A YEAR OF INNOVATION IN ITS

This paper is split into two parts:

1. The Use of Internet-Of-Things (IOT), Open Source Hardware, Software and Transmission Media for Traffic Signal Monitoring
2. Project Eboracum – Chlorus – Vehicle to Infrastructure communications utilising ODBII, Bluetooth, Android and Roadside Beacons

The Use of Internet-Of-Things (IOT), Open Source Hardware, Software and Transmission Media for Traffic Signal Monitoring

Authors:

- Robert Whiteside, Independent

1 Aims

- To develop an open source system to monitor traffic signal controller status over LoRaWAN using IOT open source technologies and interface into third party system, such as Imtrac.
- Help expand the LoraWAN network by providing gateways and coverage in the region.

The outcome is not to produce a commercially viable product but to explore what is achievable with Open Source, IOT technologies.

2 Introduction

I recently became a member of The Sheffield Hardware Hackers and Makers, a group of hobbyist, students and professionals who through membership subscriptions, fund the running of a workshop at the Portland Works, Sheffield**.** The Sheffield Hardware Hackers and Makers teamed up with The Sheffield Things Network to host several Smart Cities events with the objective to expand LoRaWAN coverage within the Sheffield City Region (SCR).

LoRaWAN coverage of a region is dependant on the number of gateways present, these gateways are installed and owned by individuals or groups; the more gateways the better the coverage. SCR has a lack of LoRaWAN Gateways available for public use; the hope would be to place gateways at traffic signal sites, with ADSL backhaul, which could be used to expand coverage of LoRaWAN in the region while also providing a link to other signal sites without communication and monitor them over LoRaWAN. The group introduced me to the LoRaWAN and IOT technologies available and soon I could see areas where this technology could be implemented in a highways asset management environment.

3 Technologies

LoRaWAN is an open source, Long Range, Low Power, Low bandwidth network using communications media operating in the EU on 868MHz. Bandwidth is limited to 51 bytes per message and 30 seconds per day, but a radius of upto 10 miles is possible. Two way communication is possible but any download data is sent after data is uploaded to the network server. LoRaWAN's current uses include measuring air quality, monitoring wildlife, flood detection sensors, traffic movement and energy consumption.

The Things Network (TTN) is a global, crowdsourced, open, free and decentralized IOT network which provides infrastructure based on LoRaWAN to connect IOT devices. Devices connect in a star topology to LoRaWAN gateways which serve to forward packets to and from the TTN server. Each device is registered to the network and also to an application using unique 64 bit keys as shown in Fig. 1 below the network diagram; the payload message is also encrypted for security. The application on the TTN handles the communication and messages between devices and server.

MQTT is a machine-to-machine (M2M) IOT connectivity protocol with an extremely lightweight publish/subscribe messaging transport, for connections with remote locations where a small code footprint is required and/or network bandwidth is at a premium.

Node-Red is an open source visual editor for wiring IOT devices in software by linking together node modules to perform functions on message payloads. It allows for rapid development with minimal coding to relay message payloads from TTN using MQTT, to other application servers such as Imtrac's Remote Monitoring System (RMS) . Node Red also provides us with a means to test applications or route data to other MQTT brokers or subscribers.

4 Hardware

The devices are based on an Arduino Uno with LoRaWAN transceiver shield which interfaces with the traffic signal controller UTC reply bit interface to monitor controller, detector and lamp fault reply bits.

The Gateways are based on a Raspberry Pi 3 (RPi3) Single Board Computer (SBC) with a multi-channel LoRaWAN hat running a Debian operating system; an additional Pi 3 would also be used as the Node Red Server.

Figure 1 Shows how the technologies and hardware are networked together.

