IPIC 2019 | 6th International Physical Internet Conference | London

JOHANNES KEPLER UNIVERSITY LINZ

Automated delivery of shipments in urban areas

Franz Kopica, Walter Morales-Alvarez and Cristina Olaverri-Monreal
Johannes Kepler University - Chair for Sustainable Transport Logistics 4.0, Linz, Austria
{franz.kopica, walter.morales_alvarez, cristina.olaverri-monreal}@jku.at

Abstract: *Fully automated delivery of goods in urban areas with small hybrid or electric vehicles can reduce CO2 emissions in cities. We contribute to the "last mile and city logistics" topic by proposing a case study for an autonomous logistics system for delivering packages and mail. For this purpose, electric vehicles and standardized autonomous transport boxes are used. The operator at the distribution center (DC) keeps track of vehicles and transport boxes based on Global Positioning Systems (GPS) information. The boxes are equipped with Radio-Frequency Identification (RFID) tags that contain a microchip to store and retrieve the information from the inventory database records in the central operating platform. Through RFID the recipient is informed about the delivery and then authorized to open the transport box with a Near Field Communication (NFC) enabled smartphone which sends a text message to the operator.*

Keywords: *Autonomous transport boxes, efficient transport system, GSM, GPS, RFID*

1 Introduction

Urban agglomerations are continuously growing, testing the limits of transport and infrastructure resources and causing extremely congested traffic areas in cities (OECD, 2010; Statistik Austria, 2019). As a consequence, travel time increases, which results in higher CO2 emissions.

Rising CO2 emissions and the expansion of transport infrastructure affect certain services and lead to circumstances that contribute to climate change and/or increased energy prices and transportation costs. In order to reduce long-term logistic cost to these systems and to limit global temperature increases (less than two degrees Celsius), sustainable transport needs to be considered as an integral part of strategies for sustainable development according to the United Nations (United Nations, 2019).

Fuel consumption and CO2 emissions reduction requires new vehicle concepts as well as efficient transportation systems. The expansion of fully electric vehicles into the market provides opportunities for sustainable mobility and a new technological era (Helmbrecht *et al.*, 2014) while also presenting a promising alternative to vehicles with internal combustion engines for urban goods mobility demands. In addition, time-critical deliveries can hardly be realized if the vehicles have to charge for hours in the middle of the delivery process, or are subject to an insufficient charging infrastructure. Therefore, small hybrid or electric vehicles are an optimal approach for the urban area (Heine, 2018). In line with this, we contribute to the International Physical Internet Conference by addressing the "last mile and city logistics" and "Autonomous Road Transports and Logistics Operation" topics proposing a case study that describes the application of an autonomous logistics system for delivering packages and mail in line with the goal of using autonomous technologies to increase efficiency of freight transport and logistics operations (Alliance for Logistics Innovation through Collaboration in Europe, 2019).

By using delivery hybrid- or electric vehicles that contain standardized autonomous transport boxes with electronic labels it is possible to optimize delivery cost and times. They drive from a strategically located logistic or central operating platform (distribution center or hubs) to a predefined destination where their load is distributed according to the information that they receive.

The delivery vehicles are stationed in a central location and are operative until all the autonomous transport boxes that they contain have been delivered. The autonomous transport boxes are able to operate the delivery vehicle's rear door to exit and enter before and after each delivery.

The proposed autonomous system for transport logistics relies on a multi-agent architecture. The different entities in the architecture are interconnected through mobile networks for the internal exchange of information (e.g. the state of the processes of each unit or its location) or for external communication to acquire data related to the surrounding road users or customer information to which the delivery is scheduled.

The remainder of the paper is organized as follows: the following section describes the related work in the area. Section 3 presents a general description of the architecture of the proposed system starting by explaining the operation of the different entities, detailing how they communicate and ending with a particular use case. Section 4 explains in detail the design of the autonomous transport box and their main hardware and software components. Section 5 describes the components of the transport management system (TMS) implemented in the distribution center. Finally, section 6 concludes the work.

2 Related Work

In the previous section we introduced the idea of an autonomous multi-agent system as an alternative for traditional shipment, whose main components (DC, vehicles, autonomous transport boxes) intercommunicate between them exchanging information (location, surrounding road users, box and package's information) with which each entity can make independent decisions to deliver a package to a particular location. Multi-agent systems (MAS) offer advantages such as increased speed, efficiency and robustness in operations, scalability, cost reduction and reusability of agents (Nikos, 2007; Balaji and Srinivasan, 2010).

Multiple intelligent, semi-autonomous agents can be applied in several contexts with a variety of finalities as for example to facilitate the process of capturing software products functionality that simultaneously represent different characters that pursue defined goals (Olaverri-Monreal *et al.*, 2013; Olaverri-Monreal *et al.*, 2014).

The advantages of using autonomous system of control in logistical processes of a multi-agent system makes this technology particularly useful and therefore its implementation over the years has been abundant as in the work of Karageorgos *et al*. (2003) who used it for logistics and planning optimization, Leung *et al*. (2016) who designed a case-based multi-agent wave picking decision support system for handling e-commerce shipments or Hribernik *et al.*, (2010) that in the context of internet of things (IoT) for transport logistics proposed an MAS-based approach to connecting information flow so that the objects were able to process information, and make decisions. The authors included in their work an extensive overview of relevant work in a number of areas, including holonic manufacturing, smart resources and intelligent products.

