COSMOS

Cultivate resilient smart Objects for Sustainable city application

Grant Agreement Nº 609043

D7.5.3 Smart events and protocols for smart public transport (Year 3 Functionality)

WP7: Use cases Adaptation, Integration and Experimentation

Version: 1.0
Due Date: 30/06/2016
Delivery Date: 30/06/2016
Nature: Report
Dissemination Level: Public

Lead partner: Empresa Municipal de Transportes de Madrid S.A. (EMT)
Authors: Sergio Fernández Balaguer (EMT)
Andrés Recio (EMT)
Iván Ledesma (Madrid City Council)
Internal reviewers: Paula Ta-Shma (IBM)
François Carrez (Surrey Univ.)
Juan Sancho (ATOS)

www.iot-cosmos.eu

The research leading to these results has received funding from the European Community's Seventh Framework Programme under grant agreement n° 609043

Version Control:

Version	Date	Author	Author's Organization	Changes
0.1	01/06/2016	Sergio Fernández Andrés Recio	EMT	First updates to the D7.5.2 (Y2), creating the initial version of D7.5.3 (Y3)
0.2	07/06/2016	Sergio Fernández Andrés Recio	EMT	Second version
0.3	20/06/2016	Sergio Fernández Andrés Recio	EMT	Third version
0.4	21/06/2016	Iván Ledesma	Madrid City Council	Updates on M-30 traffic management data info
0.5	23/06/2016	Sergio Fernández Andrés Recio	EMT	New contents on evaluation
0.6	24/06/2016	Sergio Fernández Andrés Recio	EMT	Final draft for internal review
0.7	27/06/2016	Juan Sancho	ATOS	Review Document
0.8	27/06/2016	Paula Ta-Shma	IBM	Review Document
1.0	28/06/2016	Sergio Fernández Andrés Recio	EMT	Final version

Table of Contents

Table of Figures

Acronyms

Acronym	Meaning
AENOR	Spanish Association for Standardization and Certification
AMQP	Advanced Message Queuing Protocol
CEP	Complex Event Processing
DCP	Dynamic Control Process
DDBB	Data bases
DDP	Distributed Data Protocol.
ICT	Information and Communications Technology
ITS	Intelligent Transportation Systems
JSON	JavaScript Object Notation
JSON-LD	JavaScript Object Notation Linked
MLM	MobilityLabs Madrid
NoSQL	Not only SQL
RB	Reactive Box
RDF	Resource Description Framework
SQL	Structured Query Language
TCP	Transmission Control Protocol
UC	Use-Case
UDP	User Datagram Protocol
UNE	A Spanish Standard
VE	Virtual Entity
VEProt	Virtual Entity Process for Reactive and Ontological Things
WGS84	World Geodetic System 1984

1 Introduction

As stated in the last D7.5.1 and D7.5.2, the development of intelligent systems oriented to Smart Cities requires coordination and the sum of synergies which, slowly, could lead to the expected changes. A clear example is the development of specialized sensors by companies and industries that have emerged in the wake of the niche market. This would not have been possible without setting up intelligent elements that offer business opportunities for those companies. In the meantime, these mechanisms and sensors have enabled smart cities to have physical and virtual elements on which to rely. Thus, the slow and constant work on this regard has meant to a mutual enrichment for sides, public and private sectors.

To achieve this, it has been important the investment in R&D of many entrepreneurs and managers who are anticipating the future, but also the work of theoretical researchers, docents and universities which have established guidelines and protocols for the virtualization of cities avoiding becoming chaotic. In addition to that, the role of cities, becoming city labs where to test and incorporate real projects has made possible the knowledge of the requested infrastructure to keep on working on this field.

As aforementioned before, within the objectives of COSMOS the aim is to follow the same approach, which equivalence has been expressed in D7.5.2 (the document addressing the Y2 achievements) and the current D7.5.3 deliverable for Y3, including further achievements with an additional UC approach on Madrid traffic management data for the M30 (first ring of the city) as well as exploring synergies with other European projects, such as INLIFE. This development model starts from classical systems for integrating transport data events towards a model of virtualization of such events, following guidelines for standardization, orchestration of services and public open data policies, designing protocols to be used in the exchange of information by citizens, businesses and institutions, even beyond the model of public transportation. That will be the contribution of COSMOS within the research field of Smart Cities and, more specifically, within the concept of smart transport. The transport oriented protocols which are being defined under the framework of COSMOS are the basement of the system predictive models deployed so far.

More in detail, the current document includes, in addition to the revision of the pre-existing sections, some new ones such as section 6 (Madrid UC addition: M-30 traffic management data) and section 9 (Evaluation), and also some new subsections were added regarding Y3 achievements (subsections 2.2, 5.2, 7.3, 7.5, 7.7, 8.1.2 and 8.2.7).

In summary, during Y3, different technological challenges that had been developed over the previous two years are consolidated as a real technological architecture for the city, as will be seen herein. To do this, not only efforts have been made on the part of standardization and the making of theoretical designs to build models of Smart Transport beyond the use case, but have been embodied in a real and operating platform for the city.

2 Smart Labs and research projects

Multiple projects worldwide are serving for the development of information technologies aimed at creating ecosystems that allow learning how to use Information and Communications Technology (ICT) and Intelligent Transportation Systems (ITS) in order to make smarter and more sustainable cities. The R&D efforts focus on several strategies, including:

1. Sensing systems: there are new business opportunities under multiple pathways thanks to new adapted microcontrollers oriented to signal acquisition and actuators as well as thanks to the expansion of reduced cost sensors together by an increased supply promoted by an industry in full expansion. It is therefore essential to have laboratories that can quantify and qualify the different technologies that are appearing. In Spain, scenarios such as Malaga, Zaragoza and Santander, to give some examples, provide unique opportunities for research in the field of intelligent sensing and intelligent control of objects, mainly within the fields of energy and communication. In these cities, large companies have deployed laboratories in large areas of the city including data acquisition and control of different types of elements of the city, always talking under the framework of smart labs.
2. Event management and information systems in real time: the construction of platforms oriented to the analysis of events such as queue managers and other information models based on the observation of changes, provide trained technologies for the exchange of status information among sensors. Currently, multiple commercial technologies enable the deployment of event-driven systems.
3. Storage capacity and Big Data: new storage systems, based on schemas and objects, known as No-SQL databases, are allowing scaling data solutions covering cloud infrastructures, from which status can be stored in order to apply methods of learning and experience through the use of data and the relationship between them.
4. Definition of schemes and digital information entities: multiple standards allow machines to know about entities and objects, by its ontology and semantics, providing structured repositories and methods of consultation and access to information enabling intelligent definition methods of any element of the city, especially affecting sensors and actuators.

All these elements together and in combination are being experimented in different city labs, producing a paradigm shift in the new world that is being built, which is a combination between the classic concept of a city and the integration of elements that improve its energy efficiency, mobility and accessibility, among others. In summary, Smart Cities are emerging thanks to technology and the construction of living labs in cities, and this is the approach that COSMOS is facilitating in the Madrid use case.

2.1 Smart Labs and research projects in Y2

As indicated in previous deliverables, during Y2, with the support and under the auspices of COSMOS project, the participant partners considered the possibility of making in Madrid the development of an intelligent laboratory, not only from an experimental point of view but also from a practical one. Therefore, and thanks to EMT initiative, a laboratory was deployed to serve as a testing place to ongoing projects by providing feedback coming from user experience. This laboratory has led to the complete definition (and not only under theoretical and functional aspects) of the protocol for the exchange of events oriented to smart mobility. In addition to that this laboratory has been conceived as a tool for making real analysis of events, therefore it includes real information about public transportation in Madrid, including real time data, such as the position of the buses or expected arrival times to bus stops.

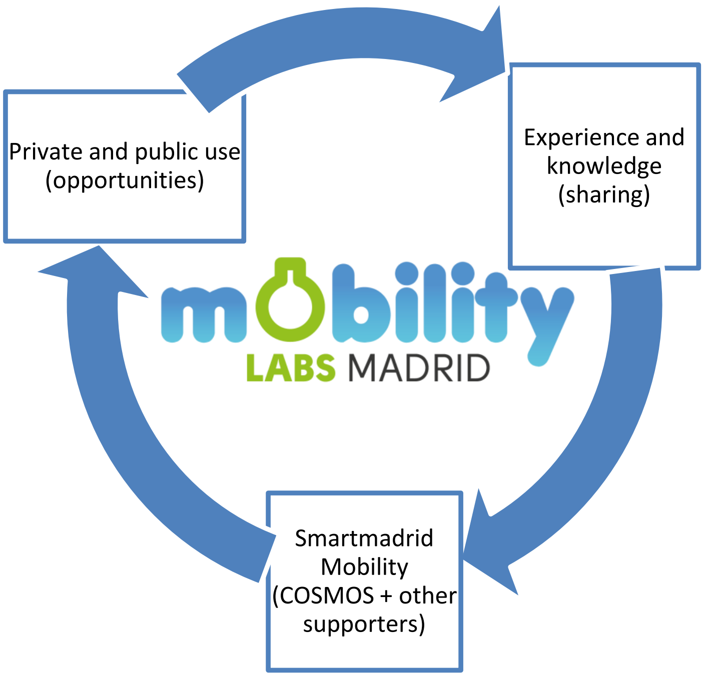


Figure 1: Madrid mobility lab cycle

This laboratory has enabled, among others:

1. Obtaining knowledge and experience of existing semantic models, as well as tool learning such as “Protége” and gain experience in IoT oriented schemes such as Semantic Sensor Network.
2. Providing an experimentation environment to universities and companies in order to investigate, learn or develop new products.
3. Creating post-graduate courses at universities offering as a learning and practical field the laboratory itself.
4. Giving talks and disseminating COSMOS project in multiple environments, events and forums.

2.2 Smart Labs and research projects in Y3

As indicated before, every aspect developed in Y2 about fixed infrastructure planning for Madrid UC are reflected in Y3 by setting a real platform specially oriented to citizens, which is also free and for open use. This platform supports a bidirectional data flow exchange mechanism, and therefore, any integrator is able to:

- Generate its own Virtual Entities, in public or private mode, or even allow other systems to update and add information in them.
- Get information about the massive historical data existent in the system.
- Get information about the entities and services in the lab.

For that purpose, two different types of infrastructure have been built:

- Processing and information storage infrastructure: This involves the installation of powerful servers visible via Internet. Through these servers, system users can exchange information about their virtual entities with the Reactive Box system. These servers are the following:

RB mobility LABS MADRID	`http://rbmobility.emtmadrid.es:3333` This server is used for observation of phenomena and information changes in real-time, that is, at the instant that data enters the system or changes its value. You can find information about their characteristics and examples of use of the infrastructure in: https://mobilitylab.emtmadrid.es/portal/index.php/reactive-box-de-mobilitylabs/
RabbitMQ	`amqp.emtmadrid.es:5672` This server is responsible for the acceptance and management of the different events coming from the clients who are integrated in the system.
myBus	`https://mybus.emtmadrid.es:8073` Through this server it is possible to access, in real time, to the activity of any EMT bus
RBdata	`https://rbdata.emtmadrid.es` Through the REST API of this server it is possible to retrieve historical data of the accumulated data collections

- Infrastructure for dissemination of information: It consists of a portal that provides information about the different storage subsystems that can interact with COSMOS. The portal contains examples of source code and instructions for the use of each component and the list of virtual entities it contains. The URL is: https://mobilitylabs.emtmadrid.es

As a summary, a complete system has been set up for the definition and storage of Virtual Entities, specially designed for COSMOS interoperability, as it is shown by the different Madrid UC scenarios.

3 The commercial approach in the Smart Cities environment

One aspect to consider in the current deployment of Smart Cities is the possibility that large companies and corporations literally take technological control of cities. This is mainly due to their privileged position in industry and research sectors, which can involve, in many cases, the deployment of various components (sensors, devices, event management systems, etc.) based on proprietary protocols and proprietary encryption systems that prevent a democratic growth approach of infrastructure. The risk is particularly high in models of municipal concessions, in which the exploitation and management of the various systems and resources of the city are delivered during a certain number of years by companies that are responsible for its management. These companies can, in turn, install sensing and control systems based on closed protocols and infrastructure, aimed basically to control their range of management and to ensure that the operation is performed in the best economic and energy efficiency conditions.

However, although the purpose for these action plans is legitimate and belongs to the normal operating field, its design has several disadvantages and risks, including:

- Prevents the constant technological adaptation: as it is based on a closed model and usually supported only in the field of the deployed system itself.
- Produces opacity phenomena in the information: as citizens and other companies have not the means to interact with the smart city infrastructure.
- It may block business models based on local industry: as local industry might have a lack of integration possibilities for expanded uses of the network of sensors or actuators.

4 Smart Transport in the field of Smart Cities

One of the objectives pursued when advancing in the implementation of models of Smart Cities is an optimized transport from both the functional and economic point of view, including (especially when talking about public sector):

- Energy efficiency for cost savings. More energy efficient vehicles pose a significant reduction in costs, since fuel and energy is one of the biggest expenses in a public transport company. Vehicles with improved aerodynamics and propulsion systems with higher performance are part of the core of the research activities of vehicle manufacturers.
- Reducing emissions to achieve cleaner cities. Using cleaner fossil fuels and new technologies (catalysts, electric vehicles, hybrid vehicles with self-production energy, etc.)
- Optimization of routes and services. Inefficient or costly routes, obstacles on public roads, delays by blockages or slow traffic are causes of poor or inefficient service level. Better services can be achieved through optimization algorithms and processes, as well as applying greater control over the regulation of the fleet. Applying new regulatory measures and policies, such as dedicated lanes for public transport, increase the efficiency of the service.
- Quick resolution of incidents to avoid possible downtime. The public transport is subject to various endogenous and exogenous factors that will prevent planning to be accomplished. The affectation of an incident will largely depend on the knowledge of the events by the managers and the ability of a quick decision making transfer.
- Management of smart ticketing, with the aim of improving co and inter-modality. One factor that highly affects the efficient use of public transport is the creation of integrated tariff systems. Furthermore, the creation of registration systems for selling and accessing to public transport guarantee the control of demand, promoting management and spending management policies based on real data. The use of large areas of multimodal exchange stations and the analysis of origin-destination matrices coupled with fast access systems and contactless technologies for accessing or paying ensure comfortable and economical use of public transport.
- Instant detection of incidences. The use of sensors and alarms (detectors) in public transport vehicles ensure the anticipation of an event with major consequences. To this end, industrial sector is increasingly introducing elements for early detection, including Machine Learning processes as predicting issues before they actually occur is also very important.
- Integration with other systems of the city. Public transport is not an isolated element of the city, and its optimum performance largely depends on its links with other city infrastructures. Therefore it is necessary to foster joining relations between different infrastructures and systems creating supra-intelligent mechanisms. Some examples are the prioritization for buses at traffic lights or the coordination with security and emergency systems.
- Customer Information Systems. Ensuring that the public transport user has reliable information that allows instantaneous trip planning or, to take instantaneous decisions if the usual trip option is failing.
- Accessibility. Public transport should not be effective only with a majority of citizens but to all citizens. For this reason, it must provide means of access and service knowledge using channels and mechanisms accessible for all.

