

Connected Motorized Riders - A Smart Mobility System to Connect Two and Three-wheelers

Sakthivel Manikandan Sundharam
FSTC/LASSY
University of Luxembourg
L-1359 Luxembourg
sakthivel.sundharam@uni.lu

Loic Fejz
RealTime-at-Work (RTaW)
4 rue Piroux, Centre Atrium
54000 Villers-les-Nancy France
loic.fejz@realtimework.com

Nicolas Navet
FSTC/LASSY
University of Luxembourg
L-1359 Luxembourg
nicolas.navet@uni.lu

Abstract—The Smart Cities Mission has been launched in India in 2015 to develop 100 cities, with smart mobility being one of the main topics in the mission. As urban areas are flooded with two (motorcycles) and three wheelers (auto-rickshaws), introducing smart control of such vehicles may reduce the congestion on the roads and the number of accidents. Indeed, over-speeding and drunken driving are common traffic violations. In this project we propose an IoT-based smart mobility system which tracks data, such as the vehicle location, vehicle speed, alcohol level of the driver, etc. efficiently over the internet. Our system has been conceived with CPAL, a high-level language meant to simulate and execute Cyber Physical Systems including IoT applications. A prototype running on ARM mbed IoT hardware, shows the feasibility of our concept. We believe that more efficient and interactive traffic management, more disciplined driving behaviors, reduction in accident rate, more controlled pollution, increased passenger safety can be achieved if systems like the one prototyped in this work deployed contributing to smarter cities.

I. INTRODUCTION AND PROBLEM STATEMENT

The data from WHO Global status report [5] on road safety 2015 states that 72% of vehicles in India are motorized two and three wheelers. Because of this, the two and three wheelers are major contributors in creating traffic congestion in urban areas. The report also states that among all the categories of road deaths, motorized two and three wheelers are predominant ones with 34% cases. The main reasons for two and three wheeler accidents are over-speeding and alcohol consumption. Indeed, the urban speed limit is often ignored by commuters and manual checks for alcohol consumption is not a systematic enough procedure to reduce the accidents.

In addition, to the best of our knowledge, there is currently no technology supported system in India to know the density of motorized riders in a particular region or an urban area. Also, there exists no automated system to know whether vehicle speed and alcohol consumption level are within the legal limits. These needs motivate us to propose a smart system where two and three-wheelers are equipped with set of sensors and are connected to the internet. Next section gives the overview of proposed solution. Remaining sections of the paper explain the solution in details.

978-1-5090-2541-1/16/\$31.00 © 2016 IEEE

II. SOLUTION: CONNECTED TWO AND THREE-WHEELERS

Two-wheelers (i.e. motorcycles) and three-wheelers (auto-rickshaws) are low-cost and easy commuting means in India. The dashboard in these vehicles is simple, available either as analog or as digital meter. But new vehicles are released in the market with digital dashboards. Our smart mobility solution can be integrated in the dashboard either as additional embedded hardware or as software in existing hardware. A prototype device has been built to demonstrate the feasibility of the idea. The device is connected to internet and the web platform captures the live data as depicted in Figure 1.

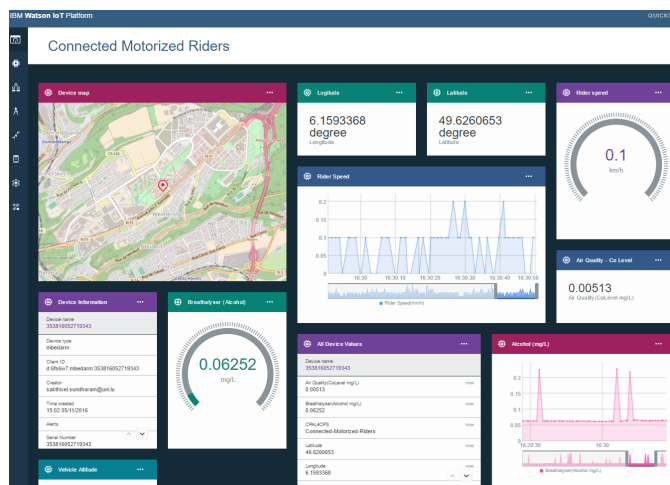


Fig. 1. Connected Motorized Riders - A smart mobility system for connecting two and three-wheelers to internet. Data such as latitude, longitude, altitude, speed, alcohol blood level are tracked and monitored in IBM Watson IoT platform as shown in the screenshot.