Most of the research in the area of transport logistics in the last decades has focused on the development of models for the management of information. Therefore we contribute to the research in the field by implementing an architecture that includes the development of autonomous entities that do not require a human operator to deliver goods from the transport vehicle to the final destination.

Although The McKinsey Institute (2016) forecasts that driverless vehicles such as delivery robots will make up 85 percent of last-mile deliveries by 2025, until the day there are no conclusive studies on their use in cities. The cases of shipping robots are scarce and mostly private, such as the prototype announced by Amazon or Fedex that seeks to transport packages from Monday through Friday during daylight hours (Mashable 2019). The company that stands out the most in the implementation of autonomous robots for sending packages is an Estonian company called Starship Technologies. The company developed autonomous food delivery bots that were capable of carrying two grocery bags (Venturebeat, 2019). Although the vehicle designed by Starship works efficiently, it only focuses on transporting food to a certain location. In our case we propose a general architecture for an urban environment, automating the delivery process as a proposal for transport logistics in a completely digital era.

3 System Overview

As mentioned in section 1, we implement a multi-agent system (MAS) using the principles of autonomous control in intelligent products that conform a hierarchical organization to meet the objective of shipping a package to a particular place. This use case was selected because of the need in modern society to implement intelligent systems with automated technology as a solution in shipping companies (e.g. Fedex, Amazon, DHL) to reduce consumption of fossil resources, decrease CO_2 emissions, increase consumer comfort and improve shipping's security. The physical smart entities that make up the proposed organizational architecture are as follows:

- The distribution center where the different packages to be transported are located;
- The autonomous transport boxes that are the containers of the packages to be transported and constitute at the same time the entities with which consumers interact personally to pick up their order and
- The electric vehicles that are in charge of transporting the boxes or smart containers to the recipient neighborhood.

In this first approach, the packages to be sent are non-perishable objects whose total weight does not exceed 10 kg.

3.1 General Architecture

Initially in the central operating platform hub or distribution center the goods are unloaded from the carriers into small hybrid or electric vehicles (e.g. vans), that are in constantly communication with the central operating platform and therefore are able to receive all the information pertaining the distribution of the load from the leading information technology (IT) system, such as delivery address, recipient's personal information, traffic, route and changes in delivery schedule.

The distribution center is in charge of the real-time supervision of the packages using servers that contain a database of the autonomous transport boxes including tables for the description of the load they contain, recipients and shipping route information. Through an electronic communication system for network connection, it monitors the surrounding road users in urban areas in order to calculate the most optimal routes for the vehicles and the transport boxes (considering an optimal balance between distance to the recipient's address and travel time but also being able to consider other parameters as in Olaverri-Monreal *et al.*, 2016).

The autonomous transport boxes are smart containers that contain integrated sensors such as RFID tags to allow the final customer to access their content. The tags can read data from the recipient phone through NFC. As autonomous units the transport boxes are capable of making decisions based on the data they acquire through sensors such as video cameras, ultrasound, GPS and RFID tags. They also have communication capabilities to inform the recipient about the delivery location and schedule.

The smart containers are capable of maintaining constant communication with the distribution center throughout the whole delivery process by using a global system for mobile communications (GSM) module, which turns the transport box into an access point to the network in an urban environment. It is important to note that we do not consider using a Wi-Fi module to connect the autonomous transport box to the network because in urban environments these kind of networks fluctuate, as they depend on routers with private access and reduced coverage, unlike cellular networks that depend on antennas that have total coverage in cities, being only necessary a SIM card from the telephone provider to access them. Figure 1 illustrates the system architecture.