To achieve the aforementioned objectives, technologies and information systems pose one of the greatest challenges and, in turn, one of the main tools in the field of transportation, and ITS and ICT initiatives must be efficient and be based in technological models that guarantee the points described above.

5 Smart Cities Public models vs. Private models

Smart Cities offer a new range of niches in which private exploitation system models of transportation, energy and other sensorial elements are not the only viable infrastructure, as are not either those exclusively municipal or public. On the contrary, mixed and juxtaposed models between private and public systems can be a way to ensure continuity, scaling and co-participation in building smarter cities.

The proposed COSMOS model is intended to provide a possible architecture equivalent to those which have been the classical ones that have come to be the benchmark in the construction of cities, that is: public bodies concerned and in charge of ensuring the transmission and exchange of information from which any organism, company or citizen can create their business models or research activities. The proposed model is based on standardization, certification and co-participation of systems emulating the traditional functioning of cities, since this model is adapting naturally to the ecosystem of human societies.

5.1 Data model dissemination during Y2

During Y2, an important effort has been put into the dissemination of the data model oriented to mobility events, mainly through two different strategies:

1. Introducing COSMOS model, thanks to several workshops, to different departments of Madrid City Council, explaining clearly the benefits and projection of using such model for the welfare of the city. Two of the aforementioned workshops were done at the highest level, to top managers and heads of the departments of IT and Traffic and Energy Management control centers of the city.
2. Publicly publishing the model in Github and publicly disseminating the repositories and the available information, together with examples about the system functioning code to capture transport and mobility events.

5.2 Data model dissemination during Y3

In addition to that, during Y3, Madrid partners, in cooperation with COSMOS partners and other external ones (Polytechnic University of Madrid, MediaLab Prado, ESRI, and MongoDB, among others) have worked hard in preparing a hackathon in order to present this new Opendata approach. With this vocation, the contest is intended for developers and data journalists, either individually or in teams, who want to work with free data or are looking for new development models, no matter whether they are students, professionals, or business startups.

http://mobilitylab.emtmadrid.es/portal/index.php/hackaton-2016/

Planning a hackathon and its dissemination through networks and specialized forums created prior knowledge of the project as a whole. In addition, it has allowed building a system of subscriptions for new users to build their own virtual entities due to its dynamic and adaptive construction.

The definition and design of data models is documented in public mode, defined by large groups. For example, the design of entities to manage Route monitoring by supervisors and the exchange messaging with the users' devices (Transport Use Case) is documented in detail on the dissemination website:

http://mobilitylab.emtmadrid.es/portal/index.php/data-model-for-route-planner/

Data Model for Route Planner

Definition of main entities

ROUTESMAD.usrtrack

This collection specifies the points for the realization of a user tracking by the route.

Name	Type	Description
layer	String	Mongo collection's name where the data is stored
idRoute	String	Unique identifier formed according to the structure of userPlan
idSesion	String	Unique identifier created for each session
nameRouteUser	String	Name of the creator of the route
dayWeek	String	Days of week to track the route. Possible values: L, M, X, J, V, S, D
dayType	String	Day's Types to track the route. Possible values: FE, LA, SA

Figure 2: Data model for route planner-Screenshot of the dissemination website information

The ultimate goal is to get a complete catalog of each of the entities that will progressively and evolutionarily take part in the MobilityLabs design and from which COSMOS will feed through their integration in native mode.

Currently, the platform has an extensive catalog of data covering many aspects of the city of Madrid (see catalog on this link: http://mobilitylab.emtmadrid.es/portal/index.php/projects-page/). This catalog has two main areas, a real-time one and a mass storage one (Big Data). This information can be read through the platform APIs. Similarly, there are APIs that allow the introduction of new data by the contestants.

An initial workshop session about the hackathon is scheduled for July 11th 2016, in order to launch it and present the basic information, and of course, to answer any questions that may arise during the nearly two months of competition.

6 Madrid UC addition: M-30 traffic management data

During Y3, an additional approach or scenario has been added in order to better address Madrid UC. This complementary component is based on the traffic management data of the primary ring of the city, the M-30 road, which has enormous influence on the overall traffic of the city.

Traffic in big cities affects important aspects of quality of life of its inhabitants and visitors, pollution and noise levels, commuting, road safety and many other aspects that affect the perception of citizens about a city.

Public bodies responsible for traffic management in large cities such as Madrid must have instruments to detect abnormal traffic conditions so that they can take immediate corrective action to mitigate the traffic problem and inconvenience to citizens. Rather than a reactive approach, a predictive one would be preferable, because actions could be taken to prevent traffic congestions and avoid CO2 and pollution emissions and wasted time.

Madrid City Council (hereafter MCC) owns and manages a large network of sensors (hereafter traffic detectors) scattered around the city that provide traffic data that enables traffic control systems to provide near real time tuning of traffic lights intervals, phases and offsets. This applies to urban roads where traffic lights are installed (urban traffic).

As well as urban roads, MCC is also responsible for an important urban ring road known as M-30; traffic conditions on this ring road highly determine traffic conditions in the rest of the city. This ring road bears an average daily traffic of 166.000 vehicles and some sections bear as many as 300.000 vehicles/day.

Most of the entrances to the city and many of the main avenues and streets of the city come or flow into the M-30, so that the traffic on the M-30 will greatly affect traffic throughout the city, as can be seen in the figure number 3, which shows the M-30 first ring which surrounds the central area of Madrid City, indicating the maximum speeds of each road segment.

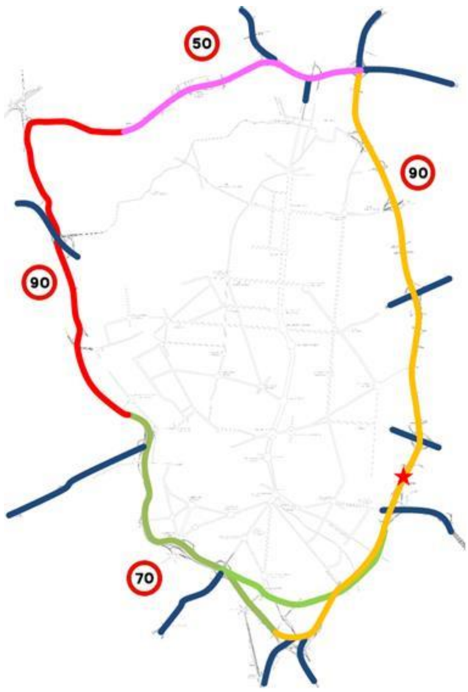


Figure 3: M-30 ring road maximum speeds

Only a small section of the ring is regulated by traffic lights and consequently less automated responses are currently possible to manage traffic in the ring as opposed to urban traffic.

M-30 traffic is monitored 24 hours a day from Madrid Mobility Management Centre control room; control room operators supervise traffic monitoring cameras, real time traffic data from traffic detectors and information provided by local police on site.

In the event of accidents, incidents, unusually heavy traffic or other unusual traffic conditions the traffic monitoring personnel would take the following actions to ease the situation:

- Inform drivers by means of message panels recommending caution, alternative routes
- Inform drivers by means of a webpage, APP
- Inform national road authorities
- Send local police
- Send tow trucks

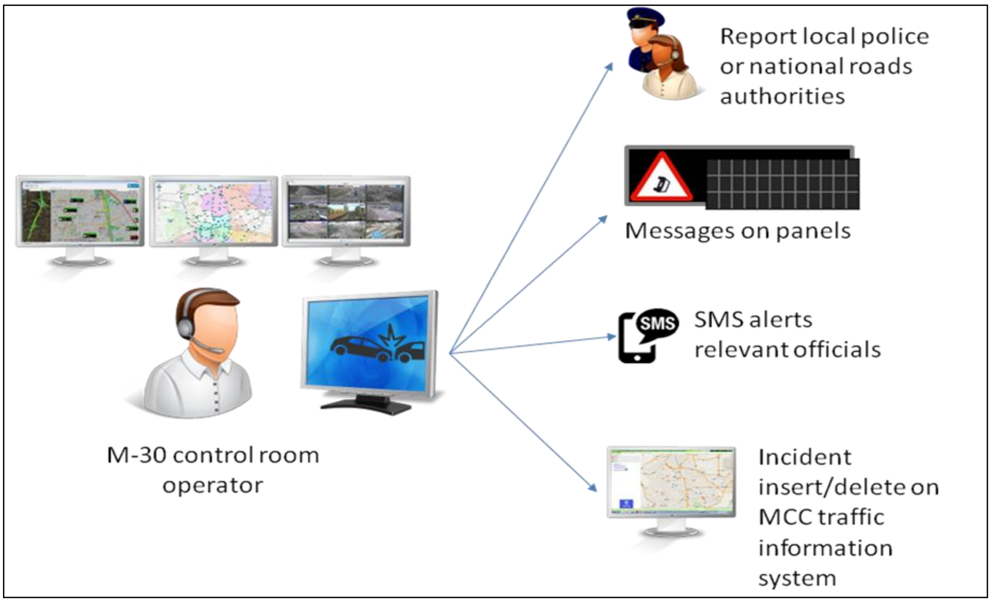


Figure 4: Actions in response to traffic conditions

Speeding these actions could avoid worsening of traffic conditions on M-30 and affecting either urban roads or motorways incoming to the city. So it would be useful if M-30 control room operators could receive alerts/alarms if traffic conditions differ from expected patterns, so that they could take quicker actions.

As mentioned above, measures taken would be much more effective if they were based on predictions rather than the current situation because traffic congestion could potentially be completely avoided.

Expected traffic patterns could be calculated from available historical M-30 traffic detectors data (data from January 2008 onwards is available and fields' definitions are provided in ANNEX 1). This information has been already provided to COSMOS developers and is now efficiently stored with Metadata Annotations in COSMOS Cloud Storage.

Alerts/alarms could be integrated either in a specific dashboard or in existing M-30 traffic control systems. COSMOS has integrated Alarm Notifications in a specific dashboard so that Traffic Controllers are able to provide feedback on those notifications.

Once traffic patterns are calculated, rules issuing alerts/alarms should be worked out, tested for a certain period in the Madrid Traffic Control Room and tuned. Thanks to the feedback provided by Traffic Operators during Y3 demos, the tuning step is now more efficient and valuable.

Some examples of patterns and rules that we are interested in:

- **Road section traffic volume pattern**

Historical data of traffic volume and speed of certain road sections containing two or more traffic detectors could be analysed to determine a traffic volume pattern.

If current traffic volume and speed are A % higher than expected for more than B consecutive minutes, a congested traffic alert for that section should be issued.

A and B should be fine-tuned in order to avoid too many alerts in the Madrid Traffic Control Room. Road sections of interest are provided in ANNEX 3; traffic detectors on access and transfer roads are not considered to define road sections of interest.

- **Prediction of road section traffic volume pattern**

Traffic Managers count with a gained knowledge and background experience that allow them to foresee future traffic status based on their own assumptions and paying attention to current city traffic status. Being aligned to the previous *Road section traffic volume pattern*, getting notifications in advance about future traffic conditions would be very useful and highly recommended. In addition, being able to toggle between different time intervals ahead of time in order to better understand the evolution of a predicted status would be also very interesting.

Other kinds of optional analysis could be worked out:

- **Typical travel time pattern**

Typical travel time between approaches and exits in the road could be determined; if current travel time is C % higher than expected for more than D consecutive minutes, issue congested traffic alert for that route.

C and D should be fine-tuned afterwards in Madrid Traffic Control Room in order to avoid too much alerts. Routes of interest are provided in ANNEX 3.

A travel time computation algorithm is provided in ANNEX 2.

- **Truck traffic pattern**

Historical data of truck traffic could be analysed to determine a truck traffic volume pattern; if current truck traffic is G % higher than expected an alert should be issued so that M-30 control room operator further analyses the situation.

7 COSMOS within the Smart Public Transport infrastructure. Standardization and protocols proposal

7.1 Virtual Entities associated or connected with public transport: schema and ontology definition

The definition of an intelligent transport model involves designing proper ontologies. Transport COSMOS model schemas and contents are being built on RDF and JSON-LD models. The definitions are proposed under European standards through the Spanish Association for Standardization and Certification, AENOR (CTN 178 AENOR group as defined in UNE 178301). The datagram message headers are being defined under schemes of context (@context) according the proposed standard by the JSON-LD community.

The Virtual Entities which are being defined under semantic schemes are:

- Line: Each journey trip made by the public transport service (bus line).
- Bus: Each of the transport units serving in a line (bus).
- Driver: Driver performing each specific service on the bus.
- Traveller: Bus user connected or integrated into the COSMOS infrastructure.
- Electronic Panel: Information device providing useful information to travellers about the service or incidents.
- Sensor: reading device or actuator connected to the transport infrastructure (bus or informative electronic panel).
- Event: Each of the messages containing information related to transportation. An event can be one bus passing by a bus stop, a traveller getting on the bus, an engine temperature alarm and, in general, any information generated in the system which gets integrated into the COSMOS infrastructure.

