REDUCTION
2011-2014

Deliverable 4.3.2

Report on second pilot

(to be used for field study 2)

22-01-2015

Public Document

D4.3.2 [Report on second pilot (to be used for field study 2)**]**

Project acronym:	REDUCTION
Project full title:	Reducing Environmental Footprint based on Multi-Modal Fleet management Systems for Eco-Routing and Driver Behaviour Adaptation
Work Package:	4
Document title:	Advanced Prototype. Report on second Pilot.
Document subtitle:	Software architecture in phase 2 of field trials
Version:	1.2
Official delivery date:	August 31, 2014
Actual publication date:	______________________
Type of document:	Report
Nature:	Public
Authors:	A. Lois, M. Morssink, W. Pover, A. Haddu, D. Zweijtzer, N.T. Larsen, C.J. Nielsen, J. Grabocka, K. Mouskos, K. Dimitrios, M.Valé
Approved by:	______________________

Version	**Date**	**Sections Affected**
1.0	31/07/2014	Initial version
1.1	29/08/2014	Internal review comments processed
1.2	22/01/2015	External review comments processed

Table of Contents

List of tables

List of figures

Executive Summary

The REDUCTION project follows a multidisciplinary approach to the challenge of reducing the CO2 emissions via intelligent transportation systems. The project is composed of several main pillars: eco-routing, multi-modal eco-routing, eco-driving and distributed data mining. Consortium partners (UHI, UTH, AU, AAU, and DDE) offer methodological contributions in the Work Packages 1-3, while industrial partners (CTL, TrainOSE, FlexDanmark) will test the developed methodologies in three concrete field trials, as described in Work Package 5.

The consortium is committed to accomplish four field studies. The FlexDanmark field trial (Task 5.2) will enable verifying the efficiency of the eco-routing algorithms developed in Work Package 3, while the test-bed will be the taxi routing platform of FlexDanmark. Similarly the TrainOSE field trial (Task 5.3) is dedicated to elaborating multi-modal eco-routing techniques and applies them to a train-bus hybrid traffic topology. Finally the Nicosia field trial (Task 5.4) will enable the validation of the eco-driving and distributed data mining approaches of Work Package 2 and the communication device and protocols designed by Work Package 1. The study will analyse the data and vehicular communication of a public bus fleet in Nicosia. In addition to the three field trials covered by the tasks of Work Package 5, a new field trial will be organised by Trinité with a focus on eco-routing for individual users through a portable "iPhone" eco-routing application.

Work Package 4 is devoted to the development of several modules elaborated in the life-span of the project and their integration. **In this deliverable (D4.3.2) the primary focus will be on describing the advances in the prototype as described in the previous deliverable (D4.3.1). The software architecture and communication protocols to be used will be elaborated more extensively, which can be used for the field trials of Work Package 5.**

The architectural description will follow a modular fashion. All field studies (FlexDanmark, TrainOSE and Nicosia) utilise specialised software stacks that implement the methodologies tailored for the focus of analysis. Naturally, the software stacks of each use case comply with the technical requirements imposed by the underlying infrastructure of the field trial companies and interact with the support systems of the respective provided services. The state-of-art specific software architecture to be used in all the field trials (FlexDanmark, TrainOSE, Nicosia and Trinité) is covered in depth throughout this deliverable, as well as the inter-modular communication and interaction between modules that is carried through web-service technologies.

Based on the experience of previous work in the Reduction project Trinité designed a layered area based architecture having multiple levels of control. Each level has its own span of control. If the limits are reached escalation to a higher level is done. On peer level requests to solve traffic challenges is offered in a generic approach.

The advantage of this structure is that information to the individual users can be addressed when they are in an area.

For the field trials Trinité established contact with the city of Amsterdam in order to use their extensive CO2 sniffer network. Amsterdam has been measuring CO2 and particulate values over a long period of time on our selected route (Jan van Galenstraat). The access to both actual and historic data is of great use to measure the effect of the field trial. This will be used to finetune the market ready product.

The mathematical models for the Amsterdam use case will be included into the current system in existing (or new) objects The Area Traffic Manager (ATM) connects these objects and coordinates the communication in order to facilitate the user.

Meetings will be established with some major transport enterprises in order to find future users for the architecture when the REDUCTION project is finished.

Usage of the standards DatexII and DVMExchange offers the possibility to communicate on regional level, where Network Management Systems (NMS system) inter-operate to address common traffic challenges. These systems are commonly built, owned and managed by different parties.

1. Introduction

The reduction of CO_2 emissions is a great challenge for the transport sector nowadays. Despite recent progress in vehicle manufacturing and fuel technology, still a significant fraction of CO_2 emissions in EU cities is resulting from road transport. Therefore, additional innovative technologies are needed to address the challenge of reducing emissions. The REDUCTION project focuses on advanced ICT solutions for managing multi-modal fleets and reducing their environmental footprint. REDUCTION collects historic and real-time data about driving behaviour, routing information, and emissions measurements that are processed by advanced predictive analytics to enable fleets enhance their current services as follows:

1) Optimising driving behaviour: supporting effective decision making for the enhancement of drivers education and the formation of effective policies about optimal traffic operations (speeding, braking, etc.), based on the analytical results of the data that associate driving-behaviour patterns with CO_2 emissions;

2) Eco-routing: suggesting environmental-friendly routes and allowing multi-modal fleets to reduce their overall mileage automatically; and

3) Support for multi-modality: offering a transparent way to support multiple transport modes and enabling co-modality.

REDUCTION follows an interdisciplinary approach and brings together expertise from several communities. Its innovative, decentralised architecture allows scalability to large fleets by combining both V2V and V2I approaches. Its planned commercial exploitation, based on its proposed cutting edge technology, aims at providing a major breakthrough in the fast growing market of services for "green" fleets in EU and worldwide, and present substantial impact to the challenging environmental goals of EU.

1.1 Objectives of work package 4

The main objective of WP4 is to have a real-time publish-subscribe distributed middleware with generic functionalities. The functionalities from WP1, WP2 and WP3 will be integrated based on different interfaces.

Requirements on software level for the envisaged final software product are collected and the software architecture is defined. The software architecture will be based on the principles of i) publish-subscribe, and ii) distributed middleware. Such architecture provides higher levels of abstraction, hiding the complexity of dealing with a variety of platforms, networks and low-level process communications. Application developers may concentrate only on the current requirements of the software to be developed, and use lower-level services provided by the middleware when necessary.

The software development of the case studies in WP5 will use the system design and architecture in this work package as a framework to integrate different functionalities.

1.1.1 Overview of WP1, WP2, WP3 and their tasks

WP1, WP2, WP3 are mainly contributed by the academic partners. The academic partners and their tasks are shown in Table 1.

University Thessaly (WP1)	Wireless communication including V2I and V2V	Duration from M1 to M30
University Hildesheim (WP2)	A predictive model to educate drivers to improve driving	Duration from M1 to M36
University Aalborg and University Aarhus (WP3)	Eco-routing algorithm, based on fuel consumption and GHG emission models	Duration from M1 to M36

Table 1: Tasks academic partners

1.2 Objectives of deliverable 4.3.2

Deliverable 4.3.2 aims to describe the pilot software and the relevant functionalities that have been offered in the second phase of the field studies. Such deliverable will cover a detailed description of the software architecture of the various modules contained by the modules of each work package. The internal dependencies and interactions of functionalities inside the module of each field trial will be elaborated. In addition interfaces to connect between functionalities across field trials will be covered through inter-connecting web-services. A summary of the objectives of this deliverable is given in Table 2.

Objective	*Result*
To describe the extended and more detailed functionalities for WP1, WP2 and WP3, to be used for phase 2 of the field trials.	The functionalities are described in chapter 4, 5, 6 and 7.
To describe the software and system architecture that is needed for phase 2 of the field trials.	The architecture is described in chapter 3, 4, 5, 6 and 7.
To show the relationships and interoperability between the different field trials.	The relationships and interoperability is described in chapter 3

Table 2: Summary table of deliverable 4.3.1

2. Targeted functionalities

Primarily the methodologies developed in the REDUCTION projects will be validated through the four designed field trials namely: FlexDanmark, TrainOSE, Nicosia and Trinité. The technologies to be deployed together with the corresponding field trial where they will be tested are depicted in Table 3. A tick (+) denotes that the respective technology of the first column will be deployed on the field trial, whose column contains the tick.

Technology (Partner(s))	*Work Packages*	*Field Trial*			
		FlexDanmark	TrainOSE	Nicosia	Trinité
Vehicular Communication (DDE)	WP1			+	
P2P Communication and Analytics (UTH, UHI)	WP1, WP2			+	
Eco-Driving (UHI,TRI)	WP2			+	+
Eco-Routing Taxi Fleet (AU, AAU, FD)	WP3	+			
Eco-Routing Individual (TRI)	WP4				+
Multimodal Eco-Routing (TrainOSE)	WP5		+		

Table 3: Associations of technologies and partners versus field trials

The reproducibility of the technologies across sites is not highly spread, as Table 3 shows. This however is not going to be a problem, since the technology of every work package will define standardised output on CO2 emission, fuel consumption and travel time, to be used as input definition of the integral software architecture of WP4. In other words, the technologies developed in the different field trials are loosely coupled to the integral software architecture of WP4.

The architecture of the software to be used in the first phase of the mentioned field trials is described in each of the following dedicated chapters:

- Chapter 3 – **FlexDanmark** field trial, second pilot architecture description
- Chapter 4 – **TrainOSE** field trial, second pilot architecture description
- Chapter 5 – **Nicosia** field trial, first and second pilot architecture description
- Chapter 6 – **Trinité** field trial, second pilot architecture description

3. Top-level design

The architecture is based on a real-time publish-subscribe distributed middleware with generic functionalities and capability to communicate with different sources using communication standards. The architecture is described in the next chapter middleware.

As an addition to the middleware cloud computing describes how to create a computation infrastructure on top of the middleware. Required for work package 2 predictive analytics models.

3.1 The middleware

The DSS-Datapool is optimised for data handling from many different sources. The European standard for traffic systems to exchange data is DatexII. In the field-test REDUCTION partners will be connected to the DSS-Datapool. Within the DSS-Datapool an Area Traffic Manager part of the generic application layer can use all the information from the partners to calculate an optimised Eco-friendly route (see

Figure 1). The next chapter will give insight of the DSS architecture.

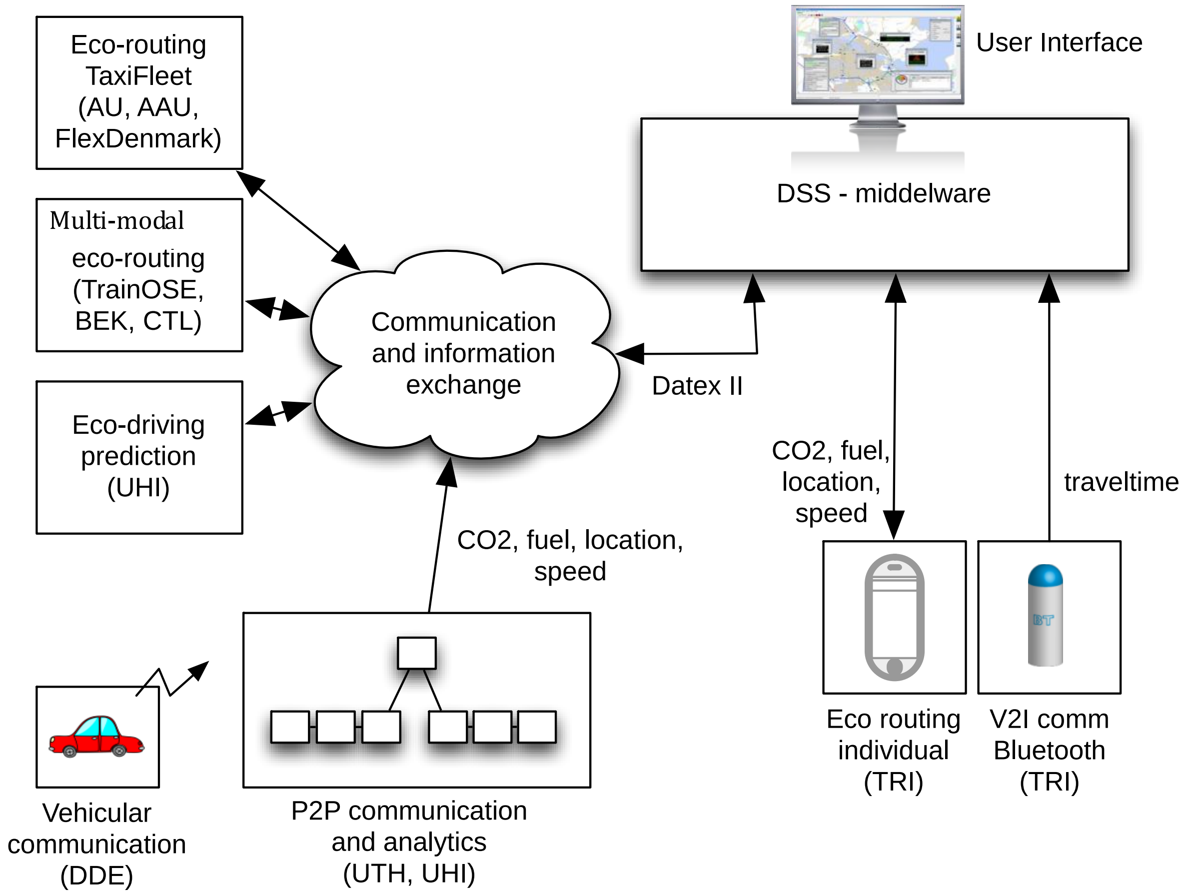


Figure 1: REDUCTION system architecture

In Table 4 , an overview is given of the different field trials and their functionalities.

Case study	***Details***	***Interoperability***
FlexDanmark	Eco-routing,	Via DatexII
TrainOSE	Multi modal eco-routing	Via DatexII
Nicosia	Eco-driving and distributed data mining	Via DatexII

Table 4 Interoperability

3.2 DSS

DSS is an abbreviation for *Dynamic Subscription Software*, and is designed as real-time publish subscribe *middle-ware*.

Middle-ware is the common name for software and system architectures that exchange data between different users in a system environment. There are different types of *middle-ware* available, like databases, CORBA, etc. All have the purpose to exchange data between different entities in a system environment. The information layer in the DSS environment is called the DSS Datapool.

The data exchange in DSS is set up by 'Subscriptions'. An object that needs specific information (*Consumer*) can subscribe to an object providing this information (*Producer*). The *consumer* only gets information from a *producer* if its status has changed. The *consumer* is able to call actions on a *producer* that might change the *producers'* status. Using the distributed middleware limitations are solved by breaking down the problem in several objects starting from measuring information to road segment to road to area.

3.2.1 Positioning of DSS

Before we look how DSS works, it would be good to understand the technical position of DSS in the world of system architectures.

3.2.2 Data-polling versus Event-driven

The DSS-Datapool is, as previously said, a *middle-ware* designed to exchange all sorts of data between different software modules. The different *middle-ware* applications use various strategies to exchange information. These can be qualified as follows:

- Object orientated versus data orientated
- *Event-driven* versus *data-polling*

Before making the choice for a particular *middle-ware,* it is important to judge whether the *middle-ware* characteristics are in line with the wanted system characteristics. The most important questions to be answered are:

- Are there many producers of information?
- Are there many consumers of information?
- What demands are their concerning status changes of producers that need to be communicated to consumers? Do all consumers need to know these changes immediately?

- Who takes the initiative to pass the changes through?

Two examples:

1. Imagine a network of cash machines connected to one central computer at a bank. There is one producer (the central server) that passes information to many consumers (the cash machines). The information wanted by the consumer is the state of account: Does Mister X still has enough money on his account? The producer receives the request and replies the consumer. This is a workable example of data polling.

2. A network with multiple loops on a road section passes the information to the operators in a Control room that a car has stopped. We have a situation where many producers pass data to a limited number of consumers. If this would be a data polling structure, we would have a huge amount of status requests, which would have an enormous effect on the network capacity. Therefore the initiative for communication should in this example be with the loop, i.e. the stopping of the car results in a message to the operators (*event-driven*).

DSS is object-orientated, *event-driven* and especially designed for a "*many to many*" system environment. Consumers can subscribe to specific producers that have relevant information. Changes within producers will be made available to all consumers.

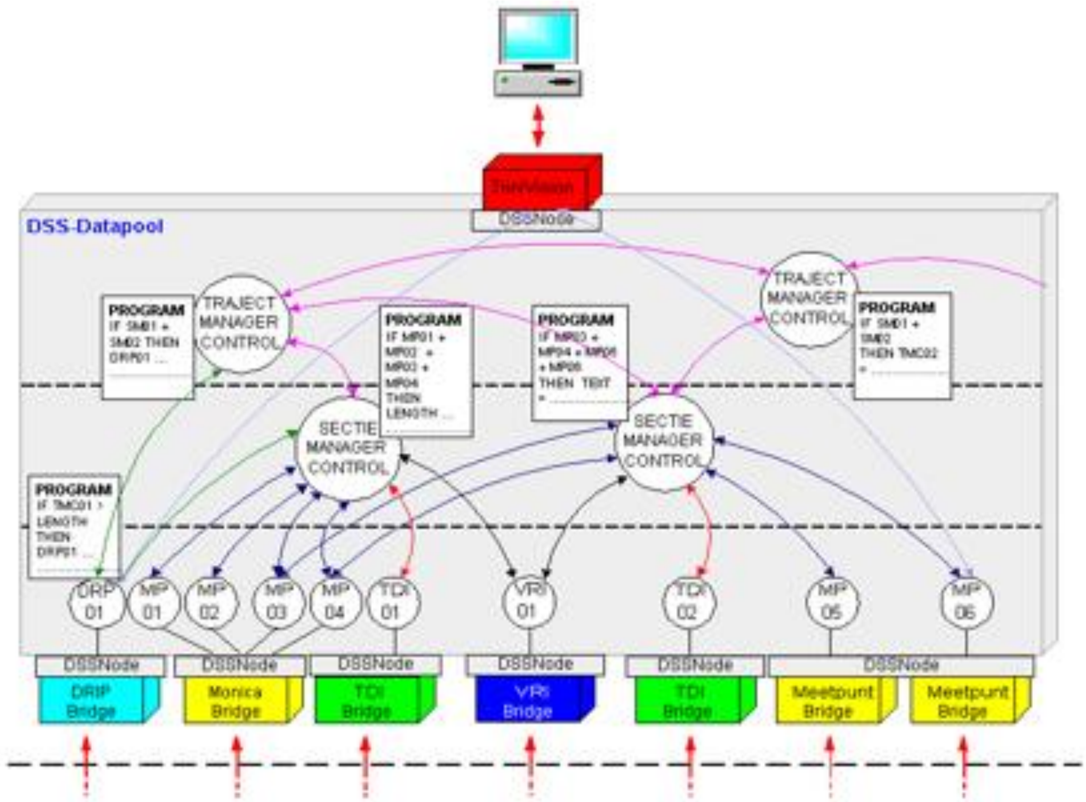


Figure 2 the DSS-Datapool

3.2.3 Specific versus Generic systems

Another important aspect of a system is the scope. Is it focussed at specific tasks and functions or should it be capable to combine multiple tasks and functions. An operating system for video matrices (and cameras) is an example of a specific system. A DCS System (Integrated Programming and Visualization Pack for PLC's) would be an example of a generic system. With DSC-systems whole factories are automated.

Specific systems have extensive functionalities available that all are performing one task. Generic systems have to perform multiple tasks and are therefore per task less complete.

DSS is a generic system, designed to support as many tasks and functionalities possible under one-system architecture.

3.2.4 The context of DSS

Although DSS can be used in all IT projects, it is currently mostly used in Traffic Control rooms. From these Control rooms traffic is being monitored and controlled, for example with the use of video cameras. DSS is used as the main source of information. It visualises loops and uses the data they produce for active traffic management. It is a full independent traffic management system that operates roadside signal devices automatically and has tools for manually operating cameras, hard-shoulder running and tunnels. It also serves as a guarding system that can generate automatic alarm messages.

3.2.5 Data Structure DSS

DSS is a *middle-ware* and therefore has no hard requirements for the internal structure of applications developed within DSS. Whether you create one large control with all functionality included or multiple controls all containing a part of the functionalities does not matter.

How to structure the applications is a choice to be made by the designers, this could be either Functional Decomposition or Object Orientated, but DSS has specifically been designed to build object orientated applications. The next choice to be made is the grouping of objects in the Datapool. The next paragraphs will discuss structures that are most common in DSS Applications.

3.2.6 The Information Layer

The first layer in most DSS environments contains the (software) images of the objects from the real world (roadside devices). We call this the information layer.

A DSS Bridge creates the relation between the virtual internal software objects and the external existing objects. The *Bridge* transfers *update*s (status change, error reports, etc.) from the external object, to the internal object (that in turn makes the information available to its subscribers). Commands that are generated from the Datapool to the internal software object will be transmitted via the DSS Bridge to the external object.

3.2.7 The Application Layer

All functionalities, except for internal controls, will preferably be placed in the Application Layer. The choice to do so is based on experience. Functionalities are subject to change in time, whilst the external objects in the Information layer hardly ever change.

If functionalities are separated from the information source, we benefit from more reliable system architecture. It also allows more applications to share the data from the Information Layer.

Different applications can (in turn) be divided:

1. Functional applications
2. Generic applications

Functional Application Layer
In a Functional Application Layer controls are grouped based on their functionality. The organization of data and controls are reflected vertically in the Information Layer.

Examples of a Functional Application Layer are the current Open VMS Architecture and Management Services Hard-shoulder Running.

The disadvantages of a Functional Application Layer are very similar to those of a Functional System. Integrating a new functionality on the same data layer often leads to adjustments on all functionalities on that layer. Altering objects from one functionality could have an influence on the next. Both functionalities share actuators and need to be able to handle these. Therefore they will need to be adjusted.

Generic Application Layer
A Generic Application Layer will be designed with the idea that more functionality will be running within the layer. The design phase will ask more of the designers and technicians initially, but will offer more possibilities eventually. In a Generic Application Layer applications are often embedded in a software environment that looks like the environment that needs to be automated. The environment that currently uses DSS, TrafficLink (network management layer), is reflected in the Application Layer as a network structure. In this network all sorts of functionalities are embedded. The generic character is stressed by no longer mentioning the specific functionalities, but by mentioning the more generic *services like:* to inform, reduce incoming flow, etc.

Although most of the existing DSS systems already have generic characteristics, most of them still have a functional structure. However we will always try to embed new functionalities in one network management layer.

The Generic Application Layer is used for the new advances in the REDUCTION project for example the Digital Road Authority in chapter 3 using the information from the information layer was designed for REDUCTION.

3.2.8 Objects in DSS

1) General Description
Objects are the bases to the functionality in the DSS-Datapool. All of the required functionality of a process that needs to be automated will eventually be part of one of the DSS objects as either input or output. Not only the software images of external objects, but also the controls that generate the interaction between the objects.

An object in de DSS-Datapool is built of:

1. A set of properties
2. A set of operations on the object

3. Particular properties such as the *Notifications*.

4. Program units and Bridges

Because several objects of a certain type may exist in the DSS, the Datapool object is identified with a unique identification code, the Object ID.

This chapter describes the structure of DSS objects, as well as the interactions between them.

2) Properties of an Object

Every object in the real world has certain properties. A car for example has wheels, an engine and fuel consumption. The objects in the Datapool mirror the objects in the real world and therefore have the same properties available.

The object-*properties* can be compared with the class-variables of object orientated programming languages like C++ and Java.

Property-types

Just like in C++-objects multiple types of *properties* are available, like:

- *Integer*
- *Double*
- *String*
- *Boolean*

But also variables of a higher level, like:

- *Array's*
- *Table*
- *Sheet*
- *Pixmap*
- *Notifications*

All properties listed above can be part of a DSS-object.

PUBLIC and PRIVATE properties

Not every property of an object needs to be published. Internal variables and definitions are not of any interest to anyone other than the object itself and the administrator. The publishing of this data would also create unnecessary system activity. Therefore only a part of the properties is published. In the object they are put in a separate table 'PUBLIC'. The rest of the properties are stored in the table 'PRIVATE'.

3.2.9 Exchange of information between objects

The DSS architecture has been designed to distribute information between different objects. As described in the previous paragraphs the information in the objects has been captured in the properties. The information to be exchanged in general concerns the condition of these properties. When the status of an object property changes, this is immediately communicated to any others that might be interested, i.e. any objects or bridges that have a subscription. The command for this is 'Link'. The subscription will make sure that the information is passed on, until the subscription has been cancelled.

Objects that offer information are called *producers*, where objects that subscribe to this information are referred to as *consumers*. DSS-objects will mostly be both consumer and producer

A. Standardised update-information: the Signal

Description of signals

The *Signal* is the way DSS exchanges information between objects. This type of update has got a specific name and contains a previously determined set of properties. The *update* of a specific set of relating *properties* is combined in one *Signal*. An object therefore has multiple *signals available*. The transmitting of a specific *Signal* within the DSS is called ***emitting*** and is triggered on *property* change.

B. A-standardised receiver: the Slot.

Description of slots

With the implementation of signals, only a part of the data transfer issue had been resolved. Filtering incoming data at the consumer object remained necessary. The *slot* was implemented as a specific receiver that can be linked to a specific *signal*. The command for this is 'CONNECT'.

A. Operations on an object: the Action

DSS is not only an information system, but also a control system. Commands from information receivers can be passed via DSS to real roadside objects.

To perform an operation on a DSS object, the corresponding *Action* of the object is activated. An *Action* can be compared with a *method* of a C++-class. The *Action* performs object specific tasks; it can manipulate both PRIVATE- and PUBLIC-properties and trigger signals with adjusted properties as a result.

B. The Reply.

The *Reply*-event will give the result on an action call. Objects receive a *Reply* when another object has sent an *Action.* In this *Reply* the result of the *Action* command is given as a return variable RETVAL (integer). RETVAL can have the following values:

- -2 = RESOURCE_TEMPORARY_UNAVAILABLE: The requested *Action* is temporarily not available. (For example because it is just performing an action for another Object).
- -1 = PENDING: The *Action* has been done but the operation at the actual device is still in progress. The sender now knows that the Object has received his request.
- 0 = FAILED: The *Action* has been done but the operation has failed, for whatever reason.
- 1 of higher = SUCCES: *Action* was successful.

C. Asynchronous Action commands

In a system like the DSS-Datapool, that distributes data; more aspects are crucial to the correct functioning of the application. One of these aspects is whether the processes are synchronous or asynchronous.

Before we can answer this question we will explain the definition of synchronous and asynchronous. If a process is completely synchronous, every action will take place after the other. In an asynchronous process actions can also take place at the same time.

Within the DSS-Datapool most processes are asynchronous. Every object can use CPU time, so any synchronisation will have to be built in by the programmer.

D. Dealing with a Reply.

How does the system deal with the replies? Object B has to be able to function as normal, when it receives a reply, to avoid 'freezing' of the system. The answer is relatively simple. If an Object doesn't have any actions to perform, it is available to receive updates and new events. The *Reply* is nothing but a special version of the *Signal*.

For the receipt of this *signal* the system uses a standard slot (embedded in every object): HANDLE_RETVAL_REPLY(..). The slot makes sure that the internal object status is updated, so the object knows whether the operation has been successful.

The remaining question is how the generic slot (HANDLE_RETVAL_REPLY(..)) identifies the right reply? The source of the *Reply* can be tracked; just like the type of reply. But since Object B can send out multiple request for the same action (an impatient user), this could lead to confusion. Therefore the internal object administration will add a unique CALLNR and a time to live to every command coming from an external object. This CALLNR will act as a reference throughout the process.

E. Locking an object.

Because processes do not run synchronized, we often see that objects (or actions for objects) have to be locked for a period of time. This happens so often that a standard mechanism has been implemented. An object can simply lock an specific *Action* by using the function DISABLE(action name). Unlocking is done using ENABLE(action name).

As a result of the DISABLE(..) command, all other object that are trying to send a command to this object, will receive the reply RETVAL = -2 (RESOURCE_TEMPORARY_UNAVAILABLE).

F. The Synchronous Action command

Besides the non-synchronous commands, DSS can also handle synchronous commands. From the previous paragraphs we have learned that this might not be applicable to all actions, but there are still plenty of actions that create no problems. One needs to realize that Object B will wait for an answer and does not perform any other action in the meantime, nor is open to receive any other event like the *Reply*-Event.

Considering the above mentioned disadvantages, synchronous actions also offer serious advantages. The most important is the reduction of administration that guards the consistency of the object.

For example: An action from Object A that triggers two other actions in Objects B and C. If this action is non-synchronous, the first action B is commanded and then the Object A will wait for the reply. To investigate the result a HANDLE_REPLY_B(...) Slot needs to be created. In this HANDLE_REPLY(..) C is called, after the action in B has been finished successfully. The *Reply* of C is again dealt with in slot: HANDLE_REPLY_C(...).

G. The Internal Object administration

In previous paragraphs the Internal Administration of a DSS Object has been mentioned a few times. This administration is meant to deal with several standard tasks that derive from *Actions* and the follow-up of these. We will not sum up all possibilities of the internal administration, but it is good to know the most important:

- Automatic CALLNR generation and registration, which makes every command unique.
- Automatic command replies if an *Action* is temporarily unavailable in the object (RESOURCE_TEMPORARY_UNAVAILABLE).
- Automatic creation of a *Reply-Event* (RETVAL=FAILED) that will be posted in a slot, if there has not been a *Reply* from the object within a certain time.
- Automatic continuation of a program when a previously set time has passed ('time-out') and no *Reply* to a synchronous command has come from an object. Without this the program would 'freeze' forever.

3.3 Interface description

In this chapter the interface communication protocol is defined, this is the advance for REDUCTION to connect the different work packages of the REDUCTION partners to the DSS architecture.

The interface description is based on the DatexII definition. There are some information-fields needed for this interface that are not defined in the DatexII standard. For those fields some information records are defined on top of the standard.

For security and authentication the standard VPN connection between two communicating systems is proposed.

There are two messages defined that needs to be exchanged between two communicating systems:

- A data message, containing data values that needs to be exchanged and
- A message reply, indicating the data message has been received successful or not successful.

The essential information that is exchanged within the data message is:

- CO2 emission, fuel consumption, travel time and velocity
- Identifier of the measured route, to relate the same route in two different systems.
- Location, length and geographical form of the route, to allow display on a geographical map.

The data message structure makes it possible to exchange one or more routes within one message and for each route one or more measurements.

The data message contains the following structure and field definition:

XML Name (DatexII)	Remarks
Record: REDUCTIONInformation	
ModelVersion	(integer) Fixed value "1"
SequenceNr	(Int64) sequence number
Record: ReductionSupplierIdentification	
Country	(string) Two letter country identification. "nl, du, fr", etc
NationalIdentifier	(string) Fixed value: "REDUCTION"
Language	(string) Fixed value: "en"

MeasurementType	Enumeration with values: traject or point
SensorType	Enumeration with values: CO2 detector, calculated, ...
Record: PolutionMeasurement	
PublicationTime	(datetime) Current time
StartTime	(datetime) Start time of the measurement series
EndTime	(datetime) End time of the measurement series
Record: ReductionLocation (there can be multiple locations or trajects)	
LocationId	(string) Identifier of the location
TrajectDistance	(double) Unit: km. Distance of the measurement traject. Only filled in when it is a traject.
Record: ReductionLocationForDisplay (a location can have multiple points when it is a traject. This can be used to visually display the traject)	
Latitude	(double) Latitude of the coordinate in WSG84
Longitude	(double) Longitude of the coordinate in WSG84
Zvalue	(double) Height in meters above sea level (optional)
Record: GeneralComment (multiple comments possible per traject)	
Comment	(string) Free text field
DateTime	(DateTime) Datetime of the comment
CommentType	(enum) Type of the comment
Record: ReductionMeasuredValue (multiple measurement records possible, for multiple start times). Each traject can have one measurement record for each minute or other time scale. The measurements are over the whole track.	
CO2Emission	(double)
FuelConsumption	(double)
Velocity	(double) Speed in km/hour, of the measured vehicle
TravelTime	(double) Traveltime over the traject in seconds.
CO2Exhaust	(double) Specific exhaust value in ? Of the measured vehicle
VehicleType	(enum) Typ of vehicle. Enum with the values: Car, Bus, Truck
MeasurementStartTime	(datetime) Start of the specific measurement

MeasureMentEndTime	(datetime) End of the specific measurement. (optional) when not filled in the start and end time are the same.

Table 5 DatexII field definition

The message reply is the answer to the previous data message, indicating that the data is received. The result can be an acknowledge or an error. In case of an error, the reason field can be used to give more information about the error condition that occurred.

The message reply contains the following structure and field definition:

XML Name (DatexII)	Remarks
Record: ReductionInformationReply	
Ackowledge	(Boolean) “0” (failed) or “1”
OrgSequenceNr	(Int64) sequence number of message for which this is a reply
Reason	(string) description of the error condition
DateTime	(DateTime) Time reply send

Table 6 DatexII reply message

The chapter 10 Appendix provides a description of the XSD Interface and the complete XML definition (XSD). The partners of REDUCTION can use this XSD for implementation of the interface.

3.4 Cloud computing (UTH)

Design, implementation and performance analysis of a vehicular cloud using Hadoop over wireless links

3.4.1 INTRODUCTION

The infrastructure provided by Trinité's publish/subscribe (pub/sub) system allows for the connection of many different components, but it is not a truly computation infrastructure. For instance, it is not possible to split a computational-intensive task into parts and execute these different pieces, say, by a few vehicles and a few backend servers. However, such kind of operation is necessary when we want to run applications such as the distributed data mining ones developed in REDUCTION's Work Package 2.

Therefore, we had to look for solutions to be used in the advanced prototype, so another advance for REDUCTION that would create a kind of a "virtual machine" over Trinité's pub/sub system that would make the underlying "hardware" (vehicles, servers) completely transparent to the application programmer. To achieve this goal we thought of seeing the whole problem as a problem of developing a cloud infrastructure over a pub/sub system.

In the following section, we will describe the design, implementation and evaluation of our cloud manager which is a layer above Hadoop framework. Hadoop can be installed over a middleware that supports Java Virtual Machine (JVM). Hadoop was designed to analyse data using nodes connected by Ethernet cables locally. So this layer will make Hadoop to work over wireless links and will give solutions to several fundamental problems in vehicle networks. All the technologies and offered services will need to operate within the local/regional/national legal and institutional frameworks.

3.4.2 Vehicular Cloud

Today, the term "cloud computing" is used to describe the concept of running remote service over the network. The backend are usually located in the data centre and services can be providing to massive amount of users concurrently in a distributed manner. The idea behind cloud computing is that computing, storage, and bandwidth are packaged as a service and can be utilised over a network, which is the Internet. The vehicular cloud differs from today's Internet cloud in the nature of its hardware and sensor platform. The vehicle cloud builds its platform opportunistically in a centralized or distributed control, from spare and idle resources made available by participating vehicles. Another major difference is the mobility nature of vehicular cloud. Vehicle cloud enables application to run on a particular location at a particular time, which is not possible for traditional cloud service. In fact, the majority of services provided by the vehicle cloud are expected to be location aware applications.

Vehicular Clouds (VCs) utilise the concept of Cloud Computing and VANETs (Vehicular Ad-hoc Networks) together. There are two types of VCs. Infrastructure-based VC and Autonomous VC (AVC). In the Infrastructure based VC, the driver accesses roadside infrastructure for services. In AVC, the vehicles are organized in VC to support emergencies and ad hoc events. The VC services can be categorized into three levels, application, platform and infrastructure.

Infrastructure as a Service (IaaS): Several types of virtualization occur in this layer. Among the other resources, computing, network, hardware and storage are also included. In the bottom layer of the framework, infrastructure devices and hardware are virtualized and provided as a service to users to install the operating system (OS) and operate software applications. Therefore, this layer is named Infrastructure as a Service (IaaS). Amazon Web Services (AWS) is a very good example of this category where Amazon provides its customers computing resources through its Elastic Compute Cloud (EC2) service and storage service through both Simple Storage Service (S3) and Elastic Book Store (EBS).

Platform as a Service (PaaS): In PaaS, Mobile operating systems such as Android, iPhone, Symbian and other OS, as well as database management and IMS are included in this section. This layer contains the environment for distributing storage, parallel programming design, the management system for organizing distributed file systems and other system management tools for cloud computing. Program developers are the primary clients of this platform layer. Google AppEngine and Microsoft Azure are good examples of that category.

Software as a Service (SaaS): Analytical, interactive, transaction and browsing facilities are included in the Application layer. SaaS delivers several simple software programs and applications as well as customer interfaces for the end users. Thus, in the application layer, this type of services is called Software as a Service (SaaS). By using the client software or browser, the user can connect services from providers via the internet and pay fees according to their consumed services, such as in a pay as you go model. IBM is a good example of this category.

Olariu et al. [1.9] argued that it is only a matter of time before the huge vehicular fleets on our roadways, streets and parking lots will be recognized as an abundant and under-utilised computational resource that can be tapped into for the purpose of providing third-party or community services. With this in mind, we have coined the term AVC to refer to: a group of largely autonomous vehicles whose corporate computing, sensing, communication and physical resources can be coordinated and dynamically allocated to authorized users.

Our cars spend significant time on the road and may face dynamically fluctuating locations. In this case, the vehicles will help local consultants resolve traffic incidents in a timely fashion which is not possible with the municipal traffic management centres alone due to the lack of adequate computational resources. We expect that, the vehicles are capable of solving problems in many situations that may require an indefinite time for a centralized system.

The cloud infrastructure consists of two parties: cloud storage and cloud computation. The data gathered by the inside-vehicle layer will be stored in the geographic information system (GIS), a road traffic control device or a storage system based on the type of applications. The computation part is used to calculate the computational tasks which provide faster performance, for example, the health recognition sensors send data to driver behaviour database in cloud storage.

Despite the existence of the well-developed information technologies, there is a significant challenge that hinders the rapid development of vehicular networks. Vehicles are normally constrained by resources, including computation, storage, and radio spectrum bandwidth. Due to the requirements of small- size and low-cost hardware systems, a single vehicle has limited computation and storage resources, which may result in low data processing capability. On the other hand, many emerging applications demand complex computation and large storage, including vehicle multimedia entertainment, vehicular social networking, and location based services. It becomes increasingly difficult for individual vehicles to efficiently support these applications. A very promising solution is to share the computation and storage resources among all vehicles or physically nearby vehicles. Another important characteristic of AVCs is the ability to offer a seamless integration and decentralized management of cyber-physical resources; an AVC can dynamically adapt its managed vehicular resources allocated to applications according to the applications' changing requirements and environmental and systems conditions. Routing in vehicular networks has been considered as a challenging task, with consideration of various factors, including vehicles mobility patterns, the ability of vehicles to carry messages to different locations, and the availability of Internet connections at roadside infrastructure. Moreover, variant communication channels have different levels of reliability in a location, which leads to various end-to-end delays and communication costs. A good routing strategy should take all these for an optimized and dynamic decision.

One of the main characteristics of VC is the mobility of the nodes which directly affects on the available computational capabilities and storage resources, for example, the number of parked vehicles in the parking is not constant. Therefore, to provide fluctuating application requirements and resource accessibility on the move, the necessary related protocol architecture and VC networking must be developed.

The existing layered network architecture, for example TCP-IP stack, is not adequate to support ongoing evolving technologies and applications. It needs to use the service-oriented and component based network architecture [1.10] with sufficient learning opportunities and monitoring facilities in order to cope with reusable and extensible applications and resources, which require to be largely deployed as common services available in VC environments.

The fundamental building blocks and structures that compose VC should be engineered and designed to face the structural stress of the unstable working situation. Olariu et al. [1.11] was the first to propose a robust dynamic architecture for VC based on Eucalyptus cloud system [1.12] and virtualization approach to aggregate the computational and storage resources. Greater emphasis and more researches are necessary for the migration of virtual machines among cars and efficient vehicle virtualization.

3.4.3 Application scenarios

The main goal of this section is to illustrate the power of the AVC concept. VCs offer cost-efficient way of services. Smart combinations of storage, price, and communication abilities of VC can be chosen. We touch upon several important scenarios illustrating various aspects of AVCs that are extremely important and that, under present-day technology are unlikely to see a satisfactory resolution.

2) Scenario I: augmenting the capabilities of small businesses.

Consider a small business employing about 250 people and specializing in offering IT support and services. It is not hard to imagine that, even if we allow for car-pooling, there will be up to 150 vehicles parked in the company's parking lot. Day in and day out, the computational resources in those vehicles are sitting idle. We envision harvesting the corporate computational and storage resources in the vehicles sitting in the parking lot for the purpose of creating a computer cluster and a huge distributed data storage facility that, with proposer security safeguards in place, will turn out to be an important asset that the company cannot afford to waste.

3) Scenario II: dynamic management of parking facilities.

Anyone who has attempted to find a convenient parking spot in the downtown area of a big city or close to a university campus where the need for parking by far outstrips the supply would certainly be interested to enlist the help of an automated parking management facility. The problem of managing parking availability is a ubiquitous and a pervasive one, and several solutions were reported recently. However, most of the existing solutions rely on a centralized solution where reports from individual parking garages and parking meters are aggregated at a central (city-wide) location and then disseminated to the public. The difficulty is with the real-time management of parking availability since the information that reaches the public is often stale and outdated.

This, in turn, may worsen the situations especially when a large number of drivers are trying to park, say, to attend a down-town event. We envision that by real-time pooling the information about the availability of parking at various locations inside the city, an AVC consisting of the vehicles that happen to be in a certain neighbourhood will be able to maintain real-time information about the availability of parking and direct the drivers to the most promising location where parking is (still) available.

4) Scenario III: autonomous mitigation of recurring congestion.

In face of traffic congestion some drivers often pursue detours and alternate routes that often involve local roads. Making the decision behind the steering wheel is often challenging. The driver does not know whether the congestion is about to ease or is worsening. In addition, when many vehicles decide to execute the same travel plan, local roads become flooded with traffic that exceeds its capacity and sometimes deadlocks take place. Contemporary ITS and traffic advisory schemes are both slow to report traffic problems and usually do not provide any mitigation plan. An AVC-based solution will be the most appropriate and effective choice. Basically, vehicles in the vicinity will be able to query the plan of each other and estimate the impact on local roads. In addition, an accurate assessment of the cause of the congestion and traffic flow can be made by contacting vehicles close to where the bottleneck is. In addition, appropriate safety precautions can be applied to cope with the incident, e.g. poor air quality due to the smoke of a burned vehicle.

5) Scenario IV: sharing on-road safety messages.