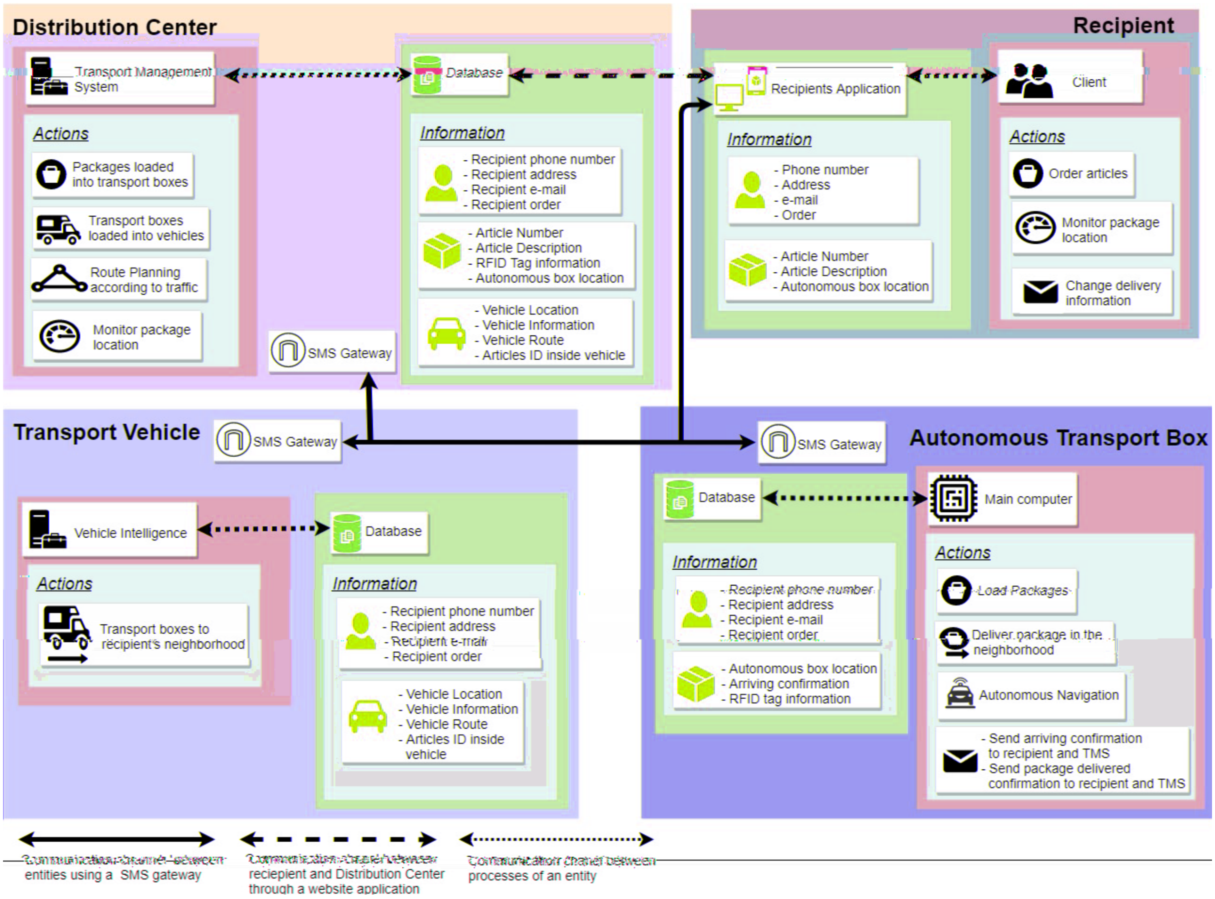

Figure 1: System architecture

3.2 Communication

As for the communication between the entities in our system, a transport management system in the hub DC included additional communication capabilities through GPS and short message service (SMS). This communication was achieved thanks to the fact that on the part of the transport box there is a SIM7000E device that works as a GSM module and that together with the program SMS Server tools generates an SMS Gateway that connects with the DC's gateway. Figure 2 illustrates the communication process.

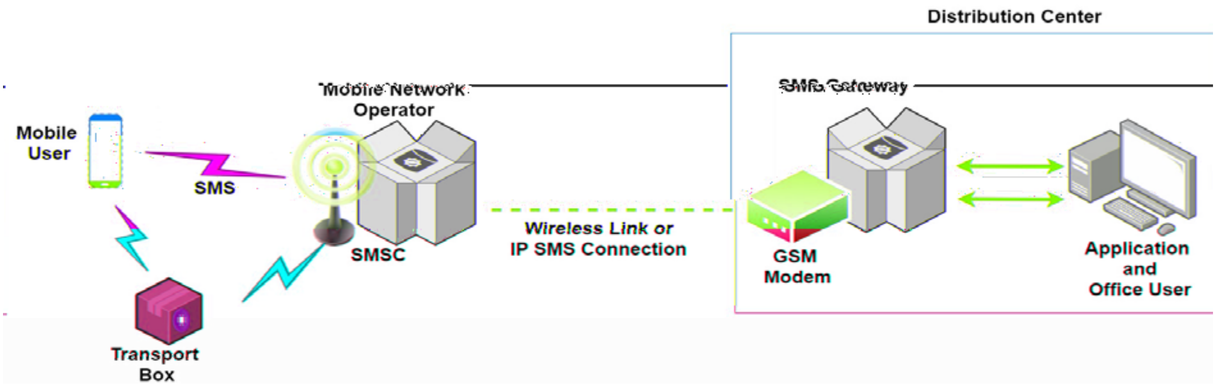

Figure 2: Communication architecture designed through a SMS gateway.

The SMS gateway is an interface for communication that uses mobile networks between smart entities. It offers a framework to process the messages transmitted between the devices, and it has functionalities such as the HyperText Transfer Protocol application programming interface (HTTP-API)[1], that handles the data acquire through the HTTP protocol from the network or the email to SMS functionality. This allows the DC to send a message to the autonomous transport box based on the emails that it receives.

The Short Message Service Centre (SMSC) is part of a global mobile network (GSM-, UMTS- or LTE) and is responsible for storing, forwarding, converting and delivering messages from the short message service.

On the other side, the TMS sends updated address information (received from the recipient) gateway to the autonomous transport boxes and vice versa using the database driven SMS as follow:

1) The information from the TMS-database is converted into a text message by a script and transmitted to the transport box through the SMS gateway.
2) The SMS is converted into a text message by a script and stored in the database of the transport box.
3) The confirmation of the change of address will be transmitted to the TMS.
4) An application from the TMS sends an email or SMS to the recipient.

As an example of messages sent between the DC and the autonomous transport box, Figure 3 shows an example of a script to send a SMS message, written in the programming language PHP that is embedded in HTTP-code.