The semantic definition of virtual entities is available in open data format in the GITHUB web site:

https://github.com/madridopenlabmobility/MOBILITY-MADRID-virtual-entities

7.2 New Virtual Entities for Y2

During Y2 new important virtual entities have been aggregated:

- Bus Stop: It contains the information and attributes of each EMT bus stop
- Bus Events.: It collects each bus position every 20 seconds
- Bus Sensor: Specific bus sensors have been connected to COSMOS system in order to provide readings for the Machine Learning

7.3 New Virtual Entities for Y3

During Y3 the entities design has evolved towards an abstract and self-definable model. The purpose that has been pursued is not having pre-defined objects within the system so it can be built from scratch. That, in practice, has not been entirely possible, as in the model there are certain intrinsic structures to the architecture functioning, such as the Users and Layers entity (required for the Reactive subsystem). However, it has been fully achievable for events that are inserted into the system from various connected clients, since all events may optionally be accompanied by a schema definition that is related to a semantic model. The design of this semantic model is built according to the same rules as the rest of the structures that support or represent the virtual entities; therefore, the systems that create virtual entities and their values can be defined in other virtual entities that define the object.

An example can be seen in this definition of a virtual temperature sensor to be installed on a bicycle in a simulation of the hackathon. The "ontology" collection is defined as a virtual entity that defines virtual entities. The result of the VE BICISENS design would be:

```
{
"collection":"BICISENS",
"attributes":[
  {
    "Attribute": "layer",
    "Value": "temperature",
    "Property": "mobilitymadrid:layers",
    "URI": "http://mobilitylab.emtmadrid.es/def/layers#sensormad",
    "Description": "kind of layer into RB or Mobility Servers. Layer
ontology describes data domain Reactive Box and other elements of
this entity.  (___) is substituted by three digit whith the class of
sensor"
  },
  {
    "Attribute": "owner",
    "Value": "[value]",
    "Property": "schemas:Text",
    "URI": "https://health-lifesci.schema.org/landlord",
    "Description": "Creator of current item definition"
  },
  {
    "Attribute": "type",
    "Value": "[value]",
    "Property": "schemas:Text",
    "URI": "http://schema.org/Text",
    "Description": "Mode of sharing data (public shared or private)"
  },
  {
    "Attribute": "version",
    "Value": "[value]",
    "Property": "schemas:Text",
    "URI": "https://schema.org/version",
    "Description": "Version of current schema"
  },
  {
    "Attribute": "_id",
    "Value": "[value]",
    "Property": "Schema:text",
    "URI": "http://schema.org/Text",
    "Description": "Unique id of sensor into system"
  },
  {
    "Attribute": "instant",
    "Value": "[date time value]",
    "Property": "dcterms:date",
    "URI": "xsd:datetime",
    "Description": "Instant of data generation"
  },
  {
    "Attribute": "geometry",
    "Value": "GEOJSON",
    "Property": "point object",
    "URI": "http://geojson.org/geojson-spec.html#id2",
    "Description": "Geo-point of current position of sensor"
  },
```

```
  {
    "Attribute": "accuracy",
    "Value": "[value]",
    "Property": "schemas:Text",
    "URI": "https://schemas.org/Text",
    "Description": "Class or characteristics of value"
  },
  {
    "Attribute": "shape",
    "Value": "[value]",
    "Property": "structure object",
    "URI": "object",
    "Description": "for painting in map (icon size color)"
  },
  {
    "Attribute": "acelerometer",
    "Value": "value",
    "Property": "schemas:Number",
    "URI": "https://schemas.org/Number",
    "Description": "Section of route where bus is position in
current time"
  },
  {
    "Attribute": "battery",
    "Value": "[idSection]",
    "Property": "schemas:Number",
    "URI": "https://schemas.org/Number",
    "Description": "Section of route where bus is position in
current time"
  },
  {
    "Attribute": "socket",
    "Value": "[value]",
    "Property": "schemas:Text",
    "URI": "https://schemas.org/Text",
    "Description": "Name of device or sensor"
  },
  {
    "Attribute": "value",
    "Value": "[value]",
    "Property": "ssn:MeasurementCapability",
    "URI":
"http://purl.oclc.org/NET/ssnx/ssn#MeasurementCapability",
    "Description": "Characteristics of meassures and context
conditions"
  },
  {
    "Attribute": "state",
    "Value": "[value]",
    "Property": "structure object",
    "URI": "object",
    "Description": "for painting in map (popup container)"
  },
  {
    "Attribute": "busData",
    "Value": "[value]",
    "Property": "JSON structure",
    "URI": "JSON structure",
    "Description": "data of bus when the user device is linked to"
  }]
  }
```

By this design, it is possible to build multiple versions of each representation of each virtual entity assigning each model and each version an owner or designer of that specific item. This is especially important in the “shared” type virtual entities, because they involve multiple users or systems storing information with different authors and different structures.

7.4 Specialization of data model in the storage system during Y2

One of the most important topics to decide standardization and data storage structure is the criteria that will affect, undoubtedly, to the scaling and diversification potential of the structure and data that support it.

A key decision in the storage and management model of data of the different Madrid VE has been to establish the advantages and disadvantages of an unstructured model with respect to the traditional model that manages the transactional information of Madrid (unstructured DDBB vs. relational DDBB). This information, stored in large SQL and Oracle servers, conditions the growth and scalability of the model, so finally it has been decided to separate the traditional model from that one oriented to VE which is the one that feeds COSMOS.

The separation of models in the field of transport means a serious challenge in establishing systems for data synchronization, especially in real-time environments. For this purpose, any model based on relational systems must contain mechanisms for detecting changes in the database and must be able to be observed in order to be transferred and transformed into an unstructured storage. Only through a homogeneous and timely flow over time exchange mechanisms can be really applied. The clearest example of this calculation is to obtain the bus arrival time a certain bus stop. This data, maintained in the kernel memory of the fleet control system must be transformed and built into a new model based on observation and IoT oriented.

Another important issue is how the storage and information management system integrates within the platform that will provide service to COSMOS model and other IoT based models. Since the final decision of Madrid Local Cloud solution for COSMOS is based on the construction of the Meteor based integrator element, it was considered that the most suitable model was the NO-SQL database together with the inclusive feature of that platform to monitor changes based in MongoDB queues. Both facts have been key in taking the decision about which database to use.

7.5 Specialization of data model in the storage system during Y3

The model is self-definable and allows input of multiple versions from multiple users to a virtual entity scheme, as already mentioned previously. However, a model of internal structure has been built to manage both safety access rules and system.

We can therefore state that, although outside the scope of COSMOS project, a new security model for Transport, called SDA, has been developed and much of its design has been adapted to house or run through the Reactive Box input and output entities flow. That implies an additional security requirement in the COSMOS subsystem within the core of Madrid Virtual entities. The relationship scheme between entities of the Security Profile would be as follows:

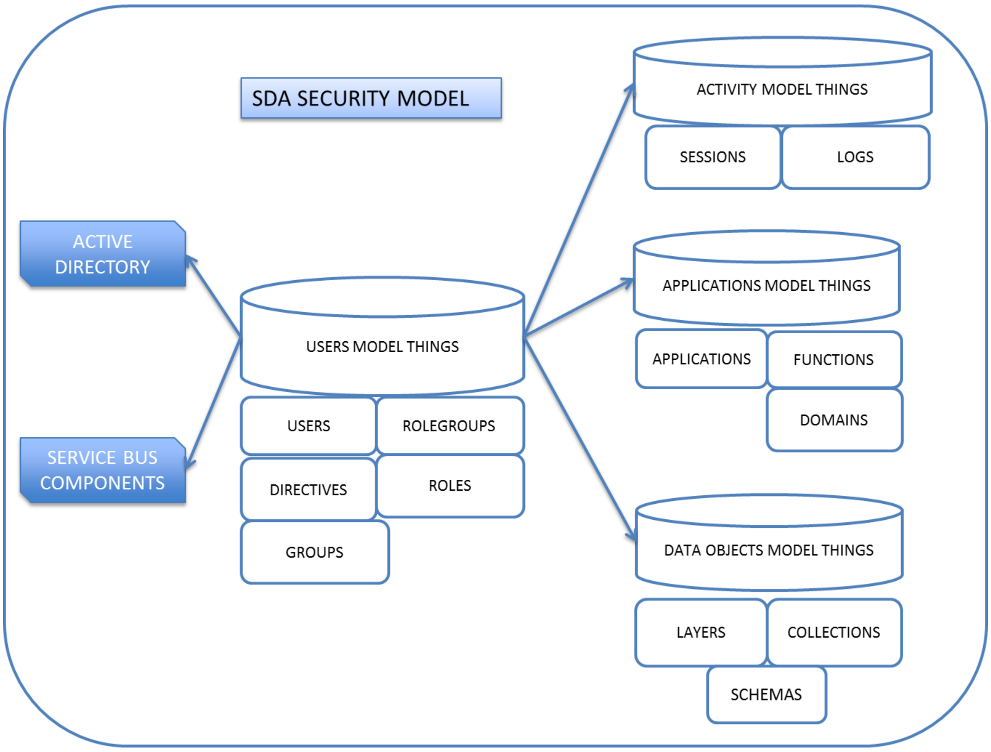


Figure 5: Security profile scheme

Within the SDA Security Model we can indicate the following most relevant schemes directly related to the COSMOS subsystem:

- Things Application Model: This model deals with defining functions and relationships with systems that use them. Every writing or reading rule of a VE is pre-defined in this model.
- Data Object Model Things. It deals with the dynamic catalog of collections that harbor VE schemes.
- Users Model Things: It defines the roles and behavior of the different system users, both referred to physical entities (person, driver, etc.) or machine entities (clients, connectors, subsystems, etc.). For this reason, the definition scope of the main entity (Users) contains relevant data indicating both access rules and human profiles, including health characteristics (healthPreferences) or disabilities (needAids), both required for the Madrid Transport UC. The scheme of VE Users would be the following:

Users Collection		
fieldName	**type**	**description**
_id	Key, String	Unique value of item (is the user id or login code)
firstName	String, mandatory	First user name
lastName	String, mandatory	Last user name
dateIni	Datetime, mandatory	Date of Creation in security system
dateEnd	Datetime, mandatory	Date end in security system (31/12/9999 is equal to infinite value)
dateLastUpdate	Datetime, mandatory	Date Last Updated of any data
dateLastLogin	Datetime, mandatory	Date last login
password	String, mandatory	Password encrypted value
timeRestrictions	Object	
daysAllow	Int	1 byte for representing days of week which the user can Access to system. The value representation is: 0111 1111. Bit less significative is Sunday and seventh bite is Saturday. 1 = allow Access 0.- deny Access if last bit is locked (= 1) everyday will be locked (full locked)
timeAccess	Object Array	Object to represent slot of times for Access or denies. Contains this format: {"timeAccess":[{“allow”:1:"from":"secs","to":"secs"},{“allow”:0,"from":"secs","to":"secs"}] }]}
phone	String, optional	Phone number
Email	String, mandatory	Mail
dateExpirePassw	Datetime, mandatory	Date of expiration password
idDirective	String, mandatory	Directive of user
idGroups	Object, optional	Array of ids of User groups
Picture	String, optional	url of photograph of user
needAids	Int, optional	Defines special needs using bits of one byte 1.- Visual disability 1.- Hearing disability 1.- Cognitive impairment 1.- infant 1.- ederly 1.- motion disability 1.- Wheel chair 1.- RFU
healthPreferences	Int, optional	Defines healthy preferences usings bits of one byte 1.- Pollen Allergy 1.- Respiratory disese 1.- Heart disese 1.- Skin disese 1.- RFU 1.-RFU 1.-RFU 1.-RFU
idDomains	Object, mandatory	Array of Domain which this users belongs to

Through this final design, the system is ready to be used under the orchestration and control of whom and how it access to the system. These usage rules should be covered by each and every one of the subsystems operating within the Madrid infrastructure, including COSMOS conception as a processing subsystem.

7.6 Big Data Model for COSMOS Machine Learning (new data model concept for Y2)

Within an intelligent transportation model, Big Data can be considered as one of the most important elements when analyzing the processing and storage of information architecture. Factors such as the system capacity, the strategy of accessing to information, the organization of data or the speed of obtaining information can pose serious problems when using the system. Within the scope of COSMOS and always thinking about Madrid scenario, and especially thinking about the specific Madrid UC, several analysis have been done until taking the decision on how the various structures on which store the information will be, always considering using the same database server model based in MongoDB. Among the most important considerations in shaping the form of collections and documents we can include:

1. Memory size of indexes in order to have at least five years storage of historic data, considering the segmentation in two data servers: the first server stores the last 12 months with a complete index for each collection in RAM memory, while a second server stores five years without any index in memory.
2. Accessing and data organization strategy:
    a. For models based on data collected by buses and managed with the shape of trips, JSON structures nested arrays are used, which principal hierarchy is the identifier of the service and its basic data: driver, vehicle, line, start, finish and other basic elements. Then there is a main array whose elements contain each of the trips made and their identification data, such as schedules, number, etc. Each of the trips, contain an array with each of the titles of transport used and the number of travelers obtained.
    b. For models based on data collected by the bus and managed as events, JSON structures separated by nature are used but grouped by a main document that contains the identifying data of the service, as specified in the previous point. Next, each object (JSON subdocument) contains an identification of the type of event that contains and associated data with a fully variable structure. Thus, there are step events by bus stop, or every time a passenger gets on the bus, or whenever there is an incident, a sensor reading, etc. All information is geo-referenced.
    c. Position event-based data obtained through the Reactive Box data are stored with the original format in which they were managed through that system. Special needs users tracking data belong to this category, as well as their check points, caregiver alarms and the historic data of each bus step by each bus stop.
3. Storage. In order to store the huge amount of data, there are special storage spaces within the Madrid storage lockers, always counting with local space, that is, without transferring to Clouds outside Madrid own infrastructure. Storage space has been reserved for five years, independently of COSMOS cloud model itself, since it is considered a part of the internal strategy of EMT for our own transport research activities.