In this work, CPAL [2] is employed as modeling and simulation environment in the design phase of the application. The final implementation has been realized using the ARM mbed development platform that includes a SDK and software libraries. In the Section III, a timing aware model-driven engineering paradigm that supports the IoT system development is presented. Section IV elaborates the hardware details of the

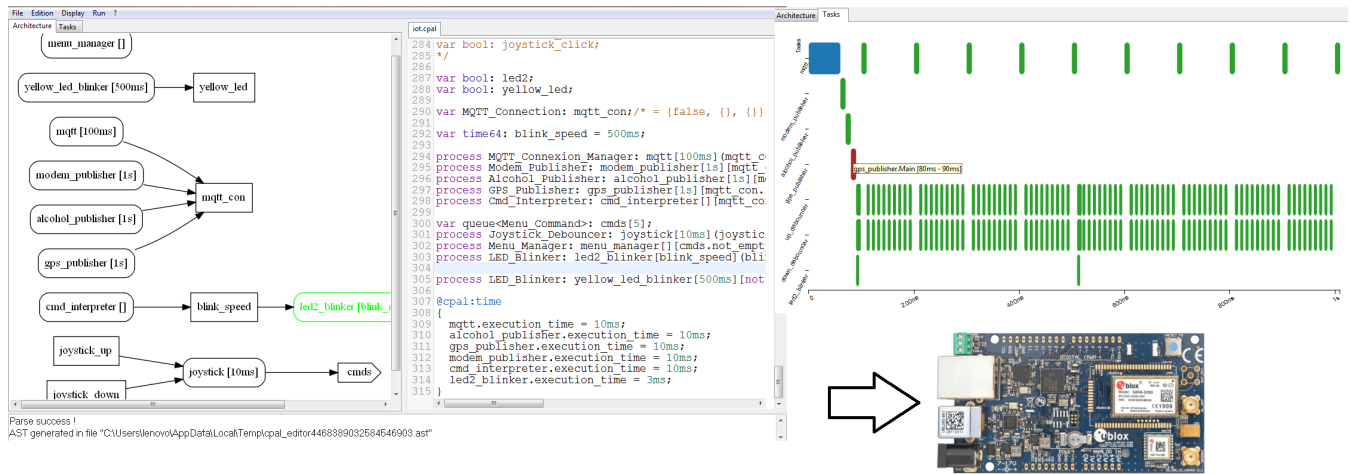


Fig. 2. CPAL model of the application. The left-hand side is the view of the functional architecture with the functions and the data flows making up the application. The CPAL code is in the center. The right-hand side shows a Gantt chart of the activation of the tasks implementing the application. This model can be executed on a workstation and without any changes interpreted (Bare-Machine Model Interpretation) on the ARM mbed Cortex M3 processor of the U-Blox C027 IoT board.

prototype, IoT software development flow, information security and device connectivity to internet. Finally the Section V concludes with results and the outlook.

III. CPAL - SYSTEM ENGINEERING TO BUILD IoT APPLICATION

A. Need for a timing aware model driven development

Innovation in the development of systems relies increasingly on software, which is disrupting complete industries like the transportation industry. Model-Driven Development (MDD) has proven to be a cost and time-effective method. But still non-functional properties such as time, safety and energy concerns are not well integrated in the existing MDD environments. The industry practice is that the design model is constructed, then the code is generated from the model and ported to the target hardware. Then run-time tests and measurements are performed during testing phase, but that is where the time and costs of fixing the problems are the largest.

This motivates the need for timing-aware design flows. Indeed, existing MDD flows are simulation-centric environments that assume that the computational resources available are infinite, or, at best, they are crudely estimated. This tends to create a gap between design phase verification and target testing. This problem motivates a new modeling paradigm. Our initiative is CPAL [7] which is a language to model, simulate, verify and execute Cyber-Physical Systems as used in automotive, aerospace, IoT, etc. These systems are typically interacting with numerous sensors, computing resources and the actuators.

B. Model-driven approach to IoT system development

Unlike the traditional MDD where execution is achieved by generating code from a model, and then compiling it, CPAL programs are executed by a real-time model interpretation engine. As shown in Figure 2, a key advantage of using CPAL for IoT application development is that it provides

during the design phase a view of the functional architecture of the application which is the set of processes, how they are activated, and the data flow among them. This is to the best of knowledge not possible with the mbed development environment. In addition, simulation possibility strengthens the early stage functional verification.

An IoT enabled CPAL model interpretation engine has been developed for the ARM Cortex M3 microcontrollers [6]. On this platform, the interpreter runs on the bare hardware, that is without an OS, what is referred to as Bare-Metal Model Interpretation (BMMI). BMMI helps to reduce the memory footprint and avoid errors that can be introduced throughout the design flow (e.g., at code generation or compilation stage, or errors at the OS level). BMMI eases also the verification, because the model is what actually executed, and not code generated from the model.