[1] RESTful-API, used to handle data via e.g. http-GET or, http-PUT

```
<!DOCTYPE html>
<html>
<body>
<?php
$text ='Adress-change:Donaustadtstrasse/30/14/18';
$url = 'https://open-source/sms-gateway/?' .
        'user='IPIC2019'.
        '&pass='md5passphrase'.
        '&to=004313317823'.
        '&text=$text;

$response = @file_get_contents ($url);

  if($response == '200') {
        echo 'SMS has been sent! ';
  }else{
        echo 'An Error occurred: '. $response;
}
?>
</body>
</html>
```

Figure 3: Example of a script to send a message through the SMS gateway

Figure 4 shows the code to convert an email to an SMS using the SMS-gateway and the PHP's mail function. To this end the phone number of the device to reach and the network's domain for the SMS Gateway are required. In this example we used the address for the network's SMS-Gateway 004313317823@sms-gateway.at.

```
<?php

if ( isset( $_REQUEST ) && !empty( $_REQUEST ) ) {
 if (
 isset( $_REQUEST['004313317823'], $_REQUEST['sms-gateway.at '], $_REQUEST['smsMessage'] ) &&
  !empty( $_REQUEST['004313317823'] ) &&
  !empty( $_REQUEST['sms-gateway.at '] )
 ) {
  $message = wordwrap( $_REQUEST['smsMessage'], 30,"<br />\n" );
  $to = $_REQUEST['004313317823'] . '@' . $_REQUEST['sms-gateway.at '];
  $result = @mail( $to, '', $message );
  print 'Message was sent to ' . $to;
 } else {
  print 'Not all information was submitted.';
 }
}
```

Figure 4: Code example to convert an email to an SMS.

3.3 Use case scenario

To illustrate the operating mode we describe in this section a potential use case in a scenario with a single family detached house with a front yard (see Figure 5). The goods are unloaded from the carriers in the distribution center into a hybrid or electric van that drives to a target destination determined by the central operating platform following the information from the leading IT system (Figure 5b). The autonomous transport boxes then receive an updated destination with new coordinates via text message (SMS) or email that is read and interpreted automatically by the box without a third party intervention, in cases where the recipient changes the destination address. In this scenario, the change is stored in the database of the transport managing system and sent to the autonomous transport box through the communication channels mentioned.

After the autonomous transport boxes have left the delivery van they drive towards the destination (Figure 5c). A confirmation that the destination has been reached and the box can take a parked position (Figure 5d) is made through the RFID tag located inside the transport box where a customer ID is stored. This information is transferred to the database application in the box and triggers the transmission of an SMS message with the delivery time information to the package's recipient (as listed on the database) (Figure 5e). The operator receives the same information (Figure 5f). The autonomous transport box is equipped with a warning system. If the autonomous transport box does not stay in its parked position for a certain pre-configured time, for example 5 seconds, an alarm is sent to the corresponding person (operator, recipient, etc.). Eligible receivers are smartphones with RFID/NFC functionality that can confirm the receipt. As a result, the autonomous transport box can be opened without an alarm and all involved actors get a confirmation of receipt (Figure 5h).

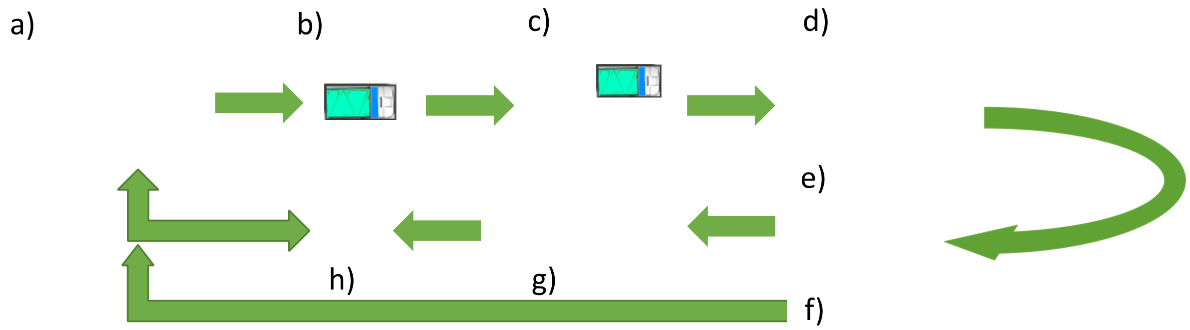

Figure 5: Use case scenario showing the process of the delivery starting with the leading IT system (a), the transport vehicles (b), the autonomous boxes traveling (c) and taking their parked position (d), the delivery of the message to the database (e) and the recipient (f), the smartphone to unblock the box (g) and the confirmation of the delivery (h)

4 Prototype: Autonomous Transport Box

As mentioned above these smart containers are equipped with all the required technology for a smooth autonomous operation. They are capable of simultaneous localization and mapping (SLAM) by constructing and/or updating a map of an unknown environment while simultaneously keeping track of their location through GPS. They are also able to perceive their surroundings through several active and passive mounted sensors (i.e. radar, cameras, ultrasonic). All units communicate changes in the delivery plan or schedule with the parent system. In addition, the autonomous transport boxes are equipped with a GPS module for localization and a global system for mobile communication (GSM) module that acts as a mobile communication modem for sending and receiving messages from the hub.