7.7 Big Data Model for COSMOS Machine Learning (data model concept for Y3)

One of the Y3 challenges faced by the storage infrastructure is the ability to accommodate the different status that the COSMOS system handles, keeping in mind that it must be adaptive in all areas. Thus, during the design of Madrid Transport Use Case, many important aspects have been taken into account in the design of exchange capabilities between the CEP and the RB so all activity and their different status can be observed reactively; therefore, each event delivered by the CEP substitutes the previous one without deleting it.

Through this storage model, the design has evolved to contain two types of parallel schemes.

- Real Time Scheme: in which the latest data version of each represented virtual entity in the system is stored. Each time a new value arrives the previous data is replaced automatically.
- Historical Outline: is a consequence of the above, since the previous value is stored in a collection in which different status and actions taken on the VE can be consulted.

In summary, it can be considered that storage contains a double inlet: On Line and Big Data, as shown in the following diagram:

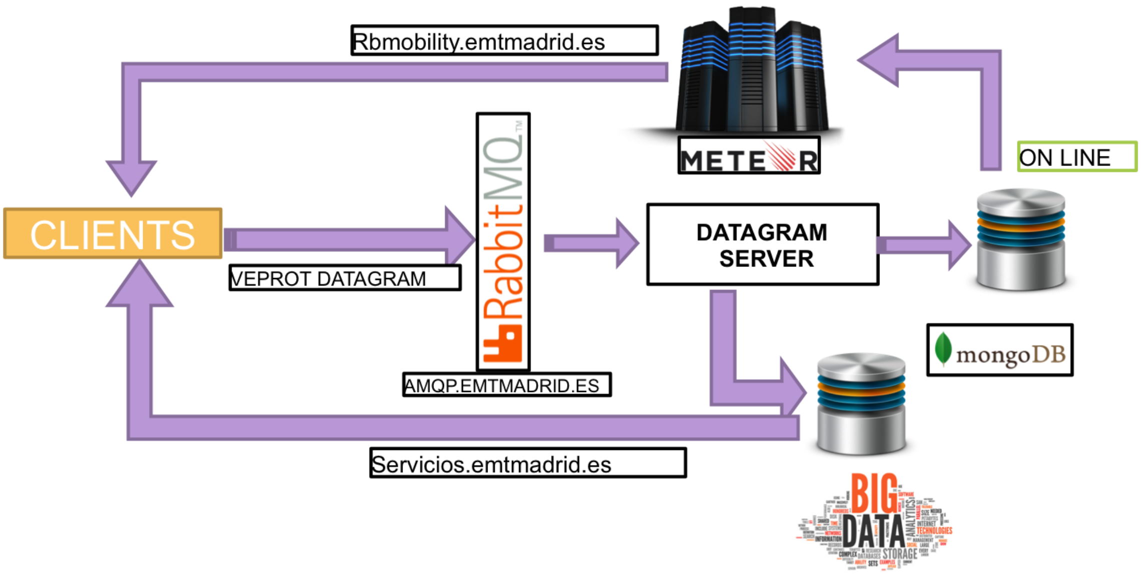


Figure 6: Data storage diagram

To achieve this, the DCP has included a decision mechanism that, from the information provided by COSMOS and other subsystems, performs actions on the Reactive Real Time information (On Line) and is stored in the moment into the historical model (concurrency). Actions designed for processing DCP model are as follows:

- INSERT: It builds data collection if it does not exist yet, and then it inserts the document into the collection with the key (_id). If the document defined with that specific _id already exists, it will ignore insertion. If inserted, it builds a parallel item with the key ObjectId into the historical model, and the value of _id will be stored in the "keyid" element.

- REPLACE: It builds the collection of data if it does not exist yet. If the document defined with that specific _id does not exist it will create it; in the event that it exists already it will replace its value with the new item defined by that _id. It builds a parallel item with key ObjectId into the historical model, and the value of _id will be stored in the "keyid" element.
- DELETE: Deletes the current value of the collection defined by that specific _id. If the operation is successful, it will store the item with the erase operation into the historical model.
- DROP: Eliminates the value defined by that specific _id if it exists within the collection; also removes all items with keyid equal to that _id from the historical model.

The result is the storage of information in MongoDB, on a massive scale (Big Data) along with the action that was performed on each item. An example can be seen in the VE that reports traffic predictions generated by COSMOS in certain measuring points of the city of Madrid. Briefly, removing objects from the VE scheme, we can see how the VE dedicated to the definition of predicting traffic, with id value PM1111, it is constructed by first time in the instant 2016-05-12 11:06:26.849000, once a “good traffic” informative message is generated.


```
{
     "_id" : ObjectId("57693fd524ca3c37d0b9a994"),
     "subsystem" : "PUTDATA",
     "function" : "REPLACE",
     "layer" : {
         "owner" : "EMT.SERVICIOS.TRAFFICMAD",
         "type" : "public",
         "name" : "TRAFFICMAD.alarms"
     },
     "levelAlarm" : "I",
     "instant" : "2016-05-12 11:06:26.849000",
     "codeAlarm" : "40",
     "geometry" : {
         "type" : "Point",
         "coordinates" : [
             -3.91630671505599,
             40.9111670272256
         ]
     },
     "textStatus" : "Good Traffic",
     "trafficState" : {
         "codeStation" : " PM1111",
         "speed" : "23",
         "intensity" : "425"
     },
     "system" : "LAYERS",
     "serviceLevel" : "0",
     "conditionAlarm" : {
         "factorSpeed" : "30",
         "factorIntensity" : "30",
         "meanSpeed" : "50",
         "meanIntensity" : "800"
     },
     "shape" : {
         "type" : "marker",
         "options" : {
             "shape" : "circle",
             "markerColor" : "blue",
             "prefix" : "fa",
             "icon" : "fa-car"
```

```
            }
        },
        "state" : {
            "instant" : "2015-11-30 12:38:35.775",
            "description" : "<b><p>Station: 15521</p><p>Good Traffic</p></b>   ",
            "value" : "1",
            "format" : "text"
        },
        "alertReceived" : "3",
        "keyid" : "PM1111"
  }
```


Subsequently, the same item undergoes a change at the instant 2016-05-12 11:06:27.701000


```
{
        "_id" : ObjectId("5769400a24ca3c37d0b9a995"),
        "subsystem" : "PUTDATA",
        "function" : "REPLACE",
        "layer" : {
            "owner" : "EMT.SERVICIOS.TRAFFICMAD",
            "type" : "public",
            "name" : "TRAFFICMAD.alarms"
        },
        "levelAlarm" : "W",
        "instant" : "2016-05-12 11:06:27. 701000",
        "codeAlarm" : "50",
        "geometry" : {
…
            ]
        },
        "textStatus" : "Bad Traffic",
        "trafficState" : {
            "codeStation" : "PM999",
            "speed" : "12",
            "intensity" : "823"
        },
```


The data is subsequently removed from the Real Time system (2016-05-12 11:09:25.246000) through a DELETE request. However, although the item disappears from the reactive system, it persists in the Big Data model storage.


```
{
        "_id" : ObjectId("5769410524ca3c37d0b9a997"),
        "subsystem" : "PUTDATA",
        "function" : "DELETE",
        "layer" : {
            "owner" : "EMT.SERVICIOS.TRAFFICMAD",
            "type" : "public",
            "name" : "TRAFFICMAD.alarms"
        },
        "levelAlarm" : "I",
        "instant" : "2016-05-12 11:09:25.246000",
        "codeAlarm" : "40",
        "geometry" : {
            "type" : "Point",

…..}
```

Finally, at 2016-05-14 12:12:21.435000 a new item is created within the collection with the same identifier that previously had, storing the new situation in the system


```
{
    "_id" : ObjectId("5769411924ca3c37d0b9a998"),
    "subsystem" : "PUTDATA",
    "function" : "INSERT",
    "layer" : {
        "owner" : "EMT.SERVICIOS.TRAFFICMAD",
        "type" : "public",
        "name" : "TRAFFICMAD.alarms"
    },
    "levelAlarm" : "I",
    "instant" : "2016-05-14 12:12:21.435000",
    "codeAlarm" : "40",
…
```


Through this mechanism, any entity and its events are stored with their different historical status within the Big Data system of the RB. Each collection of accumulated historical data is likely to be recovered through an REST API on the site https://rbdata.emtmadrid.es , providing the proper credentials. The recovery rules are:

- VE in public mode: Available to any user registered in read mode.
- VE in shared mode: Available to any registered user read and write mode.
- VE in private mode: Available only for the owner of the VE.

The recovery mechanism also allows extracting data by filtering, using BSON rules that allow selecting which criteria to use to obtain the information.

8 Transport oriented events server

The city information should flow through systems and nodal connections capable to ensure the exchange of useful information for citizens and businesses. Communication architecture should be based on open standards and be public infrastructure through which data can be consumed and reused openly (or at least, those data that are capable of being shared). Thus, the information could be exchanged in multiple modes, that is, from citizens and private sectors to municipal systems and vice versa.

In order to share this information within the COSMOS system, Madrid proposes a Local Cloud based on an event server (Reactive Box) that would act as the only point of connection and exchange of mobility oriented events. This event server must be:

- Adaptive: allowing the addition of new layers of information quickly and dynamically, compatible with dynamic information models based in objects or virtual entities
- Reactive: allowing subscription to objects or entities by clients and multiple systems.
- Safe: supporting models based on public information and private data.

The nodal server systems should be, undoubtedly, municipally controlled considering the approach of Smart Cities, and orchestration of services should correspond to infrastructures under the rule of public sector, beside the fact that private contents compliant with the protocol and standardization standards of the city could pass through it or be exchanged through it.

COSMOS proposed model for the intelligent management of public transport event, is based on JavaScript Meteor architecture, supported by Node.js. This infrastructure allows real-time client subscriptions to exchange standardized events information. The management of change for publishing queue results towards subscribers would be made by observing changes in a Mongo DB database.

8.1 Smart Events integration client

In order to provide systems that enable rapid integration of events within the transportation environment, COSMOS proposes the development of a specific piece of software that allows the exchange of information between different transport entities. This system, called VEProt (Virtual Entity Process for Reactive and Ontological Things), is a software client that will be released under an Open Source model on Github. It aims at easing the exchange of events and at consuming/sending information about transportation as well as any other useful data using the events server of the city.

8.1.1. Queues and messages for Y2

An important issue that had to be analyzed in depth is the MessageBus mechanism of interaction within the Madrid infrastructure because, although the Reactive Box system has a mechanism of subscription-watching in real time through which the different virtual entities can be obtained, this system behaves, for security reasons, as a unidirectional mechanism. This is, as a pure event generator. Due to this fact, the model considers how VEProt messages coming from the Complex Event Processing (CEP) are inserted into the RB. On this regard, the following issues have been taken into account:

- Speed response: A reactive system must contain entry mechanisms that do not involve delays or blockages in the entrance. Therefore, any concept based on web services or REST instances of any kind must be removed.
- Concurrent messages: Similarly, the system must be dynamic and absorb an amount of data dimensioned to handle the expected flow of information, which could exceed 2k events per second, also guaranteeing the continuation in case of system crashes.
- Security control: All messaging flows must be subject to strict security rules on the connection, so that no event of entry into the RB and involving event or status changes in a VE can be violated by any foreign element to the object itself or to any instance stored in the associated layers.
- Standardization: The messaging flow management must require a standardized scheme based on VEProt, since this is the datagram chosen for the exchange of VE within the city of Madrid.

Among the different models analysed for incoming messages into the RB system, including alarms managed by the CEP within the scope of Madrid UC, it has been selected Rabbit MQ for its adaptability to internal requirements rose within the internal communication model system. The server has been published in amqp.emtmadrid.es

8.1.2. Queues and messages for Y3

Advancing on queue management, during Y3 a selective automation mechanism has been applied. That allows events to be filtered during the reception through scripts which are implemented directly into the Rabbit server. Therefore, more control of the information and resources you use is provided. Among other tasks, the queue control now performs the following tasks:

- Security Check: VE input events are filtered in origin, checking on the security model if the connected customer is Owner or if the VE data collection is public or shared.
- Multiple consumers' instantiation by VE-type, separating incoming traffic in function of the nature or origin.
- VEProt Datagram Validation, rejecting the wrong messages.
- Tails Logging, recording correct and incorrect messages.
- JOBS management and validation for the execution of recursive tasks in the system.

In addition, a dual message queues system has been built, isolating for security purposes the public side (Internet) from the private side, in which consumers and events analyzers are. Thus, the architectures are isolated and the public manager becomes a message filtering proxy.

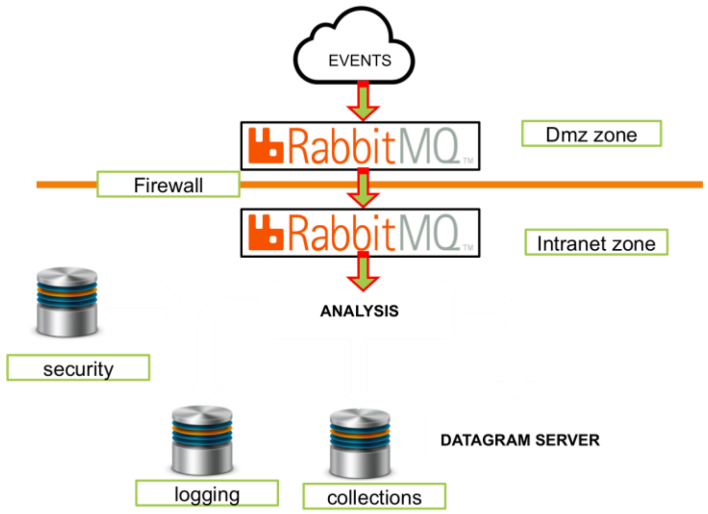


Figure 7: Messages' manager final architecture

8.1.3. VEProt description

VEProt is a logical system for the exchange of information between the different processes that can interact within an open scheme with proprietary or third parties systems. Its purpose is to provide a means of transportation and normalized information flow by which two systems or devices can communicate.

VEProt uses JSON as the exchange protocol handshaking and its functionalities are implemented through listening ports; these ports are configurable according to each subsystem model. Thus, it is possible to develop variable and adaptable configurations to each device according to port availability. VEProt sends data frames from IP ports towards the devices or subsystems which are message recipients and, in turn, can listen and receive responses via another port.