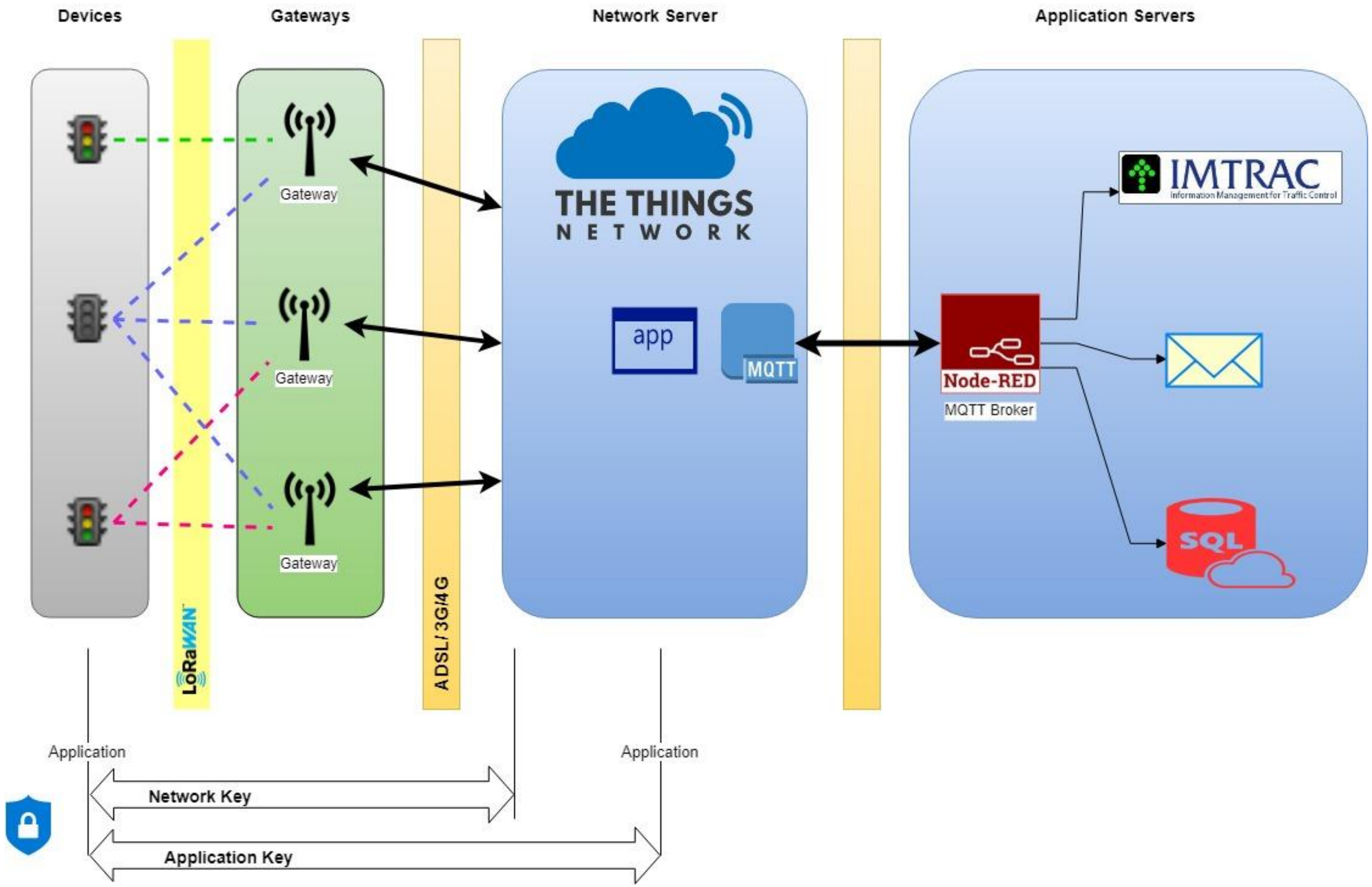


Figure 1 LoRaWAN network architecture and security

Project Eboracum: Chlorus – Vehicle to Infrastructure communications utilising ODBII, Bluetooth, Android and Roadside Beacons