A RFID reader/writer unit is installed in the autonomous transport box. The goods are equipped with a RFID-tag that contain a microchip to store and retrieve the information from the inventory database records in the central operating platform. Through this technology, the autonomous boxes are also labeled with unique identifiers that can contain a large amount of information and make inventory tracking a faster process. To complete the system, the boxes have a Raspberry pi that stores and processes the information, enabling communication between all the peripherals and processes explained above. Figure 6 depicts the relationship between all the peripherals and processes for the exchange of information.

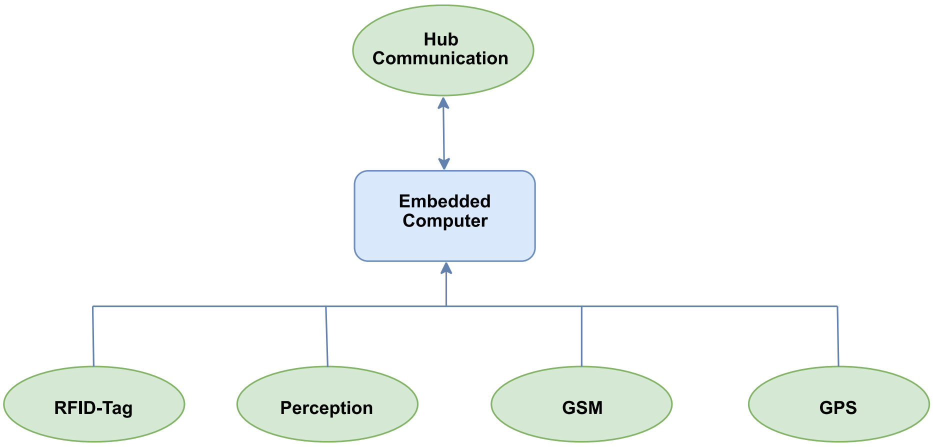

Figure 6: Relationship between all the peripherals and processes for the exchange of information pertaining to load distribution

To deliver the corresponding goods, recipients are informed via text messaging services (SMS) that are available in mobile-device systems upon arrival of the autonomous transport box. Recipients are able to open the box through the use of radio waves to read and capture information stored on the tag attached to the box using a previously installed mobile device application. This application makes it possible to operate NFC as RFID tags.

The reading distances are limited to 0.1 meters (depending on device hardware). In cases where the recipients have a mobile phone that lacks NFC functionality, the autonomous transport box has an intelligent lock with which they can open the box by providing a PIN sent via SMS.

4.1 Hardware

4.1.1 Raspberry pi

The main unit of the autonomous transport box to process data is the Raspberry pi. It consist of an embedded computer with an Advanced RISC (reduced instruction set computer) Machines (ARM) processor. Its General Purpose Input Output (GPIO) interface (extended 40-pin GPIO header) allows the transport box to communicate directly with the sensors and process the data that they acquire. This computer is compatible with a variety of Linux distributions such as Ubuntu or Raspbian, as well as Windows OS (Raspberry pi Foundation, 2018).

4.1.2 Raspberry pi module SIM7000E

The SIM7000E is a Raspberry pi module that has multi communication functionalities such as NB-IoT, eMTC, EDGE, GPRS, and GNSS. The NarrowBand-Internet of Things (NB-IoT) and enhanced Machine Type Communication (eMTC) are rising IoT communication technologies evolved from 2G to LTE (4G), with advantages including low power, low cost and wide coverage. This module allows the development of applications such as remote controlling, asset tracking, remote monitoring, and mobile POS terminals. While the GSM/GPRS, and EDGE are traditional 2G/2.5G technologies capable of sending SMS or making other wireless communications. Following protocols (TCP, UDP, PPP, HTTP, FTP, MQTT, SMS, Mail, etc.) and GNSS positioning (GPS, GLONASS, BeiDou and Galileo) are supported. These functionalities makes the SIM7000E suitable to serve as an access point for the autonomous transport boxes and to communicate these smart container with both the recipient and the distribution center using mobile networks.

4.1.3 RFID-MIFARE RC522

The MFRC522 is a highly integrated reader/writer IC for contactless communication at 13.56 MHz. This internal transmitter is able to drive a reader/writer antenna designed to communicate with ISO/IEC 14443 A/MIFARE cards and transponders without additional active circuitry. It is designed for low-power consumption and provides a robust and efficient implementation for demodulating and decoding signals from ISO/IEC 14443 A/MIFARE compatible cards and transponders. The digital module manages the complete ISO/IEC 14443 A framing and error detection functionality (parity and CRC). The MFRC522 supports contactless communication and uses MIFARE higher transfer speeds up to 424 kB/s in both directions (MFRC522 Datasheet, 2016). This module is designed for mass production and can communicate directly with any CPU board via Serial Peripheral Interface (SPI).

4.1.4 Laser sensor

The RPI Lidar A3 is a 2D laser manufactured by Slamtec with a range of 20 m, scan rate of 16 kHz and angular resolution of 0.36 degrees (Slamtec, 2016). This sensor enables real-time localization of the transport box and mapping of the surroundings (SLAM). The decision to use this sensor is based on the fact that the company that manufactures these devices has a large amount of support in SLAM, in addition to providing ROS packages for this purpose. The laser will communicate by serial protocol with the Raspberry pi through a module provided by the company. With the data acquired, the main computer will perform SLAM algorithms, by this method estimating its localization and the trajectory that it has to follow.