The VEProt system, designed globally, is the set of processes and protocols that allow a low-level object within the paradigm of IoT, to communicate status and data and to connect sensors and react instantly to changing events. This system, originally designed to work within different devices in the field of smart city mobility, can be moved naturally to any platform or system which requires interoperability and reactivity, especially in city environments, providing a unified mechanism for both communication and processing of various existing elements and also for instant sharing information with the public.

VEProt is based on a message scheme containing the rules, format and information needed so the different message subsystems know the attributes, relationships and the information to be exchanged among VEs.

VEProt can communicate remote Virtual Entities (not visible to each other) through reactive mechanisms by using DDP (into Event Server of Municipality of Madrid) or, can communicate local VE through UDP / TCP Standard protocols.

VEProt is defined in two ways, depending on its behaviour. The first definition is in terms of the set of processes that enables a device, sensor or system to operate autonomously. These processes are developed through a multi-platform programming language (Python) and they are located within a local area or ecosystem, to which sensors or devices are connected.

In summary, the transformation of a sensor or device into a VE requires:

1. A manufacturer driver or a low-level routine resident in the system that has the device connected.
2. A communication interface with VEProt, consisting of the transformation process of the different actions, status and characteristics of the device into those of a VE by defining a datagram.
3. Installing VEProt software that allows two-ways communication among the device or sensor with other systems or with other VE, as well as performing autonomous actions.

Conversion processes of events from devices/sensors to virtual entities are performed at the far end of the communication, that is, at those points where the information can be processed (generally a computer) and which have contact with the device information at a low level (communication with the driver, access to the communication port, access to the interruption, etc.).

Communication frames (datagram) are defined by a common language, through which they can indicate:

- Who is it?
- What kind of Virtual Entity is it?
- What characteristics and qualities does it have?
- What actions does it want to perform on another system?
- How can it communicate?
- What does it want to communicate?

8.2 Transport events exchange protocol

The exchange of events information within a Smart City has to be based on business rules and structures of dynamic information, but known in advance.

Currently, there are models of information in a very early stage or which are not fully developed to allow open communication of events that permit co-management participation and exchange of public-private information. Therefore, COSMOS aims to provide a new vision that guarantees the possibility of exchanging any information.

This structure must ensure the inclusion of multiple sensors and data in the system, while different observers can subscribe to get the information that is of interest to them using the event server.

The proposed datagram contains the following structure:

8.2.1. Header context

The virtual entities included in the message manifest themselves through this structure that defines the context elements. Further details about the implementation of the elements of the dictionary definition can be found in http://www.w3.org/TR/json-ld/

Example:

```
"VE:prot":
[
  {
    "@context": "http://cosmos.org/contexts/cities",
    "city": "Madrid",
    "homepage": "http://www.emtmadrid.es/",
    "domain": "http://cosmos.org/contexts/domain#transport",
    "layer": "transport.emt.bus.cpucore"
    "VE": "source",
    "@type": "bus"
  },

  {
    "@context": "http://cosmos.org/contexts/cities",
    "city": "Madrid",
    "homepage": "http://www.madrid.org/",
    "domain": "http://cosmos.org/contexts/domain#sensor",
    "layer": "transport.emt.bus.sensor"
    "VE": "target",
    "@type": "co2sensor",
  }
]
```

8.2.2. Control Header

The header contains general information of the message frame.

- VE:header. Unique identifying label indicating that the text below is a header.
- VE:version. Contains the VEProt version used for communication.
- VE:id. Allows the introduction of a unique identifying value for the entity.
- VE:datetime. Date and time of the request or generated message.
- VE:dateexpir. Message expiration date and time.
- VE:coordinateX. Geographic longitude coordinate in which the message is generated. Format must be WGS84.
- VE:coordinateY. Geographic latitude coordinate in which the message is generated. Format must be WGS84.
- VE:coordinateZ. Altitude in meters in which the message is generated.
- VE:size. Message size (for check of integrity).

Example:

```
"VE:header": {
    "VE:version": "1.00",
    "VE:id": "8CDA1667-C31B-4723-8154-90B8992EDA2A",
    "VE:datetime": "2014-04-20 13:23:05.01234 UTC",
    "VE:dateexpir": "2014-04-20 14:00:05.01234 UTC",
    "VE:coordinatey": "40.6719856",
    "VE:coordinatex": "-4.089738",
    "VE:coordinatez": "853",
    "VE:size": "243"
}
```

8.2.3. Virtual Entity Identifier or Request Original System

It allows that the message's addressed or the message's intermediate manager knows about the message sender. The data included is:

- VE:source. Fixed label that indicates that the structure contains the identification of the source system of the message.
- VE:id. Unique identifier of that object within its data layer.
- VE:systemsource. System/Entity code that generates the information. The receiver system must recognize that code as it allows the segmentation and analysis of the message contents.
- VE:subsystemsource. Information origin subsystem. It will allow the segmentation of the information of a virtual entity in multiple schemes, depending on the specialization.
- VE:versionsource. Identifier of the version of the generated information by the system source, regarding the general diagram.
- VE:functionsource. Function that emits the message.
- VE:modelistening. Listening mode for the answers.
- VE:port. Listening port
- VE:address. Listening address for the message answer.

<u>Example:</u>

```
"VE:source": {
    "VE:id": "1234",
    "VE:systemsource": "EMTBUS",
    "VE:subsystemsource": "CORECPU",
    "VE:functionsource": "INTEGRATOR",
    "VE:versionsource": "3.02.76",
    "VE:modelistening": "SKT",
    "VE:address": "82.45.133.2",
    "VE:port": "1424"
}
```

8.2.4. Identifiers of the target system

It contains relevant information about who is the final addressee of the data frame. This way, segmentations can be done in the subscription mechanisms by receiver system, so only the systems that need that frame will be the consumers of that information.

It defines the system to which VEProt must send the information. It contains the following tags within its structure:

- EV:target. Unique label that identifies this block.
- EV:systemtarget. Identifier of the destination system. This is an identifier for the subscriber through which the subscriber who is the message addressee.
- VE:subsystemtarget. Identifier of the destination subsystem. A subscriber can identify its message and recognize where it will be redirected within its system.
- VE:functiontarget. Invoking function at the addressee system. It indicates the function type that must be invoked in the destiny subsystem.

Example:

```
"VE:target": {
    "VE:systemtarget": "VALUE",
    "VE:subsystemtarget": "VALUESENSOR",
    "VE:functiontarget": "READ"
}
```

8.2.5. Identifiers of instructions and data

They define the structure of the virtual entity that contains the data from the message body (on array type structures).It contains the following values:

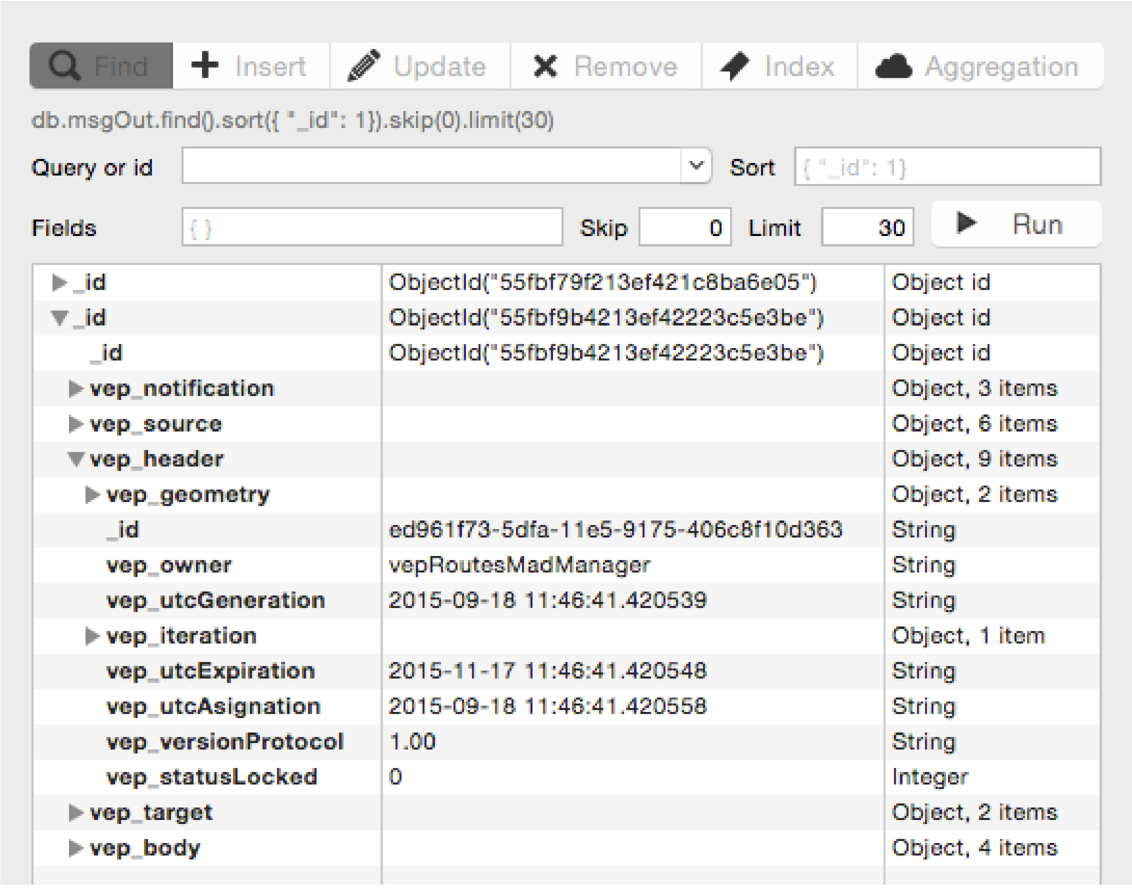
- VE:message. Unique label of the object. Message information exchanged between two virtual entities. Several of such structures may exist within the message to be exchanged.
- VE:id. Unique identifier of the data section within the overall message. It may simply be a sequential value.
- VE:confirmreq. Indicator response requirement. Value 0 = No requirement 1.- requirement.
- VE:timeout. Information or content validity time in milliseconds.
- VE:mode. Execution mode indicator. Value 0=Asynchronous (VEProt generates the call through one of its threads and continues with the execution of other processes or functions). Value 1= Synchronous (VECore stops the execution of its threads until response or timeout requested) call.
- VE:fiability. Reliability of the message; it will contain a percentage value whose maximum will be 100.
- VE:type. Defines the data type within the message.
- EV:subtype. It contains the data subtype within the type category. Some examples may be: XLS, MP3, TXT, EXE, etc.
- VE:encrypt. Encryption type (if there is any).
- VE:user. User that will be required to connect to the destiny resource.
- VE:passw. Password that will be required to connect to the destiny resource.
- VE:publickey. Public key that will be used for the encryption (coding-decoding) of the communication (if there is any).
- VE:data. Data set to be communicated by the virtual entity. The content will depend on each frame type.

Example:

```
"VE:message": [
    {
      "VE:id": "9DBC2368DA1E-56A1-895F-143A56BC87DF",
      "VE:confirmreq": "1",
      "VE:timeout": "1000",
      "VE:fiability": "100%",
      "VE:type": "BIN",
      "VE:subtype": "B64",
      "VE:encrypt": "none",
      "VE:user": "any",
      "VE:passw": "any",
      "VE:publickey": "any",
      "VE:data": {
        "port": "COM4",
        "PARITY": "true",
        "velocity": "9600"      }
    }
]
```

8.2.6. Changes done in Y2 within the VEProt datagram model

Since the implementation of the first Reactive Box system for the Madrid laboratory, certain problems were detected making incompatible the datagram protocol designed for interaction with the real software and storage elements. One of the first problems encountered was the notation "VE" of each element of the datagram produced a clear misunderstanding mistake within MongoDB schemes when using it in certain languages of instantiated classes, especially in C#. This makes that the released datagram version 1.00 which is currently working for the Y2 has changed the character ":" by "_". For example, the header object of the datagram is now represented as follows:



This has impacted directly on the notation defined originally for creating RDF definitions, since the original idea was that each element of the datagram was an element of a semantic definition. Finally, since it did not fit into the model, the ontological definition of the elements is currently being defined within the "_layers" collection of the storage database, reflecting the domain, subdomain and context of all elements of all collections.

The aforementioned definition also applies to the "_schema" collection at each element level, though this change has meant delays in the semantic definition of objects.

▼ _id	ROUTESMAD.alarms	String
_id	ROUTESMAD.alarms	String
▶ __id		Object, 3 items
▼ instant		Object, 3 items
__description	No description	String
__type	date	String
__path	.instant	String
_description	no description	String
▼ _id	ROUTESMAD.bustrack	String
_id	ROUTESMAD.bustrack	String
▼ instant		Object, 3 items
__description	No description	String
__type	date	String
__path	.instant	String
▼ geometry		Object, 5 items
__description	No description	String
__type	object	String
__path	.geometry	String
▶ type		Object, 3 items
▶ coordinates		Object, 3 items
▶ __id		Object, 3 items
_description	no description	String

Another important change while operating VEProt model in real conditions has been the redefinition of certain elements to make it operational as a whole. The most important of those changes has been the "vep_body" object whose wealth of content now responds much better to the requirements model posed by COSMOS, even in its first conception, allowing each VE not only to reveal what it is or what it contains, but also letting those VE to contain segments of encrypted and public data or even snippets of executable programs transferred into the datagram itself, stating simultaneously the VE and building at the receiver an operable and intelligent self-image, if necessary. To do this, we have introduced a content analyzer that represents in an array which data type contains every element, supporting multiple message objects within a datagram. It also contains elements for building logs and trace in remote of the activity of the entire system.

▼ vep_body		Object, 4 items
▼ vep_type		Array, 3 items
▼ 0		Array, 1 item
0	layerData	String
▼ 1		Array, 1 item
0	ebusData	String
▼ 2		Array, 1 item
0	logData	String

It has also been included a scheme (vep_requisites) so that the receiver of the message analyzes whether the VE being received through the datagram contains elements compatible with its operative system, memory, capacity, etc. Added to this, it contains an object that defines implementation plans (vep_plan) for VEs manifested as processes running on client systems.

