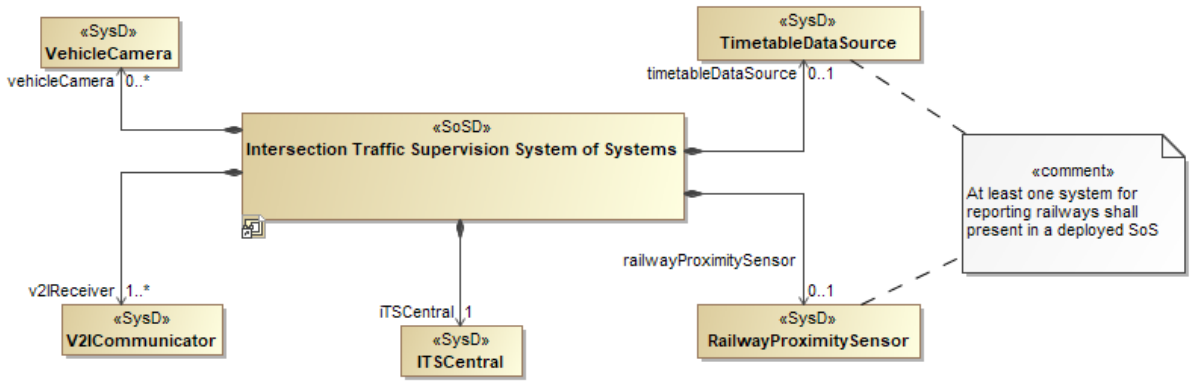

Figure 2. Systems contained in ITS-SoS

The responsibilities of these systems are summarized in Section 2.

[1] Arrowhead Framework: https://arrowhead.eu/arrowheadframework/this-is-it

Main assumptions. As managing traffic flow can be quite complex, the following assumptions are made for the current prototype of the ITS-SoS.

1. There is only one railway track in the intersection.
2. All vehicles are equipped with V2I technology.

Note: Further iterations of the development can relieve these assumptions.

2. Systems

The ITS-SoS consists of the systems described in Table 1.

Table 1. Summary of systems inside the Intersection Traffic Supervision System of Systems

System name	Summary
ITSCentral	This is the central supervisor system inside ITS-SoS that receives data from the other systems, store the current state of the intersection, analyses the current traffic situation and decides what notifications to send out to vehicles.
VehicleCamera	A system consisting of physical camera(s) for detecting road vehicles and their intended movements, and reading their license plates.
V2ICommunicator	A system managing communication with vehicles using Vehicle-to-Infrastructure (V2I) protocol. The system is capable of reporting vehicles approaching and leaving. Moreover, the system can send notifications
RailwayProximitySensor	A system encompassing the necessary hardware equipment and software to detect occupancy of railway tracks and reporting approaching and leaving railways.
TimetableDataSource	A system managing the railway timetable and reporting trains that should arrive or leave the given intersection.

The ITS-SoS is highly configurable thanks to the dynamic, service-oriented nature of the Arrowhead Framework. However, any deployed ITS-SoS shall meet the following constraints.

- There shall be exactly one *ITSCentral* system in the SoS.
- At least one *V2Icommunicator* shall be present in the SoS.
- The VehicleCamera is optional.
- At least one RailwayProximitySensor or TimetableDataSource shall be present in the SoS to have information about railway traffic.

3. Use Cases

This section defines the main actors connected to the System of Systems, and describes the high level use-cases regarding relations and information exchange between actors.

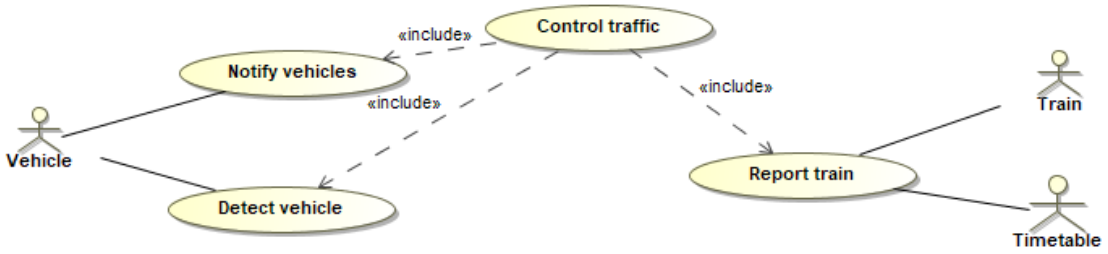

Figure 3. High-level use cases of the Intersection Traffic Supervision System of Systems

Use cases are defined in the following tables.