4.1.5 Camera

The autonomous transport box is equipped with two RGB Raspberry pi model V2 cameras to obtain images from the environment for a real time detection of objects. Together with the data acquired from the laser it will be possible to recognize and avoid obstacles. The camera also acts a security system to avoid that the transport boxes are stolen as it can take images of individuals who try to directly steal the intelligent container or open it by force.

4.1.6 Inertia Measurement Unit

The unit for measuring inertia (IMU) is a module used to estimate changes in the position and orientation of the autonomous transport box and to estimate its location in conjunction with the Lidar points and the camera images. The specific model used is the MPU9250 which has an accelerometer, gyroscope and compass which are responsible for calculating through the acceleration of the device, its angular velocity and magnetic fields to which it is subjected, changes in position and orientation of the container. The results of these calculations can be acquired by communicating with the device through the I2C protocol either with the main computer of the box or through a tertiary processor such as an Arduino, used in this specific case to decouple the acquisition of these data from the main computer.

4.1.7 Motors

The transport box must be able to withstand loads up to 10 kg per shipment, travel at a speed of 0.5 m/s, have 4 wheels 15 cm in diameter, and be able to drive along paths that incline of up to 20 degrees. With these specifications, a motor that produces a torque of at least 2 Nm at an approximate speed of 6.66 rad/s is required. In order to comply with these characteristics, we used brushless direct current motors for the high power efficiency and the torque-to-weight ratio they offer. To control the velocity of the motors in the most efficient way, an Electronic Speed Control (ESC) circuit that makes the motors follow a speed reference sent from the main computer or from a third party microcontroller such as an Arduino is needed.

In most cases, the ESC receives a Pulse Width Modulation (PWM) as input in one of its terminals, which is used as a reference to transmit the necessary signals to the motor to magnetize and demagnetize its rotor and thus generate movement. It should be noted that the ESC is also responsible for sending the necessary current signals, thus decoupling the logic circuit (microcontroller) from the actuators. In our particular case, the Raspberry pi sends a speed reference to an Arduino by serial protocol that will perform the low level control and map this speed signal to a PWM. This signal is then transmitted to the ESC, which will then send the current necessary to the motor to initiate transport box motion.

4.1.8 Power Supply

The battery to be used must be capable of supporting both the load of the computer and the load of the engines. In an approximate calculation, the transport box may need a battery with a capacity of at least 10000 mAh, a nominal voltage of at least 12V and high capacity to withstand the load of the motors. Based on these specifications and the need for a battery with relatively little weight, a lithium polymer batteries (LiPo) was chosen.

4.2 Software

The software run under the Ubuntu operating system and was stored in the SD card of the Raspberry pi.

4.2.1 Ubuntu 18.04 & NOOBS (New Out Of the Box Software)

The operative system installed on the Raspberry pi is Ubuntu 18.04 LTS using Noobs. Noobs is an installation assistant for the primary operating system for the family of Raspberry pi and it is highly optimized for the embedded computer with over 35,000 compiled software packages (Raspberry pi Foundation, 2018). This installation assistant sets up the Ubuntu 18.04 long-term support (LTS) operating system and it is officially supported on the x86, AMD64 and ARM architectures. As with all Debian derivatives, the program packages are divided into several package sources that can be installed through the terminal.

4.2.2 Ubuntu - SMS Server

SMS Server Tools 3 (smstools) are an SMS gateway software for sending and receiving text messages (SMS) using GSM modems on Linux. They are especially interesting for monitoring systems to send notifications not only by email, but also by SMS. This software is the main telecommunication interface that, through the GSM SIM7000E module, allows the autonomous transport box to exchange information with the distribution center and with the recipient via SMS or email.

4.2.3 RDBMS – Relational Database Management System

The autonomous transport box uses a database based on MySQL to store recipient’s information giving the smart container an efficient and secure storage of data sent by the distribution center or by the recipient, in addition to serving as backup in cases of eventualities.

MySQL is a free open source product and it is one of the world's most common relational database management systems. It is available as open source software as well as a commercial enterprise version for various operating systems. It has a number of tools which are available for administration. Alternatively, the included command line tools or software can be used with a graphical user interface. The entire administration of the server is done by the program PhpMyAdmin that is a third-party tool for editing (Apache, 2019).

4.3 Apache Webserver

We use this server to interact with the TMS GPS software that needs a running server on a device to access their location. The Apache HTTP Server is a modular open source Web server with convenient functionalities such as encrypting the communication between browser and Web server (mod_ssl), the capacity to be used as a proxy server (mod_proxy) or the complex manipulation of HTTP head data (mod_headers) and URLs (mod_rewrite) (Apache, 2018). The HTML code of a page, and the design (via CSS) files are retrieved from different sources such as databases and transmitted separately to a client. Script languages, such as PHP, ensure that all individual information is connected to a document. The free source code allows it to be adapted to individual needs.

4.4 Robotic Operating System (ROS)

ROS is a framework for the development of robots in which it is possible to intercommunicate isolated processes through messages transported by communication channels called topics (Quigley *et al*., 2009). This framework suits this use case perfectly because it allows the processing of the different data acquired by the systems described in section 4.1 and can control the different modules of the system. For example, with ROS it is possible to acquire the laser data, perform SLAM and with the results send signals corresponding to the motors to move to a specific location. Another important feature is that this framework has a large amount of support for its use in embedded computers such as Raspberry pi, and the great community of programmers that publish their open source packages for the developing of robots. The specific distribution configured in the transport box is ROS Melodic due to the Linux distribution installed on the Raspberry pi.