Finally, as mentioned above, the data object (vep_data) has been replaced by an array of elements that allows multiple instances, features, messages, values or actions traveling in the same datagram and can be managed as a set or following an order. A special but typical case is the property of merging two VEs, and applies especially when the person with special needs is on board the bus, as shown in the following example.

▼vep_data		Array, 3 items
▼0		Object, 9 items
layer	ROUTESMAD.usrtrack	String
instant	2015-08-10 14:53:02.462053	String
idRoute	55d1801ffb954f08f8f1489f	String
dayType	LA	String
▼geometry		Object, 2 items
type	Point	String
▼coordinates		Array, 2 items
0	-3.69138338267	String
1	40.4211505276	String
idSesion	21d1701fab914f08c8e1438e	String
nameRouteUser	jmendez	String
order	3	String
dayWeek	M	String
▼1		Object, 11 items
status	5	String
direction	1	String
bus	8811	String
stop	0	String
line	008	String
trip	14	String
name	####	String
▶geometry		Object, 2 items
altitude	626.372	String
delay	-346	String
offSet	570	String

8.2.7. Changes done in Y3 within the VEProt datagram model

The constant evolution of the datagram and the opening of Reactive Box system to various platforms that support indirect integration with the COSMOS system have resulted in the need to expand header schemes referring to models of data entry of VE related events. Consequently, the ability to manage the property of the collections has added resources information control resources in the headers. The following items have been added:

- Regarding the system: creation of a specific system for managing event data that is related to the direct charge of events from external and internal systems. It is called LAYERS.
- Regarding the subsystems: creation of two subsystems, one for writing events in the VE and another for reading, called PUTDATA and READDATA.
- Regarding functions: As indicated previously on this document, creation of the INSERT, REPLACE, DELETE and DROP functions, which allow the dynamic management of information stored by the owners, manipulating data stored in the system.
- LAYER: Through this object is defined the VE in which you want to change (name) and how you want to share the event within the collection (type); it also generates the owner's event definition within the VE, with the aim of releasing it in public domain.

Example of the VEProt added header segment:

```
"system": "LAYERS",
 "subsystem": "PUTDATA",
 "function": "REPLACE",
 "layer": {
   "owner": "EMT.SERVICIOS.TRAFFICMAD",
   "type": "public",
   "name": "TRAFFICMAD.alarms"
 }
```

Another task done during this period has been the standardization of collections of images to use the public internet marker. Thus, representation maps (including the Reactive Box one) support customizing icons and data. For this purpose the shape object is used.

```
"shape": {
  "type": "marker",
  "options": {
    "shape": "circle",
    "markerColor": "red",
    "prefix": "fa",
    "icon": "fa-car"
  }
}
```

Finally a content object has been defined in order to generate pop-ups on the maps. This content object allows the inclusion of rich content (html) and other data from other systems, such as voting systems or iFrames. This element is called "state" and contains free values to be used by the VE event data contributors.

```
"state": {
   "instant": "2015-11-30 12:38:35.775",
   "description": "<b><p>Station: 15521</p><p>Good Traffic</p></b>   ",
   "value": "1",
   "format": "text"
 }
```

As a conclusion, although the VEProt model is not yet closed and still evolving, efforts have focused on the ability to provide variable exchange mechanisms for multiple and agnostic information formats supported by the increasingly varied VE of the system.

8.2.8. Response indicators

In the area of response data the invoked subsystems will return the achieved result data either as a return message or as the required data frame, provided that a response has been requested.

The content is defined as indicated below:

- VE:returntosender. Unique defining label indicating that the object is the one informing about the result of the operation. The object contains the following values:
  - VE:errorcode: Return code about the result of the request.
  - VE:errordescription. Request result description.
  - VE:datetime. Date and time of the result of the request according to the invoked system
  - VE:response. It will be an added object to each message sent within the array
  - VE:response. Array with the content of the returned data
    - VE:id. Identifier of the individual frame of the message.
    - VE:data. Values of the message.

Example:

```
"VE:returntosender": {
      "VE:,errorcode": "000",
      "VE:errordescription": "INFO:decoded ok, datagrame allowing",
      "VE:datetime": "2014-04-20 13:23:05.06543 UTC",
      "VE:response": [
            { "VE:id":"9DBC2368DA1E-56A1-895F-143A56BC87DF",
              "VE:data": {
                    "valuelevel": "18",
                    "temperature": "5°"
                 }
            }
      ]
}
```

Regarding this section, there is an important remark regarding Y2 achievements, which is about the response indicators, as they merge as new elements of the array vep_body of the datagram.

9 Evaluation

During Y3 some activities have been carried out in order to perform the best possible evaluation of the different COSMOS project achievements and developments, and therefore, gathering this way useful feedback from, among others, final users. On this regard, we have research and development projects involving multidisciplinary integrators, including several European IoT ones beyond COSMOS project, with the aim of fostering synergies among them.

Right after Y2 review, we contacted with the local partner of INLIFE project http://www.inlife-project.eu/. This project, which acronym means “INdependent LIving support Functions for the Elderly” belongs to the Horizon 2020 program, and its concept is to address the challenge of turning existing research efforts to reality for real people across Europe.

Existing flexible ICT solutions could assist elderly users with cognitive impairment in organising, carrying out and completing everyday tasks and constitute essential factors for continuing to be and feel independent. IN LIFE will offer all-around, personalised, multi-faceted existing ICT solutions and services addressing diverse daily activities (eating, physical activity, commuting, mental stimulation, communication, social interaction, etc.) to users with cognitive impairment living in their own home or in sheltered homes, as well as to their formal and informal carers.

In addition to that, INLIFE emphasis is placed on elderly and carer interactions, communications and care scheduling and monitoring.

Considering the aforementioned aspects, both teams realized about the synergies among both projects, and therefore, we agreed to cooperate.

During last months, several meetings were held between COSMOS and INLIFE managing teams, including representatives from Polytechnic University of Madrid, Atos Research & Innovation, Madrid Handicapped Association, Madrid Autism Association, etc.

Figure 8: April 25th, 2016 COSMOS-INLIFE project meeting

In April 26th, 2016, a press released was launched including a survey to gather feedback from potential end users. The survey can be accessed in the following link: https://docs.google.com/forms/d/1PhkejCY3C7IB0Pnml1HMOfpVzOBEkkFNa9jqy73Lyxk/edit?usp=sharing

And the results can be consulted at the following link (65 responses): https://docs.google.com/forms/d/1PhkejCY3C7IB0Pnml1HMOfpVzOBEkkFNa9jqy73Lyxk/edit?pref=2&pli=1#responses

Suggestions and recommendations have been taken into account accordingly.

10 COSMOS opportunities under the Smart Transport context

There are multiple circumstances and needs within a mobility oriented city system that depends on systems or subsystems whose development and implementation is beyond the specific context of transport or the establishment of protocols and standards for the exchange of information within a city. Among the main ones, we could find the security elements to ensure that certain content can be transferred between ends with the highest guarantee and confidentiality. On this regard, COSMOS could provide a secure way to encrypt a message segment that can travel through a public infrastructure and within an open protocol.

Information such as personal data, images, health data, etc. could use the Hardware Security Board developed under the framework of COSMOS as an encryption mechanism “extreme to extreme”. Those client systems requiring encryption of a VEProt datagram segment can perform the encryption operation and then be translated only by authorized recipients. It also lets information storage through Storlets within the Cloud with segments of encrypted messages and be recovered afterwards with total security.

Another important element for Smart Transport environment is the Complex Event Processing engine which allows you to manage and anticipate situations where a special event somewhere in the city can affect very significantly to the overall mobility or specific mobility of a citizen. Such events, by analysing patterns of previously processed data through the CEP can provide predictions that can help efficiency in traffic and, ultimately, to the overall mobility of the city.

Therefore, analysing the course and passing of buses, the density of traffic and traffic light phases provides useful data for planning special situations or prevent changes to a value of a pre-established series. Thus, connecting a predictive events analysis system to the transportation infrastructure provides added value for the management and planning of public transport and traffic.

For this, the mobility systems of the city must provide those events to the CEP, and through its analysis, getting automatically outlier data to take the appropriate decisions; and all in real time!

A clear example or use case is when the expected arrival time of a bus, which carries a person with special needs, should be analysed in a particular place and this bus does not arrive within its forecast; or undergoes a change on the bus route; or the person with special needs gets off the bus at a wrong bus stop; or does not realize that he/she has arrived to his/her destination, etc. all the aforementioned situations that could alert a supervisor or caregiver. To this end, transport events and contextual data need to be sent to Big Data storage systems such as those designed in the context of COSMOS. The purpose of Madrid, as a Smart Mobility City is to analyse how COSMOS, from the point of view of a permanent events analysing system can be used to solve complex problems of the city.

In summary, efficient communication of a uniform and open data flow, integrating connectors and processes within urban public platforms, interoperating with a system like COSMOS provides the ability to dynamically compose information flows for solutions adaptable to multiple situations.

11 Conclusions

During Y2, the initial conclusions outlined in Y1 deliverable were confirmed. This is, that the final conclusion of the study leads us to believe that the integration of the various systems of public transport within the paradigm of IoT passes through the right combination of public infrastructure and private systems, within which the public component plays the following roles:

- Supplier of a unified event information exchange protocol.
- Provider of public service infrastructure, especially server architectures able to understand, integrate and connect different systems, information providers and customers under a unified architecture.
- Coordinating and managing body for the approval and of smart devices in order to ensure the adaptability of different mechanisms and virtual organizations in the city.

On the contrary, the private component (businesses, citizens, research centres), will use the public infrastructure used for the purposes that their model or task is required, either for collaborative purposes or for direct consumption of information.

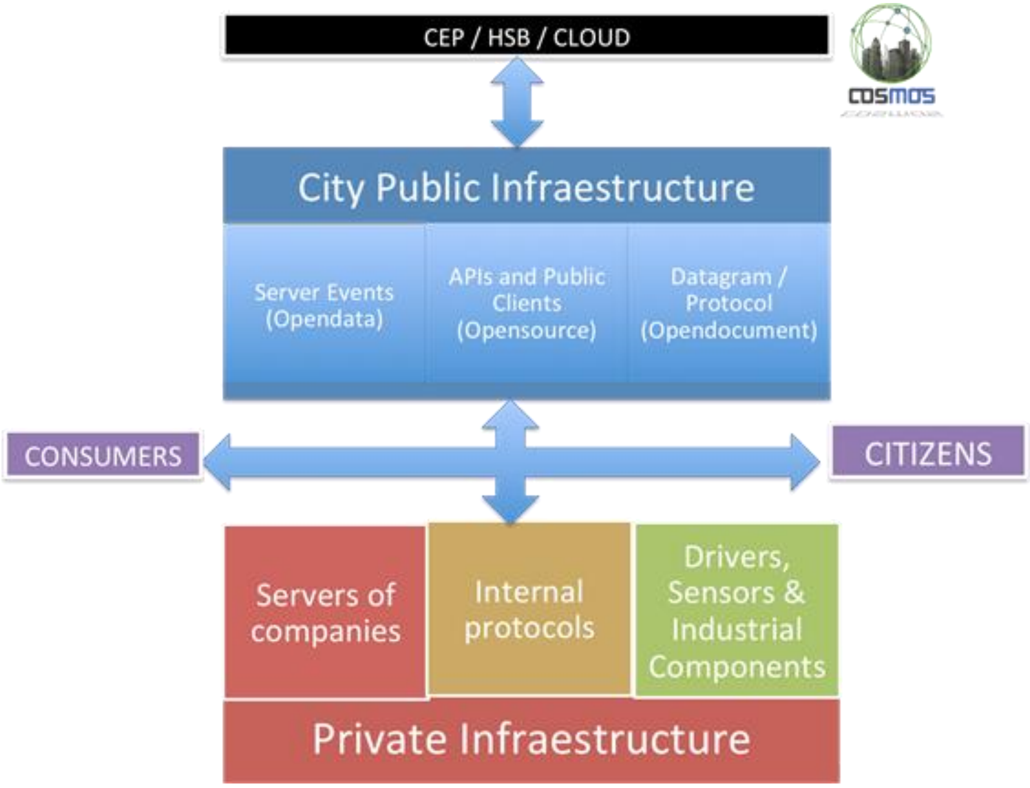


Figure 9: Public and private cooperation schema

But beyond the Smart Transport concept, we must start thinking about global mobility issues. Therefore, the creation of heterogeneous platforms on which multiple phenomena can be observed and exchanged in standard formats and under an analytical perspective, provide a very useful and valuable model for the city. COSMOS, as part of the integrated subsystems integrated into Madrid Mobility scenario becomes a factor of change and evolution.

In conclusion, tasks carried out during Y3 meet the objectives that were set during Y2 as specified into the “Recommendations for Y3” reference section of D.7.5.2.

12 Final recommendations

Among the recommendations for Y2 some topics were highlighted to ensure the success of the project. For instance, the need of establishing a multi-discipline working group aimed at creating a model definition, communication and management of objects, systems and sensors for the city of Madrid. This working group has been working already during Y2, and has been responsible for the specification, development and implementation of a Smart City Mobility Lab in Madrid, working mainly in transportation topics but with an open scope.

The main aim was to constitute a group wide enough to allow the participation of the four major groups of actors potentially involved in the development of the VEProt system. This group has been the project management committee and the "seed" for the implementation of the lab infrastructure.