The trend in the car manufacturing industry is to equip new vehicles with major sensing capabilities in order to achieve efficient and safe operation. For example, Honda is already installing cameras on their Civic models in Japan. The cameras track the lines on the road and help the driver stay in lane. A vehicle would thus be a mobile sensor node and an AVC can be envisioned as a huge wireless sensor network with very dynamic membership. It would be beneficial for a vehicle to query the sensors of other vehicles in the vicinity in order to increase the fidelity of its own sensed data, get an assessment of the road conditions and the existence of potential hazard ahead. For example when the tire pressure sensor on a vehicle reports the loss of air, vehicles that are coming behind on the same lane should suspect the existence of nails on the road and may consider changing the lane. The same happens when a vehicle changes lane frequently and significantly exceeds the speed limit; vehicles that come behind, and which cannot see this vehicle, can suspect the presence of aggressive drivers on the road and consider staying away from the lanes and/or keeping a distance from the potentially dangerous driver. The same applies when detecting holes, unmarked speed breakers, black ice, etc. Contemporary VANET design cannot pull together the required solution and foster the level of coordination needed for providing these safety measures.

6) Scenario V: Self-organized high occupancy vehicle (HOV) lanes.

For precise and predefined travel time, HOV lanes carry a vast number of cars, which carry many passengers especially during periods of high traffic congestion. However, the authorities know about the congestion and have the official power to setup HOV lanes, but they do not have sufficient resources to compute and assess the situation to establish the time frame to use the HOV lane to ease the effects of traffic jam. VCs could setup HOV lanes dynamically by stimulating the flow of traffic and reducing the travel times for HOV lanes. VCs can dynamically provide the solution by gathering data from on-board vehicle sensors, and this type of solution is not possible with the current technology.

7) Scenario VI: VC in developing countries perspective.

Due to lack of sophisticated centralized decision support systems and infrastructure, the concept of VC will be very important in developing countries as well. Moreover, VCs will play a vital role in making a vast number of computing resources accessible through a vehicular network by using many computing applications dynamically, which are not possible to use with the current infrastructure.

3.4.4 Analysing Data With Hadoop

Hadoop [1.13] is a well-adopted, standards-based, open-source software framework built on the foundation of Google's MapReduce framework and a distributed file system called Hadoop Distributed File System (HDFS) based on Google's File System papers [1.14]. MapReduce is a parallel programming technique derived from the functional programming concepts and is proposed for large-scale data processing in a distributed computing environment.

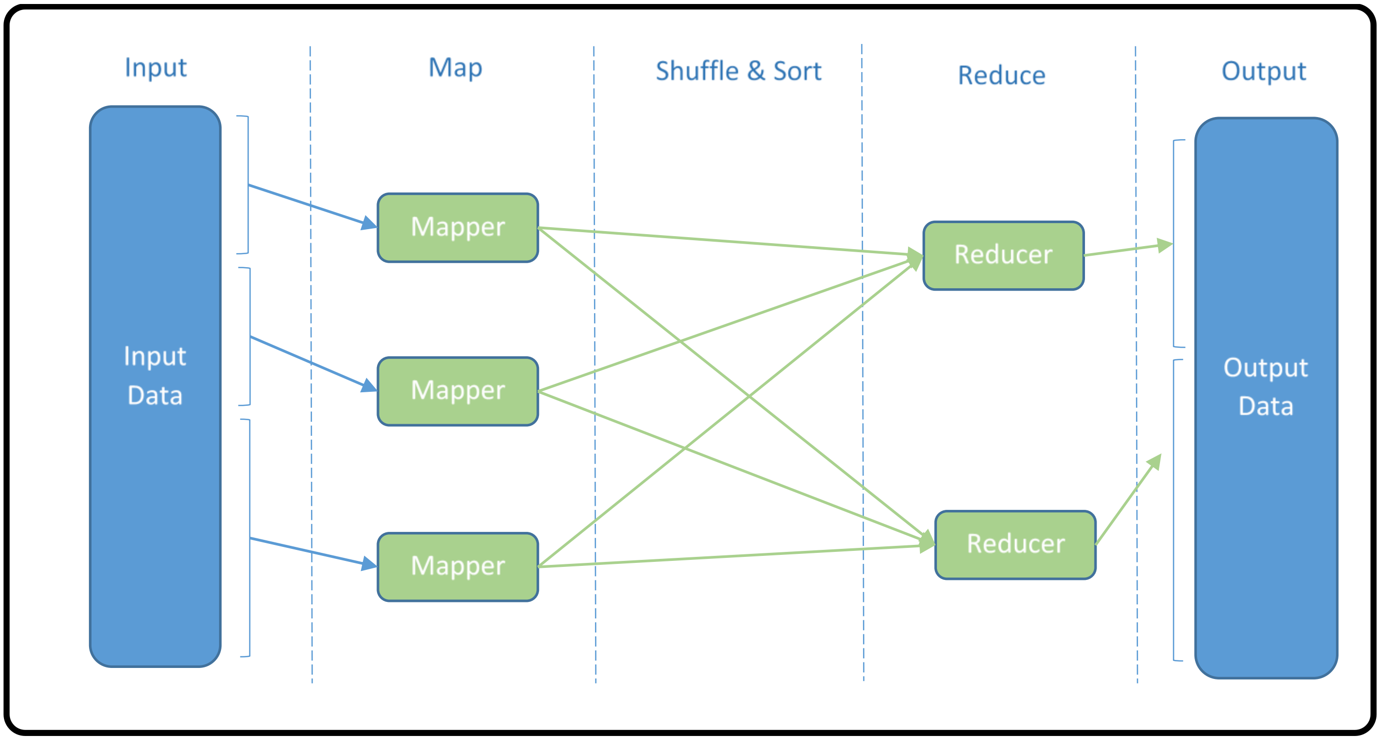


Figure 3 MapReduce Architecture

The focus is on scanning hundreds or thousands of files in parallel to run a set of repetitive actions in a batch-oriented processing fashion. Hadoop provides tools for the analysis and transformation of very large data sets using the MapReduce paradigm. It is very popular among academic institutions and real industries. It is suitable for fields such as web search, email spam detection, genome manipulation, social networks, and economic computations and for analysis of unstructured data such as log and text. An important characteristic of Hadoop is the partitioning of data and computation across many of hosts, and executing application computations in parallel close to their data. A Hadoop cluster scales computation capacity, storage capacity and IO bandwidth by simply adding commodity servers.

A. MapReduce

Hadoop leverages a cluster of nodes to run MapReduce programs massively in parallel. The computation takes a set of input key/value pairs, and produces a set of output key/value pairs. The user of the MapReduce library expresses the computation as two functions: Map and Reduce shows what is happening when a Hadoop job is running. The first step when running a Hadoop job is to process input data. Data are broken into data blocks and stored across the local files of different nodes. Data's replication is very important for reliability. Each input split contains several lines of data, each line is read as a record and then will be wrapped as key objects and value objects.

Map, written by the user, takes an input pair and produces a set of intermediate key/value pairs. The map function will consume a key and a value object and emit a key and value object as well. During this process, all the records will be executed by the same map function. Each node in order to run a job uses local system resources like CPU, memory and storage. The local files that constitute the file system are called HDFS. The number of nodes in each cluster varies from hundreds to thousands of machines. Hadoop can also allow for a certain set of fail-over scenarios. After all the map tasks finish, the reduce tasks will pull the corresponding partitions from the output of the map tasks. The MapReduce library groups together all intermediate values associated with the same intermediate key I and passes them to the Reduce function.

The Reduce function, also written by the user, accepts an intermediate key and a set of values for that key. The framework groups all the pairs, which have the same key and invokes the reduce function passing the list of values for a given key. It merges together these values to form a possibly smaller set of values. Typically, just zero or one output value is produced per Reduce invocation. All these data will be sorted and merged in the reduce tasks to make sure that all the values with the same key will be put together.

Finally reduce tasks will run and produce the output data. This last step is important to assemble intermediate results into a final result.

B. HDFS

HDFS is the storage component of Hadoop. While the interface to HDFS is patterned after the UNIX file system, faithfulness to standards was sacrificed in favour of improved performance for the applications at hand. It is a file system optimized for high throughput and works best when reading and writing large files. To support this throughput HDFS leverages unusually large for a file system block sizes and data locality optimizations to reduce network input/output (I/O).

Like in a file system for a single disk, files in HDFS are broken into block-sized chunks, which are stored as independent units. Unlike a file system for a single disk, a file in HDFS that is smaller than a single block does not occupy a full block's worth of underlying storage. HDFS stores file system metadata and application data separately. As in other distributed file systems, like GFS [1.13] [1.14], Lustre [1.15] and PVFS [1.16] [1.17], HDFS stores metadata on a dedicated server, called the NameNode. Application data are stored on other servers called DataNodes. All servers are fully connected and communicate with each other using TCP-based protocols.

In we can see HDFS architecture. HDFS maps all the local disks to a single file system hierarchy allowing the data to be dispersed at all the data/computing nodes. Scalability and availability are also key traits of HDFS. In order to provide data reliability HDFS uses block replication. Initially, each block is replicated by the client to three data-nodes. The block copies are called replicas. A replication factor of three is the default system parameter, which can either be configured or specified per file at creation time. For HDFS, the most important advantage of the replication technique is that it provides high availability of data in high demand. This is actively exploited by the MapReduce framework, as it increases replications of configuration and job library files to avoid contention during the job start-up, when multiple tasks access the same files simultaneously.

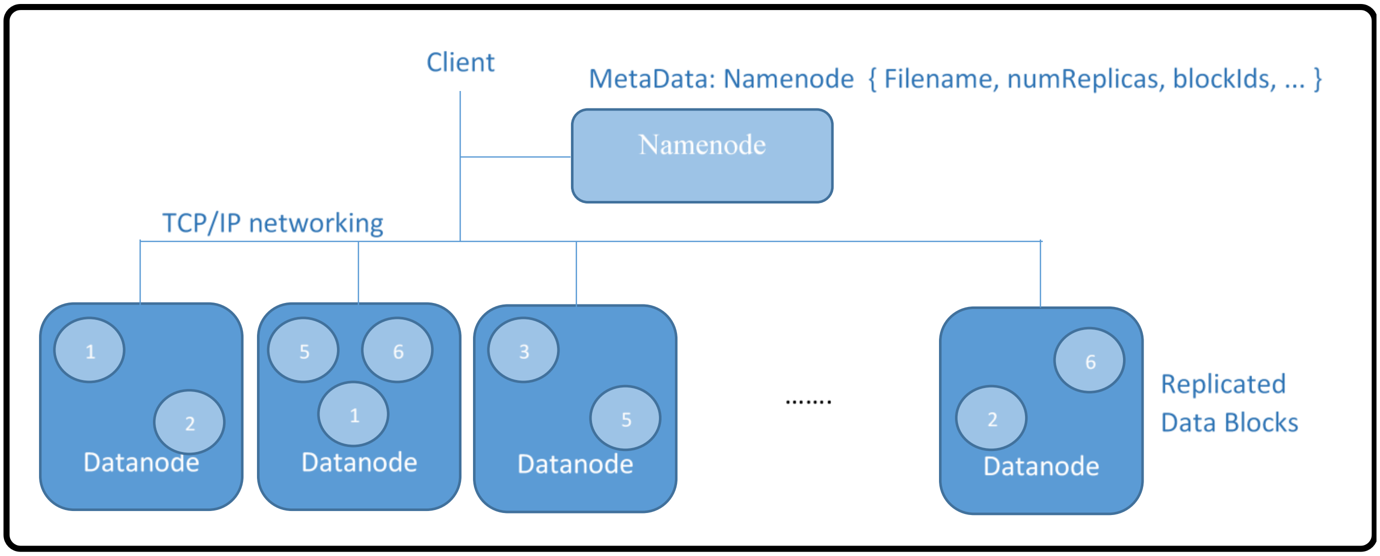


Figure 4 HDFS Architecture

Once the block is created, its replication is maintained by the system automatically. The name-node detects failed data-nodes, or missing or corrupted individual replicas, and restores their replication by directing the copying of the remaining replicas to other nodes. Hadoop schedules the MapReduce computation tasks depending on the data locality and hence improving the overall I/O bandwidth. This setup is well suited for an environment where Hadoop is installed in a large cluster of commodity machines. Hadoop stores the intermediate results of the computations in local disks, where the computation tasks are executed, and then informs the appropriate workers to retrieve them for further processing. Although this strategy of writing intermediate result to the file system makes Hadoop a robust technology, it introduces an additional step and a considerable communication overhead, which could be a limiting factor for some MapReduce computations. Different strategies such as writing the data to files after a certain number of iterations or using redundant reduce tasks may eliminate this overhead and provide a better performance for the applications.

C. NameNode

The HDFS namespace is a hierarchy of files and directories. The NameNode is the master of HDFS that directs the slave DataNode daemons to perform the low-level I/O tasks. The NameNode is the bookkeeper of HDFS; it keeps track of how your files are broken down into file blocks, which nodes store those blocks, and the overall health of the distributed file system. The function of the NameNode is memory and I/O intensive.

Files and directories are represented on the NameNode by inodes, which record attributes like permissions, modification and access times, namespace and disk space quotas. The file content is split into large blocks and each block of the file is independently replicated at multiple DataNodes. The

NameNode typically doesn't store any user data or perform any computations for a MapReduce program to lower the workload on the machine. The NameNode maintains the namespace tree and the mapping of file blocks to DataNodes.

When a HDFS client wanting to read a file first contacts the NameNode for the locations of data blocks comprising the file and then reads the specific block contents from the DataNode closest to the client. Then, the client writing data, requests the NameNode to nominate a suite of three DataNodes to host the block replicas. Finally, the client then writes data to this three DataNodes.

The current design has a single NameNode for each cluster. The cluster can have thousands of DataNodes and tens of thousands of HDFS clients per cluster, as each DataNode may execute multiple application tasks concurrently. HDFS keeps the entire namespace in RAM. The inode data and the list of blocks belonging to each file comprise the meta-data of the name system called the image. The NameNode also stores the modification log of the image called the journal in the local host's native file system. During restarts the NameNode restores the namespace by reading the namespace and replaying the journal. The locations of block replicas may change over time and are not part of the persistent checkpoint, which is the persistence record of the image.

D. DataNodes

Each slave machine in your cluster will host a DataNode daemon to perform the grunt work of the distributed file system, reading and writing HDFS blocks to actual files on the local file system. When you want to read or write a HDFS file, the file is broken into blocks and the NameNode will tell your client which DataNode each block resides in. Each block replica on a DataNode is represented by two files in the local host's native file system. The first file contains the data itself and the second file is block's metadata including checksums for the block data and the block's generation stamp.

During start-up each DataNode connects to the NameNode and performs a handshake. The purpose of the handshake is to verify the namespace ID and the software version of the DataNode. If either does not match that of the NameNode the DataNode automatically shuts down. The namespace ID is assigned to the file system instance when it is formatted. The namespace ID is persistently stored on all nodes of the cluster. Nodes with a different namespace ID will not be able to join the cluster, thus preserving the integrity of the file system. The consistency of software versions is important because incompatible version may cause data corruption or loss, and on large clusters of thousands of machines it is easy to overlook nodes that did not shut down properly prior to the software upgrade or were not available during the upgrade. A DataNode that is newly initialized and without any namespace ID is permitted to join the cluster and receive the cluster's namespace ID. After the handshake the DataNode registers with the NameNode. DataNodes persistently store their unique storage IDs. The storage ID is an internal identifier of the DataNode, which makes it recognizable even if it is restarted with a different IP address or port. The storage ID is assigned to the DataNode when it registers with the NameNode for the first time and never changes after that. A DataNode identifies block replicas in its possession to the NameNode by sending a block report. A block report contains the block id, the generation stamp and the length for each block replica the server hosts. The first block report is sent immediately after the DataNode registration. Subsequent block reports are sent every hour and provide the NameNode with an up-to-date view of where block replicas are located on the cluster.

During normal operation DataNodes send heartbeats to the NameNode to confirm that the DataNode is operating and the block replicas it hosts are available. The default heartbeat interval is three seconds. If the NameNode does not receive a heartbeat from a DataNode in ten minutes the NameNode considers the DataNode to be out of service and the block replicas hosted by that DataNode to be unavailable. The NameNode then schedules creation of new replicas of those blocks on other DataNodes. Heartbeats from a DataNode also carry information about total storage capacity, fraction of storage in use, and the number of data transfers currently in progress. These statistics are used for the NameNode's space allocation and load balancing decisions. The NameNode does not directly call DataNodes. It uses replies to heartbeats to send instructions to the DataNodes. The instructions include commands to replicate blocks to other nodes, remove local block replicas, and reregister or to shut down the node and send an immediate block report.

These commands are important for maintaining the overall system integrity and therefore it is critical to keep heartbeats frequent even on big clusters. The NameNode can process thousands of heartbeats per second without affecting other NameNode operations.

3.4.5 Vehicular Network Design

In this section, we introduce the design of our system. shows the cloud architecture for vehicular networks [19]. It is a hierarchical architecture that consists of three interacting layers: vehicular cloud, roadside cloud, and central cloud. Vehicles are mobile nodes that exploit cloud resources and services.

- Vehicular Cloud: a local cloud established among a group of cooperative vehicles. An inter-vehicle network, a VANET, is formed by V2V communications. The vehicles in a group are viewed as mobile cloud sites and they cooperatively create a vehicular cloud.
- Roadside Cloud: a local cloud established among a set of adjacent roadside units. In a roadside cloud, there are dedicated local cloud servers attached to Roadside Units (RSUs). A vehicle will access a roadside cloud by V2R communications.
- Central Cloud: a cloud established among a group of dedicated servers in the Internet. A vehicle will access a central cloud by V2R or cellular communications.

This architecture has several essential advantages. First, the architecture fully utilises the physical resources in an entire network. From vehicles to roadside infrastructures and data centre, the computation and storage resources are all merged into the cloud. All clouds are accessible to all vehicles. Second, the hierarchical nature of the architecture allows vehicles using different communication technologies to access to different layers of clouds accordingly. Hence, the architecture is flexible and compatible with heterogeneous wireless communication technologies, e.g., DSRC, LTE/LTE-Advanced and CR technologies. Third, the vehicular clouds and the roadside clouds are small-scale localized clouds. Such distributed clouds can be rapidly deployed and provide services quickly.

A. Vehicular Cloud

In a vehicular cloud, a group of vehicles share their computation resources, storage resources, and spectrum resources. Each vehicle can access the cloud and utilise services for its own purpose. Through the cooperation in the group, the physical resources of vehicles are dynamically scheduled on demand. The overall resource utilization is significantly enhanced. Compared to an individual vehicle, a vehicular cloud has much more resources. Due to vehicle mobility, vehicular cloud implementation is very different from a cloud in a traditional computer net- work. We propose two customization strategies for vehicular clouds: Generalized Vehicular Cloud Customization (GVCC) and Specified Vehicular Cloud Customization (SVCC). In GVCC, a cloud controller is introduced in a vehicular cloud. A cloud controller is responsible for the creation, maintenance, and deletion of a vehicular cloud. All vehicles will virtualize their physical resources and register the virtual resources in the cloud controller. All virtual resources of the vehicular cloud are scheduled by the cloud controller. If a vehicle needs some resources of the vehicular cloud, it should apply to the cloud controller. In contrast to GVCC, SVCC has no cloud controller. A vehicle will specify some vehicles as candidate cloud sites, and directly apply for resources from these vehicles. If the application is approved, the corresponding vehicles become cloud sites, which will customize virtual machines (VMs) according to the vehicle demand. These two strategies, GVCC and SVCC, are quite different. With respect to resource management, GVCC is similar to a conventional cloud deployment strategy in which cloud resources are scheduled by a controller. A vehicle is not aware of the cloud sites where the VMs are built up. The cloud controller should maintain the cloud resources. During a cloud service, if a cloud site is not available due to vehicle mobility, the controller should schedule a new site to replace it. In SVCC, since there is no cloud controller, a vehicle has to select other vehicles as cloud sites and maintain the cloud resources itself. In terms of resource utilization, GVCC is able to globally schedule and allocate all resources of a vehicular cloud. GVCC has higher resource utilization than SVCC. However, the operation of the cloud controller will need extra computation. Therefore, SVCC

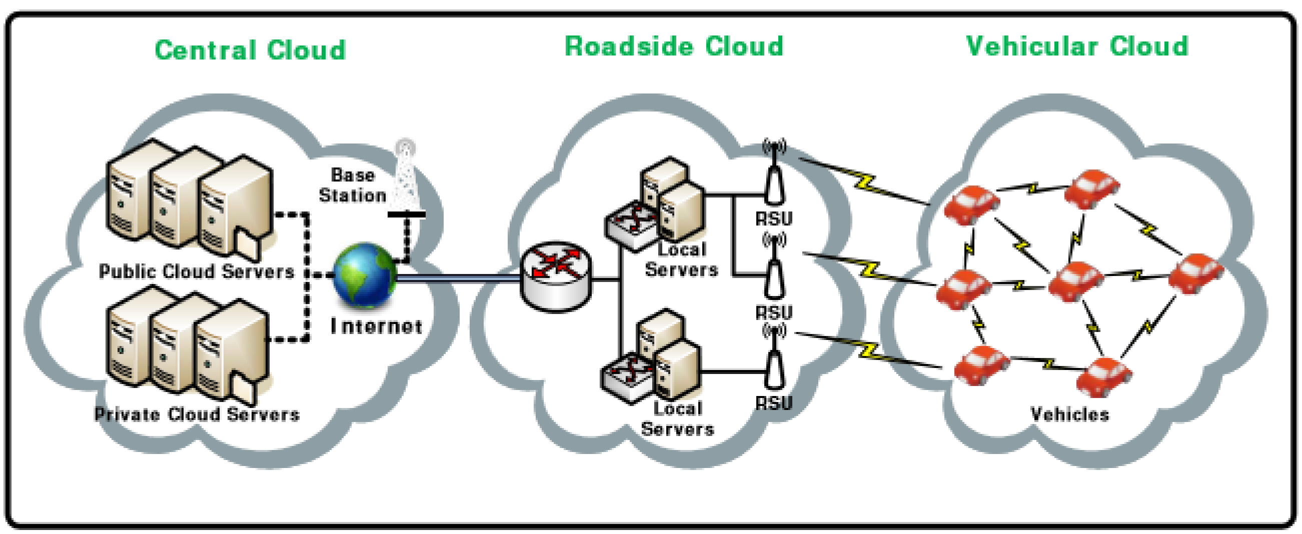


Figure 5 Proposed cloud-based vehicular networks architecture

may be more efficient than GVCC in terms of lower system overhead.

B. Roadside Cloud

A roadside cloud is composed of two main parts: dedicated local servers and roadside units. The dedicated local servers virtualize physical resources and act as a potential cloud site. RSUs provide radio interfaces for vehicles to access the cloud. A roadside cloud is accessible only by the nearby vehicles, i.e., those located within the radio coverage area of the cloud site's RSU. This fact helps us recall the concept of a cloudlet. A cloudlet is a trusted, resource-rich computer or cluster of computers that is connected to the Internet and is available for use by nearby mobile devices [18]. In this report, we propose the concept of a roadside cloudlet. A roadside cloudlet refers to a small-scale roadside cloud site that offers cloud services to bypassing vehicles. A vehicle can select a nearby roadside cloudlet and customize a transient cloud for use. Here, we call the customized cloud a transient cloud because the cloud can only serve the vehicle for a while. After the vehicle moves out of the radio range of the current serving RSU, the cloud will be deleted and the vehicle will customize a new cloud from the next roadside cloudlet in its moving direction. When a vehicle customizes a transient cloud from a road- side cloudlet, it is offered by virtual resources in terms of virtual machine (VM). This VM consists of two interacting components: the VM-base in the roadside cloudlet and the VM-overlay in the vehicle. A VM-base is a resource template recording the basic structure of a VM, while a VM-overlay mainly contains the specific resource requirements of the customized VM. Before a cloud service starts, the vehicle will send the VM-overlay to the roadside cloudlet. After combining the VM-overlay with the VM-base, the roadside cloudlet completes the customization of a dedicated VM. During a cloud service, as the vehicle moves along the roadside, it will switch between different RSUs. For the continuity of cloud service, the customized VM should be synchronously transferred between the respective roadside cloudlets. This process is referred to as VM migration.

C. Central Cloud

Compared to a vehicular cloud and a roadside cloud, a central cloud has much more resources. The central cloud can be driven by either dedicated servers in vehicular networks data centre or servers in the Internet. A central cloud is mainly used for complicated computation, massive data storage, and global decision. There already exists mature open source or commercial software platforms that could be employed for the deployment of a central cloud.

3.4.6 Our Hadoop Based Manager Architecture

Programming models used by cloud-based distributed batch processing infrastructures such as Hadoop allow parallel processing of data. For example, with Hadoop, the location and sensor data analysis algorithms can be implemented as MapReduce jobs. Scaling out the computation on a large number of machines in a cluster is simple with Hadoop. The same computation that runs on a single machine can be scaled to a cluster of machines with few configuration changes in the program. Cloud-based distributed batch processing infrastructure such as Hadoop allows scaling the data analysis jobs up or down very easily which makes analysis flexible. With this flexibility in data analysis jobs, the frequency of analysis jobs can be varied.

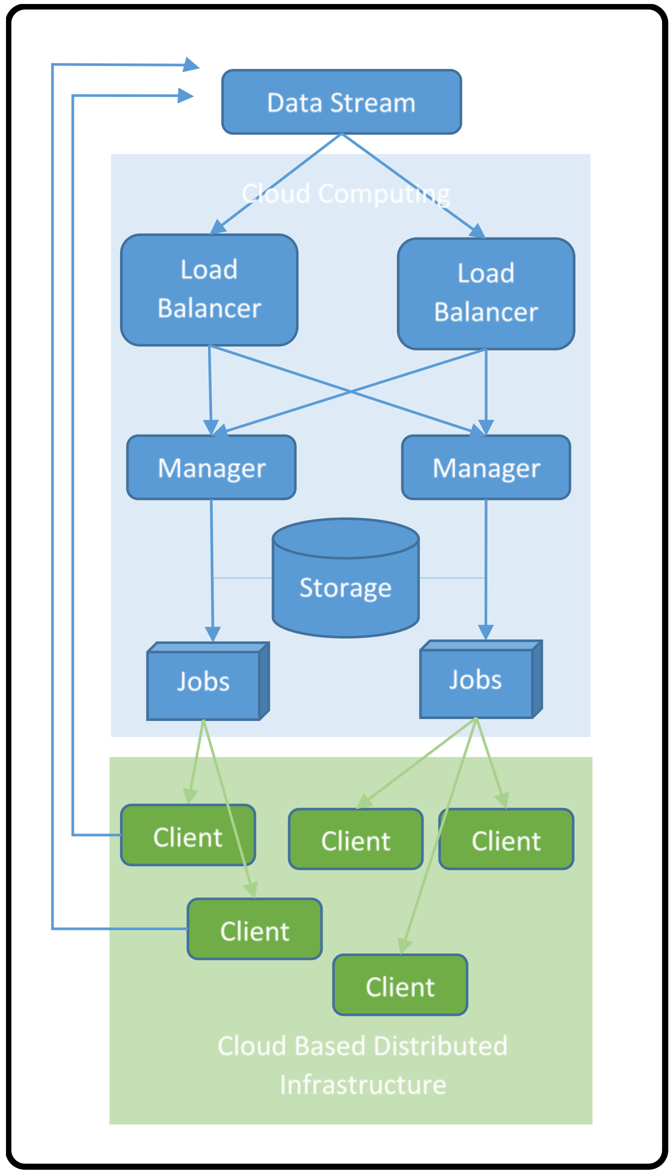


Figure 6 Cloud-based architecture used by our proposed framework

In order to design our Hadoop based cloud framework we used basic Hadoop's ideas and change them so that can be suitable for modular Datanodes with wireless connectivity. shows a cloud deployment architecture used in our proposed framework. In this deployment architecture, tier-1 consists of the web servers load balancers, tier-2 consists of application servers and tier-3 consists of a cloud based distributed batch processing infrastructure such as Hadoop. Compute intensive tasks such as data processing are formulated as MapReduce jobs which are executed on vehicles using Hadoop. This deployment is suitable for massive scale data analytics. Data is stored in a cloud based distributed storage such as Hadoop Distributed File System (HDFS).

Cloud-based deployment architecture leverages the dynamic scaling capabilities of computing clouds. Two types of scaling options are available for the cloud-based deployment, described as follows:

- Horizontal Scaling (scaling-out): Horizontal scaling or scaling-out involves launching and provisioning additional server resources for various tiers of the deployment.
- Vertical Scaling (scaling-up): Vertical scaling or scaling-up involves changing the computing capacity assigned to the server resources while keeping the number of server resources constant.

Our manager consists of two different type of nodes, master nodes and slave nodes. Master nodes works as managers. They keep track of how your files are broken down into file blocks, which nodes store those blocks, and the overall health of the distributed file system. The function of the NameNode is memory and I/O intensive. Each slave machine-vehicle in the cluster will host a slave node daemon to perform the grunt work of the distributed file system, reading and writing blocks to actual files on the local file system.

A. Master node

Master nodes work like servers. They wait for new connections from vehicles. The most important is that in our distributed system we can have more than one master node. Also during data analyses they store information about cpu and memory statistics from each slave node and data analyses results.

1) Data

The first thing that the master does is, like in a file system for a single disk, to break the files into block-sized chunks which are stored as independent files allowing the data to be dispersed at all the computing nodes. HDFS is built upon the single-node namespace server architecture. Since the name-node is a single container of the file system metadata, it naturally becomes a limiting factor for file system growth. In order to make metadata operations fast, the name-node loads the whole namespace into its memory, and therefore the size of the namespace is limited by the amount of RAM available to the name-node. In order to avoid this each time a vehicle downloads a file to analyse just appends to file's name info like what time the download happen and from whose vehicle.

In order to provide data reliability HDFS uses block replication. Initially, each block is replicated by the client to three data-nodes. The block copies are called replicas. A replication factor of three is the default system parameter, which can either be configured or specified per file at creation time. In our case vehicles do not have a lot of local storage, so we do not use this technique.

When slave nodes/vehicles want to get more data to analyse they just pull them from master node. This is very important because when we have a cloud with thousands of connected clients that request data or message exchange service processing load is very big.

2) Slaves management

Because we cannot know how many vehicles will be part of the cloud master, it keeps waiting for new connections. We can add at any time more wireless nodes for data processing.

In Hadoop during normal operation, data-nodes periodically send heartbeats to the name-node to indicate that the data-node is alive. Heartbeats also carry information about total and used disk capacity and the number of data transfers currently performed by the node, which plays an important role in the name-node's space and load-balancing decisions. The default heartbeat interval is three seconds. If the name-node does not receive a heartbeat from a data-node in 10 minutes, it pronounces the data-node dead and schedules its blocks for replication on other nodes. In our case because we use wireless moving nodes, the connections are terminated very often, so we do not use this heartbeat model. When a slave node is offline the data he analyses is sent to other nodes.

3) Slaves communication

Master node undertakes message exchange between slave nodes when a connection is available for security reasons that are analysed in the next section.

B. Slave node

Vehicular clouds have challenges that look alike to the difficulties faced in VANET and cloud computing. Slave node architecture based is a try to give solutions to challenges addressed by vehicular clouds.

1) Security and privacy

Security and privacy are very important aspects for the establishing and maintaining of the trust of users in VC. Privacy measures are required to ensure the VC communication and information in the isolated and trustworthy environment, while security procedures are needed to protect against network threats. Establishing trust relationships between several participants is a vital part of trustworthy communication and computation. It is impossible to provide a secure environment for vehicular cloud computing without considering some security requirements such as confidentiality, integrity and authentication.

- Confidentiality: Sensitive data should not be disclosed by unauthorized users,
- Integrity: Data should not be tampered with or modified. The messages must be reliable and valid,
- Availability: Data should be available whenever they are needed.
- Authentication: Determining whether someone or something is, who or what it is claimed to be,
- Privacy: The user's privacy should be preserved, and
- Real-time constraints: Some applications, such as accident alerts, require real-time or near real-time communication.

Large numbers of papers have addressed security issues related to Vehicular ad hoc network (VANET). Active and Passive location security has been proposed by Yan et al. [20]. The users are authenticated and validated using digital signatures. The messages are encrypted so as not to disclose the message contents. GeoEncrypt has been proposed in VANET by Yan et al. [21]. In the recent times, cloud security problems have been addressed. The solutions were to restrict the hardware accessibility to ensure minimal risks from insiders [22]. New platform was proposed by Santos et al. [23] to have a trustworthy conventional cloud. Santos proposed a new platform to achieve trust in conventional clouds. A trust coordinator maintained by an external third party is imported to validate the entrusted cloud manager, which makes a set of virtual machines (VMs) such as Amazon's E2C (i.e., Infrastructure as a Service, IaaS) available to users. Garfinkel et al. [24]

proposed a solution to prevent the owner of a physical host from accessing and interfering with the services on the host. Berger et.al [25] and Murray et.al [26], adopted similar solution to prevent the interference in the host services by the physical host owner. Jensen et al. [27] stated technical security issues of using cloud services on the Internet access.

The vehicular cloud has to authenticate the highly mobile vehicles. It has to verify the identity of users and check for the integrity of messages received from them. A few authentication metrics that can be adopted [28] are:

- Ownership: Unique identity like security token, identity card and user owned software token.
- Knowledge: Security questions, passwords, private identification numbers.
- Biometrics: Human biometrics like fingerprints, signature, eye scan, voice.

In VANET different attacks are possible such as, Denial of Service attack, Fabrication Attack, Alteration Attack, Replay Attack, Message Suppression Attack, Replay Attack, Sybil [29] Attack. Again, different attackers in VANET are such as, Selfish Driver, Malicious Attacker [30]. To overcome that some security requirements should be considered such as Authentication, Availability, Non-repudiation, Privacy, Real-time constraints, Integrity, Confidentiality [31].

The quality of information (QoI) retrieved from vehicles is also a concern. Data/information retrieved needs to be verified and validated before making decisions and/or publishing to the public. Authentication and authorization of the nodes accurately in the intermittent short-range communication relates to the integrity of data. The complexity increases with the increase of nodes. User identity, spoofing and tempering data could be the main threat to the network where the attacker pretends to be another user of same priority level. As a solution to this problem is suggested the use of SSH File Transfer Protocol (also Secure File Transfer Protocol, or SFTP). The SFTP is a network protocol that provides file access, file transfer, and file management functionalities over any reliable data stream. It was designed by the Internet Engineering Task Force (IETF) as an extension of the Secure Shell protocol (SSH) version 2.0 to provide secure file transfer capability. This protocol is assumed to run over a secure channel, such as SSH, that the server has already authenticated the client, and that the identity of the client user is available to the protocol. In this way, only authenticated users can connect and upload data to master node. Vehicular networks lacks the relatively long life context, so personal contact of user's device to a hot spot will require long life password and this will be impractical for securing VC. The generated public keys solve problems like that.

The attackers face many challenges as well. The high mobility has both pros and cons. The attackers will have to make repeated attempts to harm the vehicles. The access to each VM is transitory as the vehicles are bound to move across the states or districts. Moreover, the attackers have to locate the machine on which targets would be lying as it's a distributed environment. Experiments have been done to catch and compare the memory of processors, and users can find co-residence in the same physical machine [32]. The attackers must be physically co-located with the target user on the same physical machines. This will require attackers to be physically present at the same region with the target vehicles or shadow with the target vehicles at the same speed and have to collect valuable information with certain privileges or with security tokens.

2) Data collection

Vehicle-generated data or data obtained from neighbouring vehicles can be stored until the vehicle reaches a dedicated data collector or kept in the vehicle until retrieved as a reply to queries sent by data-seeking vehicles. This communication paradigm is considered a case of delay/disruption-tolerant networks. Each vehicle keeps the sensed data in local storage, and sends its stored data to the master node every time a connection is available. The vehicles are not able to access or alter the stored data on other vehicles. Therefore, the sensitive data that are stored in each vehicle should not be encrypted to protect it against the unauthorized access. The vehicles are authorized only to send to other vehicles and never to read. They can write only to master node.

3) Messages exchange

Wireless message exchange between vehicles is known as inter-vehicle communication and a network of communicating vehicles is known as a Vehicular Ad-Hoc Network (VANET). A VANET is a sub-category of wireless multi-hop networks in which a source depends on intermediate nodes to relay messages to a destination. In this communication paradigm, vehicles can be considered a resource for relaying data to other nodes out of the communication range of the source node. A vehicle can be used not only for relaying data to other neighbouring vehicles but also to/from Road Side Units (RSUs).

All nodes, such as vehicles and road-side infrastructures are able to communicate with each other based on the V2V or V2I communication models in vehicular cloud computing. Furthermore, the vehicles and road-side infrastructures are required to communicate with the cloud to store or process their data. An V2X OBU (On Board Unit) is responsible for establishing communication between vehicles or between vehicles and infrastructures. A RSU is an access point that is connected to a location server that records or processes all location data forwarded by RSUs. The location server sends the data to the cloud for processing or storage. Furthermore, a trusted certificate authority (CA) is in charge of providing authentication services for vehicles and location-based service providers. The VC's message is constructed from some fields such as: vehicle id or pseudonym to identify the driver, time, message type, length of message, data, and direction. The type of message can be:

- Short message: to send an alert message or warning messages
- Media message: to get an environment services from other vehicles or a cloud
- Priority message: to end the alert messages or urgent messages
- Acknowledge message: to confirm the delivery of messages.

Providing secure communication in a vehicular cloud plays a central role in creating safer and more efficient driving. There are several security holes in vehicular communication, which make it vulnerable against attackers, for example, preventing communication (jamming), forging messages and transmitting false hazard warnings by the attacker (forgery), dropping or modifying a message by intermediate nodes (traffic tampering), and privacy violations. In order to make safer the communication the message exchange between nodes is checked by master node.

4) Mobility

One of the main characteristics of VC is the mobility of the nodes which directly affects the available computational capabilities and storage resources, for example, the number of parked vehicles in the parking is not constant. Therefore, to provide fluctuating application requirements and resource accessibility on the move, the necessary related protocol architecture and VC networking must be developed. We design our nodes so that can request data to analyse depending in their capabilities such as channel bandwidth and available free storage. Vehicles' mobility is considered as a plus that allows vehicles to cover wider areas compared to their static counterparts. However, while pooling resources from a vehicle, the vehicle may leave the area of interest or connectivity before reporting/relaying the desired data, or finishing the task in hand. Dynamic availability of resources may also be temporal stemming, for instance, from the high need for resources during the rush hours and their idleness during late nights. Effective resource management techniques are needed for proper task assignment and retrieval.

Interruption of data transmission between faster moving nodes in the highways becomes imminent as the connectivity lasts only for a few moments. On the contrary, during traffic jam congestion of transmitter nodes will create noise and interference [33]. Obstructed by large buildings and other infrastructures in the metropolitan areas the signal attenuates at a higher rate resulting in both deterioration of signal strength and signal quality. Obstructed by large buildings and other infrastructures in the metropolitan areas the signal attenuates at a higher rate resulting in both deterioration of signal strength and signal quality [34].

To solve these problems slave node check's signal level of wireless connection. If signal level is too low or it is decreasing very fast the node does not try to read or write data at master node.

5) Vehicles capabilities

The vehicles usually have a large number of different on-board devices, including GPS, wireless transceivers, and on-board radar devices. The different vehicles are able to have a different combination of these on-board devices with different capabilities such as speed of processor, volume of memory, storage, and CPU capacity. Therefore, providing a security for these heterogeneous vehicles in VC environment is difficult because most of the cryptographic algorithms are not lightweight and the vehicles need to have certain hardware conditions. In our case due to the restrictions vehicles have, encryption is optional.

One challenge in VC design is the ability to connect to the Internet and to its resources, including Cloud Services. While Vehicle-To-Infrastructure (V2I) communications (via DSRC, WiFi and 3G/4G) are becoming increasingly more reliable in civilian settings, in environments such as battlefields and urban emergence, infrastructure access will be very limited, suggesting that Vehicle-To-Vehicle (V2V) communications will be used for critical vehicle cloud services like internal data storage, searching and sharing.

6) Network Scalability

The vehicular clouds must be capable of addressing security schemes for dynamically growing number of vehicles. The dynamics of traffic produces dynamic demands on security. They should not only handle the usual traffic but ought to take care of extra fluctuation that may occur in case of emergencies or natural disasters. Recently, Content-Based Networking (CBN) has attracted much attention as a method for searching content. In CBN, a data object (viewed as a chain of chunks) is searched and retrieved based on its identity instead of the IP address of the node on which it resides. While several independent CBN designs exist, all designs have the following common attributes:

- receiver-oriented chunk based transport
- in-network per-chunk caching
- name-based forwarding
- uniquely identifiable content naming

Intuitively, in-network caching is beneficial to highly mobile vehicular scenarios. A data retrieval failure due to intermittent connectivity can be recovered more quickly by leveraging distributed caches.

Based at CBN idea we create name-based routing. Name-based routing is different from host IP-based routing in two aspects. First, the number of content names to consider in name-based routing is significantly larger than the number of IP addresses. Second, in host-based networks, each IP address is associated with one host interface. In contrast, copies of a data object chunks may exist at different locations in CBN. Only master node stores information about connected vehicles names.

7) Data processing

Hadoop can be run in one of the three following modes: Standalone (or local) mode where there are no daemons running and everything runs in a single JVM. Standalone mode is suitable for running MapReduce programs during development, since it is easy to test and debug them. Pseudo-distributed mode where the Hadoop daemons run on the local machine, thus simulating a cluster on a small scale. Lastly fully distributed mode where the Hadoop daemons run on a cluster of machines. By default, Hadoop is configured to run in a non-distributed mode, as a single Java process.

In standalone mode that is suitable for our platform, the local file system and the local MapReduce job runner are used. Each vehicle runs Hadoop as standalone mode and waits for data to be analysed.

3.4.7 Evaluation

The Wireless Signal Propagation Emulator developed by CMU [35] accurately emulates wireless signal propagation in a physical space. The emulator senses signals generated by known wireless sources through the antenna port, subjects the signals to the same effects that occur in a real physical space (e.g. attenuation, multi-path fading, etc.), and feeds the combined signals back into wireless cards. The emulator, however, has limitations in reproducing arbitrary motion patterns. In addition, although the propagation scenario is more realistic, it is still artificially created as opposed to real life measurement.

Testbeds play a key role in network research, because a wireless medium has physical characteristics that cannot easily be simulated. Over the years, many kinds of testbeds have been developed in wireless network. Performing testbed experiments in a wireless network (WNET), however, is challenging. In order to test our developed platform we use a WNET testbed with unmovable nodes. This testbed is available for use either via remote or on site access. As for mobility support, the outdoor testbed is grounded.

Testbed approaches are the only way to conduct VANET experiments with high fidelity, but the dynamic nature of VANETs results in experimental inconsistency. WNET allows researchers to repeat experiments with different protocol configurations to ensure consistency. The whole system is under full control and thus one can adjust a few parameters while keeping the rest unchanged. However, WNET assumes accurate modelling of physical characteristics such as mobility/traffic patterns, radio propagation models, and external interferences. In the case of mobility patterns the assumption that vehicles do not move is made. In order to test this system, the nodes are being put in different distances between them every time the experiment takes place.