5 Distribution Center: Transport Management System

This section defines the components of the transport management system implemented in the distribution center. This system is in charge of constantly monitoring the autonomous transport box via GPS tracking software through the SMS-gateway communication explained in section 3.

5.1 GPS Tracking System Software

The application for the Web server, in this case Traccar, can be self-hosted in the cloud or on-premise. It is designed specifically to provide web-based GPS tracking services for a "fleet" of vehicles that run an Apache Web server. It is a fleet tracking system that is very highly configurable and scalable to larger enterprises as well. A customizable mapping service supports OpenLayers/OpenStreetMap in addition to Google Maps, Microsoft Virtual Earth, and Mapstraction (which provides mapping support for MultiMap, Map24, MapQuest, and more). It supports GPS trackers from a variety of vendors, from low- cost models to high-end quality brands (Traccar, 2019). The implementation of Traccar in the TMS allows the constant monitoring of the location of the different autonomous transport boxes in real time. Also, as can be seen in Figure 7, the Traccar architecture is similar to the general architecture proposed by us, which makes it perfect to meet our requirements, fulfilling the objective of using open source software for the development of the system. Following the architecture of Traccar, in our use case the tracking devices are the autonomous transport boxes and the small hybrid or electric vehicles, the persistent storage is a MySQL database implemented on the TMS and the Web application is for managing users, devices and other entities.

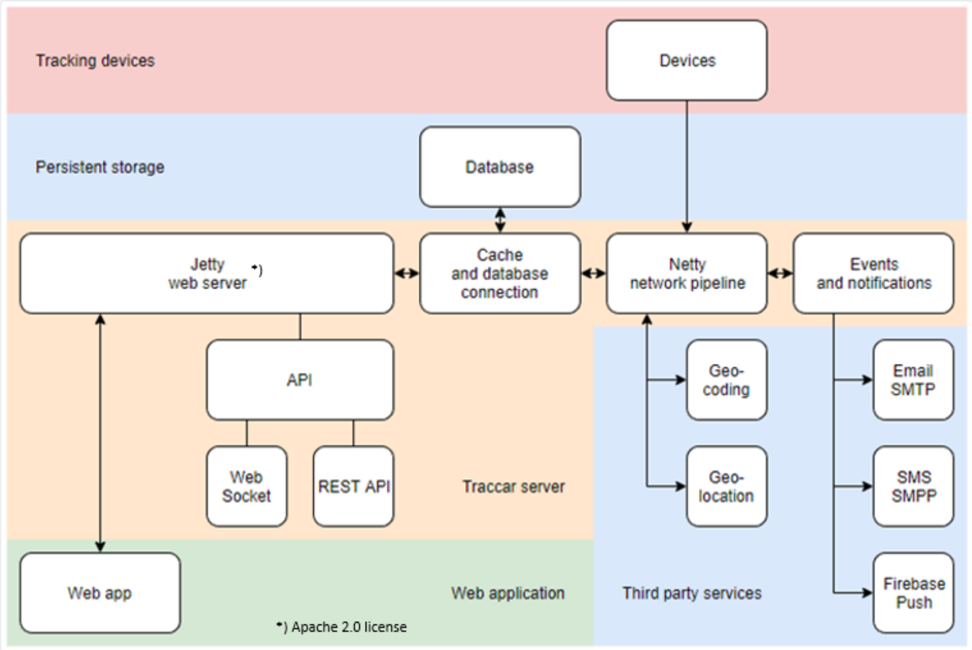

Figure 7: Traccar architecture (Traccar 2019).

5.2 SMS Gateway-Jasmin

Similarly to the autonomous transport box, the TMS includes a software application called Jasmin (Jasmin, 2014) that works as a framework to use the python SMS-Gateway that is available in the DC. Through the HTTP protocol communication with the autonomous transport boxes is established. In addition the API makes it possible an intelligent routing to connect in an efficient way to the autonomous transport boxes and the transport vehicles.

6 Conclusion

In this work we proposed an alternative way to transport packages or mail based on a multi-agent autonomously controlled system, explaining in a general way the architecture of the system designed; making special emphasis in the design of the hardware and software components of a prototype for autonomous transport boxes. A description of the implemented transport management system in the DC completes the work.

One of the main objectives of this work was to implement the prototype of the autonomous transport box to test it in real conditions. To achieve this goal we also developed the required software. After testing the platform regarding communication and monitoring tasks (servers, SMS gateways and GPS software) to detect potential conflicts, we concluded that the trial communication with the distribution center ran seamlessly.

Future work will address a more sophisticated system that will be evaluated in various scenarios including other road users.

Acknowledgment

This work was supported by the Austrian Ministry for Transport, Innovation and Technology (BMVIT) Endowed Professorship for Sustainable Transport Logistics 4.0.