A benefit of interpretation is that the low-level services needed for the product development can be offered by the interpretation engine (e.g., MQTT protocol support). Only the high-level application is to be developed as per the functional and non-functional requirements. Hence the development efforts to interface sensors, GPS and MQ-3 to hardware is reduced, leading to shorter time to market.

IV. PRODUCT PROTOTYPE AND IoT PLATFORM

A. Device overview

The speed of the vehicle is generally estimated by the wheel speed sensor and is already available in a dashboard. In the prototype we have used the GPS speed sensor which provides the speed but also the location of the vehicle. An ARM mbed-enabled IoT kit with U-Blox C027 (ARM Cortex M3 core) is used as central unit to run the application. We have used U-BLOX Max 7Q series GPS module to capture the location data and speed calculation. Global Navigation Satellite System (GNSS) antenna is used to connect the GPS module.

A MQ-3 ethanol sensor is used to detect the alcohol content in the driver's breath. A MQ-7 Carbon mono-oxide (Co) sensor is connected to the board to measure the air quality level while the vehicle is on the move. SARA-G350 GSM GPRS modem with cellular SIM card at the back side of the board enables the connectivity using standard mobile data network.

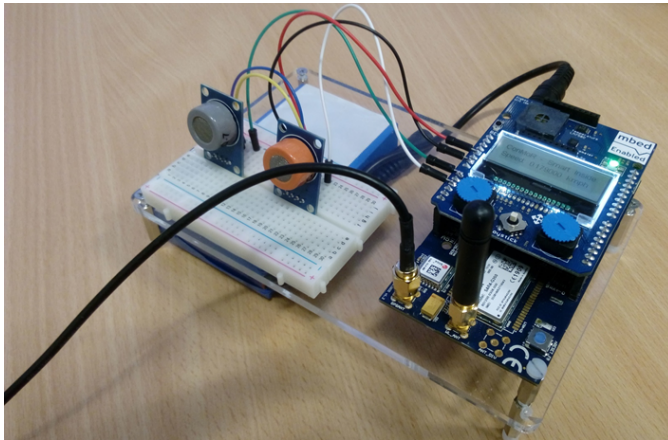


Fig. 3. Prototype set-up for interfacing sensors to ARM mbed IoT hardware (right) consisting of the application shield and the base board U-Blox C027 ARM Cortex M3, the MQ-7 Carbon mono-oxide sensor (left on the breadboard) and the MQ-3 ethanol sensor (right on the breadboard), U-BLOX-Max 7Q GPS module (not shown in the picture) with GNSS antenna, and the GSM modem with antenna

Figure 3 shows the prototype of the *connected motorized riders* device built using ARM mbed enabled IoT hardware. The IoT board consists of two parts: the application shield and the base board U-Blox C027 ARM Cortex M3. The sensors such as GPS and MQ-3 are interfaced to application shield of the board. Since the product is for mobility applications, to connect to the internet either IoT network or cellular connectivity is required. We have carried out our experiments using cellular GSM/GPRS modem to connect the device to the internet wirelessly.

B. ARM mbed IoT flow

ARM mbed IoT platform provides the OS, cloud services, tools and libraries to facilitate the creation and deployment of large-scale commercial IoT solutions based on standards. The ARM mbed device connector [1] service connects the device (i.e) motorized rider to IBM cloud efficiently and easily with inbuilt security functions enabled. IBM Watson IoT platform [3] which is used in our project to manage connected vehicles is one of the cloud-hosted services offered by the IBM Bluemix cloud infrastructure.

The mbed device connector service is connected to the IBM Watson platform to send/receive the IoT device events, using API keys and tokens. We have defined the mbed connector bridge to connect to the platform. Events (send/receive) between devices (vehicles) and the Watson IoT platform are created using Node-RED application which is a visual tool (see Figure 4) for wiring the IoT objects such as hardware devices, APIs, and online services in an intuitive way.

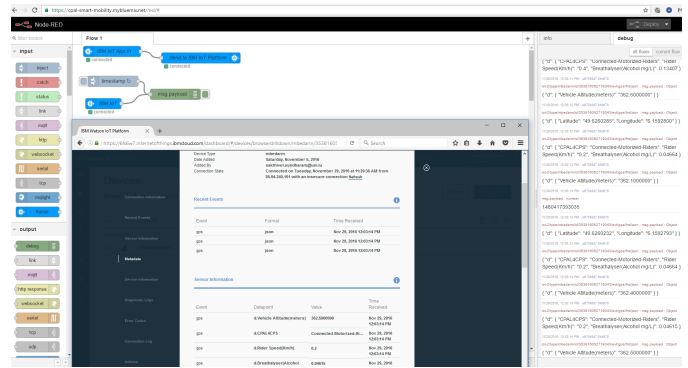


Fig. 4. Node-RED provides a browser-based flow editor that makes it easy to wire together the data flows coming from the devices using a wide range of predefined nodes. Once defined in Node-RED, the application is then deployed and executed in the IBM Watson IoT platform.