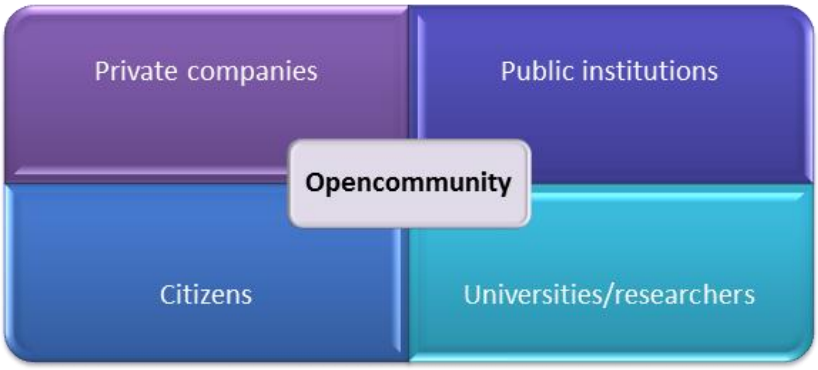


Figure 10: Proposed local working group for Y2

As stated in the deliverable D7.5.1 (Y1), the aforementioned community has been structured as follows:

- Project Management Committee: management function will be at a high level throughout the life cycle of the project and presentation, distribution and publication of the progress and results achieved over time.
- Audit and Oversight Committee: their role will be to establish the guarantees to ensure that the tasks that are planned are carried out with maximum quality as well as to verify that the repositories of code and documentation and new versions of VEProt have the highest reliability and warranty. This committee will also analyse the bug and the feedback from any source, establishing mechanisms for the evolution of the system.
- Coordination of code development: It shall arrange the analysis and specification of the source code to develop. It will also have the function to document and upload to the public repository the code developed.
- Developers: Those responsible for conducting the development of source code, using public repositories (GitHub, etc.)
- Specification-ontology and semantic entities: is responsible for defining the data model of virtual organizations in the city, as well as to maintain the information and the definition of the entities in public repositories. This function will be performed by specialists.

- Repositories, diagrams, web-semantics: the schemes will be used under any standardized specification for defining objects in public and accessible repositories.
- Protocol specification data: VEProt requires a standardized data exchange, to be defined, nurtured and developed by specialists and updated in a public repository.

More specifically, in the D7.5.2 (Y2) and regarding recommendations towards Y3, we pointed out that, within the broad and complex scenario that arises in this document, COSMOS offered endless possibilities within the scope of IoT in the city of Madrid, and not only in the area of mobility and transport but also in other areas, by conjugating the reactive elements deployed, which require expert systems in Machine Learning and CEP to operate.

Therefore, we believed that the system could be ripe to raise any use case that combine data sources and combined problems in two or more aspects or topics of the city. This could be the case of analyzing the conjunction public transport/lighting, or private transport/traffic optimization and improvement of congestion, etc. These ideas, based on the successful implementation of Madrid use case could be considered as alternatives, and that is why the complementary approach to the M-30 traffic management data has been added during Y3.

Additionally, the system should also follow the line of technical dissemination of the laboratory and the empowerment of its open and public use. On this regard, building an information portal to let proper dissemination and learning of Madrid reactive elements should be considered, using at the same time the research opportunities that COSMOS offers both within private and public solutions. The portal must contain various elements that allow both rapid deployment and content accessibility, such as GIT, CMS content (wordpress, etc), Wiki, etc.

Moreover, we also indicated that it may be useful to have specific user interfaces for startups and researchers to build new reactive elements and investigate with them, using COSMOS system as the generator of complex solutions, and as a result of Y3 work, the MadridLabs portal, the hackathon and the different evaluation processes in conjunction with other EU projects (specially with INLIFE) have been developed, fulfilling the Y2 recommendations.

13 Bibliography

1. RDF Dataset Normalization: A Standard RDF Dataset Normalization Algorithm;http://json-ld.org/spec/latest/rdf-dataset-normalization/

2. Organization of Schemas; http://schema.org/docs/schemas.html

3. Semantic Web Development Tools; http://www.w3.org/wiki/SemanticWebTools

4. Node.js Api Reference Documentation; http://nodejs.org/documentation

5. DDP, the Distributed Data Protocol; https://www.meteor.com/ddp, http://www.meteorpedia.com/read/DDP_Clients, https://github.com/foxdog-studios/pyddp

6. UNE 178301:2015, Ciudades Inteligentes. Datos Abiertos (Open Data); http://www.aenor.es/aenor/normas/ctn/fichactn.asp?codigonorm=AEN/CTN%20178#.VOBIO1OG_7p

7. Advanced Message Queuing Protocol; http://www.rabbitmq.com

8. Implantación de la Sociedad de la Información, el Conocimiento y la Innovación en la Ciudad de Madrid. Boletín Oficial de la Comunidad de Madrid Num. 6661; http://www.madrid.es/portales/munimadrid/es/Inicio/Ayuntamiento/Boletin-Oficial-del-Ayuntamiento/Buscador-Boletines/862-2012-04-18-Dec-Alcsa-Comision-Tec-Impl-Soc-de-la-Informac-el-conoc-y-la-Innov-?vgnextfmt=default&vgnextoid=9c965e7751ed6310VgnVCM2000000c205a0aRCRD&vgnextchannel=7a698db0ae967010VgnVCM1000009b25680aRCRD

9. Semantic Sensor Network definition and specifications http://www.w3.org/2005/Incubator/ssn/wiki/Main_Page

10. MongoDB https://docs.mongodb.org/manual/

14 ANNEX 1: M-30 Traffic data definition

FIELD	SPANISH DESCRIPTION	ENGLISH DESCRIPTION
IDENTIF	Código del punto de medida	PM code
CATEGORIA1	Nº de vehículos de la categoría 1 (berlinas)	Type of vehicle 1, Sedan
CATEGORIA2	Nº de vehículos de la categoría 2 (furgonetas)	Type of vehicle 2, Van
CATEGORIA3	Nº de vehículos de la categoría 3 (camiones y autobuses)	Type of vehicle 3, trucks and buses
ECOMUNICACIONES	Error de comunicaciones	Data communication error
ESTADO	Estado del punto de medida	PM condition
FECHA	Minuto en el que termina la toma de datos	date and time of end of data reception
HIOCC	Alarma de detección de incidentes. Algoritmo HIOCC	Incident detection flag if the PM is compatible with HIOCC algorithm
INTENSIDAD	Intensidad por hora (Real multiplicada por 60)	Traffic volume (vehicles per hour)
INTENSIDADA	Intensidad Alisada	Smoothed traffic volumen
INTENSIDADS	Intensidad Suavizada	Softened traffic volume
LIGEROS	Porcentaje de vehículos ligeros (< 7 m)	% of vehicles less than 7 m
LONMEDIA	Longitud media de los vehículos detectados	Average vehicle length
NS	Nivel de Servicio	Service level: 1 free flow to 4 congested flow, 0 and 5 not normal situations
NSA	Nivel de Servicio alisado	Smoothed Service Level
NSF	Nivel de Servicio filtrado	Filtered Service Level
OCUPACION	Ocupación	Occupation, % of time the sensor has a vehicle over it.
OCUPACIONA	Ocupación Alisada	Smoothed Occupation
OCUPACIONEHI	Ocupación Media de la espira integrada según encuadre horario	Average integrated occupation depending on time

FIELD	SPANISH DESCRIPTION	ENGLISH DESCRIPTION
OCUPACIONI	Ocupación Media de la espira integrada	Average integrated occupation
OCUPACIONS	Ocupación Suavizada	Softened Occupation
OCUPACIONSI	Ocupación Media de la espira suavizada integrada	Average softened integrated occupation
PESADOS	Porcentaje de vehículos pesados (> 7 m)	% of vehicles more than 7 m
SEPARACION	Separacion media entre vehículos	Average vehicle gap
TIEMPORETENCION	Tiempo de retención	Congestion time
TIEMPORETENCIONA	Tiempo de retención alisado	Smoothed congestion time
TIEMPORETENCIONS	Tiempo de retención suavizado	Softened congestion time
VELOCIDAD	velocidad	Speed
VELOCIDADA	velocidad Alisada	Smoothed speed
VELOCIDADEHI	velocidad integrada según encuadre horario	Integrated speed depending on time
VELOCIDADI	velocidad integrada	Integrated speed
VELOCIDADS	velocidad Suavizada	Softened speed
VELOCIDADSI	velocidad suavizada integrada	Softened integrated speed
VOLUMEN	Intensidad real en el periodo de agregación (1 minuto)	Traffic volume in 1 minute
VOLUMENDIA	Volumen acumulado del día	Daily traffic volume
VOLUMENEHI	Volumen de vehículos integrado según encuadre horario	Integrated traffic volume depending on time
VOLUMENI	Volumen Integrado	Integrated traffic volume
VOLUMENSI	Volumen de vehículos suavizado Integrado	Softened integrated traffic volume

15 ANNEX 2: Travel Time Algorithm

The algorithm for estimating travel time based on traffic data provided by traffic detectors considers a method of statistical and geometric calculation.

First, the variables involved in the calculation are described. Then, the geometric algorithm is specified. Finally, parameters are included for travel time calculation in the event of congestion.

Weighted average speed

In order to obtain the travel time of a section or travel route (i.e. a set of consecutive sections) you take into account the distance between the source and destination (usually from one message panel to an exit) on which you want to inform as well as speed in the sections contained in the route.

The measured speed of each section is the weighted average speed of the last n valid records (last n values of the integration periods where speed ≥ 0), where n is a configurable integer. The weighting of the records is inversely proportional to age (considered most important to the most current value):

$$Weighted\_Average\_Speed\_for\_section = \frac{\sum_{x=0}^{x=n-1}\left(Speed_i * (n-x)\right)}{\sum_{i=1}^{i=n} i}$$ Where:

- n the number of records considered.
- x the index for each record:
  - $x = 0 \Rightarrow$ the freshest record.
  - $x = n-1 \Rightarrow$ the oldest record.

The weight or relevance of the record is inversely proportional to the age of the record according to the following formula:

$$Weight_X = \frac{100 * (n-x)}{\sum_{i=1}^{i=n} i}$$

An example for obtaining the weighted average speed for a section is shown:

- If $n = 5$, the last 5 values obtained for speed.
- $\sum_{i=1}^{i=5} i = 1+2+3+4+5 = 15$

In the following table the records are ordered from the most current to the oldest and its weight is provided:

Record x	Speed	Weighting
0	48	$100*(5-0)/15 = 33{,}33\%$
1	87	$100*(5-1)/15 = 26{,}66\%$
2	68	$100*(5-2)/15 = 20\%$
3	78	$100*(5-3)/15 = 13{,}33\%$
4	82	$100*(5-4)/15 = 6{,}66\%$

The weighted average speed is:

$$Weighted\_average\_speed\_for\_section = \frac{\sum_{x=0}^{x=n-1}(speed_i*(n-x))}{\sum_{i=1}^{i=n} i} = \frac{\sum_{x=0}^{x=4}(speed_i*(5-x))}{15} = 68{,}\bar{6}Km/h$$

Invalid speed values are not considered for calculating the average speed.

Area calculation

The estimated travel time estimated between two origin-destination points, is the spatial area A between them and the linear interpolation that connects the different pairs of coordinates of the sections (1/Speed, Section (PK)) as shown in the following figure :

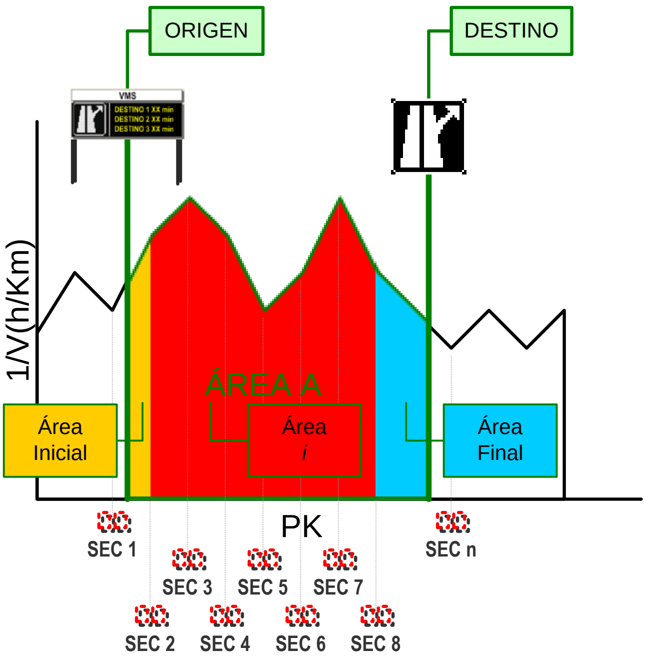


Figure 11: Area calculation for estimated travel time

Let be A the area and n sections (1..n), SEC1 corresponds section previous to the origin and SECN is the section after the destination. Therefore to obtain the travel time from a source to a destination, there must be a section before the origin and a section afeter the destination, in order to obtain the speeds in the points of origin and destination respectively by interpolation.

Each section forms a pair of coordinates (X_i, Y_i) by its location PK (X_i) and the inverse weighted average speed in a moment (Y_i).

The area A that estimates travel time from origin to destination, is the sum of the following areas:

$$A = Ainitial + \sum_{i=2}^{n-2} A_i + Afinal$$

Ainitial (yellow in the graph above) between the origin point of the area A and the first inner section A (that is, section 2). It is obtained as follows:

$$Ainitial = \frac{(X_2 - PkInic) \times \left[\left(\frac{Y_2 - Y_1}{X_2 - X_1} \times (PkInic - X_1) + Y_1 \right) + Y_2 \right]}{2}$$

where PkInic is the point (Pk) for the initial area. Simply apply the formula of the line to the calculation of the Y coordinate of a point. And then find the area of the area.

Ai (red in Figure 4 above) between each inner section to the area and the next. It is obtained by the following formula for each inner section (i = 2..N-2):

$$A_i = \frac{(X_{i+1} - X_i) \times (Y_{i+1} + Y_i)}{2}$$

That is to obtain the area of the triangle and the square between one section and the next.

Afinal (blue Figure 4 above) between the last inner section (that is, the section n-1) and the end point of area A. is obtained by the following formula:

$$Afinal = \frac{(PkFin - X_{n-1}) \times \left[\left(\frac{Y_n - Y_{n-1}}{X_n - X_{n-1}} \times (PkFin - X_{n-1}) + Y_{n-1} \right) + Y_{n-1} \right]}{2}$$

where PkFin is the end point for area calculation. It is to apply the formula for calculating straight to the Y coordinate of a point, and then find the area of the area.