References

- Alliance for Logistics Innovation through Collaboration in Europe (2019) "6th International Physical Internet Conference" [Online]. Availabe: https://www.pi.events/. [Accessed 6.06.2019].
- Apache (2018): HTTP-Server Project [Online]. Available: https://httpd.apache.org/ [Accessed 21.03.2019].
- Apache (2019): PhpMyAdmin. [Online]. Available: https://dev.mysql.com/doc/refman/8.0/en/ [Accessed 21.03.2019].
- Balaji, P.G., D. Srinivasan (2010). An Introduction to Multi-Agent Systems. 10.1007/978-3-642-14435-6_1.
- Heine. A. (2018): Future of transport logistics. [Online]. https://www.pt-magazin.de/de/ specials/mobilitaet/ [Accessed 21.03.2019].
- Helmbrecht, M., C. Olaverri-Monreal, K. Bengler, R. Vilimek, A. Keinath (2014) : How Electric Vehicles Affect Driving Behavioral Patterns., In IEEE Intelligent Transportation Systems Magazine. Special Issue on Electro-Mobility. Volume 6, Issue 3, pp. 22 - 32.
- Hribernik, K., T. Warden, K.D. Thoben, O. Herzog (2010): An Internet of Things for Transport Logistics - An Approach to Connecting the Information and Material Flows in Autonomous Cooperating Logistics Processes. 54-67.
- Jasmin (2014): Jasmin SMS-Gateway [Online]. Available: https://jasmin.readthedocs.io/en/latest/architecture/index.html [Accessed 21.03.2019].
- Karageorgos, A. N. Mehandjiev, G. Weichhart, A. Hämmerle (2003). Agent-based optimisation of logistics and production planning. Engineering Applications of Artificial Intelligence. 16. 335-348. 10.1016/S0952-1976(03)00076-9.
- Leung. K.H., K.L. Choy, M.C. Tarn, S.W.Y Cheng, H.Y. Lam ; Jason C.H. Lee ; G.K.H. Pang
- Design of a case-based multi-agent wave picking decision support system for handling e-commerce

shipments, 2016 Portland International Conference on Management of Engineering and Technology (PICMET), Honolulu, HI, 2016, pp. 2248-2256. doi: 10.1109/PICMET.2016.7806645.

- McKinsey & Company (2016): Parcel delivery -The future of last mile kids. [Online], https://www.mckinsey.com/~/media/mckinsey/industries/travel%20transport%20and%20logistics/our%20insights/how%20customer%20demands%20are%20reshaping%20last%20mile%20delivery/parcel_delivery_the_future_of_last_mile.ashx, 2019/04/16. [Accessed 21.03.2019].
- Mashable (2019): Amazon adds adorable Scout to its robot delivery fleet. [Online], Available: https://mashable.com/article/amazon-delivery-bot-scout/?europe=true#s0yRenbePZqw [Accessed 16.04.2019].
- MFRC522 Datasheet, (2016) "Standard performance MIFARE and NTAG frontend". https://www.nxp .com/docs/en/data-sheet/MFRC522.pdf.
- Nikos, V. (2007) A Concise introduction to multiagent systems and distributed artifical intelligence. Synthesis Lectures On Artificial Intelligence And Machine Learning, 1st edition, 2007.
- OECD/China Development Research Foundation (2010): Trends in Urbanisation and Urban Policies in OECD Countries: What Lessons for China?, OECD Publishing, Paris, https://doi.org/10.1787/9789264092259-en.

- Olaverri-Monreal, C., Hasan, A., Bengler, K. (2013) "Semi-Automatic User Stories Generation to Measure User Experience", Proceedings CISTI 2013 8th Iberian Conference on Information Systems and Technologies., pp. 340–345, June 2013.
- Olaverri-Monreal, C., Hasan, A., Bengler, K. (2014) "Intelligent Agent (IA) Systems to Generate User Stories for a Positive User Experience", In International Journal of Human Capital and Information Technology Professionals, pp. 26 – 40.
- Olaverri-Monreal, C., Pichler, M., Krizek, G.C., Naumann, S. (2016) "Shadow as Route Quality Parameter in a Pedestrian-Tailored Mobile Application", In IEEE Intelligent Transportation Systems Magazine. Volume: 8, Issue: 4, winter 2016, pp. 15-27.
- Quigley, M., K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, A. Ng (2009). ROS: an open-source
  Robot Operating System. ICRA Workshop on Open Source Software, 1–5.
- Raspberry pi Foundation (2017): NOOBS. [Online]. Available: https://www.raspberrypi.org/downloads/ [Accessed 15.03.2019].
- Raspberry pi Foundation (2018): Rasberry Pi 3 Model B+ [Online]. https://static.raspberrypi.org/files/product-briefs/Raspberry-Pi-Model-Bplus-Product-Brief.pdf. [Accessed 14.03.2019].
- Slamtec, (2016) RP Lidar A3. [Online] http://www.slamtec.com/en/Lidar/A3 [Accessed 15.04.2019].
- Statistik Austria (2019): Based on fact statistical material on the transport sectors of road, rail, aviation and inland waterways. [Online], Available: http://www.statistik.at/web_de/statistiken/energie_umwelt_innovation_mobilitaet/verkehr/index.html [Accessed 12.03.2019].
- Traccar (2019): GPS Tracking Platform. [Online] http://www.traccar.org/ [Accessed 21.03.2019].
- United Nations (2019) Climate Change - United Nations Sustainable Development [Online]. Available: https://www.un.org/sustainabledevelopment/climate-change-2, 14.03.2019.
- Venturebeat (2019): Starship Technologies' robots begin delivering food to college kids. [Online], https://venturebeat.com/2019/01/22/starship-technologies-robots-begin-delivering-food-to-college-kids/, 2019/04/16. [Accessed 16.04.2019].