A. NITOS Testbed

NITOS currently consists of 50 wireless nodes, deployed out-doors at the exterior of the University of Thessaly campus building. Users can perform their experiments by reserving slices (nodes, frequency spectrum) of the testbed through NITOS scheduler that together with control and Management Framework (OMF), support ease of use for experimentation and code development. Two Gigabit Ethernet switches interconnect the nodes with NITOS server, namely the Control switch that provide for control of experiment execution and measurement collection and the Experimental switch, which can be used for conducting wired experiments. A third Gigabit Ethernet, namely the Chassis Manager switch, is dedicated in controlling the operational status of the nodes through the transmission of custom http requests that control solid state relays on the Chassis Manager cards. NITOS nodes feature up to three wireless network interface cards (NICs), using the Atheros AR5424 and AR9380 chipsets. Each node use an Intel(R) Core (TM) 2 DUO processor in 2261MHz, 64bit, Ram 2Gb and 26GB Hard Disk.

B. Configuration

For the implementation of our mechanism, we used the MadWiFi open source driver. Thanks to the use of the MadWifi modules by the Linux kernel, it is possible to implement a Wireless Access Point with a Personal Computer or an Embedded Device that has a WiFi network card with an Atheros chipset. This feature is available starting with version 1.0.beta8 of Zeroshell, which introduces WiFi support in either AP (Access Point) or STA mode.

The following operational modes are supported:

- Sta: Station, a.k.a. infrastructure or managed. This device is acting as typical WLAN client station. This is the default mode if not otherwise specified. This is the mode that is used for slave nodes.
- Ap: Access Point, a.k.a. master. This device acts as the Access Point for other WLAN client stations and it is used by master nodes so that slaves can connect to them.
- Adhoc: Ad-hoc. a.k.a. IBSS mode. This device is in a peer-to-peer(s) WLAN without the need for an Access Point and it is used by the slaves when masters are not available.

The option to use the MadWifi drivers, combined with the use of wpa_supplicant and hostapd packages, is due to their ability to perform the functions of an AccessPoint with advanced features, for example:

Access authentication and wireless traffic encryption through WPA/WPA2 (RSN). It is supported either in WPA-PSK mode, in which the client, in order to be associated to an SSID, must know the Pre-Shared Key, or WPA-EAP mode, also known as WPA Enterprise, in which a user can become authenticated with Username and Password or a X.509 digital certificate validated by a Radius server. Both the TKIP encryption algorithm and the more secure CCMP, based on AES, are supported;

With management of the Multiple SSID mode, it is possible to create up to 4 virtual standalone Access Points for each WiFi network card in the system. It is clear that virtual SSIDs belonging to the same WiFi network card share the radio channel being used, and thus the available bandwidth. Moreover, for each virtual SSID it is possible to establish a standalone authentication and encryption scheme.

One of the four possible SSIDs can also work in Managed mode and associate to a WLAN as a client. For example, this is useful to extend the range of the Wireless network itself by implementing repeaters that work in WDS (Wireless Distribution System) but do not need to be interconnected by means of a Wired network.

Virtualization introduces an additional overhead, which may affect experiment results: Virtualization adds an extra software processing layer between application and hardware. In addition,

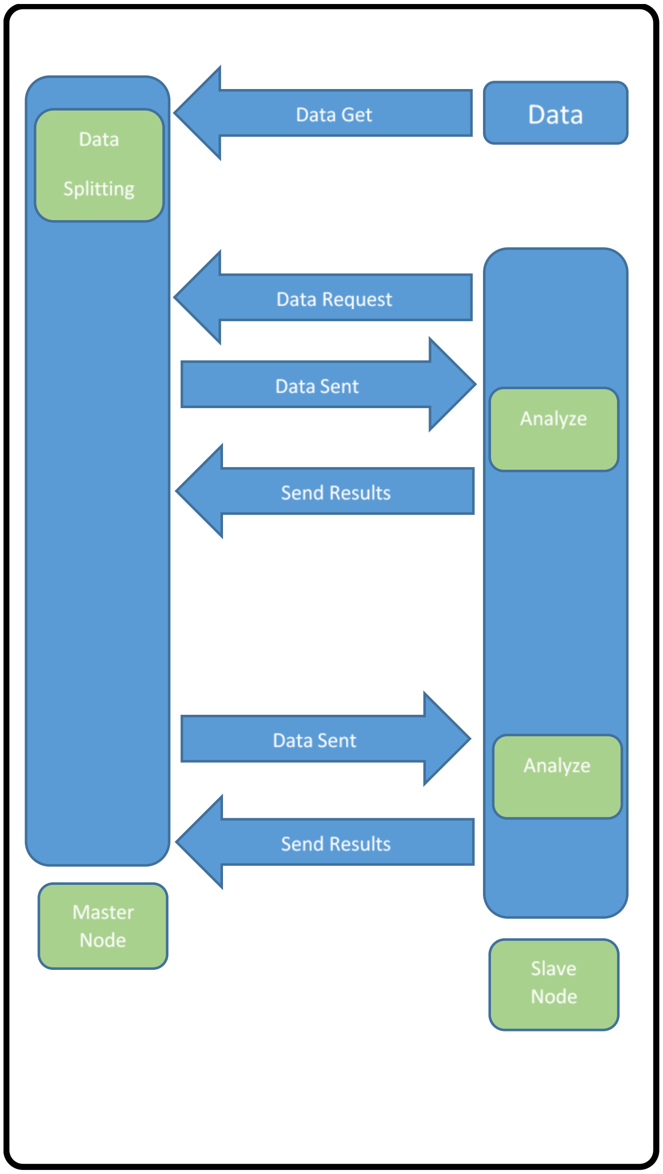


Figure 7 Stages during data analyses

hardware resources are used to run the virtual machines. Therefore, system performance is worse than a regular Linux system. For the experiments we created two Linux Distributions, one for master nodes and one for slave nodes.

In both masters and slaves Hadoop is installed in single mode with default configuration. When we start our manager in nodes additional information like file chunks size, data location, ports for message exchange and pseudo-name are required. shows stages during data analyses with a single slave node. With green colour we can see what is happening inside each node during time.

Benchmarking is an important issue for evaluating distributed systems, and extensive work has been conducted in this area. Various research and industry standard performance benchmarking solutions exist. TPC-C [36], TPCW [37], TPC-H [38], and SPEC OMP [39] are domain specific benchmarks: the first evaluates on-line transaction processing (OLTP) systems, the second applies to ecommerce web sites, the third evaluates decision support systems, and the fourth applies to parallel scientific computing applications that utilise OpenMP. SPECweb is another performance benchmark that evaluates web applications from different domains, such as banking, e-commerce and support [40].

Even though these benchmarks are useful in analysing distributed systems in general, they are not applicable to MapReduce systems. The need of MapReduce benchmarking is motivated by the many recent works that have been devoted to the study and improvement of MapReduce performance and scalability. These include task scheduling policies in MapReduce, cost-based optimization techniques, replication and partitioning policies. MRBench also provides microbenchmarks in the form of a MapReduce implementation of TPC-H queries [41]. However, these microbenchmarks are not representative of full applications with complex workloads, and they do not provide automated statistics for performance. More recent work proposed HiBench [42], a benchmark which evaluates Hadoop in terms of system resource utilization, and Hadoop job and task performance. Although low-level resource monitoring may be useful in targeting specific system features, HiBench does not avoid the pitfalls of microbenchmarks, missing multiuser workloads for batch or interactive systems.

To evaluate our platform, we present four case studies. We investigate the scalability of the system with regard to the size of clusters and data input. Also we investigate, how chunks' size and number of files slave nodes get per request, affect data processing.

C. Chunks' file size and number of files get per request

After a lot of experiments we found that the chunks' file size and the number of files gets by slave per request effect the amount of time for data processing. We conducted experiments running on one node for (a) 403MB, (b) 741MB and (c) 1.9GB data input size and 10MB, 100MB and 200MB chunk file size. presents the performance results of the experiments. shows the relationships between the amount of processed data and the execution durations for a given cluster. Response time results show that,

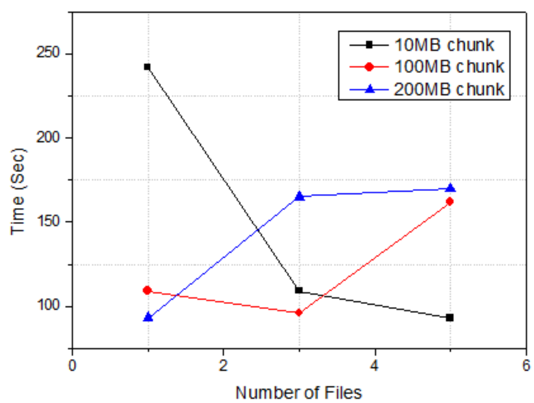
- when we can get a small number of files per request it is better to split data at bigger chunk size
- it is better to request bigger number of files when smaller chunks files are available
- in any other case the option of 100MB file chunk would gave the best results

In the case with wireless moving nodes the best scenario would be to use small chunks files, due to bandwidth limitations.

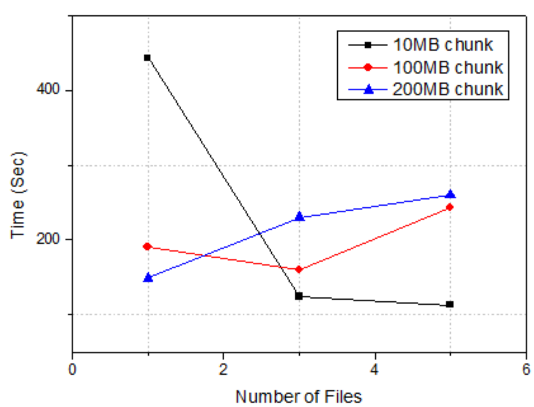
represents the relationships between the amount of chunk size and the execution durations for (a) 1 file slave request and (b) 3 files slave requests. Response time results show that, the best scenario would be the use of 100MB chunks file size and 3 file get per request.

D. Scalability With Regard To Cluster Size

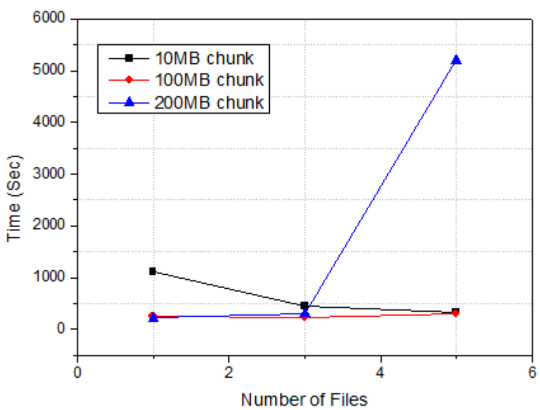
In this case study, we evaluate the scalability of our platform with regard to the size of clusters. We conducted experiments running clusters of different sizes: 3 and 7 nodes. We compare the results of these clusters with the results obtained when running on one node. We know that using more nodes every time we have and smaller processing times. So we run only one experiment just to see that this is true.



(a) 403MB Data size input.

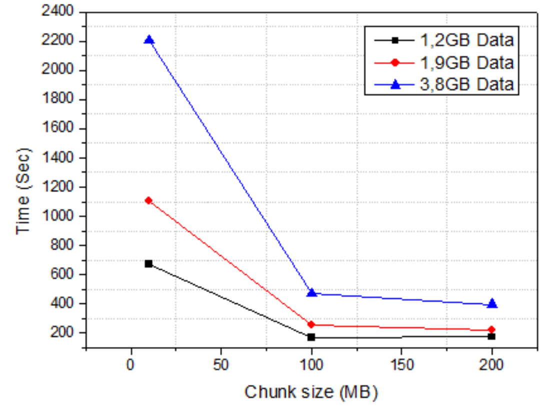


(b) 741MB Data size input.

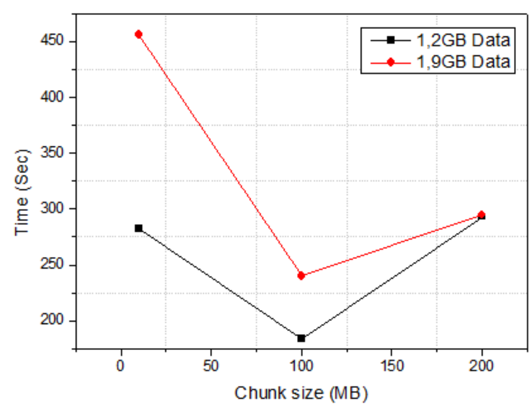


(c) 1.9GB Data size input.

Figure 9 The relationships between the amount of number of files per slave request and the execution durations for (a) 403MB, (b) 761MB and (c) 1.9GB data size input



(a) 1 file slave request.



(b) 3 files slave request.

Figure 8 The relationships between the amount of chunk size and the execution durations for (a) 1 file slave request and (b) 3 files slave requests

Here, response time results show that, using more nodes to process data the time is needed is smaller with 3.8GB data size input. We can see that in Figure 10.

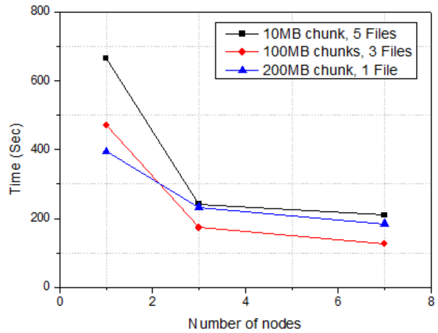


Figure 10 The relationships between the amount of number of nodes and the execution durations for 3.8GB data size input

E. Finding best case

In this case study, we investigate how chunks' file size and number of files get by slave per request effect the amount of time for data processing by the cluster. We conducted experiments with 3.8GB data in the follow scenarios:

- Small number of files per request (1 file) and big chunk size (200MB)
- Big number of files (5 files) and small chunks files (10MB)
- The best option of 100MB file and 3 files per request

Figure 10 presents performance results as functions of cluster size in the above 3 scenarios. As we can see the best option would be to use 100MB file chunks and getting 3 files each time we request data. We must notice that the differences are small because these three solutions are the best options we use.

F. Scalability With Regard To Data Size

In this case study, we investigate the scalability of our platform with respect to the size of input data. We conducted experiments with different sizes of input data: 403MB, 761MB, 1.2GB, 1.9GB and 3.8GB, 100MB file chunk size and 3 files get per slave request. The experiments were conducted on a 7 node clusters. Figure 11 presents performance results as functions of input data size.

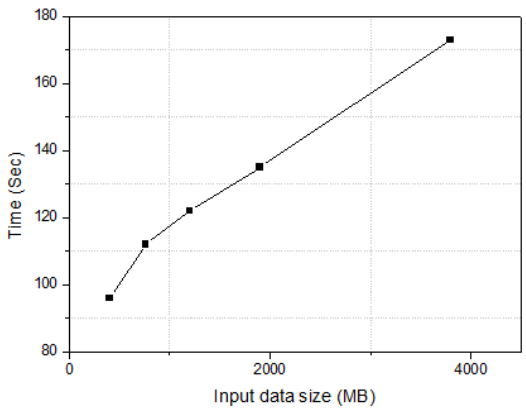


Figure 11 The relationships between the data input size and the execution durations for 403MB, 761MB, 1.2GB, 1.9GB and 3.8GB data size input.

3.4.8 Conclusion

We have described the implementation and tuning of a system running over Hadoop that can transform a set of backend servers and moving vehicles into a computational infrastructure into such a way that it is completely transparent to the application programmer. The application programmer does not need to know the specifics of the underlying hardware, but it focuses into the implementation of the application logic, and not the management of the hardware. Our system can be used to implement various types of applications, such as distributed data mining tasks, video streaming, in-network data aggregation, and so on. Apart from its usefulness into the layers stack of REDUCTION's architecture, it is also a novel addition into the literature of vehicular clouds that can fuel new research and development efforts.

4. FlexDanmark Field Trial

The following main issues will also be addressed for the advanced prototype to be used in the second field trial for FlexDanmark.

- Multi-modal transport using buses and mini-buses/taxis/trains will be covered in detail.
- Using a set of trips will be large and cover a larger geographical area; ideally the second field trial should look at one year of trips from FlexDanmark covering all of Denmark.
- A number of minor technical changes that will not be covered in this deliverable.

Compared to the first field trial at FlexDanmark the software prototype is extended with functionality to handle specialised digital road maps as described in D3.1. These specialised maps describe where buses and trains can stop. The overall architecture of the prototype is unchanged for details on the software architecture please see D4.1.

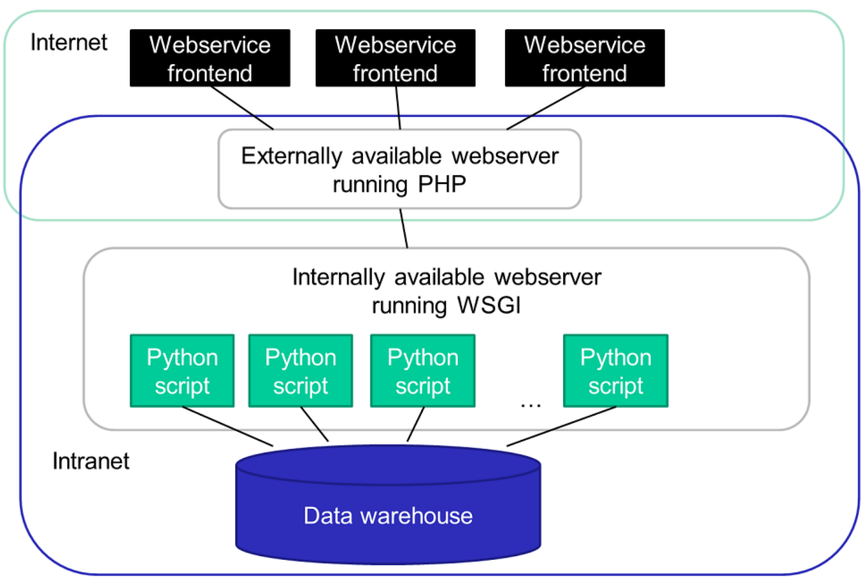


Figure 12 Architecture overview of Implementation

4.1 Multi-modal Transport

The digital networks used for trains, buses, and taxis are quite different. In addition, these networks are not digitally interconnected; special links need to be generated to connect the taxi digital network to the corresponding bus and train networks, respectively in order to achieve the use multi-modal transport, e.g., taxi + bus and/or train or bus + train. This is illustrated in Figure 13 and Figure

14 that outlines the rail network and bus network for Northern Jutland, Denmark. These two networks can be compared to the much larger and finer grained network for taxi, which can use all roads in the underlying digital network shown in both Figure 13 and Figure 14

Figure 13: Digital Network for Rails Northern Jutland, Denmark, Map from Google.

Figure 14: Digital Road Network for Busses, Northern Jutland, Denmark. Map from Google.

The networks are compared in the Table 7 below.

Type	Detail Level	Vehicle Types	GNSS Data	CANBus Data
Train	Coarse	Trains only	No	No
Buses	Medium	Buses, minibuses	Medium	Medium
Taxi/person car	Fine	Taxi, minibuses, buses, trucks person cars	Large	Medium

Table 7: Comparison of Digital Network for Vehicles types.

Table 7 shows that the detail level of the train network is coarse and in fact is a special network. The train network is only used by trains and for the second field trial neither GNSS data nor CANBus data is available for estimating travel time or fuel consumption of the trains.

For buses the level of detail of the digital network is medium. By buses we mean buses that drive on specific routes to pick up passengers typically according to a schedule. The digital road network used by buses and minibuses is typically a subset of the complete road network used by person cars/taxis with the addition of a number of roads that can only be used by buses ("traps" in road network such that taxis and person cars cannot use the special bus roads). Table 7 shows that for the second field trial a medium set of GNSS data and CANBus data is available.

For taxi/person cars Table 7 shows that a very detailed digital road network is used. This road network is used by most vehicle types. For taxis/person cars a very large set of GNSS data is available and a medium set of CANBus data.

From Table 7, Figure 13, and Figure 14 the following challenges must be addressed in the second field trial.

- Find the digital road network for trains and buses.
- Estimated the travel-time for trains from the train schedule.
- Estimate the fuel consumption of trains from the literature.
- Enable multi-modal routing.

The multi-modal routing is clearly the biggest challenge for the second FlexDanmark field trial. However the challenge with finding train/bus networks and the fuel consumption is non-trivial. The multi-modal routing will therefore focus on the route from the city of Aalborg to the city of Frederikshavn (both larger cities in Northern Jutland Denmark). The Aalborg-Frederikshavn route has been chosen for the following reasons.

- It is a route that is often used.
- It is serviced by train, buses, speedy-buses, and taxis.
- There is a substantial difference on the travel-time experienced by train compared to taxi and bus.

4.2 Changing Modality

The major new item explored in the second FlexDanmark field trials is the multi-modality, which is currently not supported in the software prototype developed. In this prototype the fundamental data structures used for computing travel-time and eco-routes is a directed graph. Graphs are very well-supported data structures in the software used and graphs can be very efficient to use. For these reasons the second field trial will reuse the graphs as the underlying fundamental data structures. However, the graph data structures must be annotated with additional information. This is illustrated in Figure 16.

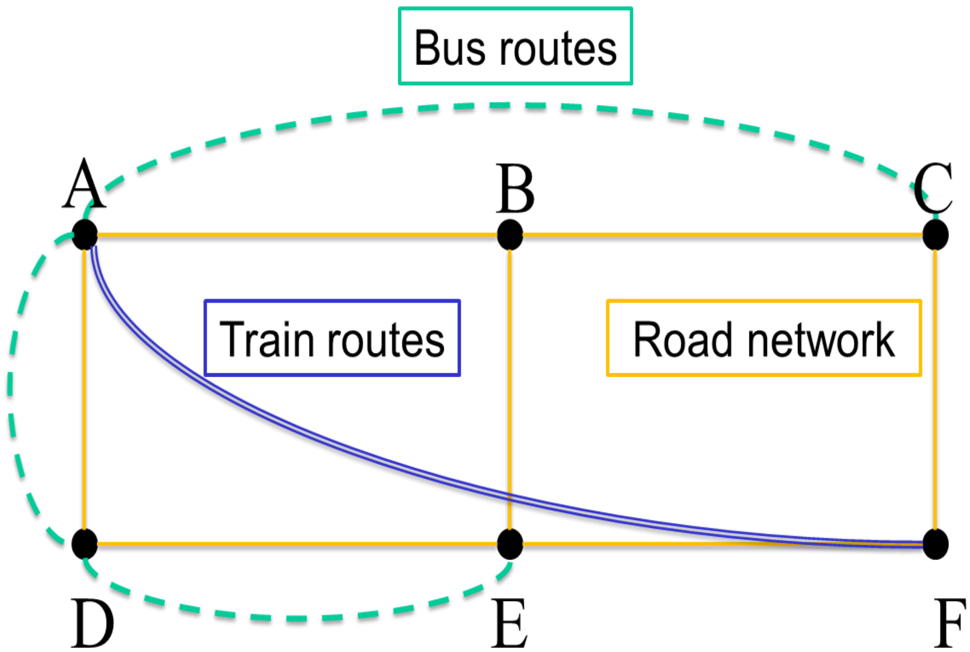


Figure 15: The Annotated Graph

In Figure 16 , the yellow road network illustrates the taxi/person car currently used. The green parts of the graph illustrate the bus road network (or bus routes) that are added to the complete graph. Finally, the blue parts of the graph illustrate the train routes. Note that is possible to change

- from taxi to bus in nodes A, C, D, and E, and
- from taxi to train in nodes A and F, and
- from bus to train in node A.

Annotating the graph makes it possible to retain the idea from the first field trial where eco-weights are added to all edges in the underlying graph. This is illustrated in Figure 16 where fuel weights have been added to all edges for all three networks.

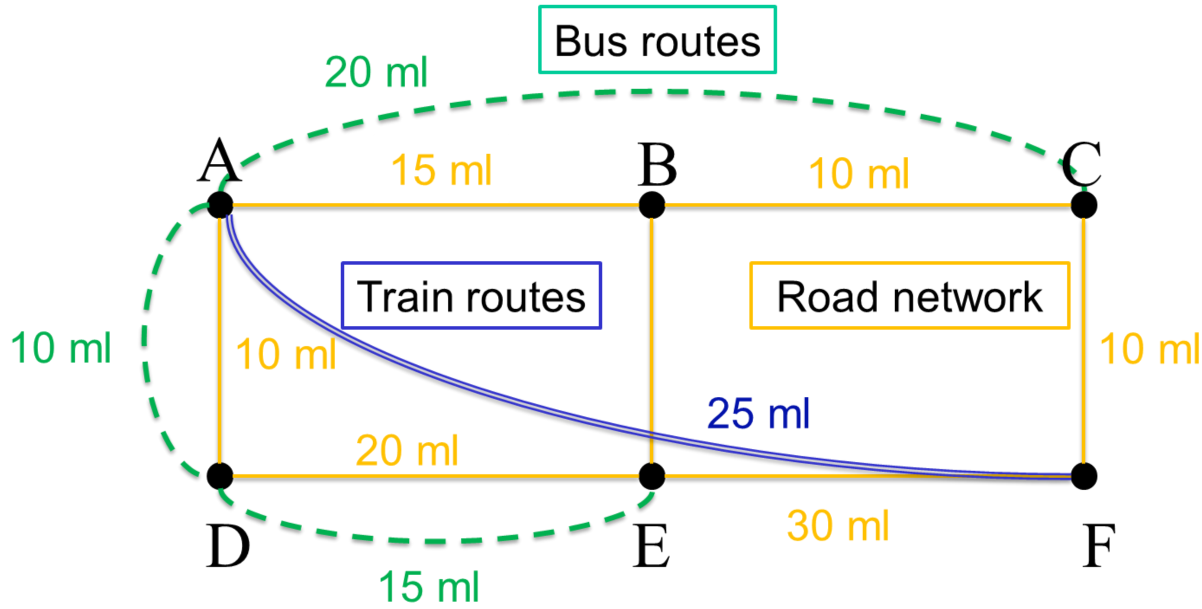


Figure 16: Annotated Graph with Weights

To implement the joined annotated graph illustrated in Figure 16 a set of independent graphs will be joined by creating locations where it is possible to change modality. The locations are basically new edges added to the joined graph. The idea with these new locations is illustrated in Figure 17. Here the upper green taxi/person car graph (or road network) is joined with the train network (Figure 13), and the bus network (Figure 14). The black edges in Figure 17 are the new locations.

Note that location edges must be manually annotated with travel time. As an example, one location edge can be annotated with a 30 minutes travel-time to illustrate that changing from a taxi to a bus takes considerable time because the bus drives according to a fixed schedule and the passenger has to walk from the taxi stop to the bus stop and wait for the bus to arrive.

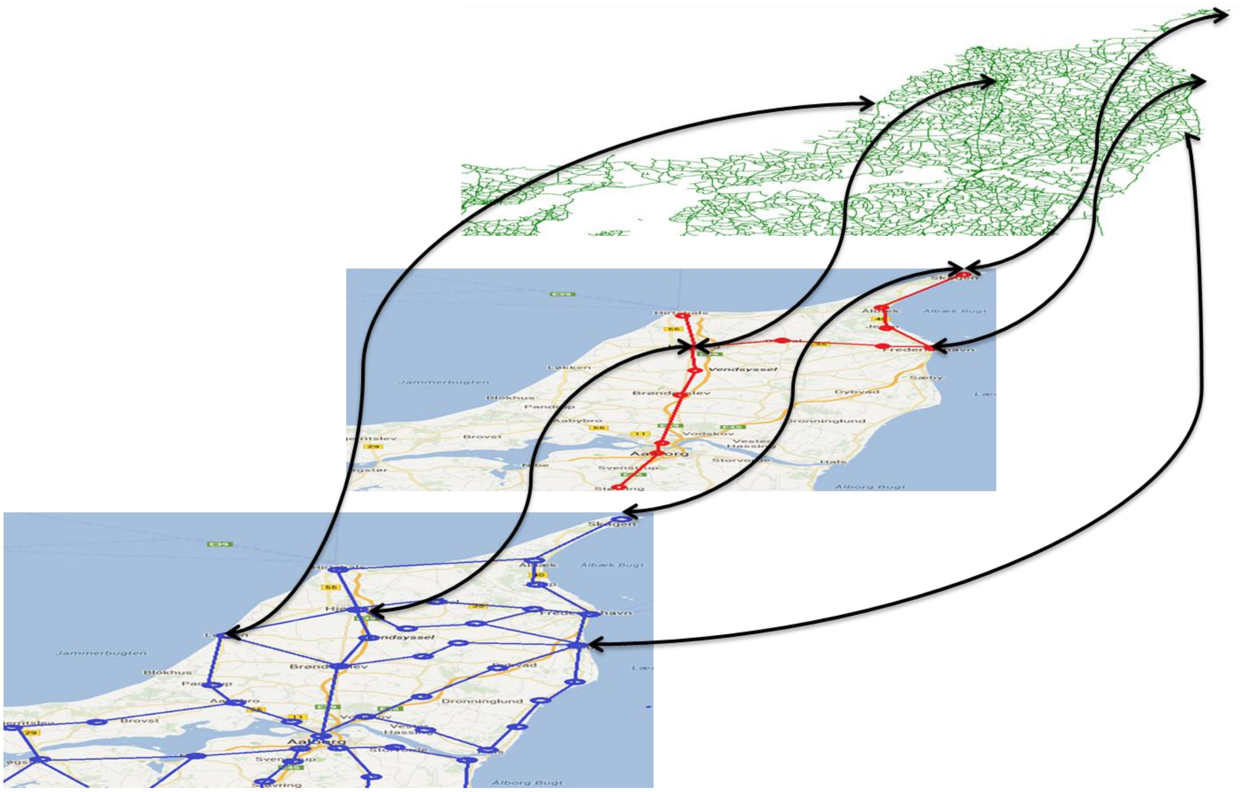


Figure 17: The joining of Independent Graphs using Location Edges (the black Arrows)

4.3 Trip-based Data

The first field trial looked at point-based measurements (low-frequent GNSS data). Where high-frequent GNSS data is available such data will be integrated in the second field trial. The main benefits of trip-based GNSS data are that real-world effects such as turn-time and signal coordination can be taken into consideration for both the travel-time and the fuel consumption. The extra data provides more accurate estimations.

Figure 18 shows how trips created from high-frequent GNSS tracks on the street Vesterbro in Aalborg, Denmark can be identified using start (S) and end (E) markers on a digital map. When the user clicks on the blue route between the start and end points the actual trip information about the route will be shown. In Figure 18, the pop-up box shows that 350 trips are found on Vesterbro on a 1,241 meter road stretch with an average travel-time of 165 seconds. The fuel consumption is here listed as Not-Available (N/A) because we for this road have plenty of GNSS data but currently no CANBus data.

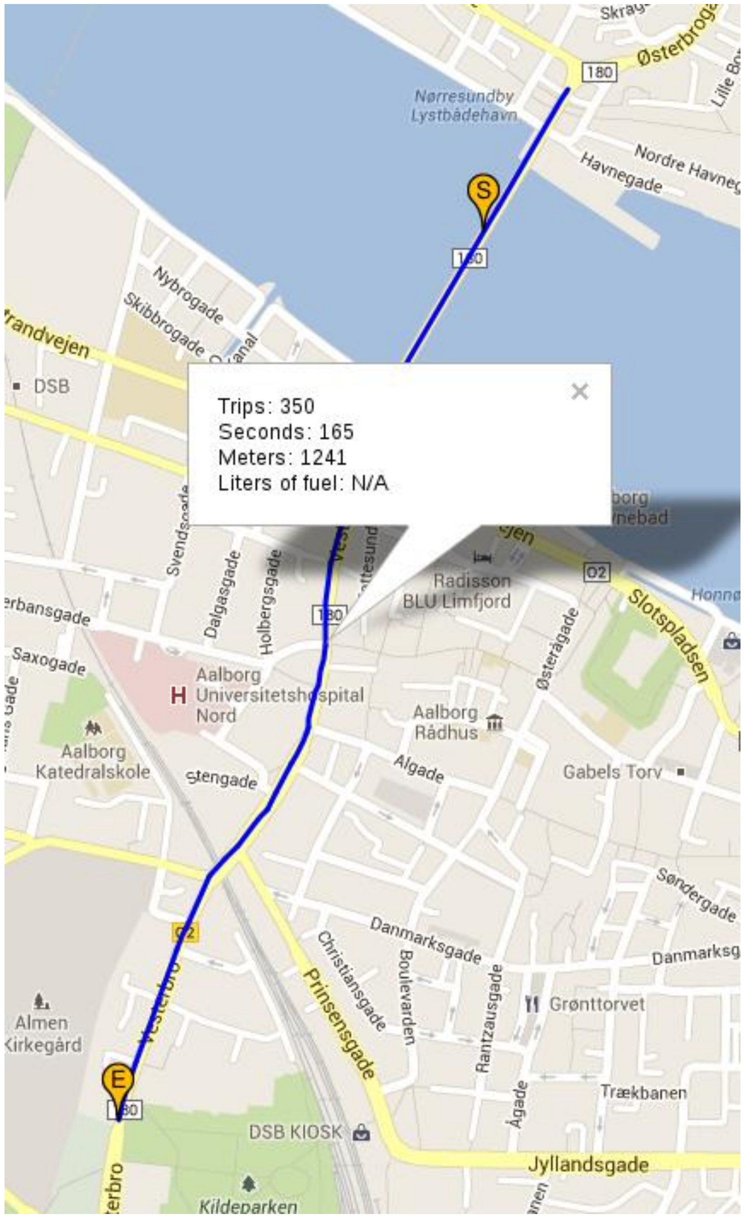


Figure 18: Finding Trips on a Route

To enable the user of the system to study the influence of for example signalised light intersections Figure 19 shows the speed details for the latest ten trips on Vesterbro in Aalborg (not exactly the same part of the road as shown in Figure 18). Figure 19 clearly shows where vehicles often stop (or lower the speed) due to traffic lights or roundabouts. The columns to the right allows the user to filter the trips examined, by for example only selecting Wednesdays between 8:00 and 9:00 in the years 2010-2013 to show the effect of morning congestion. Note that the route picked Figure 18 may contain both left and right turns.

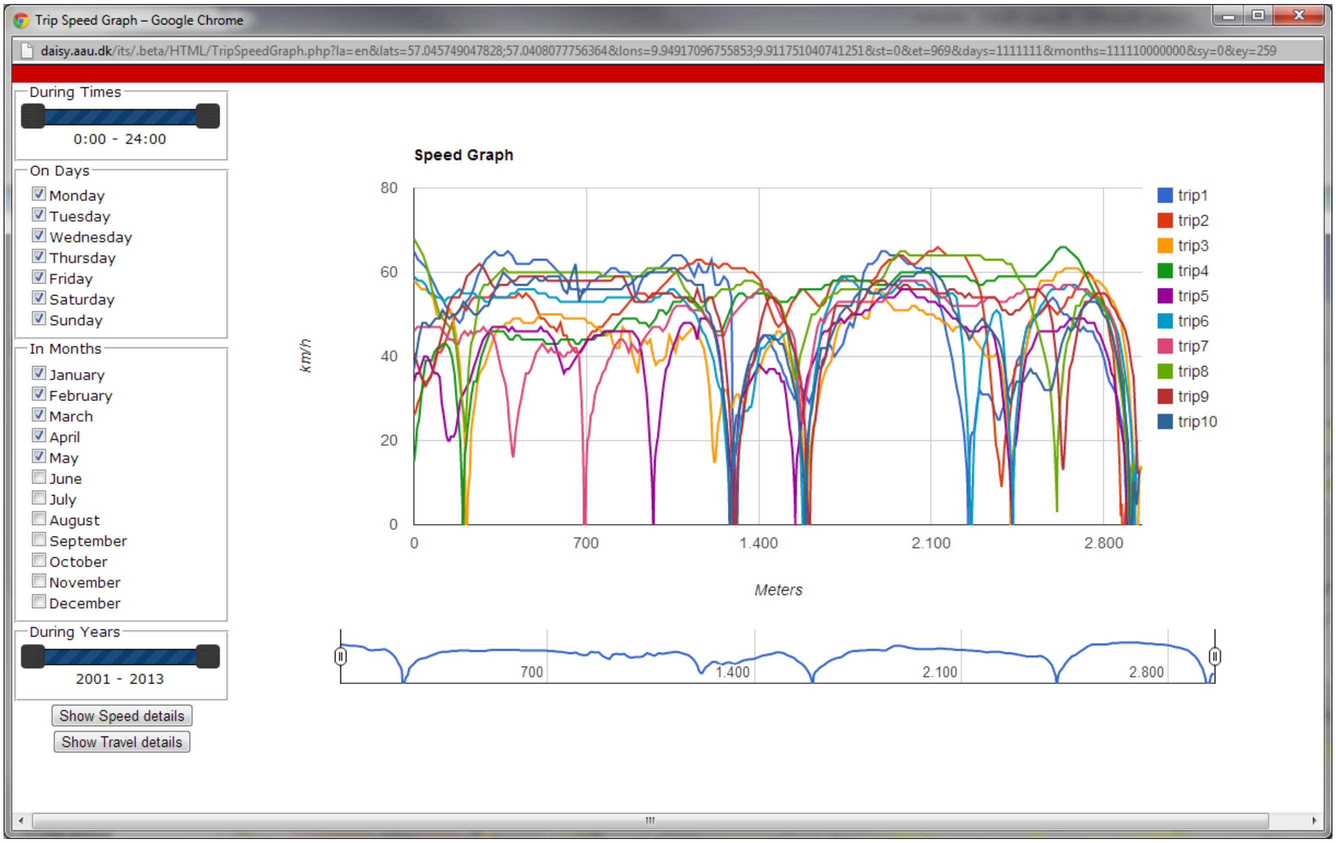


Figure 19: Using High-Frequent GNSS Data to Identify Intersections

4.4 Challenges Second Field Trial

The multi-modal scheduling is the main new issued explored in the second field trials. A number of challenges need to be addressed as follows:

- Penalty for changing modality?
- How many times can a passenger change modality on a trip?
- From which modality are switches allowed?
    - Taxi→train →taxi? Makes sense
    - Taxi→bus →taxi? Probably does not make sense!
- How will the bus/train schedule be taken into consideration?
    - This will be handled in a simplified manner in the field trial to avoid having to integrate with the complete train/bus schedule systems.

4.5 Experiences from the daily use of trip based data collected continuously

The following are the main experiences and lessons learned from the field trial.

- More efficient routes and more precise driving times.
- Reduction of disputes with drivers who disagree with the estimated driving time; more objective.
- Ability to produce an ecological estimate of the trip GHG emissions and fuel consumption.
- Set-up the framework such that in the future the GHG-emissions can be reduced based on more efficient GHG-routes.
- Generally positive experiences when it comes to drivers and companies.

A full implementation in two out five regions was to go countrywide in June, 2013; yet we are waiting for the go ahead which is expected towards the end of 2013. FlexDanmark is not in control of whether the Danish traffic companies will use the described methods but are continuously encouraging them to adopt them.

The REDUCTION results could potentially be used in the future under the following settings:

- In tenders, there can be up to 10 – 12 a year
- Driving behaviour
    - Used by taxi companies today
    - Electrical cars
    - Bonus and punishment

For FlexDanmark there will, of course, always have to be a very tight balance between achieving ecological goals and revenue. FlexDanmark's main right for existence lies in us being able to secure transport as cheap as possible within the parameters politically decided.

In terms of achieving more eco-friendly driving, no matter which method chosen, it will probably be necessary with either national incentives or restrictions. As stated earlier, we have to provide the cheapest possible transport within the parameters set and if GHC reduction is one of them we will adapt to that.

The results so far from REDUCTION that enabled us to provide much more accurate driving times have in fact made us interested for also handling ambulances in and out of our system in an efficient manner. That is a significant step forward and a sign that we are on the right track.

4.6 Multi-modal Transport in Action

FlexDanmark is engaged in a continuous process to provide safe, punctual and secure multi-modal transport in every corner of Denmark. To do so FlexDanmark has ventured into a project, that is visualised in Figure 20 below.

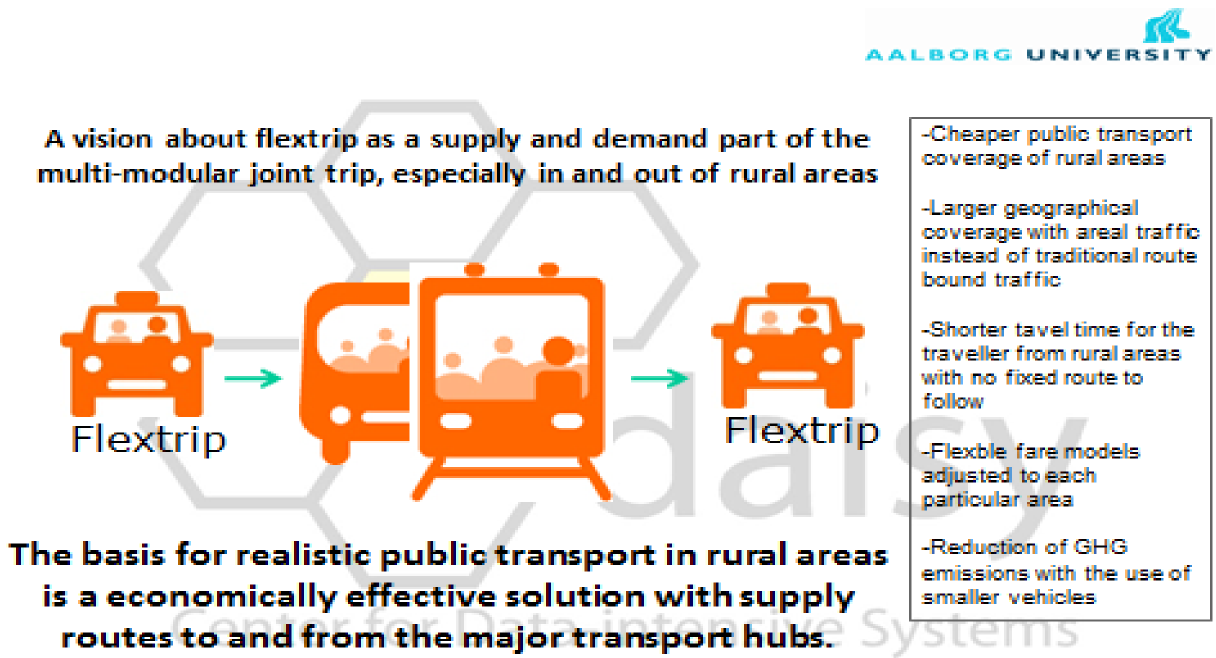


Figure 20: A vision about flextrips

The final goals for FlexDanmark are to provide transport with the following characteristics.

- A better service in rural areas
- Customers place **one** order for **one** multi-modal trip and receive **one** bill
- A better overview and understanding of multi-modal trips
- Have continuous real-time information
- Automatic trip rescheduling when delays occur

In the REDUCTION project, the fundamental steps for fulfilling these goals will be taken. This is in particular better support of rural areas using multi-modal transport and a better overview and understanding of multi-modal transport.

A goal of the second field trial is to add at least two specialised maps one for trains and one for buses. The GNSS data and CANBus data from the first field trial should be reused in the second field trial, but updated such that any new data received is also taken into consideration. The trips used to evaluate multi-modal trial should at least be the approximately 53,000 trips used in the first field.