C. Information security mechanisms

Watson IoT Platform runs within the Bluemix cloud. Watson fulfills ISO27001 standard for the information security management processes. Security is achieved in Watson platform at many levels. Like mentioned in Figure 5, secured information management within the organization, secured device and application credentials enable strong information security. Device credentials with organization ID, device type, device ID (in our case IMEI number of the cellular modem), authentication token are integrated into the embedded program. Application credentials such as API keys and authentication tokens are also hard coded into the embedded program.

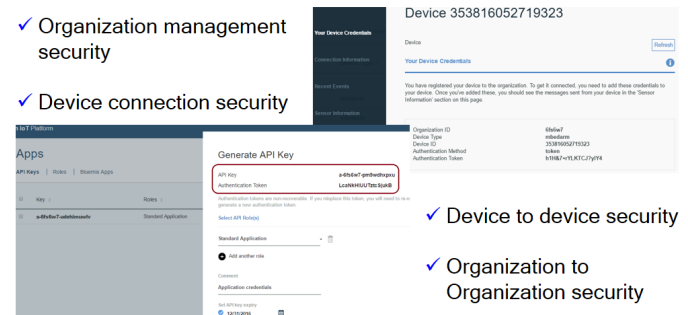


Fig. 5. Various levels of security mechanisms enabled in Watson IoT platform to ensure secured connections

D. Connectivity to internet

Since vehicles are moving objects wireless connectivity is required. Wireless LAN and cellular connectivity can be considered if power consumption of the object is not a main constraint. If the power consumption of the device is a significant requirement, then wireless connectivity is more efficiently achieved through low-power Wide Area Networks such as LoRaWAN and Sigfox. The product we prototyped can be viably deployed when low-power networks are established. Sigfox for instance uses ISM band 868MHz and comply with regulation ETSI300-220. Sigfox [4] wireless networks is meant to offer connectivity to objects such as electricity meters, smart watches and washing machines, which need

to transmit very regularly small amounts of data. During the Grand Challenge day of ISED2016, the demonstration of the product has been done with cellular (GSM) connectivity.

V. RESULTS AND OUTLOOK

The device has been built and tested on a real vehicle. Speed and location data, both on the device and in the cloud platform, were verified against the data captured on the dashboard and a smartphone respectively. Also, the sensitivity and response of the MQ-3 sensor to alcohol content has been verified. Since location and speed data are tracked online, it would be possible to identify overspeeding behaviors and possibly learn from the patterns of behaviors observed. This can be done anonymously or, if the Vehicle Identification Number (VIN) is mapped to the device ID, with the identification of the vehicle's owner.

To measure the air quality, the device is equipped with a MQ-7 sensor which measures the Carbon mono-oxide (Co) level in the air. Other emission gas sensors such as Co₂, Nox, can be interfaced to the device which would enable value added features such as real-time pollution monitoring in a particular area. The experiments presented in this work, where the data were transmitted from the device to the internet, can be extended to other use-cases where data are sent from the internet to the device. For instance, when the vehicle is near schools or hospitals, the driver could be requested to reduce the speed.

ACKNOWLEDGMENT

This research is supported by FNR (Fonds National de la Recherche), the Luxembourg National Research Fund (AFR Grant n°10053122).

REFERENCES

- [1] ARM mbed connector. <https://connector.mbed.com/>. Accessed: 2017-01-12.
- [2] Cyber Physical Action Language (CPAL). <https://www.designcps.com/model-based-design-with-cpal/>. Accessed: 2017-01-12.
- [3] IBM Watson IoT platform. <http://www.ibm.com/internet-of-things/iot-solutions/watson-iot-platform/>. Accessed: 2017-01-12.
- [4] Sigfox IoT network. <https://www.sigfox.com/en/coverage/>. Accessed: 2017-01-12.
- [5] WHO Global status report 2015. http://apps.who.int/iris/bitstream/10665/189242/1/9789241565066_eng.pdf?ua=1. Accessed: 2017-01-12.
- [6] L. Fejoz, N. Navet, S. Sundharam, and S. Altmeyer. Demo abstract: Applications of the CPAL language to model, simulate and program cyber-physical systems. In *2016 IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS)*, page 59, 2016.
- [7] N. Navet and L. Fejoz. CPAL: High-level abstractions for safe embedded systems. In *Proc. 16th Workshop on Domain-Specific Modeling*. ACM, 2016.