Detect vehicle
ID: 1
Brief description: Detect a vehicle that is approaching or leaving the intersection
Primary actors: Road vehicle
Secondary actors: -
Preconditions: A system is running in the SoS that can detect vehicles (e.g. VehicleCamera or V2ICommunicator).
Main flow: 1. Vehicle is approaching/leaving the intersection. 2. Detection system detects the vehicle. 3. Detection system obtains the license plate of the vehicle (e.g. with image recognition, V2I message). 4. Detection system reports the vehicle approaching/leaving with its license plate to the ITSCentral system.
Postconditions: The vehicle is detected and it is reported to the ITSCentral system.
Alternative flows: -

Report train
ID: 2
Brief description: Report a train that is approaching/leaving or should approach/leave the intersection.
Primary actors:

-
Secondary actors: Railway
Preconditions: A system is running in the SoS that can provide information about railways (e.g. RailwayProximitySensor or TimetableDataSource), and it registers itself with ITSCentral.
Main flow: 1. Railway is approaching/leaving the intersection or should approach/leave according to the timetable. 2. System detects the railway. 3. System reports the train approaching/leaving to the ITSCentral system.
Postconditions: The railway is reported to the ITSCentral system.
Alternative flows: -

Notify vehicles
ID: 3
Brief description: ITS-SoS notifies the vehicles inside the intersection about a change in the traffic situation and advises actions (e.g. stop due to train, proceed carefully).
Primary actors: Vehicle
Secondary actors: -
Preconditions: ITSCentral system is running and is connected to V2ICommunicator.
Main flow: 1. ITSCentral receives a new message (e.g. a train or a vehicle is detected). 2. ITSCentral analyses the current situation and calculates the notification level to send out. 3. ITSCentral asks V2ICommunicator to send out notifications.
Postconditions: Notification is sent to road vehicles.
Alternative flows: -

4. Behaviour

The high-level behaviour of ITS-SoS is depicted on Figure 4.

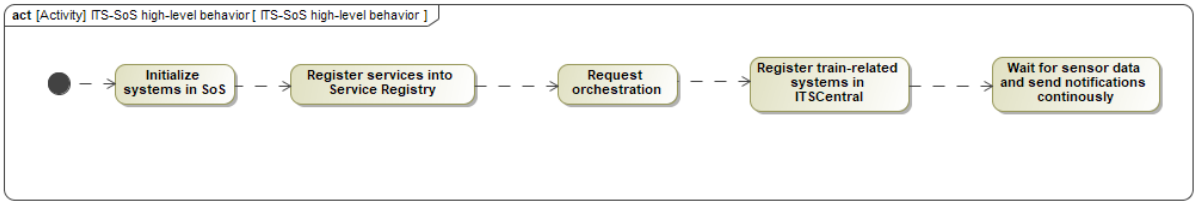

Figure 4. High-level behaviour of ITS-SoS

Functional requirements.

System of a systems initialization:

R1. All systems shall register their provided services into the *Service Registry* of the local Arrowhead Framework.

R2. All instances of the *VehicleCamera* and *V2ICommunicator* systems shall request the *ReportVehicle* service provided by the *ITSCentral* system from the *Orchestrator* core service.

R3. All instances of the *RailwayProximitySensor* and *TimetableDataSource* systems shall request the *ReportTrain* service provided by the *ITSCentral* system from the *Orchestrator* core service.

R4. All instances of the *RailwayProximitySensor* and *TimeTableDataSource* systems shall register themselves to ITSCentral using a custom message containing their type.

R5. The *ITSCentral* system shall request the *NotifyVehicles* service from a *V2ICommunicator* system from the Orchestrator core service. ITSCentral shall use this service to send alerts and notifications to the vehicles inside the intersection.

Handling vehicle data

R6. Systems detecting road vehicles (e.g. *VehicleCamera* and *V2ICommunicator*) shall report to *ITSCentral* when a vehicle approaches or leaves the intersection and send the vehicle's license plate.

R7. *ITSCentral* shall store data for the vehicles currently inside the intersection.