5. TrainOSE Field Trial

The final phase of the second field trial deals with the following issues to be inserted in the advanced prototype:

1. Web application data enrichment and the survey for multimodal trips
2. The improvement of electric locomotives energy consumption.

5.1 Multimodal Web Site – Short Description

The Web Multimodal Application is one of TrainOSE's contributions along with locomotive drivers driving behaviour models and TRAIN-TAXI Service.
The application offers multimodal transportation information to TrainOSE customers when they seek for travel information in Greek Territory. Application described shortly in the following table:

Application Purpose	To offer multimodal transportation information to TrainOSE customers in an ECO way.
Application User Interface	User interface is quite simple, since it has to be "light" enough and not confusing for its purpose. There are two main webpages. 1. Input Web Page. This web page is devoted to users input. Page is divided to two areas. Right area is dedicated to users' input information via the following UI elements. 1.1. "Travel Calculation". Travel calculation is used for the definition of how users choose to define Origin, Destination points. (it can be by address, by lat/lon pairs, by transportation Point) 1.2. "Travel Date". This field describes the desired travel date. 1.3. "Way of travel". Users can choose among ECO, Cheapest, Fastest way of traveling. 1.4. Search Button. User clicks on this button in order to get trip results. Right area contains graphical elements for better visualization of the Origin/Destination Points 2. Results Web page. This webpage contains trip results as the result of trip calculation in the Input Web Page. It is divided into three parts. 2.1. The Header part contains (Graphical / Text) information of

	how user can reach the nearest transportation station form his/her origin point. 2.2. The middle part contains detailed information related to multimodal trip, and transportation (Bus, Train, Ferry etc.) means that have been used for the trip. 2.3. The footer part contains information of how user can reach his/her destination address from the last trip leg station.
Functionalities	Application functionalities divided to concealed and unconcealed functionalities. • Unconcealed functionalities are for application users, and they are related to trip calculations and trip information. • Concealed functionalities devoted to application administrators and their main purpose is the updating/deletion/insertion of : 1. Time tables for bus, trains, and ferries. 2. Schedules for bus, trains, and ferries. 3. Network Nodes. 4. Network Arcs. 5. Ticket prices for bus, trains, and ferries. 6. CO2 emissions.
Application Data	Application data pool contains the following data. 1. Nodes (Bus/Train/Ferry Stations) 1. 105 Central Bus Stations 2. 365 Train Stations 3. 8 main Ports 4. 2 mains Airports 2. Links. 685 links each one with its own 1. Timetable 2. Ticket price 3. Travel time 4. Travel distance 5. CO2 emissions
Types of Trips	Depending on user's selection from the "way of traveling" UI element, application can produce the following trip results. 1. Minimum cost trips. Usually these trips are Train trips since trains are cheaper than busses in Greek territory. 2. Minimum time Trips. Usually these trips are bus trips since

	train trips suffering from low to medium transfer delays between different lines. 3. Minimum CO2. These trips always use trains when it is possible since the average CO2 per KM is significantly lower for trains than for busses.

Table 8 Application description Web TrainOSE

5.1 Multimodal application and multimodal Survey

Multimodal application is completed in terms of programming effort and application features. Definitely an application is never completed since there is always space for more improvements and new functionalities. However for the purposes of the Reduction Project, current functionalities fulfil project expectations. The ongoing process is the data enrichment. The ultimate goal is to cover every small village in Greek Territory, making our web application the unique platform who offers such kind of detailed travel information.

Additionally to web application, TrainOSE designed and run a detailed survey, in order to catch users (TrainOSE customers) perception for ECO (multimodal) traveling. Every customer who visits our web application asked to offer his/her help to our efforts for understanding their perception of the multimodality.

The survey (Greek version) can be found in the ANEX section. However the electronic survey format can be found in the following http link as well

https://docs.google.com/forms/d/1XoiDm_2gD3soO1CPILdxqn10qshAdu_ujqx8fI8Bg3Q/viewform

Our survey was quite simple. It has only 10 easy questions to make it attractive due to short time for filling it. First 5 questions deal with general issues about the way of traveling. Next 5 Questions deal with specific issues related to uses willingness to use ECO multimodal types of traveling.
Some statistical analysis was made over survey results. Our initial conclusions of this survey are presented in the following figures:

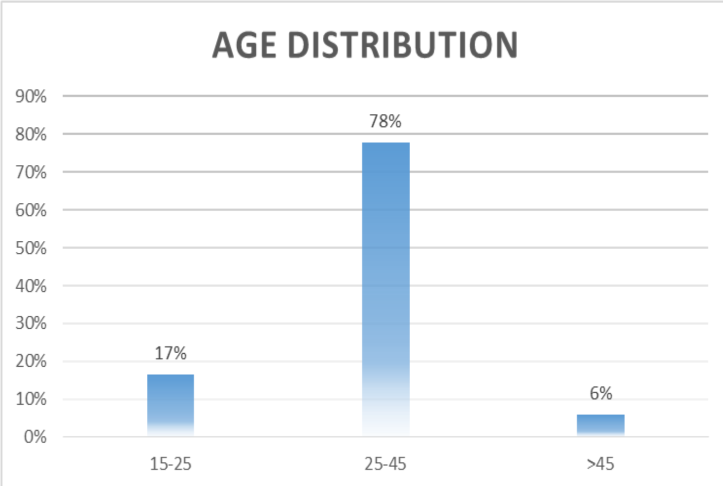


Figure 21 Age Distribution

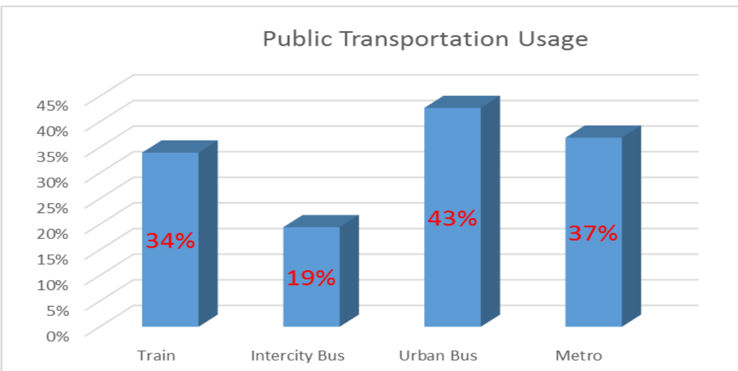


Figure 22 Public Transportation Usage

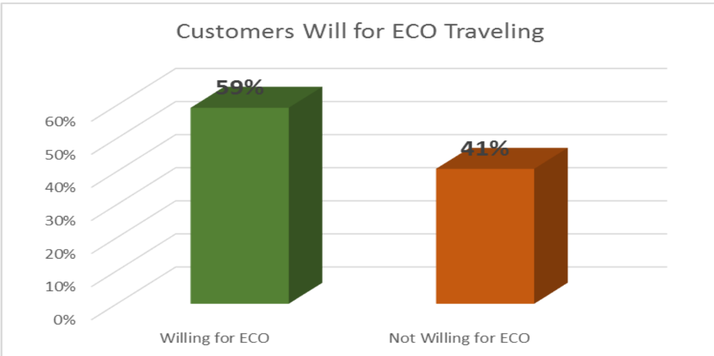


Figure 23 Customers will for ECO Traveling

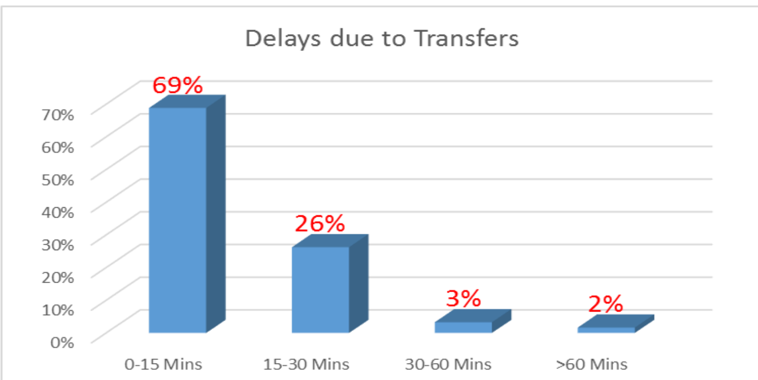


Figure 24 Delays due Transfers

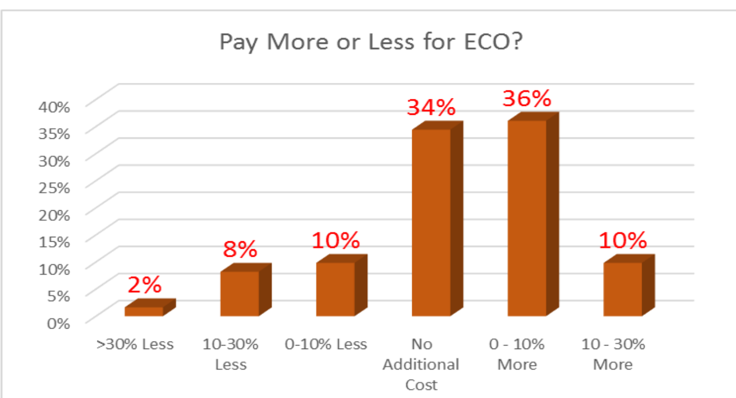


Figure 25 Pay More or Less for ECO traveling

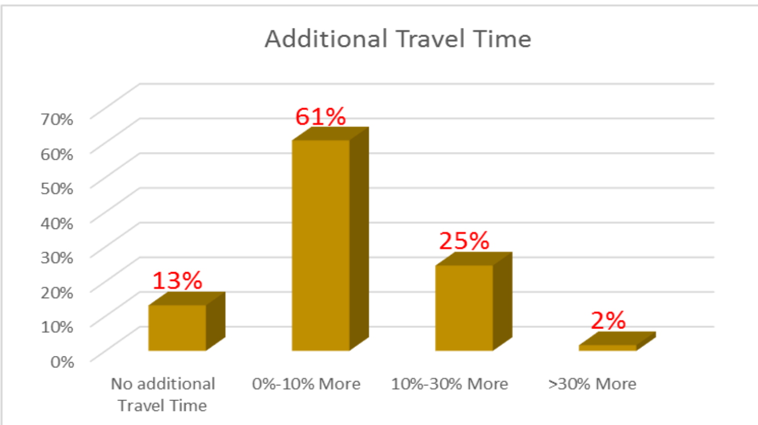


Figure 26 Additional Travel Time for ECO travelling

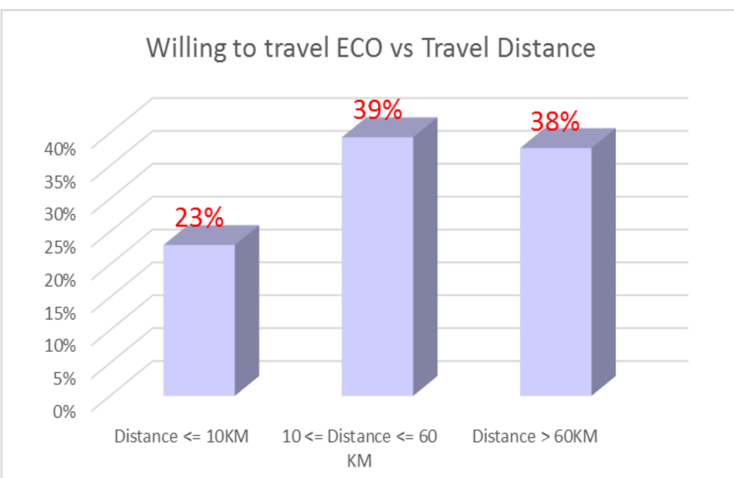


Figure 27 Willing to Travel ECO vs Travel Distance

According to the above referenced figures we conclude that

- Users accept the idea of Multimodal ECO driving in general.
- User's majority is willing to afford small transfer delays, close to 0-10 minutes for more ECO traveling.

- User's majority is willing to afford small additional cost, close to 0-10% for more ECO traveling.
- User's majority is willing to afford small additional travel time, close to 0-10% for more ECO traveling.
- Users Perception is, ECO Multimodal Traveling can fit Medium – Long trip distances.

5.2 Improving locomotives energy consumption

Improving locomotives (electric ones) energy consumption can contribute considerably in CO2 reduction given the fact that electric energy in Greece is produced by burning low quality coal (lignite). The basic idea behind this improvement was to change locomotive drivers' behaviour. The methodology we had to follow was quite obvious and can be described with the following steps:

Step 1: Collect energy consumption measurements for locomotives.

Step 2: Find the best energy consumption pattern among all measurements.

Step 3: Apply the best energy consumption pattern to test whether it is applicable under real conditions.

Step 4: Create a good driving behaviour manual.

Step 5: Disseminate the manual to locomotive drivers in order to follow written instructions when possible.

Each step of the above referenced methodology has been implemented and executed accordingly. The following paragraphs describe each step in more detail.

5.2.1 Energy Consumption Measurements

Our measurements focused on the main rail line Athens-Thessaloniki.

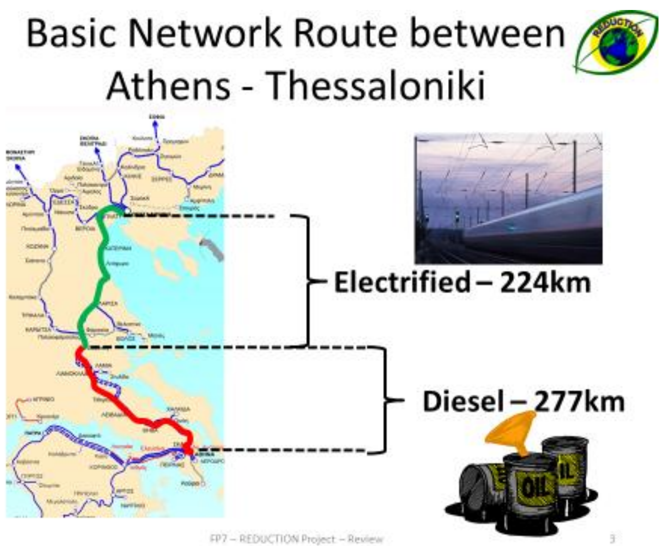


Figure 28 Main Rail Line (Athens – Thessaloniki)

Our study was focused on the electrified part of the line (Domokos-Thessaloni) with distance of 224 KMs. We took hundreds of measurements for this part, keeping all travel conditions the smae(locomotive type: Siemens 120, Train Type: Passenger Train). Figure 4.9 presents in more detail our measurements.

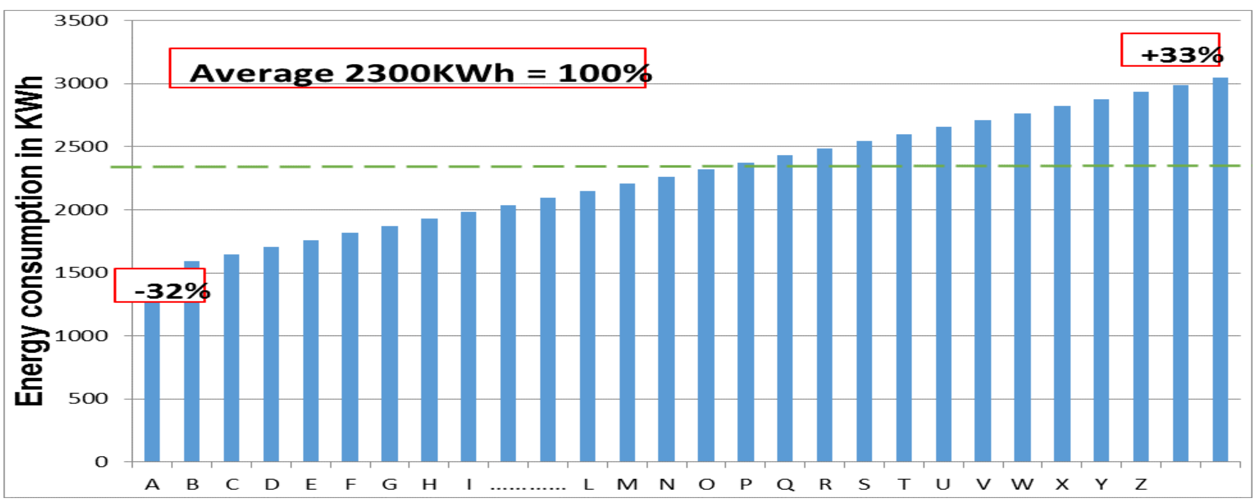


Figure 29Differences between energy consumption for passenger electrified locomotives

From figure 4.9 we conclude that:

3. The average energy consumption is close to 2300 KWs
4. less "Eco" locomotive drivers have consumption increased by 33%

5. More "Eco" locomotive drivers have consumption decreased by 32%.

Clearly there is a ground for improvement.

5.2.2 Construction of Best Energy Consumption pattern

In order to create the best energy consumption pattern, TrainOSE sent all these measurements to University of Hildesheim for further analysis.

University of Hildesheim proposed an offline driver behaviour adaptation approach (eco-driving) for Trains. An optimal driving behaviour policy was computed using Simulated Annealing optimization search over a collection of real driving behaviour data (realistic policy). Empirical findings show that if drivers would follow the recommended optimal policy, then an energy saving of up to 50 % is a realistic upper bound potential.

Finally, the University of Hildesheim send back to TrainOSE three graphs (as shown below) where the optimum driving behaviour is presented for three different scenarios.

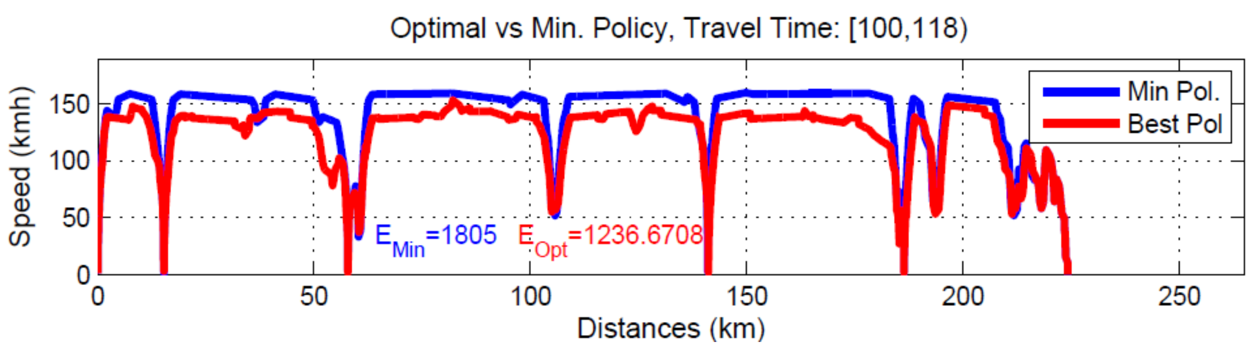


Figure 30 Optimal Driving Policy for travel time [100-118] minutes.

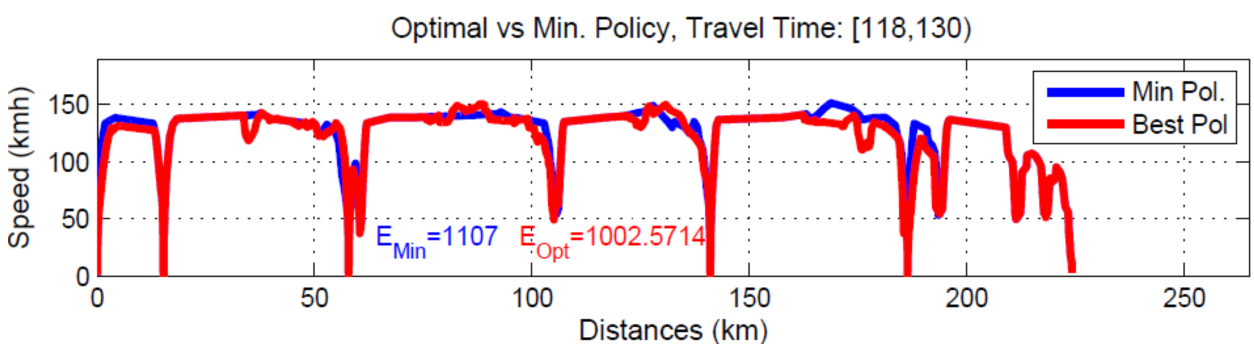


Figure 31 Optimal Driving Policy for travel time [118-130] minutes.

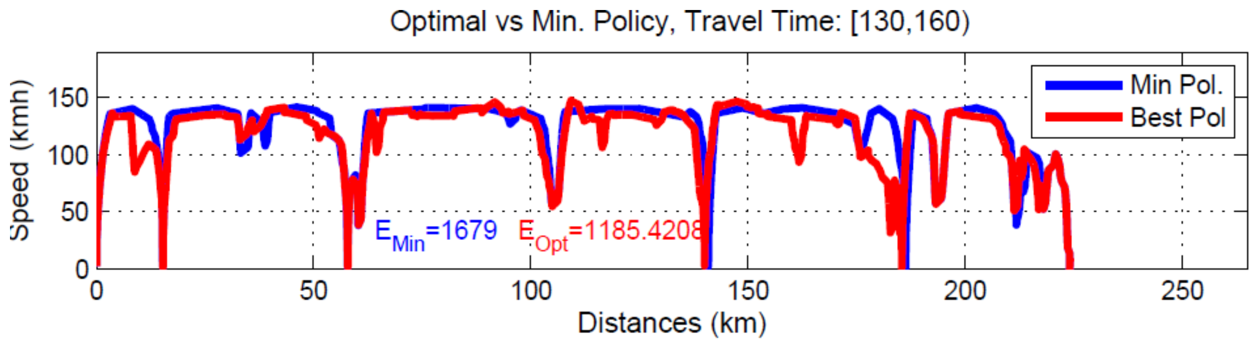


Figure 32 Optimal Driving Policy for travel time [130-160] minutes.

Based on these findings TrainOSE prepared a good driving behaviour manual.

5.2.3 ECO Driving locomotive drivers manual

The good driving behaviour manual is based on the findings of Figures 4.10,4.11,and 4.12. The most important part of this manual is the driving table. This driving table (Table 4.1) describes accurately the speed, the acceleration, the deceleration on specific route points.

Action #	**Action Description**
1	Acceleration FROM:0 Km TO:3 Km to SPEED:125
2	ROLL FROM:3 Km TO:14 Km to SPEED:120
3	Deceleration FROM:14 Km TO:15 Km to SPEED:1
4	Acceleration FROM:15 Km TO:17 Km to SPEED:126
5	ROLL FROM:17 Km TO:56 Km to SPEED:123
6	Deceleration FROM:56 Km TO:58 Km to SPEED:2
7	Acceleration FROM:58 Km TO:62 Km to SPEED:125
8	ROLL FROM:62 Km TO:101 Km to SPEED:126
9	Deceleration FROM:101 Km TO:105 Km to SPEED:49
10	Acceleration FROM:105 Km TO:107 Km to SPEED:126
11	ROLL FROM:107 Km TO:137 Km to SPEED:125
12	Deceleration FROM:137 Km TO:141 Km to SPEED:3
13	Acceleration FROM:141 Km TO:142 Km to SPEED:125
14	ROLL FROM:142 Km TO:183 Km to SPEED:125
15	Deceleration FROM:183 Km TO:185 Km to SPEED:37
16	Acceleration FROM:185 Km TO:186 Km to SPEED:62
17	Deceleration FROM:186 Km TO:186 Km to SPEED:0
18	Acceleration FROM:186 Km TO:189 Km to SPEED:121
19	Deceleration FROM:189 Km TO:192 Km to SPEED:100
20	Deceleration FROM:192 Km TO:194 Km to SPEED:55

21	Acceleration FROM:194 Km TO:195 Km to SPEED:125
22	ROLL FROM:195 Km TO:209 Km to SPEED:126
23	Deceleration FROM:209 Km TO:211 Km to SPEED:50
24	Acceleration FROM:211 Km TO:215 Km to SPEED:107
25	ROLL FROM:215 Km TO:217 Km to SPEED:91
26	Deceleration FROM:217 Km TO:218 Km to SPEED:56
27	ROLL FROM:218 Km TO:219 Km to SPEED:56
28	Acceleration FROM:219 Km TO:221 Km to SPEED:95
29	ROLL FROM:221 Km TO:222 Km to SPEED:80
30	Deceleration FROM:222 Km TO:224 Km to SPEED:3

Table 9 Optimal Driving Policy Table

The drivers' manual can be found at the ANEX section of this document.

5.2.4 Application and first results

The next step was to apply the good practice manual to real field. On 05/06/2014 we made our first trial. We delivered the good practice manual to locomotive drivers and we started the journey from Domokos to Thessaloniki. The following figures present evident of this experiment.

Figure 33 inside locomotive driving cabin minutes

Figure 34 Driver Follow our instructions for the best driving policy.

The following figures present the trip itself in terms of Geo-Location representation

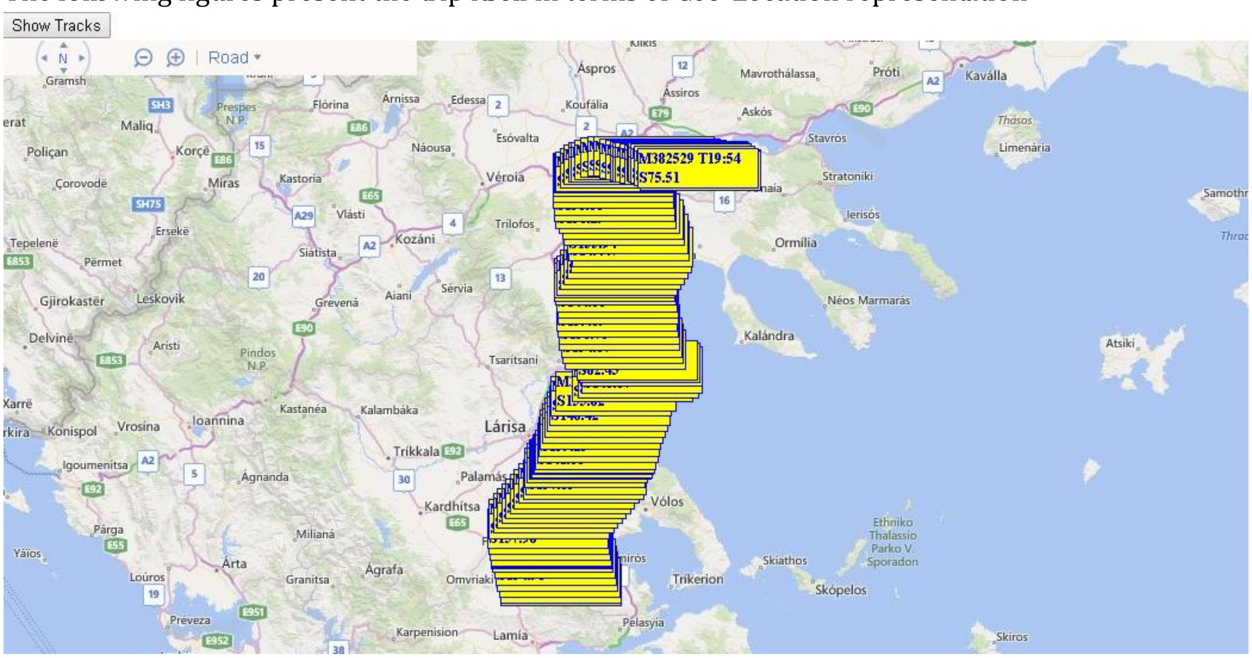


Figure 35 GPS tracks for the whole route on 06/05/2014

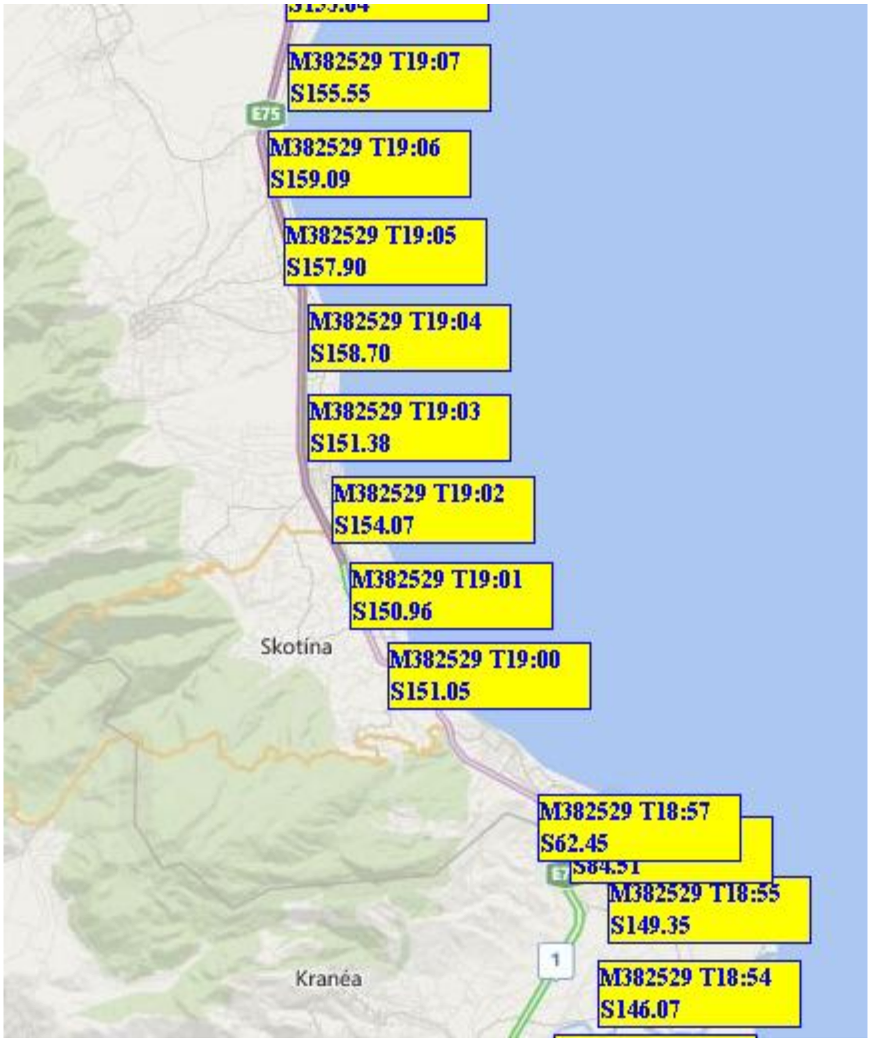


Figure 36 Detailed GPS tracks for the area outside the city of Larisa on 06/05/2014

All these tracks can be found on the following link
http://194.177.200.67/ttrack/home/index/?tpoints=0&datestr=2014-05-06

For the next four days, we made 4 more trials. Following figures describe the energy consumption for each one of these trials

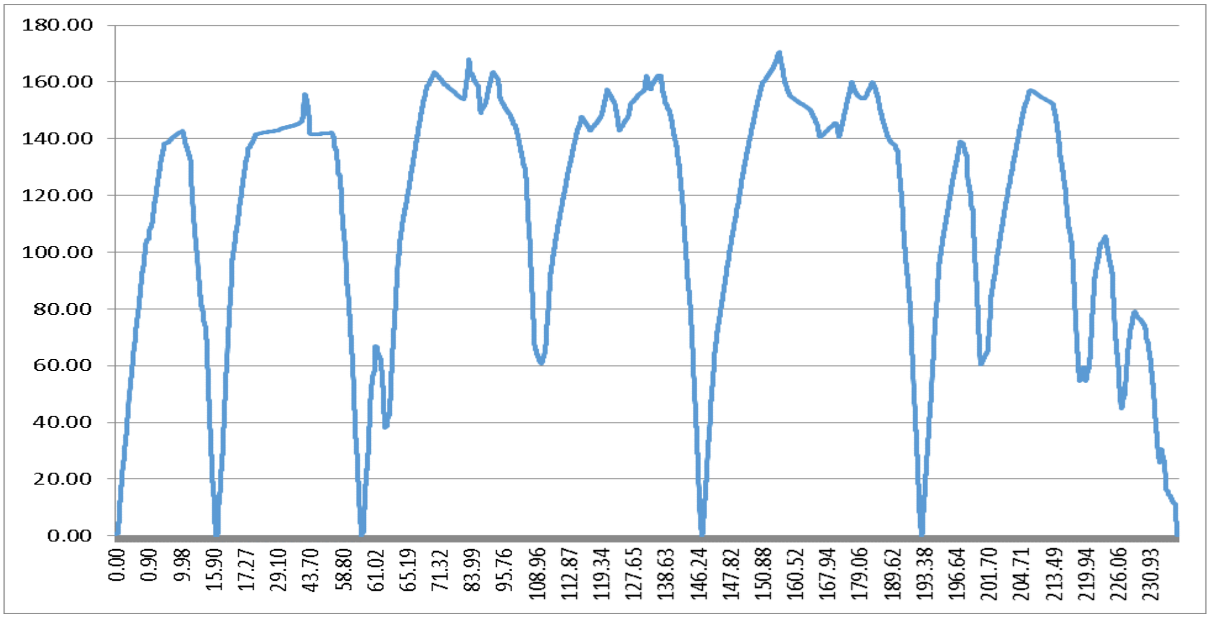


Figure 37 **First** day of trial.
Energy consumption was **1815** KWs

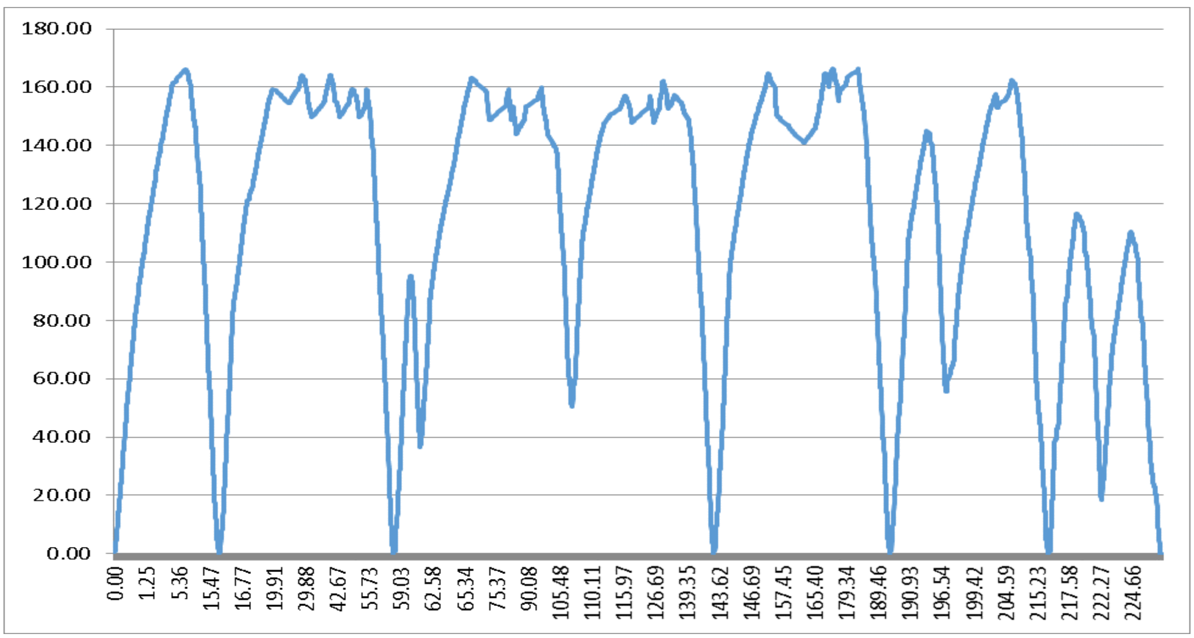


Figure 38 **Second** day of trial.
Energy consumption was **2073** KWs

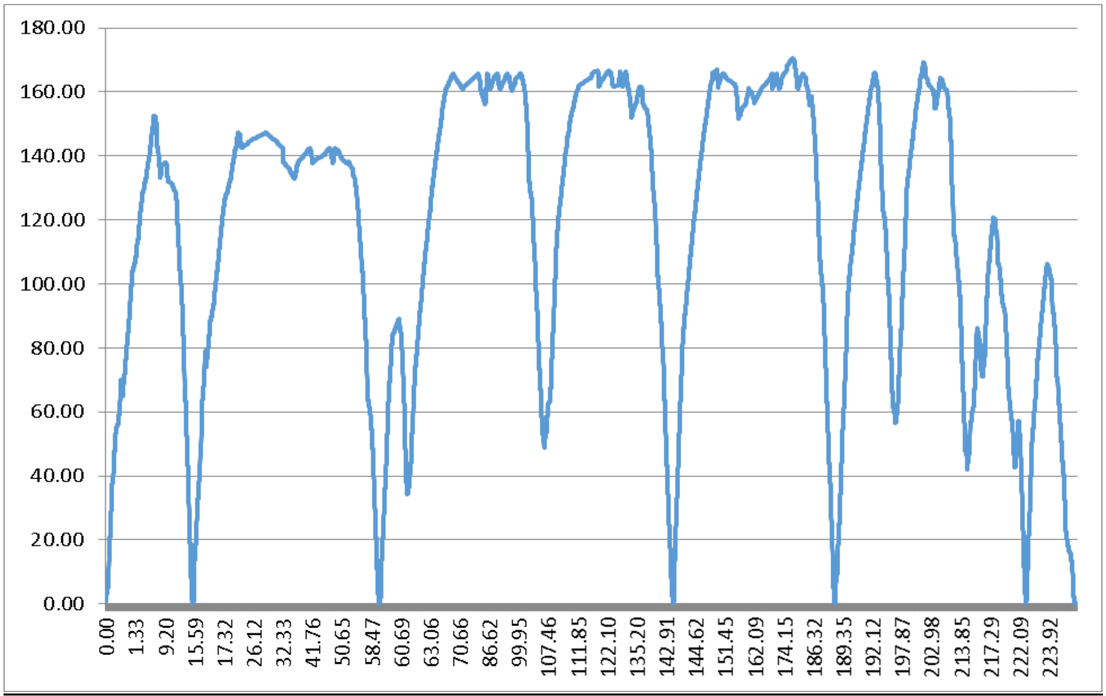


Figure 39 **Third** day of trial.
Energy consumption was **2012** KWs

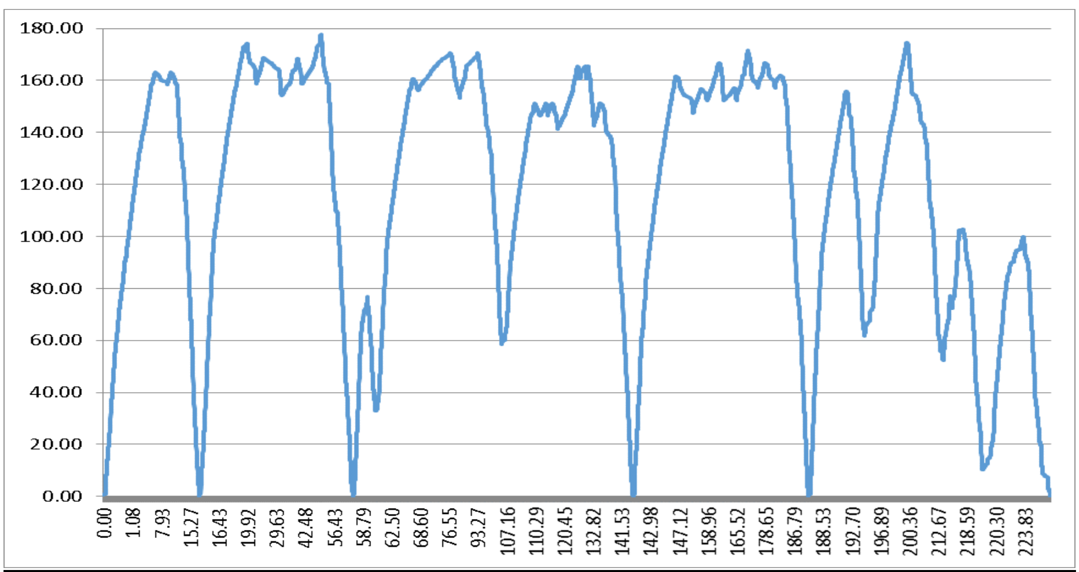


Figure 40 **Fourth** day of trial.
Energy consumption was **2135** KWs

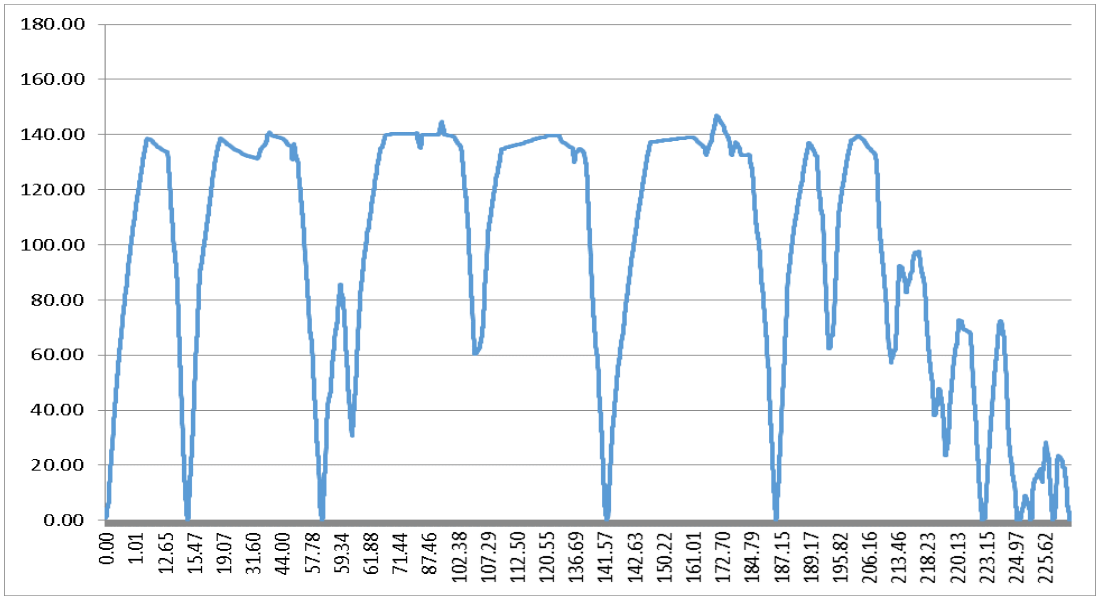


Figure 41 **Fifth** day of trial.

Energy consumption was **2135** KWs

Clearly all these experiments proved that our driving guide leads to better results for energy consumption given the fact that the average energy consumption is 2300 KWs.

The drivers' manual can be found at the ANEX section of this document.

5.2.5 Results Dissemination and Locomotive Drivers Training

Since our experiments proved that the good driving behaviour patterns from our driving manual produces better results in terms of energy consumption we proceeded to the next steps.

The first step was to gather heads of locomotive drivers unions in order to present them the manual and ask them to promote it to their locomotive drivers.

Figure 42 Presentation of our good driving manual to locomotive drivers (drivers' union).

They had their own objections over specific issues and we agreed to improve the driving manual a little bit, based on their suggestions.