An example of the area resulting from a route (set of PKs) is presented:

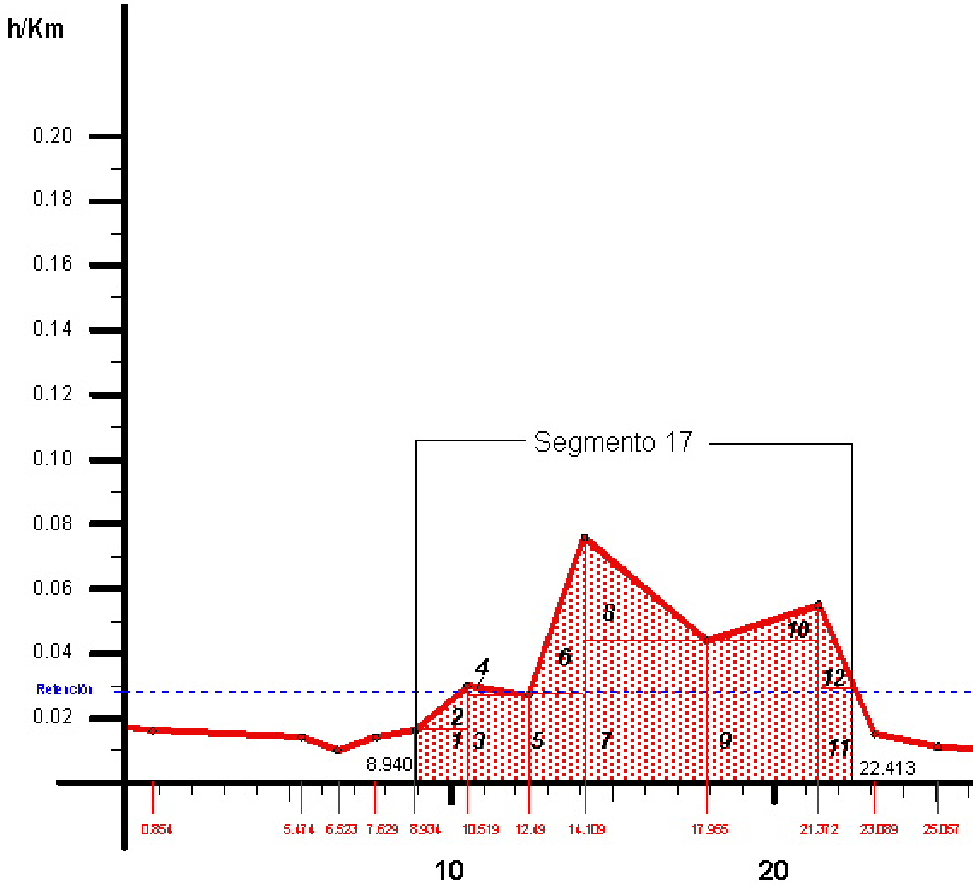


Figure 12: Estimated travel time calculation

In the vertical axis the PK (kilometer) the sections in the route.: from a message panel at PK 8 + 940, inner sections (PK represented in red) and exit point at PK 22 + 413. The thick red line results from linear interpolation of the inverse weighted average values of speed from sections represented on the abscissa.

The area represented by red dots, is the sum of the areas of geometric figures in which the surface is divided: rectangles and triangles. In the example above graph the total area has been divided into 12 figures, so the total area is the sum of the areas 12:

$$Area = Area1 + Area2 + Area3 + Area4 + Area5 + ... + Area12$$

To find the area of forms 1 and 2 in Figure 5 the interpolated value of the point source of PK 8,940 (point of the message panel) from the straight line joining the previous and following sections (PK 8,934) (10,519 PK) is needed. Analogously to find the area of forms 11 and 12 in Figure 5 the interpolated value of destination in the PK 22,413, from the straight line joining the previous and following sections as above (PK 21,372) (PK 23,089) is needed. The interpolated values are obtained from the equation of the line:

$$\frac{(y-y_1)}{(x-x_1)} = \frac{(y_2-y_1)}{(x_2-x_1)} \Rightarrow y = \left(\frac{y_2-y_1}{x_2-x_1}\right) * (x-x_1) + y_1$$

Then the interpolated values for the source and destination example route are:

$$Y_{SOURCE} = \left(\frac{0{,}029411 - 0{,}016393}{10{,}519 - 8{,}934}\right) * (8{,}940 - 8{,}934) + 0{,}016393 = 0{,}016442$$

$$Y_{DESTINATION} = \left(\frac{0{,}015151 - 0{,}055555}{23{,}089 - 21{,}372}\right) * (22{,}413 - 21{,}372) + 0{,}055555 = 0{,}031058$$

Note that a section with traffic detector is needed previous to the initial point and final point.

The resulting area is then:

$$A = 0{,}025930 + 0{,}010254 + 0{,}054749 + 0{,}001609 + 0{,}044972 + 0{,}039783 +$$
$$+ 0{,}167216 + 0{,}064314 + 0{,}148564 + 0{,}020633 + 0{,}032331 + 0{,}012750 = 0{,}623105$$

Then the estimated travel time = 0.623105 Time travel hours = 37 minutes 23 seconds.

16 ANNEX 3: Road sections of interest

Even though all the sections provided in the table below would be of interest, Main Sections 21 and 24 covering almost the complete eastern part of M-30 ring are of utmost importance to analyze M-30 behavior. Additionally, we suggest Main Section 9 to study disturbance to EMT lines since this section covers part of the route that EMT buses take from Av. Ilustración to Paseo de la Castellana and also the route buses take from Plaza de Castilla to areas of the city to the North (Las Tablas, Sanchinarro).

ROAD SECTIONS OF INTEREST ACCORDING TO ADT (AVERAGE DAILY TRAFFIC) AND TRAFFIC BEHAVIOR

Main Section	Section	Road	Direction	Description	PM					General comments
1	1	Main Road	Increasing	Manoteras - Pio XII	10013	10021	10091			
2	3	Main Road	Increasing	Pio XII - Costa Rica	10141	10211	10241			
2	5	Main Road	Increasing	Costa Rica - Av. América	10341	10401				
3	7	Main Road	Increasing	Av. América - Parque de las Avenidas	10441	10471	10501			
3	9	Main Road	Increasing	Parque de las Avenidas - Puente de Ventas	10561	10611				
4	11	Main Road	Increasing	Puente de Ventas - O'Donnell	10661	10711				
4	13	Main Road	Increasing	O'Donnell - Entrada Baipás	10831	10861				
4	15	Main Road	Increasing	Entrada Baipas - Av. Mediterráneo	10901	10941				
5	17	Main Road	Increasing	Av. Mediterráneo - Méndez Álvaro	10981	11071				
5	19	Main Road	Increasing	Méndez Álvaro - Nudo Sur	11101	11161	11201			
5	21	Main Road	Increasing	Nudo Sur - Entrada Túneles	11301					
6	23	Main Road	Increasing	Salida túneles - Puente de los Franceses	11981	12061				
7	25	Main Road	Increasing	Puente de los Franceses - Bus-VAO Moncloa	12121	12211	12241			
7	27	Main Road	Increasing	Bus-VAO Moncloa - M-40	12331	12391	12471			
8	29	Main Road	Increasing	M-40 - Salvador Maella	12571	12641	12721	12781		
8	31	Main Road	Increasing	Salvador Maella - Colmenar	12851	12921	12961	12981	13011	

Main Section	Section	Road	Direction	Description	PM					General comments
9	**33**	**Main Road**	**Increasing**	**Colmenar - Nudo Norte**	**13041**	**13101**				
9	**35**	**Main Road**	**Increasing**	**Nudo Norte - Manoteras**	**13181**	**13211**				
10	37	Side Road	Increasing	Pio XII - Arturo Soria	10142	10212				
10	39	Side Road	Increasing	Arturo Soria - Ramón y Cajal	10242	10342				
10	41	Side Road	Increasing	Ramón y Cajal - Av. América	10402	10442	10472			
10	43	Side Road	Increasing	Av. América - Parque de las Avenidas	10502					
11	45	Side Road	Increasing	Parque de las Avenidas - Puente de Ventas	10562	10662				[Section 45] The PM10612, with 2 lanes, has been rejected because its upstream and downstream measurement points (PM10562 and PM10662) are measuring 4 lanes instead of 2.
11	47	Side Road	Increasing	Puente de Ventas - Entrada A-3	10712	10832				
11	49	Side Road	Increasing	Entrada A-3 - Av. Mediterráneo	10862	10902				
11	51	Side Road	Increasing	Av. Mediterráneo - Av. Ciudad de Barcelona	10942	10982				
12	53	Side Road	Increasing	Méndez Álvaro - A-4	11108	11168	11208			
13	55	Side Road	Increasing	Colmenar - Nudo Norte	13044	13104				
14	2	Main Road	Decreasing	Manoteras - Nudo Norte	20026	23221	23181			
14	4	Main Road	Decreasing	Nudo Norte - Colmenar	23111	23071				
15	6	Main Road	Decreasing	Colmenar - Betanzos	23011	22971	22901			
15	8	Main Road	Decreasing	Betanzos - M-40	22851	22781	22721	22641	22571	
16	10	Main Road	Decreasing	M-40 - Bus-VAO Moncloa	22471					
16	12	Main Road	Decreasing	Bus-VAO Moncloa -Bus-VAO Moncloa Sur	22391	22341	22241			

Main Section	Section	Road	Direction	Description	PM					General comments
16	14	Main Road	Decreasing	Bus-VAO Moncloa Sur - Puente de los Franceses	22211	22121	22081			
17	16	Main Road	Decreasing	Puente de los Franceses - Entrada Túneles	21941					
18	18	Main Road	Decreasing	Salida Túneles - Nudo Sur	21301	21231				
18	20	Main Road	Decreasing	Nudo Sur - Av. Ciudad de Barcelona	21181	21081				
19	22	Main Road	Decreasing	Av. Ciudad de Barcelona - Salida Baipás	20971	20911				
19	24	Main Road	Decreasing	Salida Baipás - O'Donnell	20851	20831				
19	26	Main Road	Decreasing	O'Donnell - Puente de Ventas	20721					
20	28	Main Road	Decreasing	Puente de Ventas - Av. América	20661	20571	20471			
20	30	Main Road	Decreasing	Av. América - Ramón y Cajal	20441	20411				
21	**32**	**Main Road**	**Decreasing**	**Ramón y Cajal - Arturo Soria**	**20321**	**20261**	**20231**			
21	**34**	**Main Road**	**Decreasing**	**Arturo Soria - Pio XII**	**20151**	**20041**				
21	**36**	**Main Road**	**Decreasing**	**Pio XII - Manoteras**	**20021**					
22	38	Side Road	Decreasing	Manoteras - Nudo Norte	23187					
22	40	Side Road	Decreasing	Nudo Norte - Colmenar	23073					
23	42	Side Road	Decreasing	Nudo Sur - Méndez Álvaro	21192					
24	**44**	**Side Road**	**Decreasing**	**Av. Ciudad de Barcelona - Av. Mediterráneo**	**20952**	**20912**				
24	**46**	**Side Road**	**Decreasing**	**Av. Mediterráneo - O'Donnell**	**20852**					
25	48	Side Road	Decreasing	O'Donnell - Puente de la Estrella	20832	20742				
25	50	Side Road	Decreasing	Puente de la Estrella - Parque de las Avenidas	20722	20662	20572			
26	52	Side Road	Decreasing	Parque de las Avenidas - Av. América	20472	20442				
26	54	Side Road	Decreasing	Av. América - Costa Rica	20412	20322				
26	56	Side Road	Decreasing	Costa Rica - Arturo Soria	20262	20232				
26	58	Side Road	Decreasing	Arturo Soria - Pio XII	20152	20042				

ROAD SECTIONS OF INTEREST ACCORDING TO TRAVEL TIME

Section	Road	Direction	Origin	Destination	PM								
1	Main Road	Increasing	Manoteras	Av. América	10021	10091	10141	10211	10241	10341	10401		
2	Main Road	Increasing	Av. América	Puente de Ventas	10441	10471	10501	10561	10611				
3	Main Road	Increasing	Puente de Ventas	Av. Mediterráneo	10661	10711	10831	10861	10901	10941			
4	Main Road	Increasing	Av. Mediterráneo	Entrada Túneles	10981	11071	11101	11161	11201	11301			
5	Main Road	Increasing	Salida Túneles	M-40	11981	12061	12121	12211	12241	12331	12391	12471	
6	Main Road	Increasing	M-40	Colmenar	12571	12641	12721	12781	12851	12921	12961	12981	13011
7	Main Road	Increasing	Colmenar	Manoteras	13041	13101	13181	13211					
8	Side Road	Increasing	Pio XII	Parque de las Avenidas	10142	10212	10242	10342	10402	10442	10472	10502	
9	Side Road	Increasing	Parque de las Avenidas	Av. Ciudad de Barcelona	10562	10662	10712	10832	10862	10902	10942	10982	
10	Side Road	Increasing	Méndez Álvaro	A-4	11108	11168	11208						
11	Main Road	Decreasing	Manoteras	Colmenar	20026	23221	23181	23111	23071				
12	Main Road	Decreasing	Colmenar	M-40	23011	22971	22901	22851	22781	22721	22641	22571	
13	Main Road	Decreasing	M-40	Entrada Túneles	22471	22391	22341	22241	22211	22121	22081	21941	
14	Main Road	Decreasing	Salida Túneles	Av. Ciudad de Barcelona	21301	21231	21181	21081					
15	Main Road	Decreasing	Av. Ciudad de Barcelona	Puente de Ventas	20971	20911	20851	20831	20721				
16	Main Road	Decreasing	Puente de Ventas	Ramón y Cajal	20661	20571	20471	20441	20411				
17	Main Road	Decreasing	Ramón y Cajal	Manoteras	20321	20261	20231	20151	20041	20021			
18	Side Road	Decreasing	Av. Ciudad de Barcelona	O'Donnell	20952	20912	20852						
19	Side Road	Decreasing	O'Donnell	Parque de las Avenidas	20832	20742	20722	20662	20572				
20	Side Road	Decreasing	Parque de las Avenidas	Pio XII	20472	20442	20412	20322	20262	20232	20152	20042	