Notifications

R8. *ITS-SoS* shall send the following notifications to road vehicles using its services. The notification levels are ordered (Stop having the highest severity).

a) *Stop*: a train is approaching the intersection or already inside the intersection, thus stop and do not cross.

b) *TrainMightArrive:* according to the timetable, a train will pass the intersection, but the system has not detected it.

c) *LookAround:* the system has no means to detect the train physically, therefore road vehicles can cross the intersection only after verifying visually that there is no train.

d) *PassSlowly:* there is no train, vehicles can cross the intersection slowly.

R9. Handling information from *RailwayProximitySensor:*

a) If *ITSCentral* receives from *RailwayProximitySensor* that a train is arriving, then *ITSCentral* shall send a *Stop* notification to the vehicles.

b) If *ITSCentral* receives from *RailwayProximitySensor* that a train is leaving, then *ITSCentral* shall send a *PassSlowly* notification to the vehicles.

R10. Handling information from *TimeTableDataSource:*

a) If *ITSCentral* receives from *TimeTableDataSource* that a train is arriving, then *ITSCentral* shall send a *TrainMightArrive* notification to the vehicles.

b) If *ITSCentral* receives from *TimeTableDataSource* that a train is leaving, then *ITSCentral* shall send a *LookAround* notification to the vehicles.

R11. ITSCentral shall send notifications in the following ways.

a) When ITSCentral initializes, it shall send a *LookAround* notification.

b) New notifications shall be broadcasted always.

c) If there are vehicles in the intersection whose data is stored in ITSCentral, then ITSCentral shall send also the notification to each vehicle directly.

d) If a new vehicle is reported to ITSCentral, then ITSCentral shall send the current notification level to the vehicle.

e) If there is a new train sensor data, but it does not change the notification level, then ITSCentral shall not send a new notification.

R12. If there is both a *RailwayProximitySensor* and a *TimeTableDataSource* in the deployed ITS-SoS, and ITSCentral receives conflicting information from them, then the more severe notification shall have priority.

a) For example, if *RailwayProximitySensor* has reported that a train is arriving and then *TimeTableDataSource* reports a train leaving, then ITSCentral shall not send *LookAround.*

Scenarios. Figure 5 presents the simple scenario, where there are no cars yet in the intersection, and a RailwayProximitySensor reports a train approaching.

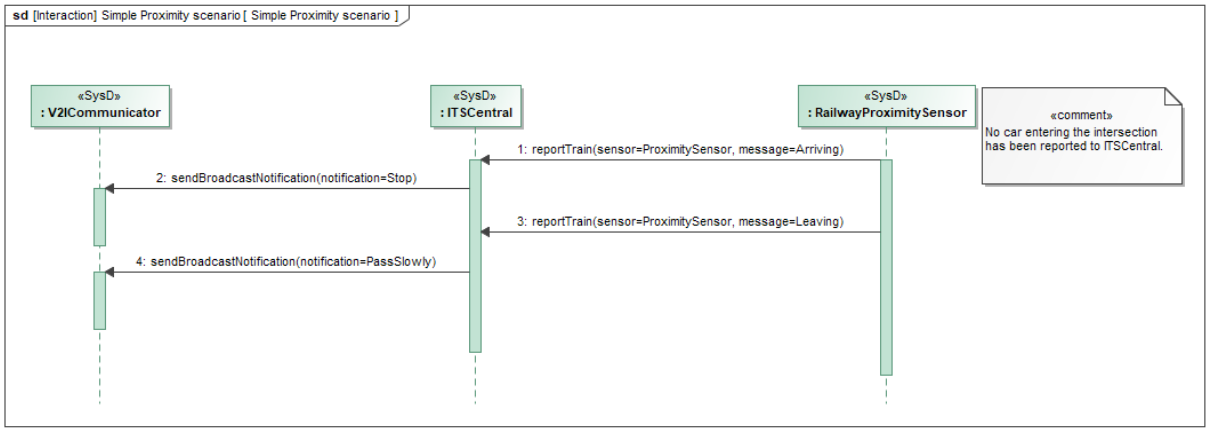

Figure 5. Basic scenario of detecting a train approaching using a RailwayProximitySensor

5. Non-functional requirements

(TBD in future iterations)

Table 2. Non-functional requirements description

Name	Description	Type	Value	Use-case

6. Revision history

6.1 Contributing and reviewing partners

Contributions	Reviews	Participants	Representing partner
Initial version of document		Zoltán Micskei, Márton Elekes	BME
First review		András Vörös	BME

6.2 Quality assurance

No	Date	Version	Approved by