The final step is to make our manual part of locomotives drivers' formal training in our local TrainOSE schools for locomotive drivers.

6. Nicosia Field Trial

The Nicosia fleet field trial architecture is described in D4.2.

The main objectives of the field trial are:

- Demonstrate the Delphi Delco Electronics GMBH (DDE) Vehicle to Vehicle and Vehicle to Infrastructure (V2V/V2I) device capabilities – this field test was decided not be done in Nicosia as it was deemed not feasible due to the problems encountered with the V2X OBU devices.
- Demonstrate a fuel efficiency and/or emissions reduction through a driver bus – OSEL bus company - and delivery fleet monitoring system – CPO delivery company - using the REDUCTION technologies:
    1. Read, store and send to a server CANbus data
    2. Analyse the driving pattern of fleet drivers (bus and delivery)
    3. Develop a fleet drivers' eco-guide to reduce fuel consumption and emissions
    4. Validate the drivers' eco-guide at a field trial.

Based on the changes to organization of the field trial and the following update is provided for each of the two types of fleets, the OSEL buses and the CPO delivery vehicles:

1) The Nicosia OSEL bus driver behaviour field trial is designed to test the capabilities of the DELPHI V2X OBU devices in retrieving CANbus fuel consumption, GHG emissions and GPS vehicle location and speed data, storing them on its solid state disk and disseminating this data remotely to a server for post processing. The bus field trial is conditional on the ability of the V2X OBU devices to read the Citaro Mercedes-Benz CANbus data. Updated full status:

    a. A preliminary test that was conducted in March 2013 was not successful in reading the CITARO bus CANbus data.

    b. A second test conducted in late August 2013 was successful and was determined that the V2X OBU could read the data from the Citaro FMS port.

    c. A new software was developed and installed on the V2X OBU.

    d. Various tests were conducted on one bus with the V2X OBU installed. Also various tests were conducted in the lab by CTL. The V2X OBU device could not operate properly under the environment of the Citaro bus as the device was found to stall without having the ability to store data – in some cases some data were stored but nothing significant. The main potential problems that were identified are: the V2X OBU could not handle the high temperatures found in Cyprus; The V2X OBU stalled when the voltage dropped below 11v. Various different configurations were attempted without any resolution. The main changes involved the use of a battery to ensure power supply for a few hours and a protective relay to ensure a voltage of 12v (the relay solution is now under testing by CTL – relays arrived on the 28 of July in Cyprus).

e. DDE and CTL are discussing whether it is feasible to carry out the OSEL field trial during the Fall of 2014 – REDUCTION ends in the31st of August 2014 – outside of REDUCTION - if the problems of the V2X OBU are resolved by the end of August, 2014.

2) The Cyprus IST-CPO fleet field trial involves a total of 68 delivery vehicles*.* CANbus-data were retrieved for four months, from September to December 2014. CTL retrieved the data from IST and map-matched to a GIS-link shape file. These data were then sent to UHI for analysis to identify fuel efficient or non-efficient driving patterns. UHI delivered the first analysis which is included in deliverable D53 based on the four month long data. The architecture for the IST-CPO remained the same as the previous year.

Nicosia OSEL Bus Field Study Summary

REDUCTION contribution to OSEL field trial: This architecture is all new for OSEL.

Test Bed: Nicosia Greater Region
Number of Buses: 5 Mercedes-Benz CITARO buses
Number of Routes: 5 buses operating on the following potential routes with final destination the centre of the City at Solomou Square which is the main bus terminal: 116 (Syn. Strovolou), 158 (Pera Horio Nesou), 160 (Geri), 110 (Pano-Lakatameia), 112 (Tseri), 157 (Arediou)
Operational times: ranging from 5:00 to 21:00 and frequencies of 15 to 30 minutes during weekdays and up to 90 minutes on the weekends
Length of route: ranges from 4 to 20 Km as most of the routes cover communities outside the greater Nicosia region while all arriving at the centre of the City of Nicosia at the main bus terminal of Solomou Square.
Clients: five (5) V2X OBU devices; to be installed at each bus
Servers: one server and one back-up server provided by Istognosis Ltd. company
Wireless communication: From/To V2X OBU devices, To/From server via 3G MTN network

Development of OSEL fuel efficient driving profiles: REDUCTION contribution to OSEL.

Nicosia CPO Fleet Field Trial Summary

REDUCTION contribution to IST-CPO field trial: The CPO already implemented a fleet management system. The main new aspect of the IST-CPO field trial will be the development of fuel efficient driving profiles that will be based on the methodologies that will be developed by the REDUCTION partners.

Test Bed: Nicosia Greater Region with potential of utilizing the entire Cyprus region
Number of delivery vehicles: up to 68
Operational times: ranging from 7:00 to 18:00
Length of routes: ranges from 2 to110 Km; where the longer trips refer to deliveries from Nicosia to Paphos area and the shorter trips within the Nicosia region.

Clients: up to 68 FMS devices; already installed by the system integrator Istognosis Ltd. company (external to REDUCTION)
Servers: one server and one back-up server provided by Istognosis Ltd. company (external to REDUCTION)
Wireless communication: From/To FMS devices, To/From server via 3G MTN network (external to REDUCTION)
Development of CPO fuel efficient driving profiles: Main REDUCTION contribution to CPO.

6.1 Architecture of the OSEL and IST-CPO fleet field trial prototype system

The architecture of the OSEL Bus and IST-CPO Fleet Field trial system advanced prototype will be based on the paradigm of client server realized in four layers, namely: data capture layer that will be realized by a number of clients, application layer, database layer and web interface as an additional layer. A detailed pictorial description of the architecture used to conduct the Nicosia field trial is shown in Figure 8.

The system's clients will mainly concentrate on data capture activities through monitoring the on-board sensors via sniffing the transmission lines of the bus's local area network. Each client will be composed of a number of modules. Depending whether the transmission of the monitored data will be realized in real-time or batch mode the analogous data transmission modules will be activated on the clients. Specifically for real time transmission a 3G mobile network will be utilised. For batch processing the data will be stored on the clients' data store and uploaded on the data server using an ftp connection with the server-side, upon entering the Wi-Fi zone of the bus depot. Client sub-systems will be installed on five OSEL CITARO Mercedes-Benz buses. Clients will be equipped with the Delphi Delco Electronics GMBH (DDE) DDE DELPHI V2X OBU device (OBU). The CPO has already installed its own fleet management system. Under REDUCTION, an API software will be installed that will gather CANbus data from each CPO delivery vehicle and send to the CPO server.

Data sniffing is achieved through a connection of the V2X OBU to the Fleet Management System port of the Citaro bus via a custom made connector developed by DDE. This approached was explored tested and a solution was found during Year 2 of the project. The CANBus data that are read by the V2X OBU via the FMS port are: GNSS at one-second time intervals, fuel consumption, GHG emissions and other engine data. The CANBus data that will be read by the CPO FMS are: GNSS at 10 sec time steps, fuel consumption at 2-second time steps; no GHG emissions are readily available.

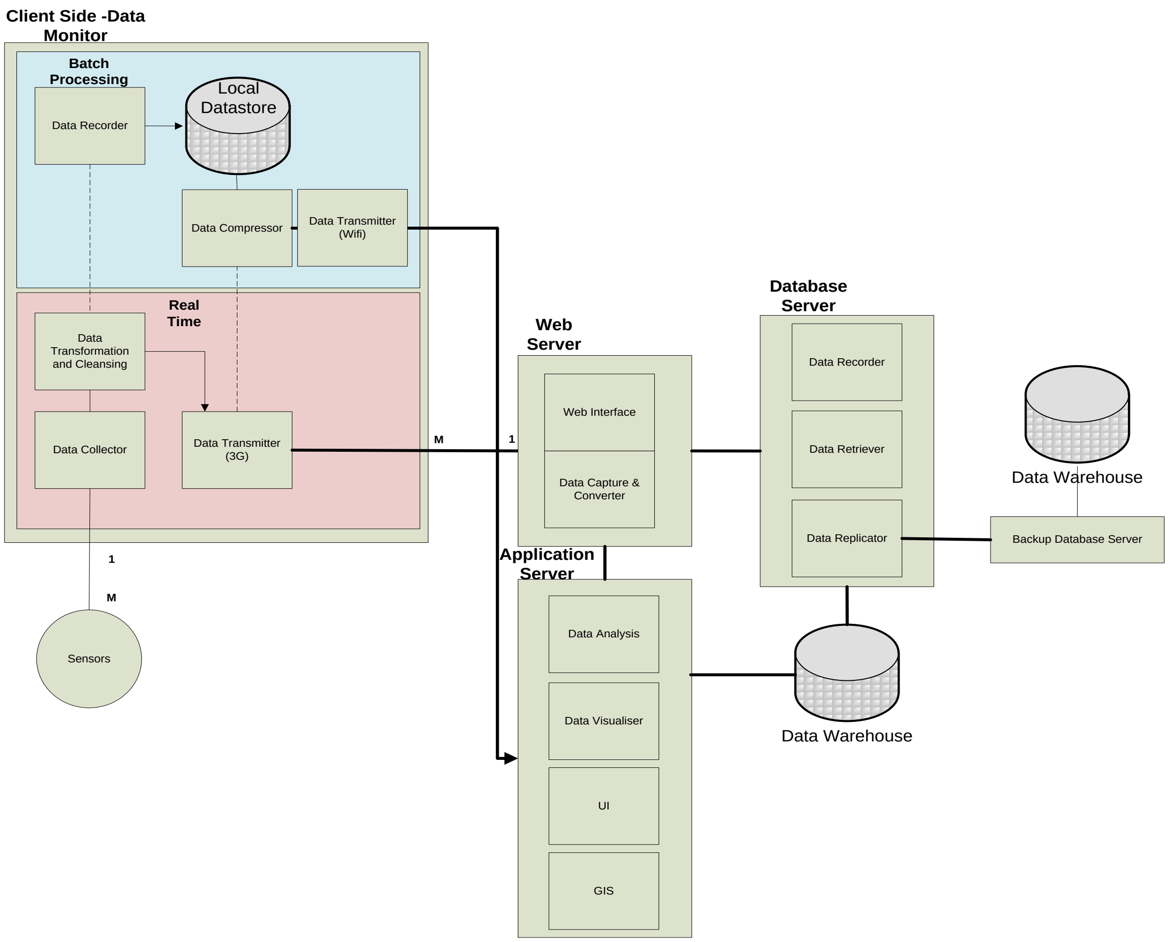


Figure 43. Cyprus Fleet Field Trial Pilot System Architecture

The server side of the architecture will mainly address the data warehousing functionality and will be composed of: (a) a web server for real time communication with the clients, (b) an application server accountable for data analysis and visualization using (c) an embedded GIS subsystem and (d) UI functionalities. The back end of the architecture will mainly focus on data storage and retrieval functions along with data replication. Pre-processing of the raw data from the on-board sensors will be performed by the clients hence the data will be ready for storage upon arriving to the data server. Data transmitted to the server via 3G will be compared with batch-mode data, firstly for validation purposes, secondly to test the robustness of the communications network and thirdly to check the capabilities of V2X OBU to operate in real-time environments. The populated data warehouse will be mined using relevance analysis and association rules techniques to identify possible links between driving behaviours and fuel/CO2 emission plus, the identification of driver profiles using cluster analysis. Given the various test conducted by DDE/CTL it was determined that the use of a powerful battery would be needed together with a protective relay to ensure a voltage of 12v.

This is the main change to the architecture for the OSEL field trial. The IST-CPO system architecture remained the same as it was designed during the second year.

Data Processing:

The main data processing functions that will take place at the server-side include:

1) All GNSS, fuel consumption and GHG emissions data will be temporarily stored at the V2X OBU SSD and if feasible will be sent in real-time using a 3G wireless service connection to the webserver; similarly all data read by the CPO API software will be sent to the CPO server for data analysis.

2) Data stored at the V2X OBU SSD will be uploaded to the webserver periodically using an ftp connection on a daily basis (at a specific time at night); the data from the CPO delivery vehicles will be uploaded at one-minute time intervals using the MTN 3G network.

3) All GNSS data will be map-matched to the Nicosia GIS (obtained from the Ministry of Communications and Works) and will be also superimposed on Google maps. The *openGIS* software will be used; the GIS will be divided into links and nodes to represent intersections and roadway links. The links will form the basis for setting up the fuel consumption profiles for each driver.

4) Evaluate the V2V/V2I capabilities of the V2X OBU device and the wireless communication system based on the data quality, route black spots, data recording by the Webserver. The V2V communications test will be conducted on a closed course under a controlled experiment. It is noted that the CPO fleet management system does not have V2V communication capabilities.

    a. V2I Communication: Data transmitted from the V2X OBU to the Webserver

        i. Identify "dead spots" of the GNSS and/or the wireless service,

        ii. Report all communication "errors" as reported by the V2X OBU

        iii. Compare the data stored at the SSD with the data send in real-time to the web-server via 3G.

    b. V2V Communication: Data from one V2X OBU installed in one bus to the V2X OBU of other buses operating at the same time. It is emphasized that the V2V evaluation will be limited to data transmission as no V2V actions will be undertaken due to the limited budgetary constraints for this field trial. The consortium decided not to carry this test in Nicosia as it was offering no substantial value to the project. Instead the simulation case study conducted by the UTH offers more meaningful results.

5) Development of a set of driving profiles per driver using vehicle location, speed fuel consumption and GHG:

    a. Vehicle Speed Profile (Distance vs. Vehicle Speed)

    b. Vehicle/Driver Fuel Consumption (Distance vs. Fuel Consumption); aggregated at the link level

    c. Vehicle/Driver GHG Consumption (Distance vs. Fuel Consumption); aggregated at the link level

6) Identification of efficient and non-efficient driving patterns for specific roadway sections with respect to fuel consumption and GHG emissions.

6.2 Specifications of the OSEL Fleet Trial Technologies

The system architecture depicted in figure 10 will be used as the blue print for the realization of the pilot system. The components used in this architecture represent generic subsystems the details of which are described below:

6.2.1 DELPHI V2X OBU Technical Data Summary

<u>On-board features:</u>

1. Industrial grade hardware compliancy
2. Standard x86 architecture INTEL® ATOM 1GHz with ext. temp. range -40°C to ~85°C
3. On-board 1GB DDR2 RAM
4. On-board 4GB Solid State Disk
5. 1x DSRC radio
6. 1x GPS (Fastrax)
7. antenna setup:
    1. dual-antenna support for DSRC
        1. (can be reconfigured for standard 802.11a/b/g/n WLAN)
    2. Enclosure prepared for multiple antennas (e.g. GPS, DSRC, 3G/4G/LTE)
8. wide-range power supply (8 ~ 32V, 20W)
9. internal protection against wrong polarity on connector
10. customized enclosure for enhanced heat dissipation
11. automotive grade connectors
12. Operating System:
    3. customized & ruggedized DELPHI-blend Linux
    4. customized & ETSI compliant 11p stack and drivers

<u>Exposed interfaces (via automotive connector):</u>

13. 1x IEEE 802.3 ETHERNET (10/100 Mbit)
14. 1x USB2.0
15. 1x CAN (High-Speed)
16. power supply lines (PWR, GND, IGN)

6.2.2 DDE V2X OBU Installation Requirements

The DELPHI V2X OBU devices will be connected to the CANbus of the Citaro Mercedes-Benz through the FMS port.

Hardware needs:

17. Five OSEL Citaro Mercedes-Benz buses
18. Five DELPHI V2X OBU devices
19. Five hard disks of 10 TB each (one for each bus)
20. Five lead acid batteries[1] (7 Ah or 25 Ah; both were tested in order to provide some independence to the CCU from 2- to 8-10 hours)
21. Five Electronic DC Under and Over Voltage Protective Relay (to supply 12Volt to the CCU without hanging); under testing
22. One PC server to receive the wireless data from the five *DELPHI V2X OBU device*s
23. Multi-meter (voltage/amperes/resistance)
24. Oscilloscope (optional)
25. Soldering iron and solder
26. Pliers, knife, isolation tape
27. 0.8 sqr-mm copper-wire

Software installation needs:

28. DELPHI V2X OBU device customized software to read the CANbus data from the Citaro Mercedes-Benz buses and send them
29. Cellular data plan
    The initial data plan was purchased from the MTN cellular company. However it was found out that it had many black spots that made the V2X OBU device to stall operation. We therefore decided to switch from MTN to CYTA during the summer months of 2014, which has much more robust signal coverage throughout Cyprus.

Personnel needs:

30. DDE: DELPHI V2X OBU device expert
31. OSEL: five bus drivers, one manager
32. CTL: One manager, two IT experts

[1] The battery is included to provide some hours of autonomy (8-10 hours) to the CCU in case the power supply is interrupted. The protective relay is added to provide 12v to the CCU without hanging. These two items were not included in the original architecture.

33. UTH: One Communications scientist
34. UHI: Project Manager, One data mining expert
35. MCW-PWD: One official representative

6.3 Istognosis Ltd.- Costas Papaellinas Organization (IST-CPO) fleet technical summary

IST-CPO's involvement in the field trials concentrated on the analysis of driving patterns of fleet drivers with the aim to reduce fuel consumption. The technologies used by IST-CPO for this purpose are detailed below:

IST-CPO Fleet Management System

36. OBD Guard
    37. OBD-II Reader (Castel Group)
    38. Optional GPS module (uBlox)
    39. GSM/GPRS module (Telit)

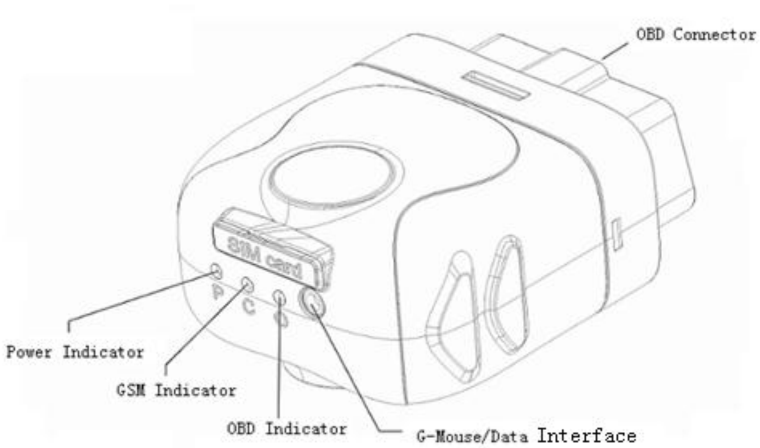


Figure 44. CPO OBD Guard

40. GSM/GPRS Specification
    41. GSM module：Telit
    42. GSM/GPRS：Quad band 850/900/1800/1900Mhz
    43. Communication protocol：Embedded TCP/IP protocol
44. GPS Specification
    45. GPS chipset：SIRF Star III u-Blox
    46. Channels：20
    47. Receiver frequency：1575.42MHz
    48. Cold start：approx 42s

49. Warm start：approx. 38s
50. Hot start：approx. 1s
51. Antenna：Built-in ceramic antenna

52. Protocol Supported for ODB-II
    53. J1850-VPW, J1850-PWM, KWP2000, ISO9141-2, CAN-BUS
54. Features
    55. OBD-II compliant.
    56. Integrated GSM/GPRS, GPS optional.
    57. Real-time vehicle diagnostic for monitoring and reporting.
    58. Integrated 300 hrs of data storage. Data is stored on remote servers.
    59. Built-in 3-axis acceleration G-Sensor module for towing alarm.
    60. Alarms can be send via SMS.
    61. User access via USB or via web platform.
    62. Alarms
        63. Speeding alarm setting
        64. Hard acceleration alarm setting
        65. Hard braking alarm setting
        66. Temperature alarm setting
        67. High RPM alarm setting
        68. Low-voltage alarm setting
        69. Extended Engine Idling alarm setting
        70. Quick change line alarm setting
        71. Sharp turn alarm setting
    72. PIDs
        73. Engine Coolant Temperature(°C)
        74. Engine RPM(rpm)
        75. Vehicle Speed Sensor(km/h)
        76. Mass Air Flow Sensor(g/s)
        77. Calculated Load Value(%)
        78. Intake Manifold Absolute Pressure(kPa)
        79. Intake Air Temperature(°C)
        80. OBD Require To Which Vehicle Designed
        81. Distance Travelled While MIL Activated(km)
        82. Fuel Rail Pressure(gauge)(10kPa)
        83. Commanded EGR(%)
        84. Fuel Level Input(%)
        85. Barometric Pressure(Kpa)
        86. Accelerator Pedal Position D(%)
        87. Accelerator Pedal Position E(%)

88. Software
    89. API to read CANbus data from the OBD port
90. Wireless Communication Data Plan
    91. The MTN 3G network will be utilised to send the data from each fleet vehicle to the server
    92. Frequency: from 2 to 10 seconds (the data plan will be decided by the end of August, 2013); currently the data frequency as set-up by IST for the CPO is one minute. The same data plan will be utilised for both OSEL and CPO.

6.3.1 IST-CPO Installation Requirements

The IST-CPO fleet management devices have been installed independently from REDUCTION by IST on the CPO delivery vehicles during July, 2013. Under REDUCTION, IST installed an API to read the raw data from the vehicle CANbus using the OBD port. IST delivered four months of data to CTL which were then map-matched and sent to UHI for analysis and determination of fuel efficient and non-efficient driving profiles.

Hardware needs:

93. Up to 68 CPO fleet vehicles equipped with the fleet management system of IST
94. Data storage server. The IST storage data server is utilised, which is the same that it will be used for the CPO fleet management system.

Software installation needs:

95. API interface to gather data from the CANbus and store on the IST server. The API is installed on each of the 68 CPO vehicles.

Personnel needs:

96. DDE: DELPHI V2X OBU device expert (if the CPO fleet is utilised instead of the OSEL buses)
97. CTL: One manager, one IT expert
98. IST: One IT expert with the CPO fleet management system
99. CPO: Up to 68 fleet drivers, one manager
100. UTH: One Communications scientist
101. UHI: Project Manager, One data mining expert
102. MCW-PWD: One official representative.

CPO Fleet Coverage area:

103. The greater Nicosia network is the primary focus of the field study

Data extracted from the OBD port:

104. GNSS data: vehicle location, speed; Frequency: every 10 sec (default); it was not feasible to gather data at a more frequent time step.

105. Fuel Consumption Frequency: every 10 sec (to be gathered by the API); it was not feasible to gather data at a more frequent time step.

6.4 Obstacles in running the Cyprus Fleet field trial

OSEL field trial obstacles:

The Nicosia OSEL bus field trial has been delayed due to the inability of the V2X OBU devices to operate under the field test's environmental conditions. Next we summarize the main issues identified during Year 3 of the project:

First OSEL Bus Diagnostic Test; 6th of March, 2013: DDE during this test with the cooperation of OSEL and CTL tried to gather data streams by sniffing data from the bus wiring system. This test was not successful.

Second OSEL Bus Diagnostic Test: A second diagnostic test was conducted in late August, 2013 based on changes conducted on the V2X OBU devices. This test was successful and the CCU device could read the data from the FMS port of the Mercedes-Citaro buses.

Subsequently DDE developed a new software and installed it on the V2X OBU device during the Fall of 2013.

The V2X OBU device was then installed on one OSEL bus and a set of tests were conducted to determine the ability of the device to read, store and send the data to a server via a 3G connection. The main changes to the architecture were: 1) Add a battery (7Ah and 26Ah) in order to provide independence to the V2X OBU during the bus operations for a few hours and examine whether the device will be able to store and retain the data, 2) Changed the 3G provider from MTN to CYTA to obtain a more stable signal and more coverage, 3) Added a protective relay controller that switched on/off the system depending on pre-specified voltage levels - Electronic DC Under and Over Voltage Protective Relay. Used the relay in sequence with the 25Ah battery to guarantee 12v supply to CCU (this configuration will be tested in early August 2014).

Main conclusions from the OSEL field trial:

1. **High temperature**
    - **a.** The bus is in operation between 06:00 -24:00 hence the V2X OBU is on all these hours. It was determined that high temperatures maybe one of the potential reasons that the CCU operation stopped operating – when the CTL team tried to gather the data at the end of the day from the bus it had to wait for at least ten minutes for the device to cool down in order to touch it . The outside temperature during the test was ranging between 30-40 Celsius.
2. **Voltage drop**
    - **a.** Lab tests prove that when the voltage drops below 11 Volts forced the CCU to stop operating and not recovering. According to the CCU specification the module should operate between 8 to 24 volts. Bus voltage fluctuations (ignition), voltage drops to below 11 volts, and it is hypothesized that this maybe one of the reasons that the V2X OBU stopped working.

3. **3G connectivity**
   a. When the CCU loses its 3G connection the module hangs and cannot restart itself unless the bus ignition switch is turned off completely.
4. **CTL and DDE continuing trials**
   a. CTL and DDE will conduct more tests during August 2014 to see if it is feasible to resolve the problems of the V2X OBU and report the results
   b. If a solution is found during August, 2014 then the two parties may decide to conduct the test during the Fall of 2014 without burdening the REDUCTION project.

IST-CPO fleet field trial obstacles:

The first data results were stored on the IST server in late December, 2014. The data processing was found to be challenging due to the limited capabilities of the server and the available bandwidth. While downloading the process will hang causing substantial delays. The data were successfully gathered by CTL by the end of February , 2014. CTL proceeded with map-matching the data using the ArcGIS software and then transformed them into a database which was then sent to UHI for analysis in April, 2014. UHI and CTL finalized the data set to be used for the analysis by the end of May, 2104. UHI proceeded with the first analysis of the data which is reported in deliverable D53.

7. Trinité Field Trial

7.1 Introduction

The Trinité field trial will be based on TrafficLink's Digital Road Authority and the streetwise App advanced prototype.

The Digital Road Authority has been designed from the vision to improve the service to the drivers if all parties (marketing, government and the travellers) combine and tune their travel(information) and intentions. Not so long ago the routing and informing of drivers was primarily a task of the government. This situation changed drastically when navigation systems and apps also advised and informed the drivers, a great advantage the introduction of these systems/apps has is that the service to an individual traveller greatly increases. An unfortunate disadvantage to the introduction of these systems/apps is the lack of synergy between all advices.

Research shows that if 15-20% of the users follows the advice that they receive from the system/app, then the network will become unstable – during a malfunction (e.g. incident), a large part of the users (travellers) will drive in a non-coordinated manner by following what they consider the most logical alternate route resulting in the failure of the system as the alternate route and the surrounding corridor may become oversaturated. In contrast, if users follow a coordinated alternate path assignment then the impact of the incident may be reduced and the system will recover more rapidly. The changing playing field where we are now requires cooperation; traffic management changes based on this into mobility management, a co creation of public and private parties with the goal to facilitate the travellers. Meanwhile the collective interests have to be guarded. The hearth of this strategy is a traffic management system that optimises the facilitation of this cooperation by a quick and flexible connection of drivers, traffic agencies and in-car systems seeing the need to align intentions and recognise that this creates a better service to passengers, the question arises: How and by whom is this entity set?

Our answer: *The Digital Road Authority*!

Digital Road Authority is a tool that merges all traffic data from various sources and processes into a smart travel advice. Connecting governments and market with each other electronically. In practice, a physical road authority (of a region) is associated with a virtual version the Digital Road Authority. Where it serves as the coordinator for all actors, and tune their real time intentions.

All parties that are connected to the Digital Road Authority can ask each other's help to more effectively deploy their own measures. Ultimately benefits are not only for these parties, but benefits the quality of traffic management as a whole because of this, the concept loops seamlessly with the objective of the initiative "better informed on their way" to do more with less.

The Digital Road Authority solution will be a part of TrafficLink.

TrafficLink is the system that gathers relevant information from the field and takes actions. The actions are executed via the available roadside equipment, DVM Exchange and the streetwise app. The streetwise app is *the* Digital Road Assistant under coordination by the Digital Road Authority.

In the Digital Road Authority trial three pilots involved:

1. IJburg – Time based demand distribution
2. Jan van Galenstraat (S105) – Truck time window optimal assignment
3. Emergency services in Amsterdam- Real time optimal routing

Pilot 1 – IJburg: The first pilot will introduce a tool for citizens in the neighbourhood of IJburg in Amsterdam. This area is a special region in Amsterdam with only two connecting bridges to leave the neighbourhood. Especially in the morning and evening rush hours many people want to leave and enter IJburg. Traffic lights at the entrances of these bridges do not have the capacity to handle this large demand. Therefore, in rush hours, large queues are formed. To improve the throughput the residents will have the possibility to get a personalised departure advice by means of an app. The recommended departure time will be based on the occupation of the roads in order to arrive at a given destination at a desired time. The main focus of this pilot is the occupancy of the roads to exit the neighbourhood. To make optimal use of the road capacity, drivers have to be distributed over time slots in an efficient and smart manner. Thereby the VRI (traffic lights) will be automatically adjusted to optimise the maximal throughput. The goal of this specific pilot is to test the user's adaptation to a personal departure time.

Pilot 2 - Jan van Galenstraat, Distribution centre: For the second pilot our focus group is freight traffic. Many distribution centres located in the Nether- lands deal with congestion of trucks during busy hours. Most of the time these distribution centres are situated in or near city centres. Trucks have to drive through narrow roads to reach their destination. Due to uncoordinated arrivals, lack of space causes trucks to line up before the entrance. This causes dangerous situations, congested city centres and environmental standards that exceed maximum values. Coordination of these trucks by means of time slot management could improve this situation tremendously.

As a main focus point, a reduction of CO_2 and NO_2 emissions for (freight) traffic has to be established. In the Netherlands traffic causes a large part of the total amount of emissions, especially in city centres. Figure 45 shows the amount of NO_2 levels over 2013. As can be seen, the areas

Amsterdam and Rotterdam have increased amounts of NO2. In the city centre of Amsterdam multiple streets exceed the maximum allowed emissions rates from the Euro standards, which is set to $40\mu/m^3$.

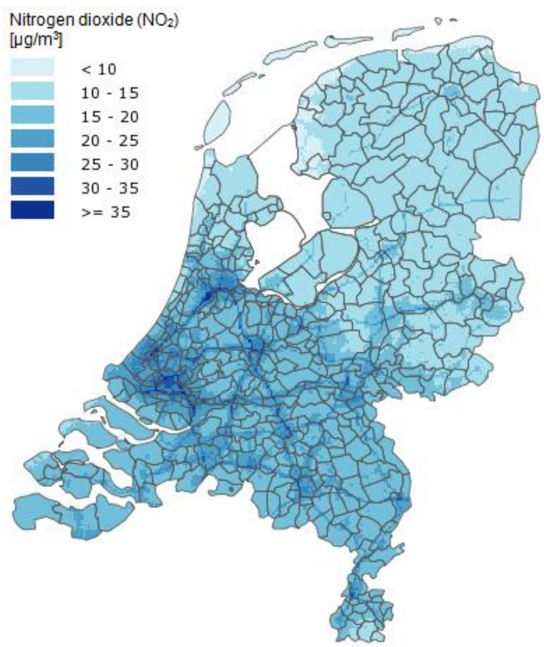


Figure 45 An illustration of the concentration of NO2 levels over 2013.

By organising the freight traffic we want to create a better spread of truck arrivals and thereby reduce CO2 and NO2 emissions. Truck drivers will be rewarded with shorter travel times when participating in this pilot. In this pilot adaptation to the current situations and optimally solve conflicts will be the main focus.

Pilot 3 - Emergency services in Amsterdam: In the third pilot we will focus on safety. The Digital Road Authority can make a contribution to the safety in an area by communicating with emergency services. The system can make contributions to reduce and optimise the travel time of an ambulance. A real-time advice of the optimal positions for ambulances to cover an area can be calculated. A second improvement would be to pass information of the current situation via in-car systems to the ambulance. Thereby one might also think of forecasting developments in a traffic situation after the occurrence of an incident. There might appear congestion due to an accident and the optimal route for the ambulance leading through this congested road. Another feature of the Digital Road Authority would be to influence traffic lights. In the future it might be possible to program traffic lights in order to create (a non-stop route) 'a green wave' between victim and ambulance. Our main focus point of this pilot is to test the ability to forecast the development of traffic after the occurrence of an incident.

7.2 The Digital Road Authority

The Digital Road Authority is a part of the TrafficLink functionality where TrafficLink is the state-of-art traffic management architecture build in the DSS-datapool. The Digital Road Authority solution is based on the AreaTrafficManager (ATM) responsible for controlling a specific part of the network. The ATM can inform app users about departure time, arrival time and incidents using the available information and architecture of the ATM.
Traffic can be found everywhere roads exist (private transport, freight transport etc). Traffic streams are nowadays undoubtedly a normal part of society, e.g. freight traffic for shop deliveries or the commuter traffic of Dutch professionals.

Where we find traffic, situations can occur, e.g. accidents. Accidents create traffic jams which decreases the flow in a certain area. If a traffic jam last longer it can have major consequences for adjacent areas.

In Figure 46 a schematic view can be seen on how a situation looks when an accident happens on one side of the Ring.

Figure 46 Accident on the Southern Ring

Figure 47 Accident on the other side

In the above situation, by using a traffic scenario, traffic can be diverted via the other side of the Ring. This makes it possible to send traffic to another direction and decrease the impact of the accident. When an accident also happens on the other side of the Ring, a scenario can be re-directed. So, 2 scenarios are currently active which both lead traffic to one location in the network where the road is (partially) blocked. This means that traffic blocks entirely and cannot proceed either way: game over.

Traffic scenarios are not flexible enough to handle these complex situations. Reason why the Area Traffic Manager (ATM) system has been designed.

ATM will use the information from Links (i.e. Links; a piece of road between an entry and an exit) and segments [100 meter section from the Link] as a representation of the road infrastructure.

In ATM NwBuildingblocks and SwitchPoints where introduced. The SwitchPoints are switches in the Network where traffic streams can be directed towards different directions (e.g. on intersections of different highways and / or entrance-points and exits). The ingoing and exit points are called AccessPoint.

Between two SwitchPoints, NwBuildingsblocks are placed on connection roads, which manage the traffic streams in this area by using the road side equipment beside the road.

Figure 48 AccessPoints and SwitchPoint

Through this structure a Base Layer of elements, SwitchPoints and NwBuildingblocks, that jointly can monitor, control and direct the traffic-streams.

Roadside equipment can be used as a traffic service to reduce the flow from an upstream to a downstream location. The streetwise app can be viewed in a manner similar to roadside equipment, which opens the door to explore many potential scenarios within the field trial."

This Base Layer needs to know which areas are important and less important in order to redistribute the traffic an optimal manner. Priorities need to be defined in the Network to keep important routes as free as possible.

To be as effective as possible Higher Layers are introduced

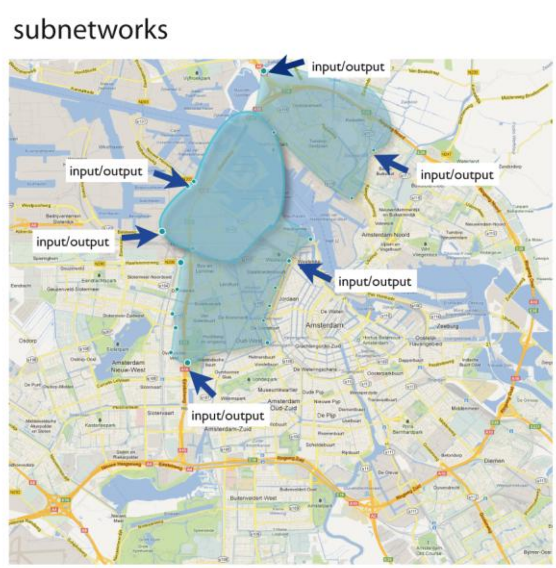


Figure 49: Number of SubNetworks

that will use the Base Layer. These Higher Layers are subdivided under area-type i.e. SubNetwork, Network, Region and Country.

Figure 49 shows several SubNetworks. These are smaller areas that manage the traffic streams for this geographical area.

Figure 50 shows how several SubNetworks form a Network. This means that each Network has several underlying SubNetworks (areas).

By adding several layers on top of each other a geographical area can be built up.

Within a Network (consisting of several SubNetworks) a specific SubNetwork can be of higher importance than another SubNetwork (e.g. because it entails an intersection of highways). This means that that specific SubNetwork has a higher priority which allows the structure to be informed of this higher priority. Based on the priority, the measures of this SubNetwork receive precedence above a SubNetwork or any other road element with a lower priority.

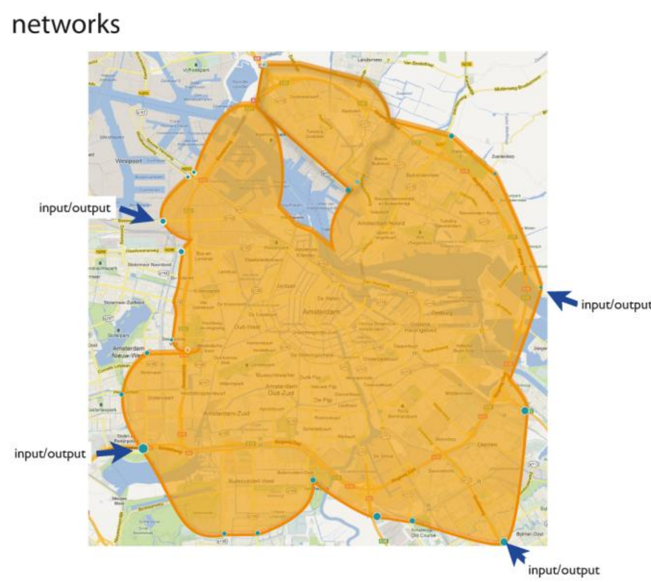


Figure 50: Network consisting SubNetworks

Within an area often more roads lead to the same point; a road user can use different routes to go from point A to B. Of all the options there is a most likely route, e.g. because the distance and / or the travel-time is shorter than any alternative routes. This route is called Preferred Route which gets a higher priority than the alternative routes. The purpose is to keep the Preferred Route free of traffic jams. This is done by informing the Base Layer of the higher priority of this route so measure on this route gets precedence.

When the Measures on the Preferred Route don't have the desired effect and traffic jams, the alternative routes will become the main route, by giving them a higher priority in order to keep the traffic flowing. When alternative routes are preferred re-routing measures can be activated.

In order to manage the Preferred and Alternative Routes we use an OD Manager (Origin-Destination) that maintains the overview of the collaboration of the routes. This OD Manager also manages the Measures for the re-routing.

Within an area multiple Origins and Destination can exist. In order to coordinate these Origins and Destinations effectively an area consist of multiple OD Managers that each manage their own Origins and Destinations through their underlying routes.

In summary, TrafficLink-ATM consists of two layers:

- Base Layer which consists of SwitchPoints and NwBuildingblocks. This Base Layer is responsible for Measures, Services and the distribution of these Measures and Services. The necessary Measures are defined by the performance of the NwBuildingblock. Performance is the indicator value for a NwBuildingblock and can be based of the available information like speed, flow CO2 measurements or travel-time
- Higher Layers which consist of Subnetwork, Network, Region and Country. These layers use the underlying Base Layer. Each Higher Layer is built up by one or multiple OD Manager, where each OD manager uses one or more routes. Within the higher layers control of Measures is done for the scope of the level. When the level off control is reached it will escalate to the next level.

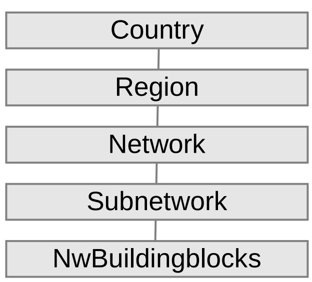


Figure 51 Layered architecture ATM

7.3 The streetwise app

Using the streetwise app as a Measure can be done from all layers in the network structure. So the Measure is a push message to the App where depending, of the effect to be achieved, different measures will be in place.

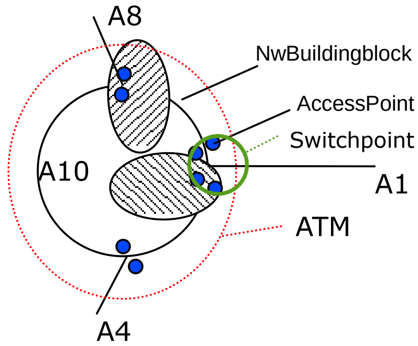


Figure 52 Schematic overview

To inform the right app users (the ones inside the area), the app must register to a specific area. This is achieved on the lowest level, the Nwbuildingblocks. The Nwbuildingblocks are configured with an exit / entrance point called AccessPoints.

The app collects a list with the configured AccessPoints from the TrafficLink environment. When the app is moving the app will calculate when it passes an AccessPoint. This will trigger an AccessPoint update (event-driven). This update will be stored in the database. On every given time an area knows (ATM) what apps are in the area.

For incident detection a mechanism will be included to map the incident location to a DSS-Object representing the incident. Incidents are received with three parameters (Id, Lon, and Lat). To use the incident messages in DSS we need to know in what area (ATM) they are occurring. This mechanism will try to map the incident to the lowest entity from the network structure, the NwBuildingblocks. If there is no match for the NwBuildingblocks the mechanism will try to find the Subnetwork, or Network and so on.

For calculating real time travel-time the following approach is used:

A route travels through multiple ATM's, the ATM is responsible for calculating the exact travel-time. While driving the app crosses the AcessPoints. The moment the app crosses the AccessPoint the responsibility to inform the app shifts to the next ATM. For the total route the default route will be used. In the database the calculated route will be merged with existing objects from the environment.

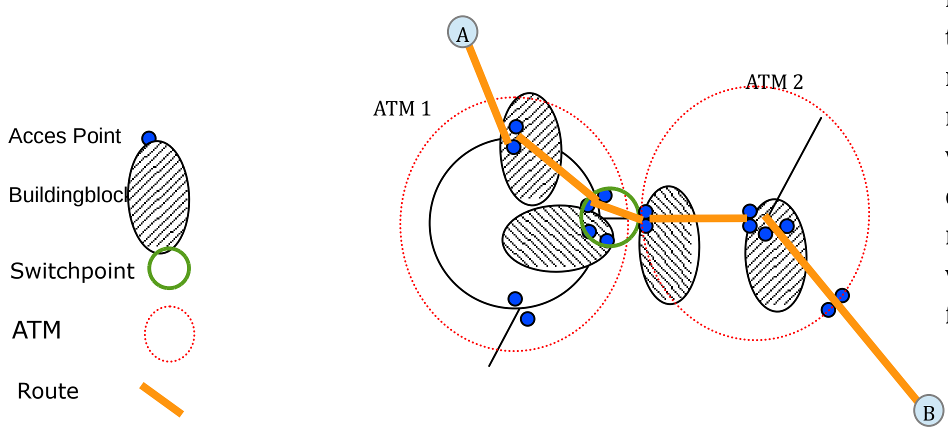


Figure 53 Path through network

Process for the IJburg pilot.

1. User enters a new appointment in the phone
2. Total route is calculated inclusive travel-time
3. Calculated route is mapped with the path through the network (NwBuildingblocks) for real time data, mapping stored in DB.
4. With real time information travel time is tuned.
5. Passing AccessPoint, travel-time for remaining route is recalculated,

Evaluate Departure time

The ATM will periodically evaluate witch app users have to leave. If a user needs to leave the ATM will send a push message to the specific user with the information to leave. See for more details D5.2

Process Incident for the pilot Amsterdam ambulances

This function will inform the current app users located inside a specific area about incidents. Informing the app users is done by the push messages to the users.

Location To Object

The incident signal from the App has three parameters (DeviceId, lon, lat).

In order to process the distress signal in the appropriate ATM, the location must be mapped to a DSS object. This method will try to match a DSS object from bottom-up. Starting if available with the Segments, Links, NwBuildingblock, Subnetwork and Network. To achieve this the following solution will be used:

I. Segment, Links and Nwbuildingblocks

These objects have X,Y information stored in the database. Using the first and last point a square is defined.

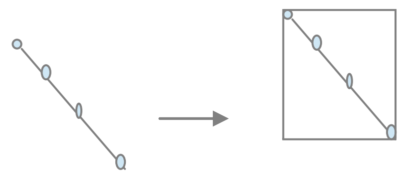


Figure 54 Mapping objects

If the given lon/lat are located "inside" the square the object is found, the location is "mapped".

II. SubNetwork and network elements

These elements have SwitchPoints / AccessPoints to define where the in- / outputs are. With this information the same procedure as described for the segments and links can be used to identify the area.

If no match is found the top layer element will handle the distress signal.

This will probably be the "Country" object.

For an efficient way of processing the point information, all information will be stored in a database. Databases are optimised for searching inside a large data pool. To make sure that the process is optimised for looking big datasets (all segments of The Netherlands) a new table will be used with only the basic information:

locationToObject_TBL

- ObjectId
- Lat(a)
- Lon(a)
- Lat(b)
- Lon(b)

7.3.1 Communication

Figure 55 App communication architecture

Connection between the APP and the backend is done thru http. The bridge has functions to support the Measures in the ATM.

For delivering the information to ATM HTTPBridge is also in place. With this bridge real time (event based) messages can be delivered to ATM. All information gathered is stored in the DB.

An example of the messages defined for communication with the app:

- accessPointUpdate: informs that the app is entering/exiting an area
- sendDistressSignal: informs about a distress signal
- cancelDistressSignal: informs about s distress signal that is cancelled.

7.4 Results

The ATM architecture has been further improved. There is now an advanced product ready to be used.

After intensive testing using the acceptance test protocol of Trinité the ATM was ready for the next step. In order to keep the time schedule some less important developments were postponed to a second release. These parts concern functionality in the higher layer objects. The pilots do not need this functionality.

Installation is done via the service department and goes along with an installation plan approved by the customer.

The base layer for the Digital Road Authority was installed in the Amsterdam Traffic Control-room with connections to the real data. The Measures to be set on a (graphical) VMS and the Traffic-control system were available in simulation form.

This will have the benefit of configuring test cases with use of real traffic situations based on the data. The advantage is that experiments with control can be done without influencing the real traffic.

The output can be seen on the simulated measures and reproduced on the live measures to see the effect.

- The base layer was prepared with the configuration for the Amsterdam environment with as top layer the city of Amsterdam with Subnetworks and NwBuildingblocks.
- A manual was delivered to the traffic manager.
- An instruction was given to the traffic manager to qualify him for testing and evaluating the system.
- There are already some additional projects starting to add information from parking places and guide drivers using this information to available parking places. For this the latest version DVM exchange must be installed with additional information from the parking places. This is not in the scope of Reduction.
- In October the second release will be installed too.
- Currently will be worked on the second release items and improvements to the first release. The improvements are meanly focussed on the user interface.
- The app will be developed with all knowledge from the previous work packages supporting the Pilot requirements.
- Next step will be connecting the Streetwise APP to the Digital Road Authority.

8. Risk Assessments

The field trials involving FlexDanmark, Nicosia and TrainOSE do not forecast deviations from the plan. The respective software stacks are being implemented, which means that the functionality is present in accordance with the description of work.

The Trinité field trial also doesn't forecast any deviations from the plan. Trinité detected draw backs on the initial architecture like excessive power consumption, user-unfriendly way of operating the App and the large amount of data to be handled and stored and made the necessary modifications to the architecture to overcome the problems. Some less important functionalities of the architecture are postponed to a second release.

The interface description has been added to connect different functionalities between the modules of the work packages. Differences in interpretation on the interface description between the different partners could constitute a certain risk for future implementation and thus needs to be monitored carefully.

The OSEL bus field trial could not be completed until August 2014 due to the V2X OBU malfunctions. DDE is working to fix the software and hardware issues and the trial will continue beyond the official deadline of August 31st, 2014. DDE has provided a new software version to CTL on August 26, 2014. This new software will be flashed onto the V2X OBU and the trial will continue. DDE is confident that the hardware/software issues will be resolved by mid-September, 2014. CTL agreed to carry out the test during the months of September through December, 2014 – subject to the condition that the units will be operational.

The IST-CPO Nicosia field trial is completed for the first part – the analysis of the driving patterns. CTL in cooperation with UHI will advise IST and CPO on which driving patterns are inefficient and will carry out the after study during the months of September and October, 2014.

Risk no.	*Description*	*Impact*	*Strategy*
1)	Problem with power consumption Trinité field tail	Low	Modification in architecture. Adapted in current version.
2)	Time schedule Tinité	Medium	Focus on the necessary functionalities of the architecture.
3)	Interface connection between the modules	Medium	Monitoring. Tuning of interaction in case

			interpretation differences.
4)	Nicosia field trial	Medium	Complete the field trial during the fall of 2014.

Table 10 Risk overview

9. Conclusion

Conclusions for the FlexDanmark field trial:

The field trial of FlexDanmark expands its architecture and geographical areas, based on the results of the first field study. Multi-modality is covered in detail and a larger set of trips, covering larger areas will be targeted. The experiments have shown that it is complex to work with CANbus data and that implementation of equipment to monitor the vehicles is not simple. The first field trials have also shown that we can gain valuable data from observing vehicles in operation. The experience of this round will be used to expand the number of vehicles and scope of the study in the next trial. The FlexDanmark field trial provides travellers serviced by FlexDanmark to select a route based on travel time, fuel consumption or emissions for taxis only or for multi-modal trips (taxi plus train, taxi plus bus/minibus). The architecture covers multimodal transport and eco-routing.

Conclusions for the TrainOSE field trial:

TrainOSE for the last phase of the second trial fulfilled the following goals and the architecture covers multimodal eco-routing.

1. Finished multimodal web application. However, Data enrichment process is ongoing.
2. Made a survey over our customers in order to get their perceptions over multimodal trip using trains as well
3. Collect measurements for energy consumption of the electrified locomotives for the route part Domokos-Thessaloniki
4. Made an initial statistical analysis over these measurements in order to see if there is ground for improvement
5. Cooperated with University of Hildesheim for the production of good driving patterns.
6. Prepared a good behaviour driving manual.
7. Put on test the driving manual for five trials and proved that the driving manual when applied gives better results for energy consumption
8. Presented the driving manual to the locomotive drivers' union
9. Move to the next phase, which is to make drivers' manual part of locomotive drivers' formal education

Conclusions for the Nicosia field trial:

The Nicosia field trial were comprised of two parts: The first part two different fleets were expected to be tested, the OSEL Nicosia bus fleet using five buses equipped with the DDE V2X OBU devices and the Costas Papaellinas Organization delivery fleet (up to 68 vehicles – about 6 months of delivery route data) equipped with their own fleet management system. During the first part of the trial, the driving profile of each of the drivers was estimated using CANbus data (GNSS location, speed and acceleration; fuel consumption). The data was analysed using data mining techniques to develop a set of guidelines for drivers in order to reduce fuel consumption. When these guidelines were developed after a satisfactory data sample was collected and analysed (about eight weeks) then the Second Phase of the field trial wolud start. The drivers were instructed to follow the set-up guidelines and they were monitored for another 8-week time period. The results will be summarised in a report. In addition, a feedback questionnaire will be distributed to the OSEL management, the CPO management, the OSEL and CPO drivers, and the Ministry of Communications - Public Works Department of Cyprus and summarised to capture the impact of the REDUCTION technologies as implemented in Cyprus for eco-driving.

The OSEL bus field trial could not be completed until August 2014 due to the V2X OBU malfunctions. DDE is working to fix the software and hardware issues and the trial will continue beyond the official deadline of August 31st, 2014. DDE has provided a new software version to CTL on August 26, 2014. This new software will be flashed onto the V2X OBU and the trial will continue. DDE is confident that the hardware/software issues will be resolved by mid-September, 2014. CTL agreed to carry out the test during the months of September through December, 2014 – subject to the condition that the units will be operational.

The IST-CPO Nicosia field trial was completed for the first part – the analysis of the driving patterns. CTL in cooperation with UHI advised IST and CPO on which driving patterns are inefficient and CTL will carry out the after study during the months of September and October, 2014.

The architecture covers vehicular communication and P2P communication.

Conclusions for the Trinité field trial:

The first field trial of Trinité showed some drawbacks on the REDUCTION App. The constant communication caused a significant amount of power consumption, selecting routes was not user friendly and a lot of data needs to be stored and computed. By adding the Area Traffic Manager to the architecture and combined with that the Digital Road Authority, vehicles can sign in and sign out when they are entering or leaving the area. This architectural change solves most of the drawbacks from the first field trial and is developed to be used in the second field trial. Also the interface description between the multi modal eco routing, eco driving, prediction and V2V communication system is being defined. This interface description can be implemented to integrate the information of CO_2 emission, fuel consumption, location, speed and travel time into the DSS-Datapool architecture and Traffic Management System of Trinité Automation, state-of-art TrafficLink. TrafficLink can be used to take active measures on the traffic flow of highways, provincial and urban roads, to reduce the CO_2 emission in the designated areas.

The ATM is installed in the Amsterdam Traffic Control-room. A manual was delivered to the traffic manager and an instruction was given.

For the field trials Trinité established contact with the city of Amsterdam in order to use their extensive CO_2 sniffer network. Amsterdam has been measuring CO_2 and particulate values over a long period of time on our selected route (Jan van Galenstraat). The access to both actual and historic data is of great use to measure the effect of the field trial. This will be used to finetune the market ready product.
The architecture achieved to cover eco-driving and eco-routing individual.
Meetings will be established with some major transport enterprises in order to find future users for the product. Based on the REDUCTION results.

Usage of the standards DatexII and DVMExchange offers the possibility to communicate on regional level, where network management systems inter-operate to address common traffic challenges. These systems are commonly built, owned and managed by different parties.

10. Glossary

Term	**Description**
AAU	Aalborg University
AP	Access Point
ATM	Area Traffic Manager
AU	Aarhus University
AVC	Autonomous Vehicular Cloud
AWS	Amazon Web Services
CANbus	Controller Area Network bus
CBN	Content Based Networking
CO2	Carbon dioxide
CTL	Cyprus Transport Logistics
DDE	Delphi Delco Electronics GMBH
DSRC	Dedicated Short Range Communication
DSS	Dynamic Subscription Software
EBS	Elastic Book Store
EC2	Elastic Compute Cloud
ETSI	European Telecommunications Standards Institute
FD	FlexDanmark
GHG	Greenhouse gas
GIS	Geographic Information System
GNSS	Global Navigational Satellite System
GPS	Global Positioning System
GVCC	Generalized Vehicular Cloud Customization
HOV	High Occupancy Vehicle
HDFS	Hadoop Distributed File System
IaaS	Infrastructure as a Service
IETF	Internet Engineering Task Force
JVM	Java Virtual Machine
OS	Operating System
PaaS	Platform as a Service
P2P	Peer to peer
RSU	Roadside Unit
QoI	Quality of Information
TRI	Trinité Automation
SaaS	Software as a Service
SFTP	Secure File Transfer Protocol

SSD	Solid State Drive
SSH	Secure Shell
SVCC	Specified Vehicular Cloud Customization
S3	Simple Storage Service
UHI	University of Hildesheim
UTH	University of Thessaly
VANETS	Vehicular Ad-hoc Networks
V2I	Vehicle to infrastructure
V2R	Vehicle to Roadside
V2V	Vehicle to vehicle
VC	Vehicular Cloud
WDS	Wireless Distribution System
VMS	Variable Message Sign
VM	Virtual Machine
WAVE	Wireless Access in Vehicular Environments
WP	Work package

References

[1] O. Andersen, B. B. Krogh og K. Torp, »An Open-source Based ITS Platform,« *First International Workshop on Next-Generation Location-Based Services (LBS n.0),* p. 6, 2013 (to appear).

[2] The PostgreSQL Global Development Group, [Online]. Available: www.postgresql.org. [Senest hentet eller vist den 01 03 2013].

[3] Refractions Research, [Online]. Available: http://postgis.refractions.net/. [Senest hentet eller vist den 01 03 2013].

[4] Python Software Foundation, [Online]. Available: www.python.org. [Senest hentet eller vist den 01 03 2013].

[5] ubuntu, [Online]. Available: ubuntu.com. [Senest hentet eller vist den 01 03 2013].

[6] F. Pereira, H. Costa og N. Pereira, »An off-line map-matching algorithm for incomplete map databases,« *European Transport Research Review,* nr. 3, pp. 107-124, 2009.

[7] National Marine Electronics Association, »NMEA 0183 Standard,« [Online]. Available: www.nmea.org. [Senest hentet eller vist den 01 03 2013].

[8] D. DePriest, »NMEA data,« [Online]. Available: www.gpsinformation.org/dale/nmea.htm. [Senest hentet eller vist den 01 03 2013].

[9] O. Andersen og K. Torp, »An Open-Source ITS Platform,« Aalborg University DBTR-32, 2012.

[10] D. Varrazzo. [Online]. Available: initd.org/psycopg. [Senest hentet eller vist den 01 03 2013].

[11] C. Thomsen, »ETL programming in Python,« [Online]. Available: www.pygrametl.org. [Senest hentet eller vist den 01 03 2013].

[12] NetworkX Developers, »NetworkX v1.4 documentation,« [Online]. Available: networkx.github.com. [Senest hentet eller vist den 01 03 2013].

[13] Souptik Datta, Kanishka Bhaduri, Chris Giannella, Ran Wolff, and Hillol Kargupta. Distributed Data Mining in Peer-to-Peer Networks. IEEE Internet Computing, 10(4):18--26, 2006

[14] Xie, Chao, Yi Pan. 2006. Analysis of large-scale hybrid peer-to-peer network topology. GLOBECOM. IEEE

[15] Khilar, Pabitra Mohan, Sudipta Mahapatra. 2007. Heartbeat based fault diagnosis for mobile ad-hoc network. Proceedings of the third conference on IASTED International Conference: Advances in Computer Science and Technology. ACST'07, ACTA Press, Anaheim, CA, USA, 194{199.

[16]Bhaduri, Kanishka, Ran Wolf , Chris Giannella, Hillol Kargupta. 2008. Distributed decision-tree induction in peer-to-peer systems. Stat. Anal. Data Min. 1 85-103.

[17] Barabsi, Albert-Lszl, Rka Albert. 1999. Emergence of scaling in random networks. Science 286 509-512.

[18] Alberto Medina, Ibrahim Matta John Byers, Anukool Lakhina. 2002. Brite: Boston university representative internet topology generator.

[19] Lars Schmidt-Thieme. Reducing environmental footprint based on multi-modal fleet management system for eco-routing and driver behavior adaptation. 2012.

[20] Josif Grabocka, Umer Khan, and Lars Schmidt-Thieme. Deliverable d2.2. 2012.

[21] Chenjuan Guo(AU), Bin Yang(AU), Christian S.Jensen(AU), and Kristian Torp (AAU). Deliverable d3.2. 2012.

[22] K. Ahn, A.A.Trani, H. Rakha, and M. Van Aerde. Microscopic fuel consumption and emission models. 78th Annual Meeting of the Transportation Research Board, 1999.

[23] M. T. Oliver-Hoyo and G. Pinto. Using the relationship between vehicle fuel consumption and co2 emissions to illustrate chemical principles. Journal of Chemical Education, 2008.

[1.1] J. Chen, X. Cao, Y. Zhang, W. Xu, Y. Sun, "Measuring the performance of movement - assisted certificate revocation list distribution in VANET", Wireless Communications and Mobile Computing, 11 (7), 888-898, 2011.

[1.2] W.J. Fleming, "New automotive sensors—A review," in *IEEE Sensors Journal*, vol. 8, no. 11, 2008, pp. 1900-1921.

[1.3] US FEDERAL COMMUNICATIONS COMMISSION (FCC) (2003, September) "Standard Specification for Telecommunications and Information Exchange Between Roadside and Vehicle Systems—5 GHz Band Dedicated Short Range Communications (DSRC) Medium Access Control (MAC) and Physical Layer (PHY) Specifications" (Washington, DC).

[1.4] R. A. Uzcategui, and G. Acosta-Marum, "WAVE: A Tutorial", IEEE Communications Magazine, vol. 47, no. 5, pp. 126-133, May 2009.

[1.5] R. Yu, Y. Zhang, Y. Liu, S. Xie, L. Song and M. Guizani, "Secondary Users Cooperation in Cognitive Radio Networks: Balancing Sensing Accuracy and Efficiency", IEEE Wireless Communications Magazine, vol. 19, no. 2, April 2012, pp.2-9.

[1.6] S. Xie, Y. Liu, Y. Zhang, and Yu, "A Parallel Cooperative Spectrum Sensing in Cognitive Radio Networks", IEEE Transactions on Vehicular Technology, vol. 59, no. 8, pp.4079 C 4092, 2010.

[1.7] W. Tantisiriroj, S. Patil, G. Gibson. "Data-intensive file systems for Internet services: A rose by any other name ..." Technical Report CMU- PDL-08-114, Parallel Data Laboratory, Carnegie Mellon University, Pittsburgh, PA, October 2008.

[1.8] S. Ghemawat, H. Gobioff, S. Leung. "The Google file system," in Proc. of ACM Symposium on Operating Systems Principles, Lake George, NY, Oct 2003, pp. 29–43.

[1.9] Brown A., Johnston S, Kelly K. "Using service-oriented architecture and component-based development to build web service applications", Rational Software Corporation, 2002.

[1.10] Olariu S, Hristov T, Yan G. "The next paradigm shift: from vehicular networks to vehicular clouds," in Mobile ad hoc networking: cutting edge directions, Basagni MC S, Giordano

[1.11] S, Stojmenovic I, 2nd ed. NJ, USA, John Wiley & Sons, Inc., Hoboken, 2013.

[1.12] T. Wang, L. Song, Z. Han, and B. Jiao "Popular Content Distribution in CR-VANETs with Joint Spectrum Sensing and Channel Access," to appear, IEEE Journal on Selected Areas in Communications.

[1.13] Nurmi D, Wolski R, Grzegorczyk C, Obertelli G, Soman S, Youseff L, et al., "The eucalyptus open-source cloud-computing system," in Proceedings of the 9th IEEE/ACM international symposium on cluster computing and the grid, Shang-hai, 2009, pp. 124–31.

[1.14] A. S. Foundation, "Apache hadoop." [Online]. Available: {http: //hadoop.apache.org/}.

[1.15] M. K. McKusick, S. Quinlan. "GFS: Evolution on Fast-forward," ACM Queue, vol. 7, no. 7, New York, NY. August 2009.

[1.16] Olariu, S., Khalil, I. and Abuelela. "Taking VANET to the Clouds". Pervasive Comput. Commun. 7, pp. 7–21.
Lustre File System. http://www.lustre.org.

[1.18] J. Dean and S. Ghemawat, "Mapreduce: simplified data processing on large clusters," in Proceedings of the 6th conference on Symposium on Operating Systems Design & Implementation, 2004, pp. 10–10. [Online].Available:http://portal.acm.org/citation.cfm?id=1251254.1251264.

[1.19] M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies. "The Case for VM-based Cloudlets in Mobile Computing", IEEE Pervasive Computing, vol. 8, no. 4, 2009.

[1.20] S. Olariu, M. Eltoweissy, and M. Younis, "Toward autonomous vehicular clouds," ICST Trans. Mobile Commun. Comput., vol. 11, no. 7–9, pp. 1–11, Jul.–Sep. 2011.

[1.21] G. Yan, S. Olariu, and M. C.Weigle, "Providing VANET security through active position detection," Comput. Commun., vol. 31, no. 12, pp. 2883–2897, Jul. 2008, Special Issue on Mobility Protocols for ITS/VANET.

[1.22] J. Sun, C. Zhang, Y. Zhang, and Y. M. Fang, "An identity-based security system for user privacy in vehicular ad hoc networks,"IEEE Trans. Parallel Distrib. Syst., vol. 21, no. 9, pp. 1227–1239, Sep. 2010.

[1.23] A. Friedman and D. West, “Privacy and security in cloud computing,” Center for Technology Innovation: Issues in Technology Innovation, no. 3, pp. 1–11, Oct. 2010.

[1.24] N. Santos, K. P. Gummadi, and R. Rodrigues, “Toward trusted cloud computing,” in Proc. HotCloud, Jun. 2009.

[1.25] T. Garfinkel, B. Pfaff, J. Chow, M. Rosenblum, and D. B. Terra, “Virtual machine-based platform for trusted computing,” in Proc. ACM SOSP, 2003, pp. 193–206.

[1.26] S. Berger, R. Cáceres, K. A. Goldman, R. Perez, R. Sailer, and L. van Doorn, “VTPM: Virtualizing the trusted platform module,” in Proc. 15th Conf. USENIX Sec. Symp., Berkeley, CA, 2006, pp. 305–320.

[1.27] D. G. Murray, G. Milos, and S. Hand, “Improving XEN security through disaggregation,” in Proc. 4th ACM SIGPLAN/SIGOPS Int. Conf. VEE, New York, 2008, pp. 151–160.

[1.28] M. Jensen, J. Schwenk, N. Gruschka, and L. L. Iacono, “On technical security issues in cloud computing,” in Proc. IEEE Int. Conf. Cloud Comput., 2009, pp. 109–116.
Fed. Fin. Inst. Examination Council, Authentication in an Internet banking environment 2009. [Online]. Available: http://www.ffiec.gov/pdf/authentication_guidance.pdf

[1.29] J. Douceur, “The sybil attack,” in Proc. Rev. Papers 1st Int. Workshop Peer-to-Peer Syst., 2002, vol. 2429, pp. 251–260.

[1.30] Levente Butty´an, Tam´as Holczer, and Istv´an Vajda, “On the Effectiveness of Changing Pseudonyms to Provide Location Privacy in VANETs” F. Stajano et al. (Eds.): ESAS 2007, LNCS 4572, pp. 129–141, 2007. c_Springer-Verlag Berlin Heidelberg 2007.

[1.31] Mousannif H, Khalil I, Al Moatassime H. Cooperation as a Service in VANETs. Journal of Universal Computer Science 2011; 17:1202-18.

[1.32] T. Ristenpart, E. Tromer, H. Shacham, and S. Savage, “Hey, you, get off of my cloud: Exploring information leakage in third-party compute clouds,” in Proc. 16th ACM Conf. CCS, 2009, pp. 199–212.

[1.33] Samara, G., Al-Salihy, W.A.H., Sures, R. “Security issues and challenges of Vehicular Ad Hoc Networks”, Proceedings of The 4th International Conference on New Trends in Information Science and Service Science (NISS), 2010, pp.393-398.
[1.34] Grilli,G., “Data Dissemination in Vehicular Networks”, PhD dissertation in Computer Science and Automation Engineering, June 2010. CMU wireless emulator, “http://www.cs.cmu.edu/emulator/”.

[1.35] Xen, http://www.xen.org/.

[1.36] “TPC-C: an on-line transaction processing benchmark”, http://www.tpc.org/tpcc/.

[1.37] “TPC-W: a transactional web e-Commerce benchmark”, http://www.tpc.org/tpcw/.

[1.38] "TPC Benchmark H - Standard Specification", http://www.tpc.org/tpch/.

[1.39] Standard Performance Evaluation, "SPEC OpenMP Benchmark Suite", http://www.spec.org/omp/.

[1.40] "SPECweb2009", http://www.spec.org/web2009/.

[1.41] K. Kim, K. Jeon, H. Han, S.-g. Kim, H. Jung, and H. Y. Yeom, "MRBench: A Benchmark for MapReduce Framework", in 14th IEEE International Conference on Parallel and Distributed Systems (ICPADS '08), 2008.

[1.42] S. Huang, J. Huang, J. Dai, T. Xie, and B. Huang, "The HiBench Benchmark Suite: Characterization of the MapReduce-Based Data Analysis", in 22nd International Conference on Data Engineering Workshops (ICDE 2010), Los Alamitos, CA, 2010.

Appendix

A. SURVEY FOR MULTIMODAL TRAVELING

B. LOCOMOTIVE DRIVERS MANUAL

C. XSD DatexII Interface description

A. SURVEY FOR MULTIMODAL TRAVELING

Reduction Survey for ECO-Multimodal Transportation

1. **Ποια είναι η ηλικία σας** *
   - (•) 15-25
   - ( ) 25-45
   - ( ) 45+
2. **Χρησιμοποιείτε τακτικά, μέσα μαζικής μεταφοράς?** *
   - (•) Ναι
   - ( ) Όχι
3. **Ποια μέσα μεταφοράς χρησιμοποιείτε τακτικά?** *
   - [x] Τραίνο
   - [ ] ΚΤΕΛ
   - [ ] Αστικό λεωφορείο
   - [ ] Μετρό
   - [ ] Κανένα
4. **Έχετε αυτοκίνητο?** *
   - (•) Ναί
   - ( ) Όχι
5. **Είστε πρόθυμοι να χρησιμοποιήσετε πολλαπλά μεταφορικά μέσα για τις μετακινήσεις σας προκειμένου το ταξίδι σας να είναι περισσότερο οικολογικό?** *
   - (•) Ναί
   - ( ) Όχι

6. **Ποιος ο μεγαλύτερος χρόνος καθυστέρησης που είστε πρόθυμοι να ανεχθείτε για την μετεπιβίβασή σας από ένα μεταφορικό μέσο σε ένα άλλο, έτσι ώστε η μετακίνησή σας να είναι περισσότερο οικολογική με συνδυασμό πολλών μεταφορικών μέσων?** *
   - ( ) 0-15 λεπτά
   - ( ) 15-30 λεπτά
   - ( ) 30-60 λεπτά
   - ( ) 60-120 λεπτά
   - ( ) >120 λεπτά

7. **Ποιο είναι το ποσό των χρημάτων για το οποίο είστε πρόθυμοι να διαθέσετε επιπλέον (η και λιγότερο) σε σχέση με τον παραδοσιακό τρόπο μετακίνησης έτσι ώστε η μετακίνησή σας να είναι περισσότερο οικολογική με συνδυασμό πολλών μεταφορικών μέσων? ***

- 0 - 10% περισσότερο
- 10 - 30% περισσότερο
- >30 % περισσότερο
- καμία επιβάρυνση
- 0-10% λιγότερο
- 10-30% λιγότερο
- >30% λιγότερο

8. **Ποιός είναι ο πρόσθετος χρόνος ταξιδιού, σε σχέση με τον αρχικό χρόνο, που είστε πρόθυμοι να ανεχτείτε ετσι ώστε το ταξίδι σας να είναι περισσότερο οικολογικό με συνδυασμό πολλών μεταφορικών μέσων? ***

- 0%-10% περισσότερο
- 10%-30% περισσότερο
- >30% περισσότερο
- Καθόλου πρόσθετος χρόνος

9. **Για ποιο τύπο μετακίνησης είστε πρόθυμοι να ταξιδέψετε με συνδυασμό πολλών μεταφορικών μέσων έτσι ώστε το ταξίδι σας να είναι περισσότερο οικολογικό? ***

- Εργασία
- Κοινωνικές εκδηλώσεις
- Εκπαίδευση
- Αγορές
- Τουρισμός
- Προσωπικοί λόγοι
- Εργασιακά ταξίδια
- Άλλο

10. **Για ποιο χιλιομετρικό μέγεθος μετακίνησης είστε πρόθυμοι να ταξιδέψετε με συνδυασμό πολλών μεταφορικών μέσων, έτσι ώστε το ταξίδι σας να είναι περισσότερο οικολογικό? ***

- Μικρά (έως 10 χιλιόμετρα)
- Μεσαία (μεταξύ 10 και 60 χιλιομέτρων)
- Μεγάλα (πάνω από 60 χιλιόμετρα)

D4.3.2 [Report on second pilot (to be used for field study 2)**]**

B. LOCOMOTIVE DRIVERS MANUAL

Οδηγός Οικονομικής Οδήγησ

Eco Driving Guide

Πίνακες

Εικόνες

1. Εισαγωγή / Introduction

Η έννοια της οικονομικής οδήγησης παρουσιάστηκε έντονη τα τελευταία χρόνια με ως αποτέλεσμα της οικονομικής κρίσης και άρα μείωσης κόστους λειτουργίας σε επίπεδο χώρας, αλλά και της ανάγκης μείωσης των εκπομπών CO2 σε πανευρωπαϊκό επίπεδο. Η ανάγκη μείωσης του κόστους λειτουργίας είναι προφανής για κάθε μη κερδοσκοπικό οργανισμό ή εταιρεία. Ωστόσο η ανάγκη μείωσης των εκπομπών CO2 δεν είναι προφανής και σε άμεση σχέση με την μείωση του λειτουργικού κόστους. Συνδέεται κυρίως με γενικότερους περιορισμούς περιβαλλοντικής ρύπανσής και συμμόρφωσης με ευρωπαϊκούς κανονισμούς. Η μείωση των εκπομπών CO2 για μια μεταφορική εταιρεία και όχι μια εταιρεία παραγωγής ενέργειας μπορεί να συνδεθεί άμεσα με την κατανάλωση καυσίμων (πετρελαιοειδή κυρίως) και την κατανάλωση ηλεκτρικής ενέργειας .

Για την εταιρεία ΤΡΑΙΝΟΣΕ η μείωση των εκπομπών CO2 συνδέεται άμεσα

1. Με το ποσό καυσίμων που χρησιμοποιείται για την κίνηση των πετρελαιοκίνητων αμαξοστοιχιών στις κεντρικές γραμμές του σιδηροδρομικού δικτύου
2. Με το ποσό της ηλεκτρικής ενέργειας που χρησιμοποιούνται στις γραμμές του προαστιακού σιδηρόδρομου καθώς και σε κάποια μέρη τη κυρίου δικτύου.

Με το παρόν εγχειρίδιο επιχειρούμε να προσφέρουμε ένα οδηγό καλής πρακτικής αναφορικά με τον τρόπο οδήγησης σε όσους χειρίζονται ελκυστήρα αμαξοστοιχίας ηλεκτροκίνητο ή ντιζελοκίνητο.

2. Παράμετροι ενεργειακής Κατανάλωσης / Factors affecting energy consumption

Η κατανάλωση καυσίμων / ηλεκτρικής ενέργειας υπόκειται σε μια σειρά από παράγοντες που μπορούν την επηρεάσουν δραματικά. Εάν καταφέρουμε να προσδιορίσουμε επακριβώς αυτούς του παράγοντες τότε μπορούμε να προχωρήσουμε και στις αντίστοιχες μετρικές, στατιστικές αναλύσεις και τέλος στην ανάλυση ευαισθησίας τους. Το τελικό αποτέλεσμα θα είναι ένα πίνακας παραγόντων κατανάλωσης καυσίμων και ηλεκτρικής ενέργειας.

Παράγοντες που επηρεάζουν την κατανάλωση καυσίμων/ενέργειας είναι :

1. Γεωγραφία επι της οποίας κινείται η αμαξοστοιχία.
2. Τύπος ελκυστήρας αμαξοστοιχίας – Υγρών καυσίμων ή Ηλεκτρικού Ρεύματος
3. Καιρικές συνθήκες (κατεύθυνση και ένταση ανέμου)
4. Αεροδυναμική των αμαξοστοιχιών και των συρόμενων βαγονιών
5. Οδηγική συμπεριφορά
6. Χρονοδιάγραμμα Δρομολογίων

Στο παρόν έγγραφο θα ασχοληθούμε κυρίως με τους παράμερους οι οποίοι επηρεάζουν την κατανάλωση και οι οποίοι μπορούν αν μεταβληθούν χωρίς οικονομικό κόστος.

2.1 Γεωγραφικοί Παράγοντες / Geography

Η γεωγραφία της σιδηροδρομικής γραμμής επηρεάζει δραματικά την κατανάλωση καυσίμων / ενέργειας. Η παρουσία μεγάλων κλίσεων (ανωφέρειες, κατωφέρειες) αυξάνουν την κατανάλωση λόγω του έξτρα φόρτου που απαιτείται για την έλξη στις ανωφέρειες και επιπλέον της μη μηδενικής κατανάλωσης στις κατωφέρειες, δεδομένου ότι η δηζελάμαξα θα πρέπει πάντα να διαθέτει ισχύ. Ωστόσο για τις ηλεκτράμαξα υπάρχει η δυνατότητα για παραγωγή ενέργειας μέσω της διαδικασίας πέδησης. Θα πρέπει να επισημάνουμε εδώ ότι η γεωγραφία της σιδηροδρομικής δεν είναι μια παράμετρος η οποία μπορεί να μεταβληθεί χωρίς κόστος.

2.2 Τύποι Ελκυστήρων / Locomotive Types

Ο τύπος ελκυστήρα (δηζελοκίνητος ή ηλεκτροκίνητος) επηρεάζει επίσης και την σχετική κατανάλωση ενέργειας (είτε θερμική είτε ηλεκτρική). Η εταιρεία ΤΡΑΙΝΟΣΕ διαθέτει για το κεντρικό δίκτυο της, δηζελοκίνητους ελκυστήρες για το μέρος της γραμμής από Αθήνα μέχρι και Λαμία, και στην συνέχεια ηλεκτροκίνητους ελκυστήρες . Ο τύπος των ελκυστήρων δεν είναι δυνατό να αλλάξει στο προσεχές μέλλον.

2.3 Καιρικές συνθήκες / Weather conditions

Οι καιρικές συνθήκες αν και επηρεάζουν την κατανάλωση καυσίμων/ηλεκτρικής ενέργειας δεν είναι μια παράμετρος η οποία δύναται να ελεγχθεί.

2.4 Αεροδυναμική αμαξοστοιχιών / Aerodynamics

Η αεροδυναμική κατάσταση των αμαξοστοιχιών δύναται να επηρεάσει δραματικά την κατανάλωση καυσίμου/ενέργειας. Ωστόσο δεν είναι μια παράμετρος η οποία μπορεί να μεταβληθεί χωρίς κόστος

2.5 Χρονοδιάγραμμα δρομολογίων / Time schedule

Το χρονοδιάγραμμα των δρομολογίων μπορεί επηρεάσει την κατανάλωση καυσίμων/ ηλεκτρικής ενέργειας. Δρομολόγια τα οποία έχουν μεγάλες αναμονές για ανταποκρίσεις, δρομολόγια τα οποία έχουν ελάχιστες ανοχές σε καθυστερήσεις μπορούν να επηρεάσουν σημαντικά την κατανάλωση. Για την περίπτωση των μεγάλων αναμονών η κατανάλωση αυξάνεται λόγω των ενεργών μηχανών σε που βρίσκονται σε λειτουργική κατάσταση αδράνειας. Για την περίπτωση δρομολογίων με μικρές ανοχές σε καθυστερήσεις, αυτό οδηγεί σε αυξημένες καταναλώσεις μέσω της ασκούμενης πίεσης επι των μηχανοδηγών να κρατηθούν εντός ορθού χρονικού πλαισίου.

2.6 Τρόπος Οδήγησης / Driving Behavior

Ο τρόπος οδήγησης είναι μια σημαντική παράμετρος που επηρεάζει την κατανάλωση καυσίμου/ενέργειας. Για την ίδια διαδρομή δύο διαφορετικοί μηχανοδηγοί μπορούν να παρουσιάσουν σημαντικές αποκλίσεις στην κατανάλωση ενέργειας. Ετσι για παράδειγμα ένα «επιθετικός» οδηγός μπορεί να αυξήσει την κατανάλωση χωρίς τελικά να επιτύχει καλύτερους χρόνους για το δρομολόγιο του. Ο τρόπος οδήγησης μπορεί να επηρεασθεί χωρίς κόστος για τον οργανισμό / εταιρεία. Εξαιτίας αυτού του γεγονότος έχει νόημα να μελετήσουμε τους τρόπους οδήγησης και να προσπαθήσουμε μέσω κάποιας μεθοδολογίας αφενός να εντοπίσουμε τον ποιο οικονομικό τρόπο, αφετέρου να το διαφημίσουμε /

προωθήσουμε καταλλήλως. Με το παρόν πόνημα επιχειρούμε να προσφέρουμε ένα οδηγό καλής οδήγησης πιστοποιημένο με μαθηματικά μοντέλα.

3. Οικονομική Οδήγηση / Eco Driving

3.1 Εισαγωγή / Introduction

Η οικονομική οδήγηση μπορεί να αποτελέσει ένα σημαντικό παράγοντα βελτίωσης της κατανάλωσης καυσίμων/ενέργειας. Μπορεί να βασισθεί σε καλές πρακτικές οδήγησης οι οποίες μπορούν εξαχθούν με χρήση εργαλείων στατιστικής σε μια σειρά από διάφορες μετρήσεις. Ωστόσο πριν προχωρήσουμε σε ανάλυση μετρήσεων θα πρέπει να προσδιορίσουμε τους παράγοντες οι οποίοι επηρεάζουν την οικονομική οδήγηση. Σε γενικές γραμμές οι παράμετροι επηρεάζουν στον μεγαλύτερο βαθμό την σχετιζόμενη με την οδηγική συμπεριφορά κατανάλωση καυσίμου/ενέργειας είναι α. ταχύτητα κύλισης, β. επιβράδυνση, γ. επιτάχυνση. Στην παράγραφο που ακολουθεί περιγράφονται με μεγαλύτερη ακρίβεια οι προαναφερθείσες παράμετροι και αναλύονται αντιστοίχως.

3.2 Παράμετροι Οικονομικής οδήγησης / Factors Affecting ECO Driving

3.2.1 **Ταχύτητα / Speed**

Η ταχύτητα με την οποία κινείται η αμαξοστοιχία επηρεάζει την κατανάλωση καυσίμου/ενέργειας.. Όπως θα παρουσιασθεί και στην παράγραφο των Μετρικών η διαφορά της ταχύτητας μπορεί να αυξήσει την κατανάλωση κατά 50%. Το ποια ταχύτητα θα χρησιμοποιείται κάθε φορά από την αμαξοστοιχία είναι άμεση συνάρτηση του περιορισμών του δρομολογίου καθώς και της παρεχόμενης υπηρεσίας στους πελάτες της εταιρείας. Αυτό που μπορεί να βελτιωθεί είναι, να περιοριστούν οι άσκοπα μεγάλες ταχύτητες οι οποίες αυξάνουν την κατανάλωση χωρίς να προσφέρουν σημαντική βελτίωση της περιεχόμενης υπηρεσίας.

3.2.2 Επιβράδυνση / Deceleration

Η περιττή επιβράδυνση επηρεάζει την κατανάλωση εμμέσως ως αποτέλεσμα της ακολουθούμενης επιτάχυνσης, προκειμένου η αμαξοστοιχία να επανακάμψει στην αρχική ταχύτητα κύλισης.

3.2.3 Επιτάχυνση / Acceleration

Οι επιτάχυνση επιφέρει εγγενώς αύξηση της κατανάλωσης. Ωστόσο είναι εξαιρετικά σημαντικό η ρυθμός επιτάχυνσης. Έντονοι ρυθμοί επιτάχυνσης επιφέρουν δραματική αύξηση στην κατανάλωση καυσίμων από ότι ήπιοι ρυθμοί επιτάχυνσης για μεγαλύτερο χρονικό διάστημα. Σκοπός του εγχειριδίου είναι να εντοπίσουμε τα ποσοτικά χαρακτηριστικά της «κακής» - ενεργοβόρας επιτάχυνσης και να προτείνουμε την αντικατάστασή της με «καλές» επιταχύνσεις που επιφέρουν το ίδιο αποτέλεσμα.

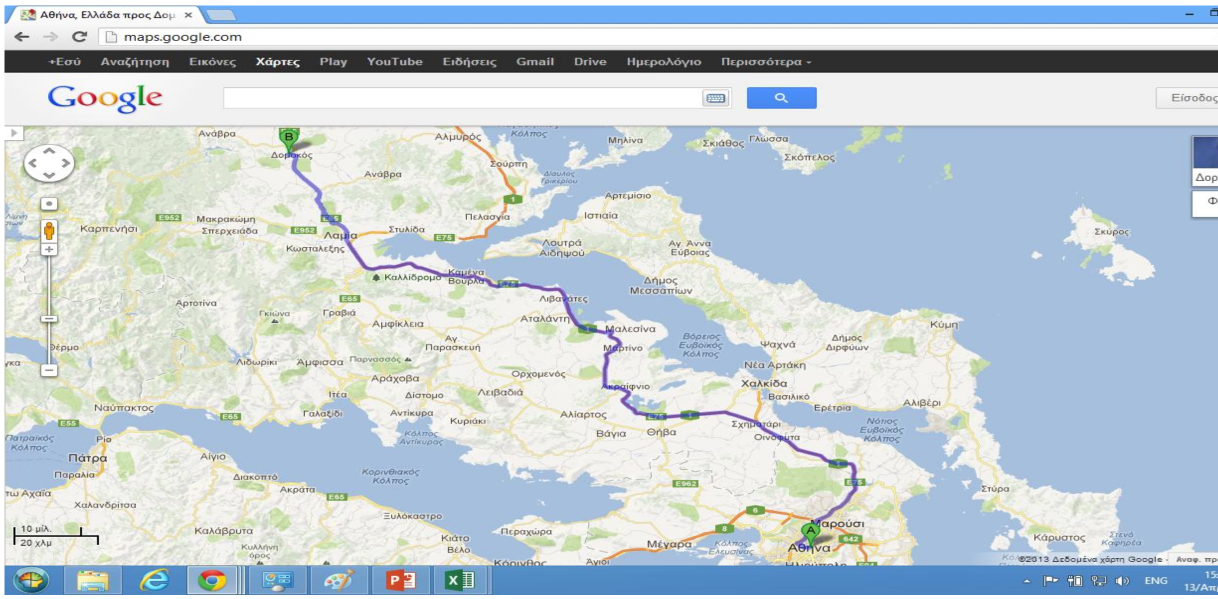
3.3 Μετρικές και Ανάλυση/ Metrics and Analysis

Για την εξαγωγή συμπερασμάτων στα ποσοτικά χαρακτηριστικά της κατανάλωσης προχωρήσαμε σε μια σειρά από μετρήσεις κατανάλωσης επι της βασικής γραμμής Αθήνας Θεσσαλονίκης. Υπάρχουν δύο βασικά μέρη της διαδρομής τα οποία διαφοροποιούνται με βάση το είδος της αυτοκινητάμαξας(Δηζελοκίνητη, ηλεκτροκίνητη).

Για κάθε ένα από τα δύο μέρη έχουμε και τις αντίστοιχες μετρικές.

3.3.1 Μετρικές Δηζελάμαξας – Αθήνα – Δομοκό /Diesel Locomotives Athens-Domokos

Η διαδρομή Αθήνα – Δομοκό με χρήση δηζελάμαξας περιγράφεται με σχετική ακρίβεια στην εικόνα 1.



Εικόνα 1 – Δρομολόγιο Αυτοκινητάμαξας στο τμήμα Αθήνα - Δομοκού

3.3.2 Μετρικές Δηζελάμαξας για απλή τροχοδρόμηση.

Ο πίνακας που ακολουθεί περιγράφει τις αποστάσεις κατά τις οποίες η αυτοκινητάμαξα τροχοδρομεί χωρίς επιταχύνσεις – επιβραδύνσεις. Παρατηρούμε ότι η κατανάλωση κείται στο όριο των 2lt/Km

Route Parts	Part-1	Part-2	Part-3	Part-4	Part-5	Part-6
Distance (normal Roll)	7,26Km	58 Km	34 Km	60 Km	30 Km	25 Km
Consumption	2,07lt/km	1,81	1,6	2,31	2,02	2,47

Diesel - Lt/per Km		lt/km	lt/km	lt/km	lt/km	lt/km
Total Consumption - Liters	15,03 lt	104,98 lt	54,4 lt	138,6 lt	60,6 lt	61,75 lt
Total Distance	214,26 Km					
Total Consumption	435,36 lt					
Average Consumption	2,03 lt					

Πίνακας 1 – Απλή τροχοδρόμηση

3.3.3 Μετρικές Δηζελάμαξας για Επιταχύνσεις

Ο πίνακας περιγράφει έξι σημεία στα οποία παρατηρήθηκαν επιταχύνσεις. Η πρώτη γραμμή περιέχει την απόσταση που απαιτείται προκειμένου η αυτοκινητάμαξα να προσεγγίσει την ταχύτητα των 100 Χλμ / Ωρα. Η δεύτερη γραμμή περιγράφει την μέση κατανάλωση καυσίμου σε lt/Km.

Αυτό που παρατηρούμε είναι ότι η κατανάλωση είναι σχεδόν 300% (μέση κατανάλωση περί το 2lt/Km) περισσότερη από την περίπτωση όπου η αυτοκινητάμαξα κινείται με σταθερή ταχύτητα και σε διαδρομή χωρίς θετικές κλίσεις εδάφους

Stops	STOP-1	STOP-2	STOP-3	STOP-4	STOP-5	STOP-6
Distance to reach max speed of 100kms/h	3.6Km	5.7 Km	8 Km	17 Km	7.3 Km	5.2 Km
Consumption Diesel - Lt/per Km	6.6 lt/km	5.8 lt/km	5.3 lt/km	7.4 lt/km	4.6 lt/km	7.5 lt/km
Total Consumption - Liters	23.76lt	33.06lt	42.4lt	125.8lt	33.58lt	39lt
Total Distance for all accelerations	46.8Km					
Total Consumption for all accelerations	297.6lt					
Average Consumption (lt/Kms)	6.3lt/Km					

Πίνακας 2 – Επιταχύνσεις

3.3.4 Μετρικές Δηζελάμαξας για Επιβραδύνσεις

Ο πίνακας περιγράφει έκι σημεία στα οποία παρατηρήθηκαν επιβραδύνσεις (αντίστοιχες των επιταχύνσεων). Η πρώτη γραμμή περιέχει την απόσταση που απαιτείται προκειμένου η αυτοκινητάμαξα να μειώσει την ταχύτητα από 100 Χλμ / Ωρα σε 0 Χλμ/Ωρα. Η δεύτερη γραμμή περιγράφει την μέση κατανάλωση καυσίμου σε lt/Km.

Αυτό που παρατηρούμε είναι ότι η κατανάλωση είναι σχεδόν ίδια(μέση κατανάλωση περί το 2lt/Km) με την περίπτωση όπου η αυτοκινητάμαξα κινείται με σταθερή ταχύτητα και σε διαδρομή χωρίς θετικές κλίσεις εδάφους

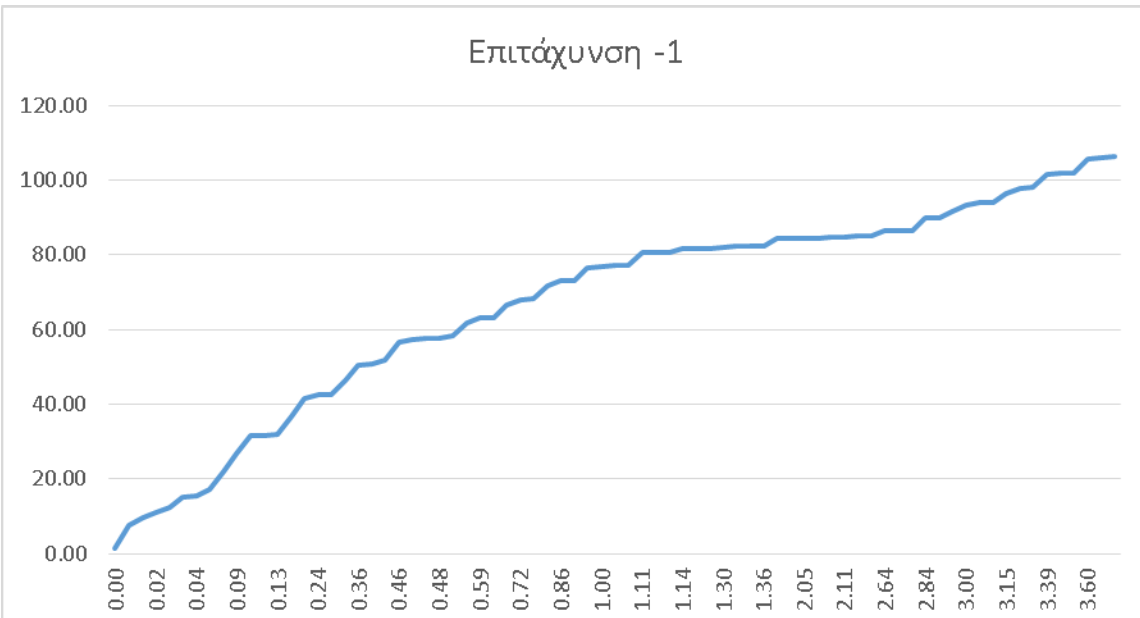
Stops	STOP-1	STOP-2	STOP-3	STOP-4	STOP-5	STOP-6
Distance to reduce speed to 0kms/h	2.95	0.92	1.68	1.89	2.00	1.35
Consumption Diesel - Lt/per Km	4.06	2.17	1.19	2.11	1.37	2.21
Total Consumption - Liters	11.98	2.00	2.00	3.99	2.74	2.98
Total Distance for all decelerations	10.79					
Total Consumption for all decelerations	25.69					
Average Consumption (lt/Kms)	2.38					

Πίνακας 3 – Επιβραδύνσεις

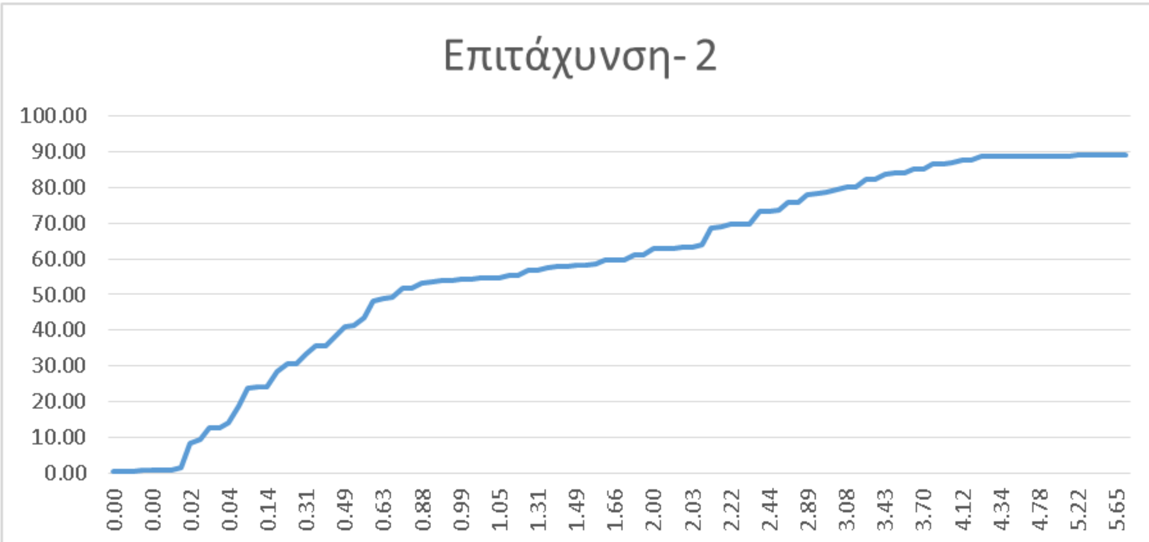
3.3.5 Συμπεράσματα από την ανάλυση των Μετρικών Δηζελάμαξας.

Από τους παραπάνω πίνακες προκύπτει αβίαστα το συμπέρασμα ότι οι επιταχύνσεις επιβαρύνουν δραματικά την κατανάλωση καυσίμων. Με αυτό το δεδομένο η συμβουλευτική κατεύθυνση προς τους μηχανοδηγούς είναι η προσπάθεια για μείωση των έντονων επιταχύνσεων και αντικατάσταση αυτών με ηπιότερες. Ο ιδεατός μηχανοδηγός που θα προταθεί ως πρότυπο μηχανοδήγησης θα διαθέτει την καλύτερη (μικρότερη) δυνατή κατανάλωση καυσίμου μέσω της αποφυγής των άσκοπων και έντονων επιταχύνσεων.

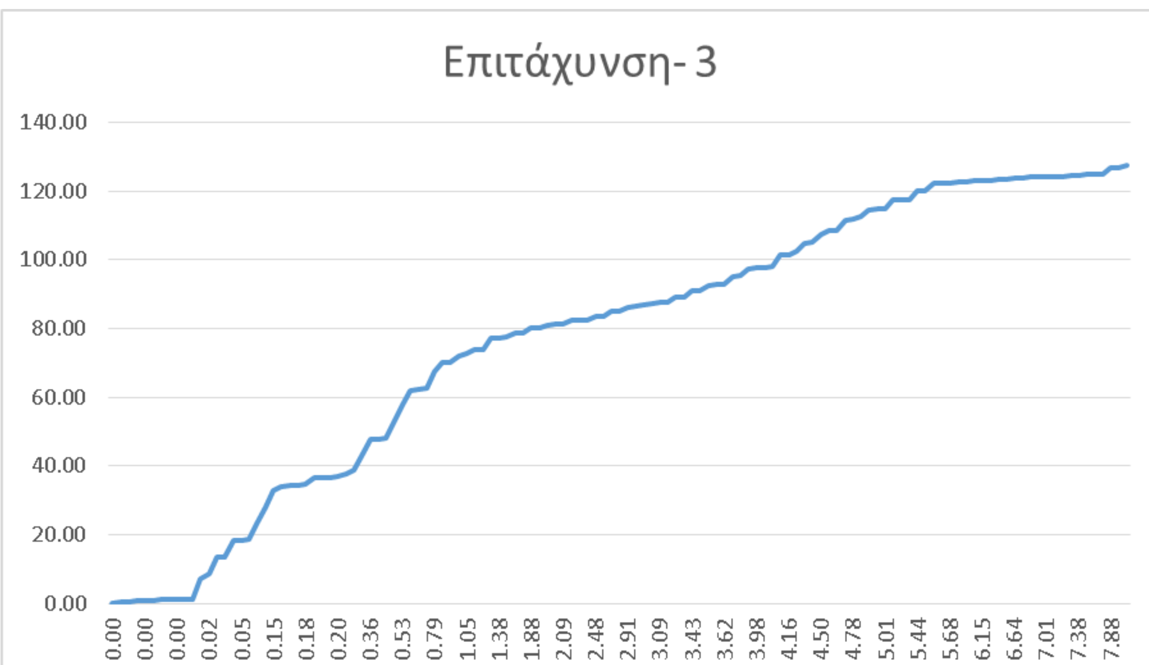
Οι παρακάτω εικόνες περιγράφουν τα σημεία της έντονης επιτάχυνσης και για τα οποία μπορεί να υπάρξει σημαντική βελτίωση εάν αυτές οι επιταχύνσεις ομαλοποιηθούν. Ο οριζόντιος άξονας περιγράφει την χιλιομετρική απόσταση κατά την οποία επιχειρείται η επιτάχυνση, ενώ ο κάθετος πίνακας περιγράφει την ταχύτητα της αυτοκινητάμαξας – συρμού.



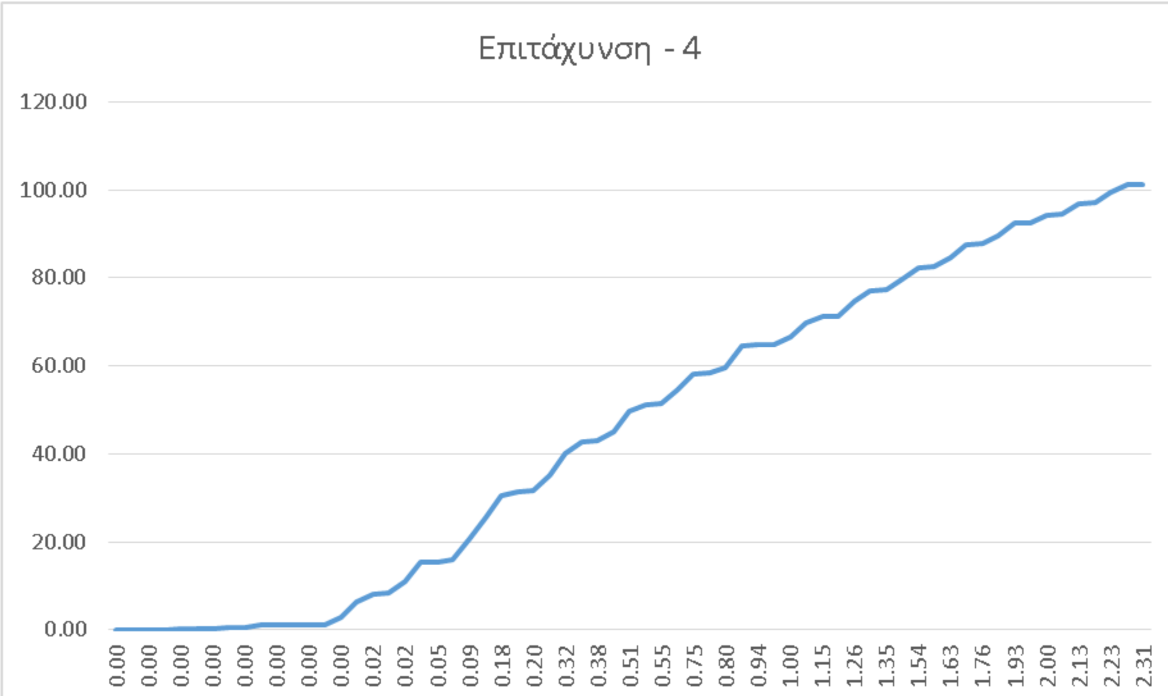
Εικόνα 2 – 1η Επιτάχυνση στο τμήμα Αθήνα - Δομοκού



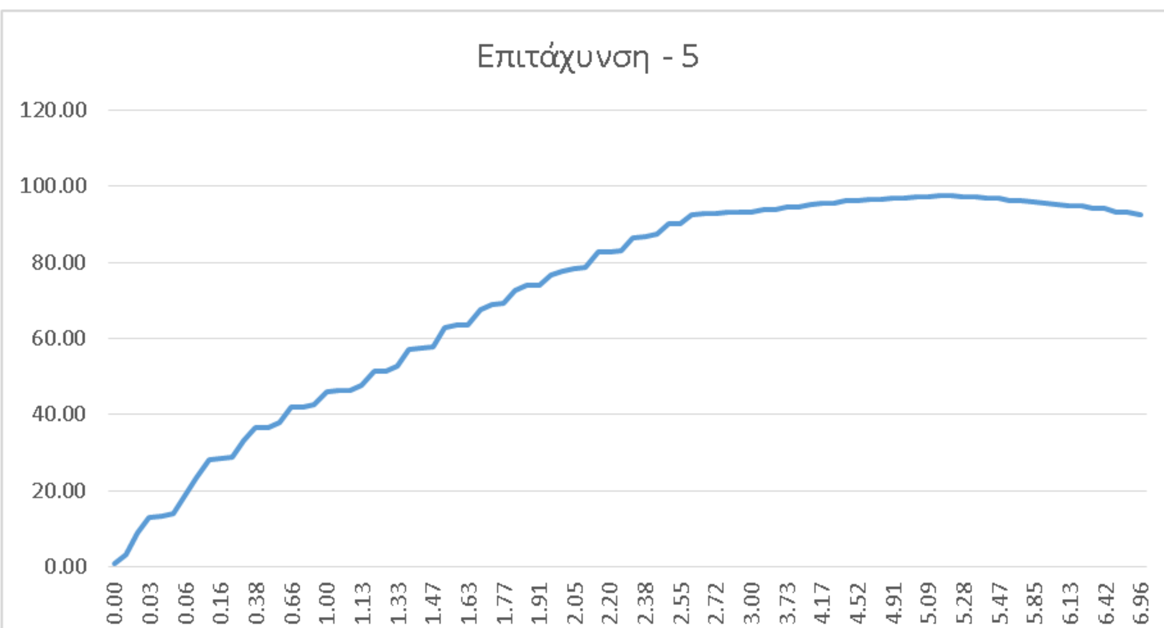
Εικόνα 3 – 2η Επιτάχυνση στο τμήμα Αθήνα - Δομοκού



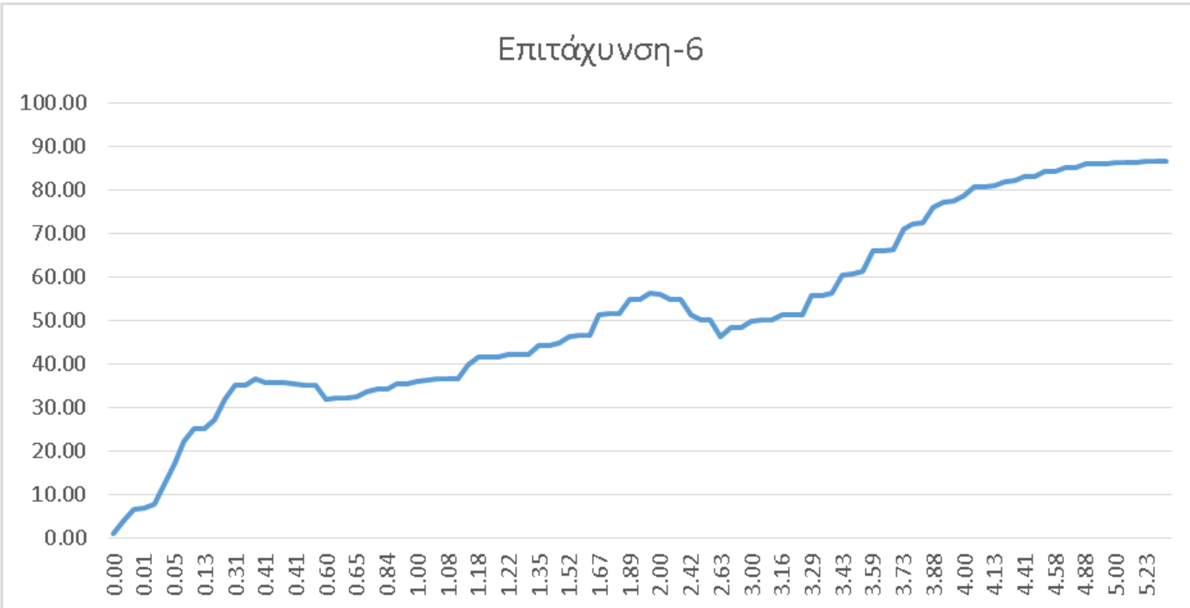
Εικόνα 4 – 3η Επιτάχυνση στο τμήμα Αθήνα - Δομοκού



Εικόνα 5 – 4η Επιτάχυνση στο τμήμα Αθήνα - Δομοκού



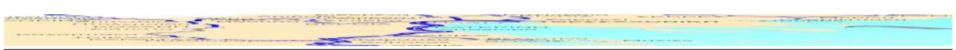
Εικόνα 6 – 5η Επιτάχυνση στο τμήμα Αθήνα - Δομοκού



Εικόνα 7 – 6η Επιτάχυνση στο τμήμα Αθήνα - Δομοκού

3.3.6 Μετρικές ηλεκτράμαξών Δομοκό - Θεσσαλονίκη.

Αντιστοίχως με τις δηζελοκίνητες υπάρχουν και οι ηλεκτροκίνητες αυτοκινητάμαξες για τις οποίες διαθέτουμε μετρήσεις για την συνολική κατανάλωση ηλεκτρικής ενέργειας της αυτοκινητάμαξας για όλο το μήκος του της γραμμής Δομοκού – Θεσσαλονίκης για το οποίο οι αυτοκινητάμαξες κινούνται με ηλεκτρική ενέργεια.



Εικόνα 8 – Ηλεκτροδοτούμενη Γραμμή Δομοκός Θεσσαλονίκη

Οι μετρικές αφορούν μια μεγάλη ομάδα ταξιδιών για το συγκεκριμένο δρομολόγιο όπου όλες οι συνθήκες της μέτρησης παραμένουν οι ίδιες . Δηλαδή :

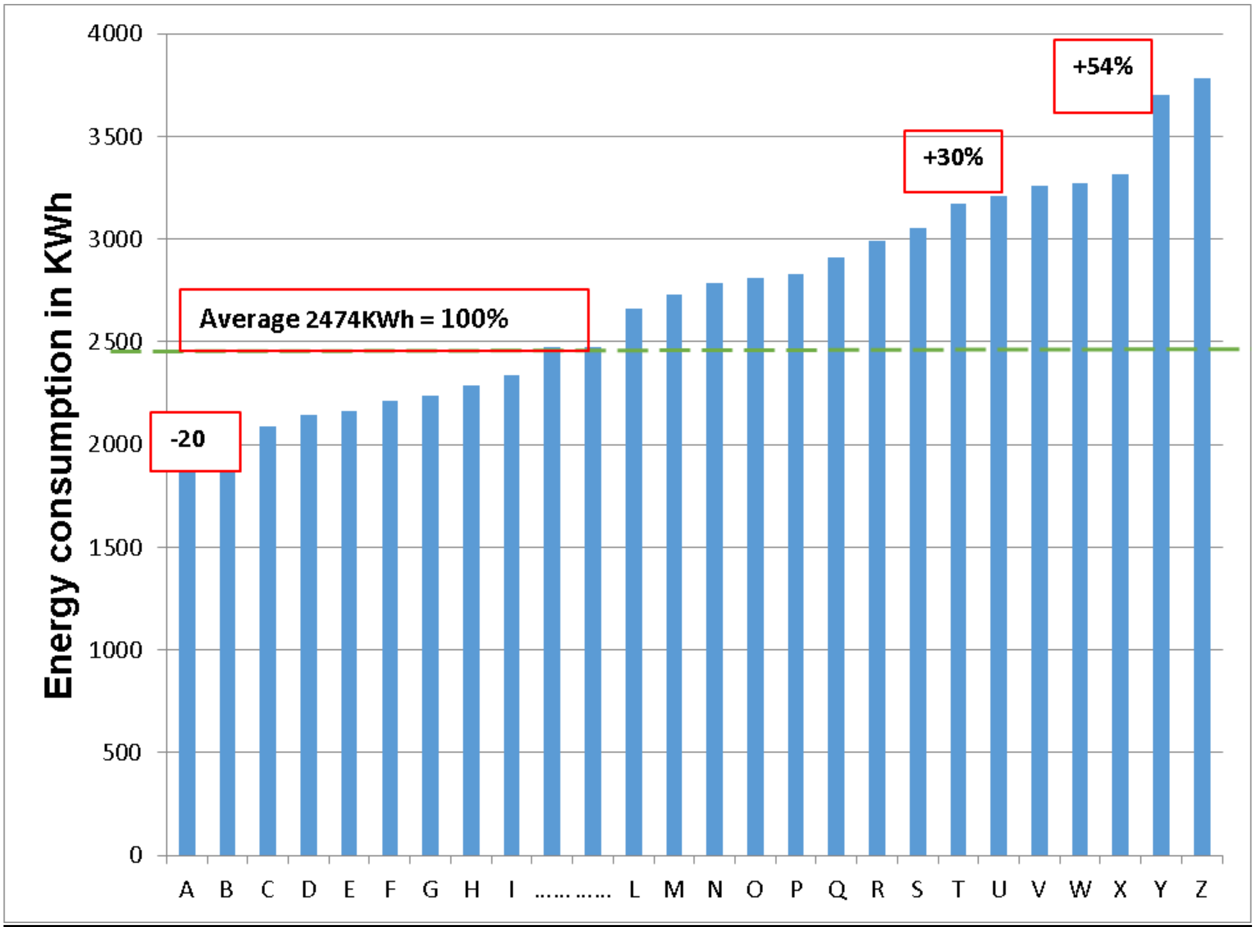
1. Ο τύπος της αυτοκινητάμαξας είναι πάντα ο ίδιος((Siemens 120)
2. Το φορτίο (αριθμός βαγονιών) είναι πάντα ο ίδιος((passenger trains – 200tn)
3. Απόσταση δρομολογίου (224Km)

3.3.7 Συμπεράσματα από την ανάλυση των Μετρικών Ηλεκτραμαξών.

Οι πίνακες που ακολουθούν περιγράφουν αναλυτικά τα διάφορα στοιχεία τα οποία έχουν συλλεχθεί και αναλυθεί.

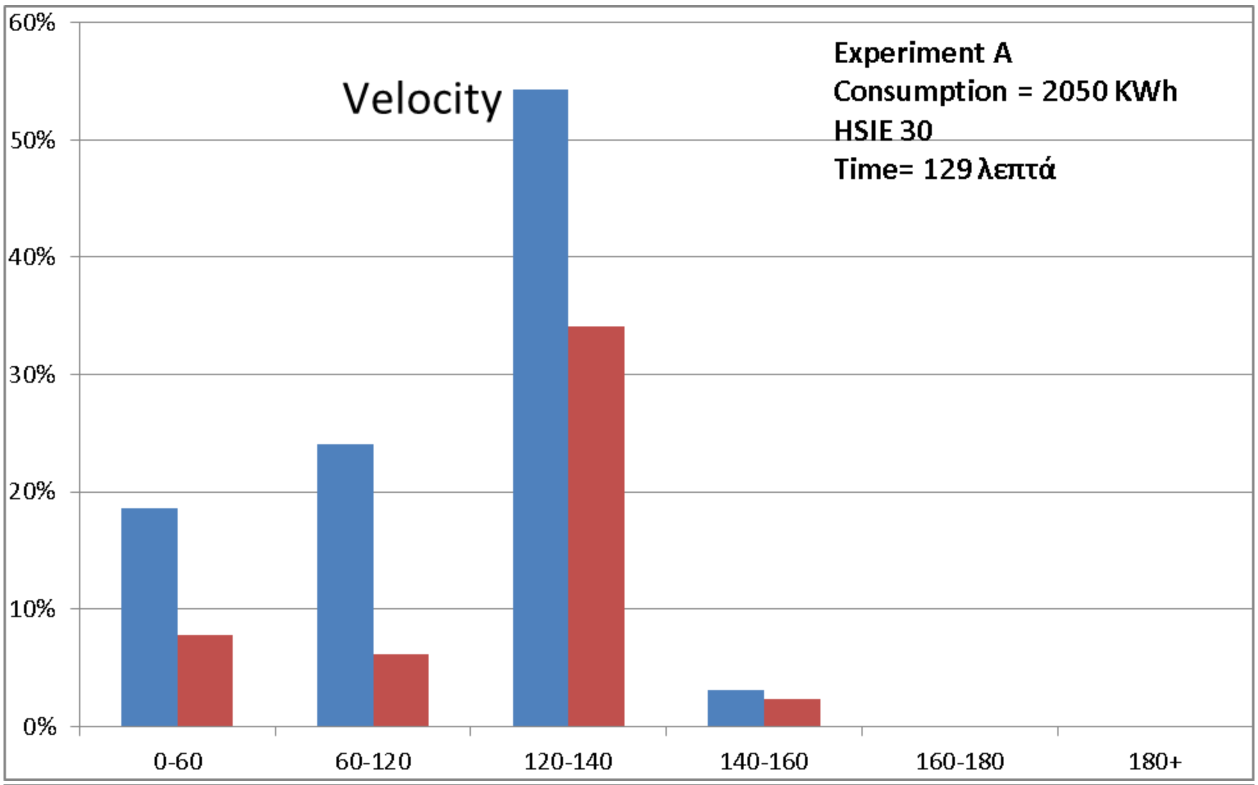
Με βάση την εικόνα 9 παρατηρούμε ότι οι καταναλώσεις της ηλεκτρικής ενέργειας ποικίλουν από ένα μέγιστο στην περιοχή των 2000Kwh έως και 3875Kwh. Είναι προφανές ότι οι διαφορές οφείλονται κατά κύριο λόγω στον τρόπο οδήγησης των συρμών.

Ο μέσος όρος κατανάλωσης είναι 2474Kwh όπως προκύπτει και από την οριζόντια γραμμή της εικόνας 9.Οι καταναλώσεις ποικίλουν από -20% έως και 54%

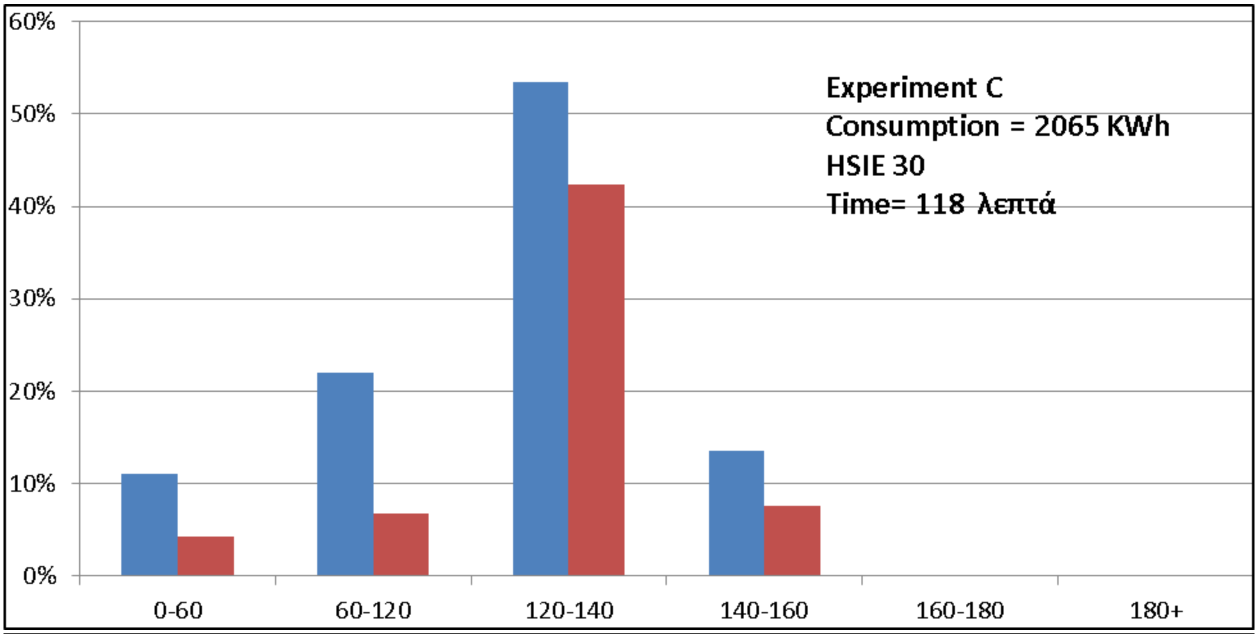


Εικόνα 9 – Γράφημα Κατανάλωσης Δρομολογίου

Τα γραφήματα των εικόνων 10,11 αφορούν ταξίδια χαμηλής ενεργειακής κατανάλωσης. Σε κάθε σήμανση ταχύτητας του οριζοντίου άξονα παρουσιάζονται 2 στήλες. Η αριστερή στήλη προβάλει το ποσοστό του χρόνου τα ιδίου κατά το οποίο η αμαξοστοιχία κινούνται στην συγκεκριμένη ταχύτητα ενώ η δεξιά στήλη περιγράφει το ποσοστό του χρόνου κατά το οποίο η αυτοκινητάμαξα βρίσκονταν σε κατάσταση έλξης. Παρατηρούμε ότι εφόσον η αμαξοστοιχία κινείται με ταχύτητες στο όριο των 120-140 Χλμ/Ωρα η κατανάλωση είναι μικρή. Κάποιες μικρές διαφορές κατανάλωσης δύναται να προκύψουν επίσης και εκ του αυξημένου ποσοστού χρόνου όπου η αμαξοστοιχία βρίσκεται σε κατάσταση έλξης αλλά πάντα για την ταχύτητα των 120-140 Χλμ/Ωρα

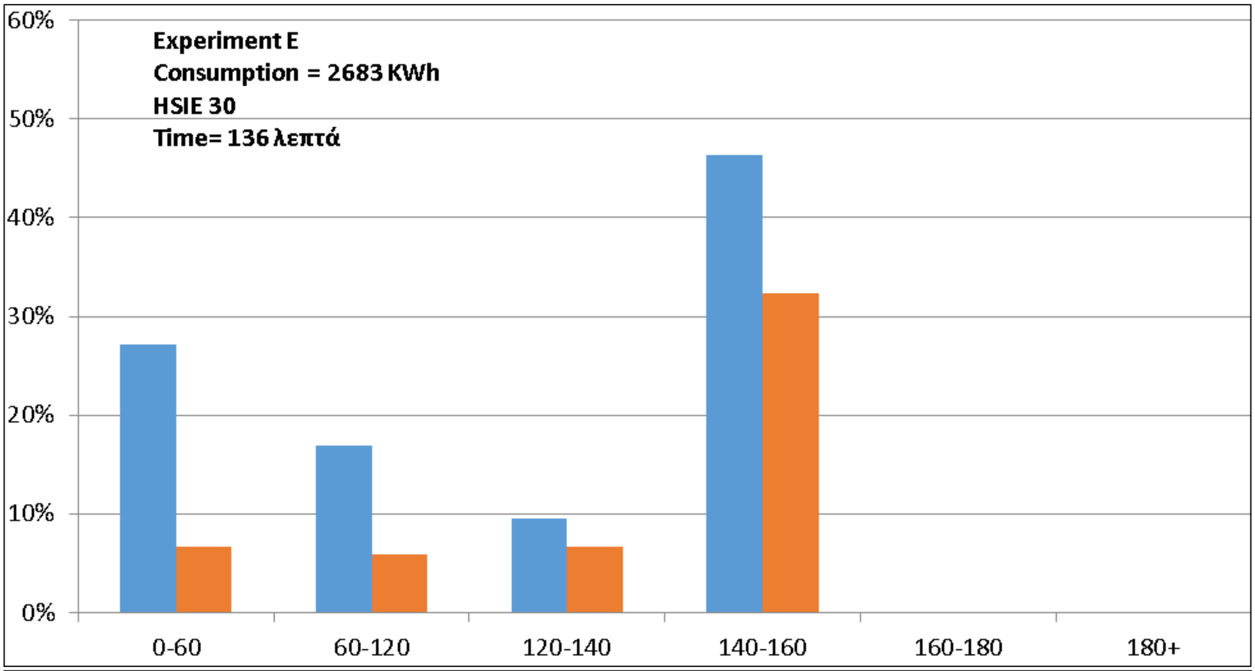


Εικόνα 10 – Γράφημα #1 - Χαμηλής Κατανάλωσης Δρομολογίου

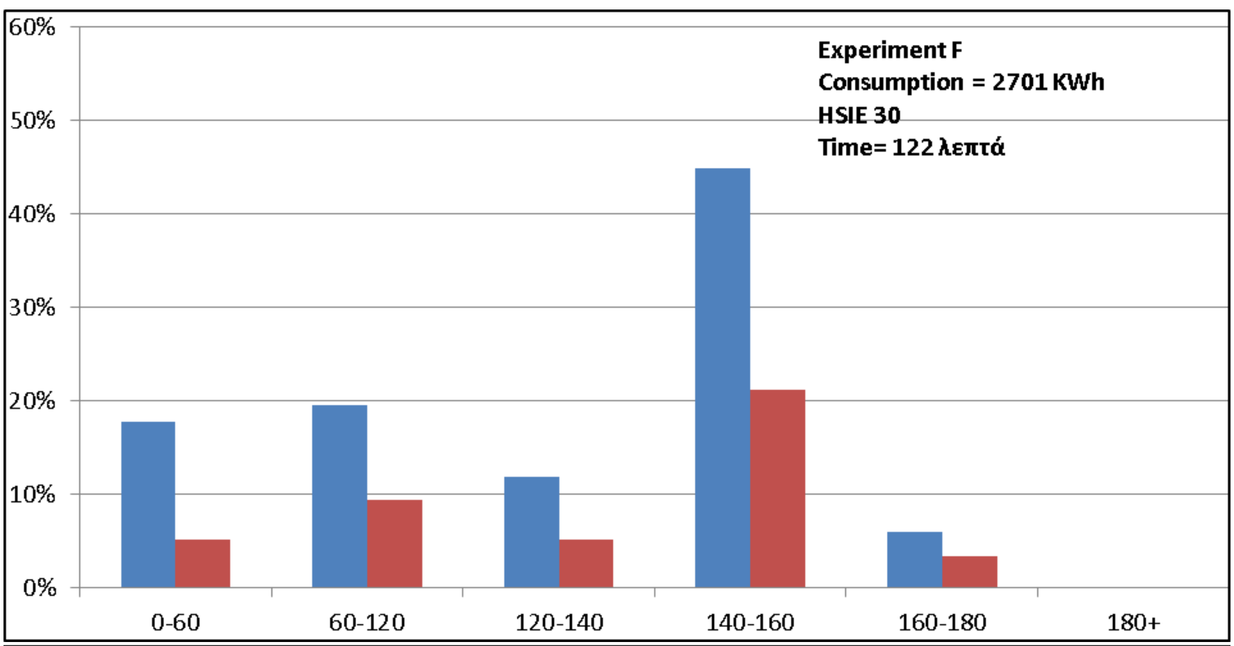


Εικόνα 11 – Γράφημα #2 - Χαμηλής Κατανάλωσης Δρομολογίου

Τα γραφήματα των εικόνων 12,13 αφορούν ταξίδια υψηλής ενεργειακής κατανάλωσης. Σε κάθε σήμανση ταχύτητας του οριζοντίου άξονα παρουσιάζονται 2 στήλες. Η αριστερή στήλη προβάλει το ποσοστό του χρόνου τα ιδίου κατά το οποίο η αμαξοστοιχία κινούνται στην συγκεκριμένη ταχύτητα ενώ η δεξιά στήλη περιγράφει το ποσοστό του χρόνου κατά το οποίο η αυτοκινητάμαξα βρίσκονταν σε κατάσταση έλξης. Παρατηρούμε ότι εφόσον η αμαξοστοιχία κινείται με ταχύτητες στο όριο των 140-160 Χλμ/Ωρα η κατανάλωση είναι μεγάλη και πάνω από τον μέσο όρο. Αν και το ποσό του χρόνου που καταναλώνεται για έλξη σε αυτές τις ταχύτητες δεν είμαι μεγαλύτερο από το αντίστοιχο ποσό για τις ταχύτητες των 120-140 Χλμ/Ωρα.

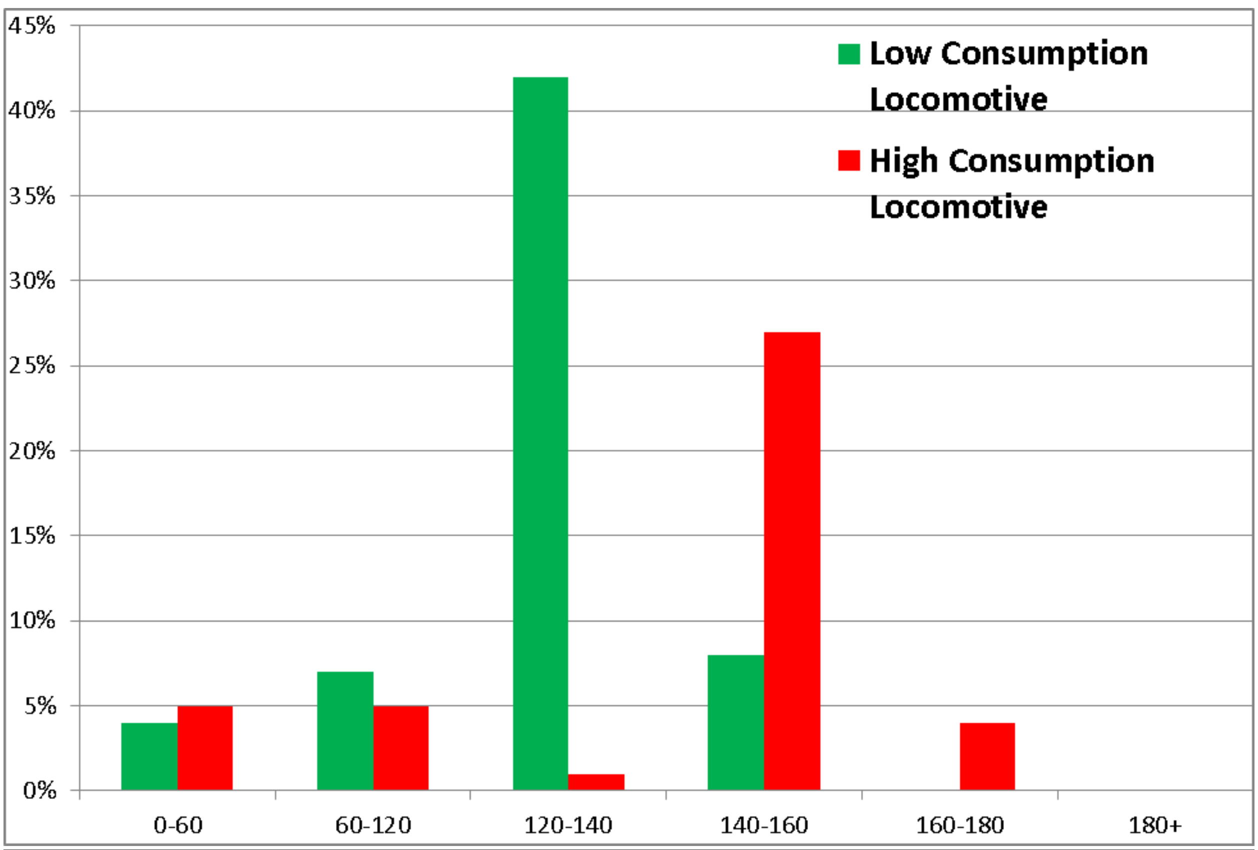


Εικόνα 12 – Γράφημα #1 – Υψηλής Κατανάλωσης Δρομολόγιο



Εικόνα 13 – Γράφημα #2 – Υψηλής Κατανάλωσης Δρομολόγιο

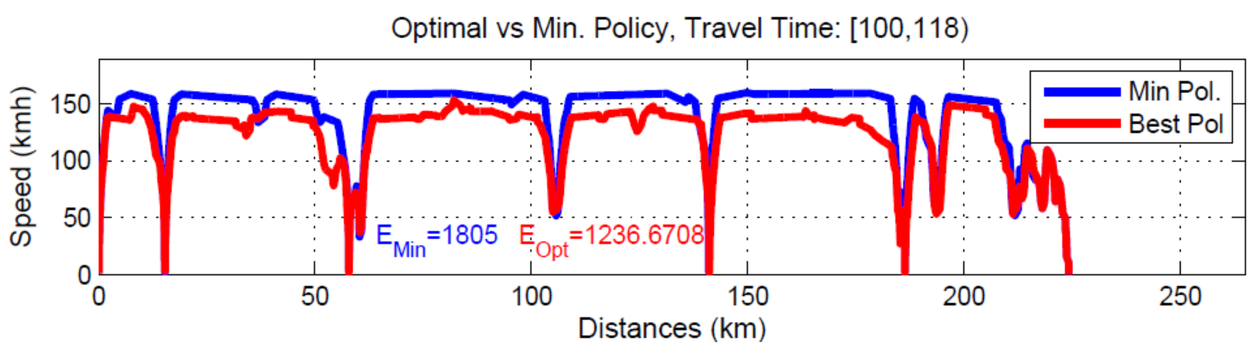
Τα γραφήματα των εικόνων 10,11,12,13 είναι απλά στιγμιότυπα συγκεκριμένων ταξιδιών. Ωστόσο το γράφημα της εικόνας 14 περιγράφει αναλυτικά σε ποιο εύρος ταχυτήτων συγκεντρώνονται οι μεγαλύτερες καταναλώσεις ενέργειας. Από το γράφημα προκύπτει ότι τα ποιο ενεργοβόρα δρομολόγια είναι αυτά για τα οποία η μέση ωριαία ταχύτητα κυμαίνεται μεταξύ 140-160 Χλμ/Ωρα. Αντιθέτως τα ποιο οικονομικά δρομολόγια είναι αυτά για τα οποία η μέση ωριαία ταχύτητα κυμαίνεται μεταξύ 120-140 Χλμ/Ωρα.



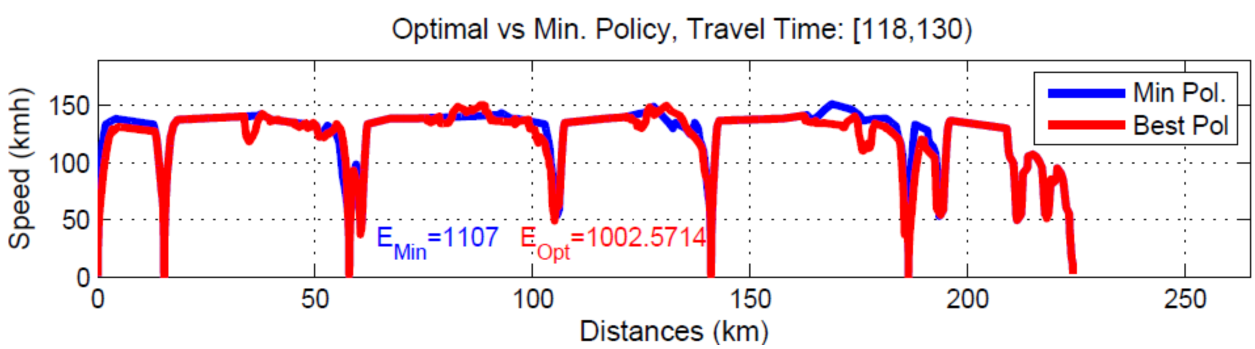
Εικόνα 14 – Γράφημα Συγκριτικής Κατανάλωσης

3.3.8 Προσομοίωση πολιτικής οδήγησης για τις Ηλεκτράμαξες στο δρομολόγιο Δομοκός-Θεσσαλονίκη.

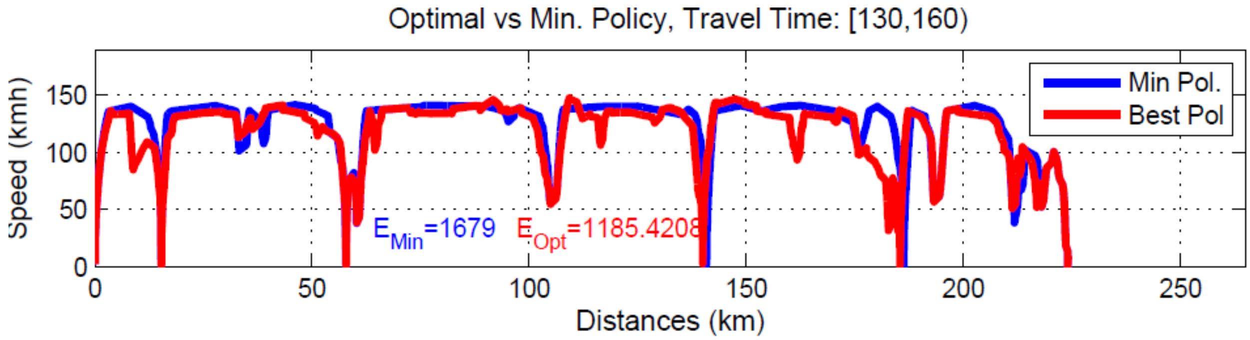
Με βάση τα αναλυτικά στοιχεία κατανάλωσης δύναται να κατασκευασθεί διάγραμμα βέλτιστης πολιτικής οδήγησης κατά στην οποία θα περιγράφονται πλήρως η ταχύτητα κύλισης για κάθε μέρος της διαδρομής καθώς και οι επιταχύνσεις / επιβραδύνσεις.. Οι εικόνες 15,16,17 περιγράφουν τις βέλτιστες πολιτικές για τα δρομολόγια διαφορετικών ταχυτήτων. Με βάση αυτές τις πολιτικές μπορούμε να κατασκευάσουμε τον οδηγό καλής συμπεριφοράς.



Εικόνα 15 – Γράφημα Βέλτιστης Πολιτικής για ταχύτητες [100,118]



Εικόνα 16 – Γράφημα Βέλτιστης Πολιτικής για ταχύτητες [118,130]



Εικόνα 17 – Γράφημα Βέλτιστης Πολιτικής για ταχύτητες [130,160]

4. Οικονομική Οδήγηση Πιλότος / Eco Driving Pilot

4.1 Εισαγωγή / Introduction

Ο πιλότος της οικονομικής οδήγησης θα περιέχει δύο διαφορετικές εκδοχές. Η πρώτη θα αφορά την οικονομική οδήγηση στις ηλεκτροκίνητες αυτοκινητάμαξες για το μέρος του δρομολογίου από Δομοκό προς Θεσσαλονίκη.

Η δεύτερη εκδοχή θα αφορά την οικονομική οδήγηση στις δηζελοκίνητες αυτοκινητάμαξες για το μέρος του δρομολογίου από Αθήνα προς Δομοκό

Θα κατασκευαστούν πίνακες οδηγικής συμπεριφοράς για κάθε μέρος της διαδρομής.

4.2 Πίνακες καλής οδηγικής πρακτικής / Good driving policy table

Ο πίνακας που ακολουθεί αφορά την οικονομική οδήγηση στις ηλεκτροκίνητες αυτοκινητάμαξες.

ΗΛΕΚΤΡΑΜΑΞΕΣ

Αρ. Ενέργειας	Περιγραφή Ενέργειας
1	Επιτάχυνση Από:0 Χιλ, Μέχρι:3 Χιλ. Σε Ταχύτητα:125
2	Κύλιση Από:3 Χιλ, Μέχρι:14 Χιλ. Σε Ταχύτητα:120

3	Επιβράδυνση Από:14 Χιλ, Μέχρι:15 Χιλ. Σε Ταχύτητα:1
4	Επιτάχυνση Από:15 Χιλ, Μέχρι:17 Χιλ. Σε Ταχύτητα:126
5	Κύλιση Από:17 Χιλ, Μέχρι:56 Χιλ. Σε Ταχύτητα:123
6	Επιβράδυνση Από:56 Χιλ, Μέχρι:58 Χιλ. Σε Ταχύτητα:2
7	Επιτάχυνση Από:58 Χιλ, Μέχρι:62 Χιλ. Σε Ταχύτητα:125
8	Κύλιση Από:62 Χιλ, Μέχρι:101 Χιλ. Σε Ταχύτητα:126
9	Επιβράδυνση Από:101 Χιλ, Μέχρι:105 Χιλ. Σε Ταχύτητα:49
10	Επιτάχυνση Από:105 Χιλ, Μέχρι:107 Χιλ. Σε Ταχύτητα:126
11	Κύλιση Από:107 Χιλ, Μέχρι:137 Χιλ. Σε Ταχύτητα:125
12	Επιβράδυνση Από:137 Χιλ, Μέχρι:141 Χιλ. Σε Ταχύτητα:3
13	Επιτάχυνση Από:141 Χιλ, Μέχρι:142 Χιλ. Σε Ταχύτητα:125
14	Κύλιση Από:142 Χιλ, Μέχρι:183 Χιλ. Σε Ταχύτητα:125
15	Επιβράδυνση Από:183 Χιλ, Μέχρι:185 Χιλ. Σε Ταχύτητα:37
16	Επιτάχυνση Από:185 Χιλ, Μέχρι:186 Χιλ. Σε Ταχύτητα:62
17	Επιβράδυνση Από:186 Χιλ, Μέχρι:186 Χιλ. Σε Ταχύτητα:0
18	Επιτάχυνση Από:186 Χιλ, Μέχρι:189 Χιλ. Σε Ταχύτητα:121
19	Επιβράδυνση Από:189 Χιλ, Μέχρι:192 Χιλ. Σε Ταχύτητα:100
20	Επιβράδυνση Από:192 Χιλ, Μέχρι:194 Χιλ. Σε Ταχύτητα:55
21	Επιτάχυνση Από:194 Χιλ, Μέχρι:195 Χιλ. Σε Ταχύτητα:125
22	Κύλιση Από:195 Χιλ, Μέχρι:209 Χιλ. Σε Ταχύτητα:126
23	Επιβράδυνση Από:209 Χιλ, Μέχρι:211 Χιλ. Σε Ταχύτητα:50
24	Επιτάχυνση Από:211 Χιλ, Μέχρι:215 Χιλ. Σε Ταχύτητα:107
25	Κύλιση Από:215 Χιλ, Μέχρι:217 Χιλ. Σε Ταχύτητα:91
26	Επιβράδυνση Από:217 Χιλ, Μέχρι:218 Χιλ. Σε Ταχύτητα:56
27	Κύλιση Από:218 Χιλ, Μέχρι:219 Χιλ. Σε Ταχύτητα:56
28	Επιτάχυνση Από:219 Χιλ, Μέχρι:221 Χιλ. Σε Ταχύτητα:95
29	Κύλιση Από:221 Χιλ, Μέχρι:222 Χιλ. Σε Ταχύτητα:80
30	Επιβράδυνση Από:222 Χιλ, Μέχρι:224 Χιλ. Σε Ταχύτητα:3

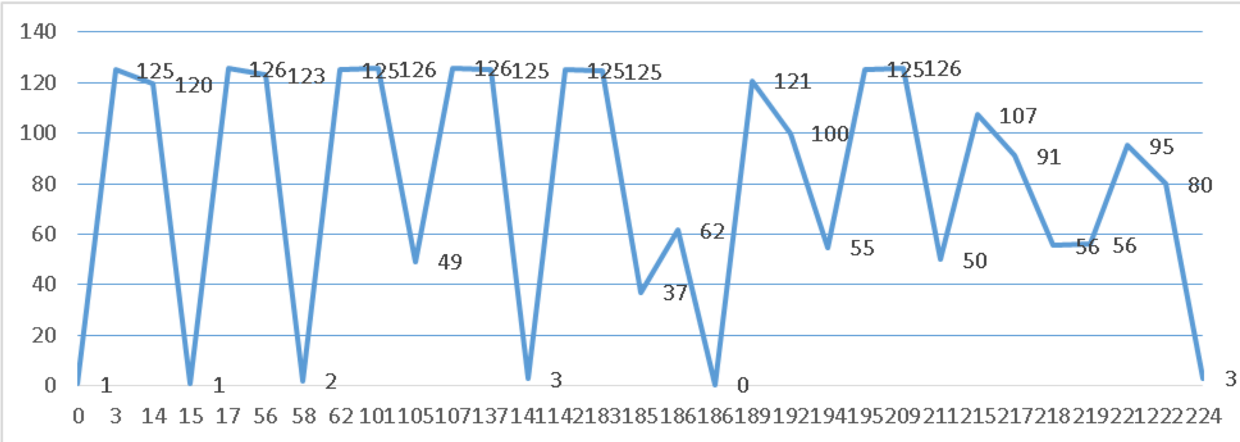
Πίνακας 4 – Οδηγική Συμπεριφορά Ηλεκτράμαξας

E-LOCOMOTIVES

Action #	Action Description
1	Acceleration FROM:0 Km TO:3 Km to SPEED:125
2	ROLL FROM:3 Km TO:14 Km to SPEED:120
3	Deceleration FROM:14 Km TO:15 Km to SPEED:1
4	Acceleration FROM:15 Km TO:17 Km to SPEED:126
5	ROLL FROM:17 Km TO:56 Km to SPEED:123

6	Deceleration FROM:56 Km TO:58 Km to SPEED:2
7	Acceleration FROM:58 Km TO:62 Km to SPEED:125
8	ROLL FROM:62 Km TO:101 Km to SPEED:126
9	Deceleration FROM:101 Km TO:105 Km to SPEED:49
10	Acceleration FROM:105 Km TO:107 Km to SPEED:126
11	ROLL FROM:107 Km TO:137 Km to SPEED:125
12	Deceleration FROM:137 Km TO:141 Km to SPEED:3
13	Acceleration FROM:141 Km TO:142 Km to SPEED:125
14	ROLL FROM:142 Km TO:183 Km to SPEED:125
15	Deceleration FROM:183 Km TO:185 Km to SPEED:37
16	Acceleration FROM:185 Km TO:186 Km to SPEED:62
17	Deceleration FROM:186 Km TO:186 Km to SPEED:0
18	Acceleration FROM:186 Km TO:189 Km to SPEED:121
19	Deceleration FROM:189 Km TO:192 Km to SPEED:100
20	Deceleration FROM:192 Km TO:194 Km to SPEED:55
21	Acceleration FROM:194 Km TO:195 Km to SPEED:125
22	ROLL FROM:195 Km TO:209 Km to SPEED:126
23	Deceleration FROM:209 Km TO:211 Km to SPEED:50
24	Acceleration FROM:211 Km TO:215 Km to SPEED:107
25	ROLL FROM:215 Km TO:217 Km to SPEED:91
26	Deceleration FROM:217 Km TO:218 Km to SPEED:56
27	ROLL FROM:218 Km TO:219 Km to SPEED:56
28	Acceleration FROM:219 Km TO:221 Km to SPEED:95
29	ROLL FROM:221 Km TO:222 Km to SPEED:80
30	Deceleration FROM:222 Km TO:224 Km to SPEED:3

Πίνακας 5 – Driving Behavior of the Electric Locomotive



Εικόνα 18 – Γράφημα Βέλτιστης Πολιτικής για Πολιτικής για ταχύτητες [118,130]

Ο πίνακας που ακολουθεί αφορά την οικονομική οδήγηση στις δηζελοκίνητες αυτοκινητάμαξες.

ΔΗΖΕΛΑΜΑΞΕΣ

Αρ. Ενέργειας	**Περιγραφή Ενέργειας**
1	**ΗΠΙΑ** Επιτάχυνση Από:0 Χιλ, Μέχρι:3.64 Χιλ. Σε Ταχύτητα:**106**
2	**ΗΠΙΑ** Επιτάχυνση Από:10 Χιλ, Μέχρι:15 Χιλ. Σε Ταχύτητα: **89**
3	**ΗΠΙΑ** Επιτάχυνση Από:66 Χιλ, Μέχρι: 74 Χιλ. Σε Ταχύτητα: **127**
4	**ΗΠΙΑ** Επιτάχυνση Από:140 Χιλ, Μέχρι: 143 Χιλ. Σε Ταχύτητα: **101**
5	**ΗΠΙΑ** Επιτάχυνση Από:167 Χιλ, Μέχρι: 174 Χιλ. Σε Ταχύτητα: **92**
6	**ΗΠΙΑ** Επιτάχυνση Από:228 Χιλ, Μέχρι: 233 Χιλ. Σε Ταχύτητα: 86

Πίνακας 6 – Οδηγική Συμπεριφορά Δηζελάμαξας

D-LOCOMOTIVES

Αρ. Ενέργειας	Περιγραφή Ενέργειας
1	**SOFT** Acceleration Από:0 Km TO:3.64 Χιλ. Σε Ταχύτητα:**106**
2	**SOFT** Acceleration FROM:10 Km TO:15 Km to SPEED: **89**
3	**SOFT** Acceleration FROM:66 Km TO: 74 Km to SPEED: **127**
4	**SOFT** Acceleration FROM:140 Km TO: 143 Km to SPEED: **101**
5	**SOFT** Acceleration FROM:167 Km TO: 174 Km to SPEED: **92**
6	**SOFT** Acceleration FROM:228 Km TO: 233 Km to SPEED: 86

Πίνακας 7 – Driving Behavior of the DIESEL Locomotive

5. Συμπεράσματα / Conclusions

5.1 Γενικά συμπεράσματα / General Conclusions

Ο πιλότος της οικονομικής οδήγησης περιέχει δύο διαφορετικές εκδοχές.

Η πρώτη θα αφορά την οικονομική οδήγηση για τις ηλεκτράμαξες όπου υπάρχει πλήρη ανάλυση καθώς και πολιτική βέλτιστης οδήγησης όπως αυτή παράχθηκε με την ανάλυση των μετρήσεων κατανάλωσης, ταχύτητας, απόστασης. Μπορούμε να εικάσουμε σε αυτό το σημείο ότι η ανάλυση είναι σχεδόν ολοκληρωμένη και υπολείπεται μόνο η δοκιμή του μοντέλου σε πραγματικές συνθήκες προκειμένου να επιβεβαιωθεί το μοντέλο βελτιστοποίησης.

Η δεύτερη περιλαμβάνει την οικονομική οδήγηση για τις δηζελάμαξες για τις οποίες δεν έχουμε σημαντικό αριθμό δεδομένων. Η βελτιστοποίηση βασίζεται στην ανάλυση των σημείων υψηλής κατανάλωσης Ντίζελ τα οποία είναι τα σημεία της έντονης επιτάχυνσης. Αυτό που προτείνεται είναι στην ουσία η αποφυγή των έντονων επιταχύνσεων έτσι ώστε μειωθεί η κατανάλωση.

Θα πρέπει σημειωθεί ότι η εκδοχή για τις δηζελάμαξες είναι κυρίως εμπειρική και δεν μπορεί να θεωρηθεί ως αυστηρά επιστημονική. Θα απαιτηθούν πολύ περισσότερα δεδομένα για να έχουμε επαρκώς αποδεδειγμένα συμπεράσματα.

Ο παρόν οδηγός βρίσκεται υπο συνεχή αναμόρφωση κάθε στιγμή που μια καλύτερη εκδοχή βελτιστοποίησης επιφέρει καλύτερα αποτελέσματα εξοικονόμησης ενέργειας.

C. XSD Interface description

```
<xs:schema elementFormDefault="qualified" attributeFormDefault="unqualified" version="2.2"
targetNamespace="http://datex2.eu/schema/2/2_0_modified">


<xs:complexType name="Comment">
 <xs:annotation>
  <xs:documentation>A free text comment with an optional date/time stamp that can be used by the operator to convey
        un-coded observations/information.</xs:documentation>
 </xs:annotation>
 <xs:sequence>
  <xs:element name="comment" type="D2LogicalModel:MultilingualString" minOccurs="1"
        maxOccurs="1">
   <xs:annotation>
    <xs:documentation>A free text comment that can be used by the operator to convey un-coded
              observations/information.</xs:documentation>
   </xs:annotation>
  </xs:element><xs:element name="commentDateTime" type="D2LogicalModel:DateTime"
                          minOccurs="0" maxOccurs="1">
   <xs:annotation>
    <xs:documentation>The date/time at which the comment was made.</xs:documentation>
   </xs:annotation>
  </xs:element>
  <xs:element name="commentType" type="D2LogicalModel:CommentTypeEnum"
                          minOccurs="0" maxOccurs="1">
   <xs:annotation>
    <xs:documentation>A classification of the type of comment.</xs:documentation>
   </xs:annotation>
  </xs:element>
```

```
    <xs:element name="commentExtension" type="D2LogicalModel:_ExtensionType"
                                        minOccurs="0"/>
  </xs:sequence>
</xs:complexType>

<xs:simpleType name="CommentTypeEnum">
  <xs:annotation>
    <xs:documentation>Classification of comment types.</xs:documentation>
  </xs:annotation>
  <xs:restriction base="xs:string">
    <xs:enumeration value="abnormalValueNote">
      <xs:annotation>
        <xs:documentation>A free text human oriented note describing details of abnormal
                                                  value associated with the Measurements.</xs:documentation>
      </xs:annotation>
    </xs:enumeration>
    <xs:enumeration value="dataProcessingNote">
      <xs:annotation>
        <xs:documentation>A free text human oriented note describing the way the
                                                  information in the Record has been or should be
                                                  processed.</xs:documentation>
      </xs:annotation>
    </xs:enumeration>
    <xs:enumeration value="description">
      <xs:annotation>
        <xs:documentation>A free text human oriented description of the measurement.</xs:documentation>
      </xs:annotation>
    </xs:enumeration>
    <xs:enumeration value="internalNote">
      <xs:annotation>
```

```
        <xs:documentation>A free text human oriented note that supports internal traffic
                                                control operations relating to the Record.</xs:documentation>
      </xs:annotation>
    </xs:enumeration>
    <xs:enumeration value="locationDescriptor">
      <xs:annotation>
        <xs:documentation>A free text human oriented description of the location of the
                                                elements defined by the Record.</xs:documentation>
      </xs:annotation>
    </xs:enumeration>
    <xs:enumeration value="warning">
      <xs:annotation>
        <xs:documentation>A free text human oriented warning relating to the
                                                Record.</xs:documentation>
      </xs:annotation>
    </xs:enumeration>
    <xs:enumeration value="other">
      <xs:annotation>
        <xs:documentation>Other than as defined in this enumeration.</xs:documentation>
      </xs:annotation>
    </xs:enumeration>
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name="CountryEnum">
  <xs:annotation>
    <xs:documentation>List of countries.</xs:documentation>
  </xs:annotation>
  <xs:restriction base="xs:string">
    <xs:enumeration value="at">
```

```
  <xs:annotation><xs:documentation>Austria</xs:documentation></xs:annotation>
</xs:enumeration>
<xs:enumeration value="be">
  <xs:annotation><xs:documentation>Belgium</xs:documentation></xs:annotation>
</xs:enumeration>
<xs:enumeration value="bg">
  <xs:annotation><xs:documentation>Bulgaria</xs:documentation></xs:annotation>
</xs:enumeration>
<xs:enumeration value="ch">
  <xs:annotation><xs:documentation>Switzerland</xs:documentation></xs:annotation>
</xs:enumeration>
<xs:enumeration value="cs">
  <xs:annotation>
    <xs:documentation>Serbia and Montenegro</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="cy">
  <xs:annotation><xs:documentation>Cyprus</xs:documentation></xs:annotation>
</xs:enumeration>
<xs:enumeration value="cz">
  <xs:annotation><xs:documentation>Czech Republic</xs:documentation></xs:annotation>
</xs:enumeration>
<xs:enumeration value="de">
  <xs:annotation><xs:documentation>Germany</xs:documentation></xs:annotation>
</xs:enumeration>
<xs:enumeration value="dk">
  <xs:annotation><xs:documentation>Denmark</xs:documentation></xs:annotation>
</xs:enumeration>
<xs:enumeration value="ee">
  <xs:annotation><xs:documentation>Estonia</xs:documentation></xs:annotation>
```

```
</xs:enumeration>
<xs:enumeration value="es">
  <xs:annotation><xs:documentation>Spain</xs:documentation></xs:annotation>
</xs:enumeration>
<xs:enumeration value="fi">
  <xs:annotation><xs:documentation>Finland</xs:documentation></xs:annotation>
</xs:enumeration>
<xs:enumeration value="fo">
  <xs:annotation><xs:documentation>Faroe Islands</xs:documentation></xs:annotation>
</xs:enumeration>
<xs:enumeration value="fr">
  <xs:annotation><xs:documentation>France</xs:documentation></xs:annotation>
</xs:enumeration>
<xs:enumeration value="gb">
  <xs:annotation><xs:documentation>Great Britain</xs:documentation></xs:annotation>
</xs:enumeration>
<xs:enumeration value="gg">
  <xs:annotation><xs:documentation>Guernsey</xs:documentation></xs:annotation>
</xs:enumeration>
<xs:enumeration value="gi">
  <xs:annotation><xs:documentation>Gibraltar</xs:documentation></xs:annotation>
</xs:enumeration>
<xs:enumeration value="gr">
  <xs:annotation><xs:documentation>Greece</xs:documentation></xs:annotation>
</xs:enumeration>
<xs:enumeration value="hr">
  <xs:annotation><xs:documentation>Croatia</xs:documentation></xs:annotation>
</xs:enumeration>
<xs:enumeration value="hu">
  <xs:annotation><xs:documentation>Hungary</xs:documentation></xs:annotation>
```

```
</xs:enumeration>
<xs:enumeration value="ie">
  <xs:annotation><xs:documentation>Ireland</xs:documentation></xs:annotation>
</xs:enumeration>
<xs:enumeration value="im">
  <xs:annotation><xs:documentation>Isle Of Man</xs:documentation></xs:annotation>
</xs:enumeration>
<xs:enumeration value="is">
  <xs:annotation><xs:documentation>Iceland</xs:documentation></xs:annotation>
</xs:enumeration>
<xs:enumeration value="it">
  <xs:annotation><xs:documentation>Italy</xs:documentation></xs:annotation>
</xs:enumeration>
<xs:enumeration value="je">
  <xs:annotation><xs:documentation>Jersey</xs:documentation></xs:annotation>
</xs:enumeration>
<xs:enumeration value="li">
  <xs:annotation><xs:documentation>Lichtenstein</xs:documentation></xs:annotation>
</xs:enumeration>
<xs:enumeration value="lt">
  <xs:annotation><xs:documentation>Lithuania</xs:documentation></xs:annotation>
</xs:enumeration>
<xs:enumeration value="lu">
  <xs:annotation><xs:documentation>Luxembourg</xs:documentation></xs:annotation>
</xs:enumeration>
<xs:enumeration value="lv">
  <xs:annotation><xs:documentation>Latvia</xs:documentation></xs:annotation>
</xs:enumeration>
<xs:enumeration value="ma">
  <xs:annotation><xs:documentation>Morocco</xs:documentation></xs:annotation>
```

</xs:enumeration>

<xs:enumeration value="mc">

<xs:annotation><xs:documentation>Monaco</xs:documentation></xs:annotation>

</xs:enumeration>

<xs:enumeration value="mk">

<xs:annotation><xs:documentation>Macedonia</xs:documentation></xs:annotation>

</xs:enumeration>

<xs:enumeration value="mt">

<xs:annotation><xs:documentation>Malta</xs:documentation></xs:annotation>

</xs:enumeration>

<xs:enumeration value="nl">

<xs:annotation><xs:documentation>Netherlands</xs:documentation></xs:annotation>

</xs:enumeration>

<xs:enumeration value="no">

<xs:annotation><xs:documentation>Norway</xs:documentation></xs:annotation>

</xs:enumeration>

<xs:enumeration value="pl">

<xs:annotation><xs:documentation>Poland</xs:documentation></xs:annotation>

</xs:enumeration>

<xs:enumeration value="pt">

<xs:annotation><xs:documentation>Portugal</xs:documentation></xs:annotation>

</xs:enumeration>

<xs:enumeration value="ro">

<xs:annotation><xs:documentation>Romania</xs:documentation></xs:annotation>

</xs:enumeration>

<xs:enumeration value="se">

<xs:annotation><xs:documentation>Sweden</xs:documentation></xs:annotation>

</xs:enumeration>

<xs:enumeration value="si">

<xs:annotation><xs:documentation>Slovenia</xs:documentation></xs:annotation>

```
    </xs:enumeration>
    <xs:enumeration value="sk">
      <xs:annotation><xs:documentation>Slovakia</xs:documentation></xs:annotation>
    </xs:enumeration>
    <xs:enumeration value="sm">
      <xs:annotation><xs:documentation>San Marino</xs:documentation></xs:annotation>
    </xs:enumeration>
    <xs:enumeration value="tr">
      <xs:annotation><xs:documentation>Turkey</xs:documentation></xs:annotation>
    </xs:enumeration>
    <xs:enumeration value="va">
      <xs:annotation>
        <xs:documentation>Vatican City State</xs:documentation>
      </xs:annotation>
    </xs:enumeration>
    <xs:enumeration value="other">
      <xs:annotation>
        <xs:documentation>Other than as defined in this enumeration.</xs:documentation>
      </xs:annotation>
    </xs:enumeration>
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name="DateTime">
  <xs:annotation>
    <xs:documentation>A combination of integer-valued year, month, day, hour, minute
                      properties, a decimal-valued second property and a time zone property from
                      which it is possible to determine the local time, the equivalent UTC time and
                      the time zone offset from UTC.</xs:documentation>
  </xs:annotation>
```

```
  <xs:restriction base="xs:dateTime"/>
</xs:simpleType>

<xs:simpleType name="Integer">
  <xs:annotation>
    <xs:documentation>An integer number whose value space is the set {-2147483648,
                  -2147483647, -2147483646, ..., -2, -1, 0, 1, 2, ..., 2147483645, 2147483646,
                  2147483647}.</xs:documentation>
  </xs:annotation>
  <xs:restriction base="xs:integer"/>
</xs:simpleType>

<xs:simpleType name="Language">
  <xs:annotation>
    <xs:documentation>A language datatype, identifies a specified language by an ISO 639-1
                          2-alpha / ISO 639-2 3-alpha code.</xs:documentation>
  </xs:annotation>
  <xs:restriction base="xs:language"/>
</xs:simpleType>

<xs:simpleType name="MeasurementTypeEnum">
  <xs:annotation>
    <xs:documentation>Type of measurement. This enumeration indicates if the measurement
                          are done on one point, or on a route, or in several related points in
                          an area.</xs:documentation>
  </xs:annotation>
  <xs:restriction base="xs:string">
    <xs:enumeration value="PointMeasurement"/>
    <xs:enumeration value="TrajectMeasurement"/>
    <xs:enumeration value="AreaMeasurement"/>
```

```
  </xs:restriction>
</xs:simpleType>

<xs:complexType name="PollutionMeasurement">
  <xs:annotation>
    <xs:documentation>Record containing the overall time of the
                                        measurement</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="PublicationTime" type="D2LogicalModel:DateTime"
                        minOccurs="1" maxOccurs="1"/>
    <xs:element name="StartTime" type="D2LogicalModel:DateTime"
                        minOccurs="1" maxOccurs="1"/>
    <xs:element name="EndTime" type="D2LogicalModel:DateTime"
                        minOccurs="1" maxOccurs="1"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="ReductionInformation">
  <xs:annotation>
    <xs:documentation>Main reduction message, containing polution measurements or
                                        calculations of several locations over several times</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="ModelVersion" type="D2LogicalModel:Integer"
                        minOccurs="1" maxOccurs="1"/>
    <xs:element name="SequenceNr" type="D2LogicalModel:Integer"
                        minOccurs="1" maxOccurs="1"/>
    <xs:element name="SupplierIdentification"
                        type="D2LogicalModel:ReductionSupplierIdentification"
```

```
                        minOccurs="1" maxOccurs="1"/>
    <xs:element name="PolutionMeasurement" type="D2LogicalModel:PolutionMeasurement"
                        minOccurs="1" maxOccurs="1"/>
    <xs:element name="MeasurementType" type="D2LogicalModel:MeasurementType"
                        minOccurs="1" maxOccurs="1"/>
    <xs:element name="SensorType" type="D2LogicalModel:SensorType"
                        minOccurs="1" maxOccurs="1"/>
    <xs:element name="Location" type="D2LogicalModel:ReductionLocation"
                        minOccurs="1" maxOccurs="UNBOUND"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="ReductionInformationReply">
  <xs:annotation>
    <xs:documentation>Acknowledge reply to the ReducttionInformation
                              message</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="Acknowledge" type="D2LogicalModel:Boolean"
                        minOccurs="1" maxOccurs="1"/>
    <xs:element name="OrgSequenceNr" type="D2LogicalModel:Integer"
                        minOccurs="1" maxOccurs="1"/>
    <xs:element name="Reason" type="D2LogicalModel:string"
                        minOccurs="0" maxOccurs="1"/>
    <xs:element name="PolutionMeasurement" type="D2LogicalModel:DateTime"
                        minOccurs="0" maxOccurs="1"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="ReductionPointCoordinates">
```

```
  <xs:annotation>
   <xs:documentation>A pair of coordinates defining the geodetic position of a single
                    point using the European Terrestrial Reference System 1989
                    (ETRS89).</xs:documentation>
  </xs:annotation>
  <xs:sequence>
   <xs:element name="latitude" type="D2LogicalModel:Float" minOccurs="1" maxOccurs="1">
    <xs:annotation>
     <xs:documentation>Latitude in decimal degrees using the European Terrestrial
                       Reference System 1989 (ETRS89).</xs:documentation>
    </xs:annotation>
   </xs:element>
   <xs:element name="longitude" type="D2LogicalModel:Float" minOccurs="1" maxOccurs="1">
    <xs:annotation>
     <xs:documentation>Longitude in decimal degrees using the European Terrestrial
                       Reference System 1989 (ETRS89).</xs:documentation>
    </xs:annotation>
   </xs:element>
   <xs:element name="Altitude" type="D2LogicalModel:Float" minOccurs="0" maxOccurs="1">
    <xs:annotation>
     <xs:documentation>Altitude is the Z-Value of the coordinate, the position in meters
                       of the point above sea level.</xs:documentation>
    </xs:annotation>
   </xs:element>
  </xs:sequence>
</xs:complexType>


<xs:complexType name="ReductionLocation">
  <xs:annotation>
```

```
    <xs:documentation>Location of a measurement device, or measurement route, with
                                        measurements attached to it</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="LocationId" type="D2LogicalModel:String"
                              minOccurs="1" maxOccurs="1"/>
    <xs:element name="TrajectDistance" type="D2LogicalModel:Float"
                              minOccurs="1" maxOccurs="1"/>
    <xs:element name="LocationForDisplay" type="D2LogicalModel:ReductionPointCoordinates"
                              minOccurs="1" maxOccurs="UNBOUND"/>
    <xs:element name="GeneralComment" type="D2LogicalModel:Comment"
                              minOccurs="0" maxOccurs="UNBOUND"/>
    <xs:element name="MeasuredValue" type="D2LogicalModel:ReductionMeasuredValue"
                              minOccurs="1" maxOccurs="UNBOUND"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="ReductionMeasuredValue">
  <xs:annotation>
    <xs:documentation>Single measurement or single calculation of CO2</xs:documentation>
  </xs:annotation>
  <xs:sequence>
    <xs:element name="CO2Emission" type="D2LogicalModel:Float"
                              minOccurs="0" maxOccurs="1"/>
    <xs:element name="FuelConsumption" type="D2LogicalModel:Float"
                              minOccurs="0" maxOccurs="1"/>
    <xs:element name="Velocity" type="D2LogicalModel:Float"
                              minOccurs="0" maxOccurs="1"/>
    <xs:element name="TravelTime" type="D2LogicalModel:Float"
                              minOccurs="0" maxOccurs="1"/>
```

```
    <xs:element name="CO2Exhaust" type="D2LogicalModel:Float"
                         minOccurs="0" maxOccurs="1"/>
    <xs:element name="VehicleType" type="D2LogicalModel:VehicleType"
                         minOccurs="0" maxOccurs="1"/>
    <xs:element name="MeasurementStartTime" type="D2LogicalModel:DateTime"
                         minOccurs="1" maxOccurs="1"/>
    <xs:element name="MeasurementEndTime" type="D2LogicalModel:DateTime"
                         minOccurs="0" maxOccurs="1"/>
  </xs:sequence>
    </xs:complexType>

<xs:simpleType name="SensorTypeEnum">
  <xs:annotation>
    <xs:documentation>If the measurement is based on a calculation or a measurement with a
                               device</xs:documentation>
  </xs:annotation>
  <xs:restriction base="xs:string">
    <xs:enumeration value="CO2Sensor"/>
    <xs:enumeration value="CalculatedCO2"/>
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name="String">
  <xs:annotation>
    <xs:documentation>A character string whose value space is the set of finite-length
                               sequences of characters. Every character has a corresponding Universal
                               Character Set code point (as defined in ISO/IEC 10646), which is an
                               integer.</xs:documentation>
  </xs:annotation>
  <xs:restriction base="xs:string">
```

```
    <xs:maxLength value="1024"/>
  </xs:restriction>
</xs:simpleType>

<xs:complexType name="SupplierIdentifierType">
  <xs:complexContent>
    <xs:annotation>
      <xs:documentation>Indentification of the manufacturer of the measurement data,
                                        country, name and language</xs:documentation>
    </xs:annotation>
    <xs:sequence>
      <xs:element name="Country" type="D2LogicalModel:CountryEnum" minOccurs="1"
                                maxOccurs="1"/>
      <xs:element name="ManufacturerIdentifier" type="D2LogicalModel:String"
                                minOccurs="1" maxOccurs="1"/>
      <xs:element name="Language" type="D2LogicalModel:Language"
                                minOccurs="1" maxOccurs="1"/>
    </xs:sequence>
  </xs:complexContent>
</xs:complexType>


<xs:simpleType name="VehicleTypeEnum">
  <xs:annotation>
    <xs:documentation>Types of vehicle.</xs:documentation>
  </xs:annotation>
  <xs:restriction base="xs:string">
    <xs:enumeration value="agriculturalVehicle">
      <xs:annotation>
        <xs:documentation>Vehicle normally used for agricultural purposes, e.g. Tractor,
```

```
combined harvester etc.</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="anyVehicle">
  <xs:annotation><xs:documentation>Vehicle of any type.</xs:documentation></xs:annotation>
</xs:enumeration>
<xs:enumeration value="articulatedVehicle">
  <xs:annotation><xs:documentation>Articulated vehicle.</xs:documentation></xs:annotation>
</xs:enumeration>
<xs:enumeration value="bicycle">
  <xs:annotation><xs:documentation>Bicycle.</xs:documentation></xs:annotation>
</xs:enumeration>
<xs:enumeration value="bus">
  <xs:annotation><xs:documentation>Bus.</xs:documentation></xs:annotation>
</xs:enumeration>
<xs:enumeration value="car">
  <xs:annotation><xs:documentation>Car.</xs:documentation></xs:annotation>
</xs:enumeration>
<xs:enumeration value="caravan">
  <xs:annotation><xs:documentation>Caravan.</xs:documentation></xs:annotation>
</xs:enumeration>
<xs:enumeration value="carOrLightVehicle">
  <xs:annotation>
    <xs:documentation>Car or light vehicle.</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="carWithCaravan">
  <xs:annotation>
    <xs:documentation>Car towing a caravan.</xs:documentation>
  </xs:annotation>
```

```
</xs:enumeration>
<xs:enumeration value="carWithTrailer">
  <xs:annotation>
    <xs:documentation>Car towing a trailer.</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="constructionOrMaintenanceVehicle">
  <xs:annotation>
    <xs:documentation>Vehicle normally used for construction or maintenance purposes, e.g.
                        digger, excavator, bulldozer, lorry mounted crane
                        etc.</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="fourWheelDrive">
  <xs:annotation>
    <xs:documentation>Four wheel drive vehicle.</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="highSidedVehicle">
  <xs:annotation>
    <xs:documentation>High sided vehicle.</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="lorry">
  <xs:annotation>
    <xs:documentation>Lorry of any type.</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="moped">
  <xs:annotation>
```

```
    <xs:documentation>Moped (a two wheeled motor vehicle characterised by a small engine
                                  typically less than 50cc and by normally having
                                  pedals).</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="motorcycle">
  <xs:annotation>
    <xs:documentation>Motorcycle.</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="motorcycleWithSideCar">
  <xs:annotation>
    <xs:documentation>Three wheeled vehicle comprising a motorcycle with an attached side
                                  car.</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="motorscooter">
  <xs:annotation>
    <xs:documentation>Motorscooter (a two wheeled motor vehicle characterised by a step-
                                 through frame and small diameter wheels).</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="tanker">
  <xs:annotation>
    <xs:documentation>Vehicle with large tank for carrying bulk liquids</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="threeWheeledVehicle">
  <xs:annotation>
    <xs:documentation>Three wheeled vehicle of unspecified type.</xs:documentation>
```

```
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="trailer">
 <xs:annotation>
   <xs:documentation>Trailer.</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="tram">
  <xs:annotation>
   <xs:documentation>Tram.</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="twoWheeledVehicle">
  <xs:annotation>
   <xs:documentation>Two wheeled vehicle of unspecified type.</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="van">
  <xs:annotation>
   <xs:documentation>Van.</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="vehicleWithCatalyticConverter">
  <xs:annotation>
   <xs:documentation>Vehicle with catalytic converter.</xs:documentation>
  </xs:annotation>
</xs:enumeration>
<xs:enumeration value="vehicleWithoutCatalyticConverter">
  <xs:annotation>
   <xs:documentation>Vehicle without catalytic converter.</xs:documentation>
```

```
      </xs:annotation>
    </xs:enumeration>
    <xs:enumeration value="vehicleWithCaravan">
      <xs:annotation>
        <xs:documentation>Vehicle (of unspecified type) towing a caravan.</xs:documentation>
      </xs:annotation>
    </xs:enumeration>
    <xs:enumeration value="vehicleWithTrailer">
      <xs:annotation>
        <xs:documentation>Vehicle (of unspecified type) towing a trailer.</xs:documentation>
      </xs:annotation>
    </xs:enumeration>
    <xs:enumeration value="withEvenNumberedRegistrationPlates">
      <xs:annotation>
        <xs:documentation>Vehicle with even numbered registration plate.</xs:documentation>
      </xs:annotation>
    </xs:enumeration>
    <xs:enumeration value="withOddNumberedRegistrationPlates">
      <xs:annotation>
        <xs:documentation>Vehicle with odd numbered registration plate.</xs:documentation>
      </xs:annotation>
    </xs:enumeration>
    <xs:enumeration value="other">
      <xs:annotation>
        <xs:documentation>Other than as defined in this enumeration.</xs:documentation>
      </xs:annotation>
    </xs:enumeration>
  </xs:restriction>
</xs:simpleType>
</xs:schema>
```