Author:
- Peter Routledge, Ian Routledge Consultancy

1 Introduction

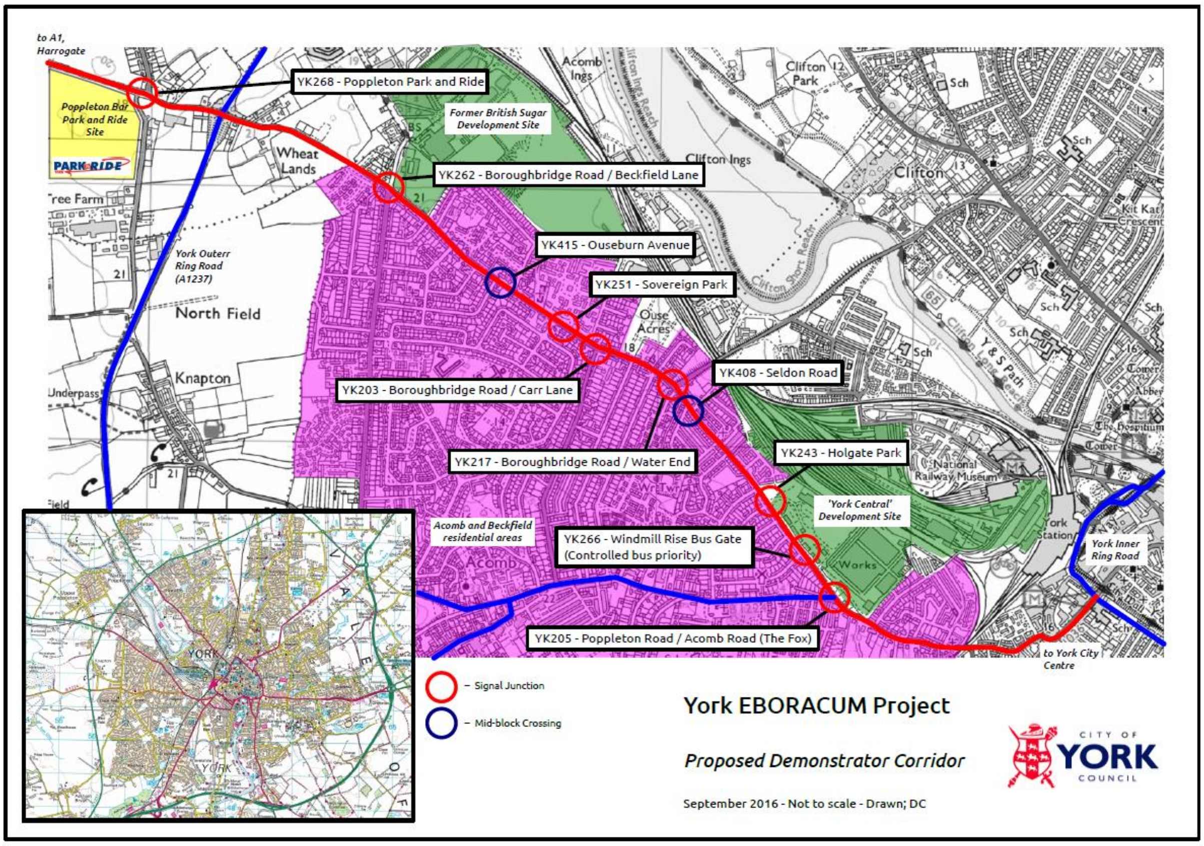
The Eboracum project funded by DfT under the C-ITS programme, aims to understand if and how probe vehicle data from both beacon and cellular approaches can improve a typical UK commuter corridor for all road users, including park and ride buses.

This paper describes a method of utilising existing technologies in an effort to provide low cost probe vehicles that in theory could be retrofitted into existing fleets.

2 Chlorus

In the spirit of York's Roman history and to tie in with the name of the project the Android application that was developed to facilitate communications between vehicles and roadside infrastructure has been named Chlorus after Constantius Chlorus who died in York on 25 July 306.

The study area incorporated the A59 corridor into York from the VMS located approximately 1.2KM to the West of the outer ring road to the traffic signal crossing located at Holgate Road / Chancery Rise a total distance of approximately 4.5KM. There were also 2 nodes in the city centre located in the vicinity of York Railway Station.



Equipment utilised to 'equip' vehicles

All equipment used is off the shelf and included:

a. Samsung Galaxy TAB A 7.0;
b. Modified Samsung Galaxy TAB A 7.0 – with wifi antennae modified to permit an SMA connector to facilitate an external aerial; and

c. OBDII (On Board Diagnostics 2) Bluetooth dongle.

App Development - Chlorus

From the outset it was opted to utilise Android devices to facilitate communications from the vehicle to the roadside beacon as:

a. The Android environment provided:
    - The lowest barrier to access with respect to development;
    - Greater access to elements of the Android kernel with regard to managing WiFi parameters from within an app;
    - Existing libraries for ODBII integration such as OBD-II Java API; and
b. Device acquisition costs were lower.

The app was developed in version 3 of Android Studio targeting versions Jelly Bean (4.1) through to Oreo (8.0). The intention was to rely on minimal third party modules however for ease the app was compiled with:

a. Google Play Services 15;
b. OKHTTP 3.8.1; and
c. Unirest 1.4.9.

A key requirement was that the app would not be interacted with whilst driving, thus although an output screen was provided whilst the app was running all data was also logged for review in the office.

The app itself was split into two parts:

a. Communications module to facilitate connecting to the IDT roadside beacons and transmitting data from the vehicle and the app to the AWS database; and
b. Vehicle interrogation module to manage the transmission over Bluetooth of requests and associated responses from the vehicle's ODBII port.

The app itself provided:

a. An overview screen that provides a visual output of the events the app is managing (Figure 1):
    - Startup OBDII data once the Bluetooth connection was established with the dongle and the vehicles ignition is started;
    - Access points in the vicinity (only those associated with the project (iEac)); and
    - The current location; and
b. Configurable settings that allow:
    - Core application parameters to be specified including end point, vehicle ID, etc (Figure 2); and
    - ODBII commands to be specified (Figure 3).

Figure 1

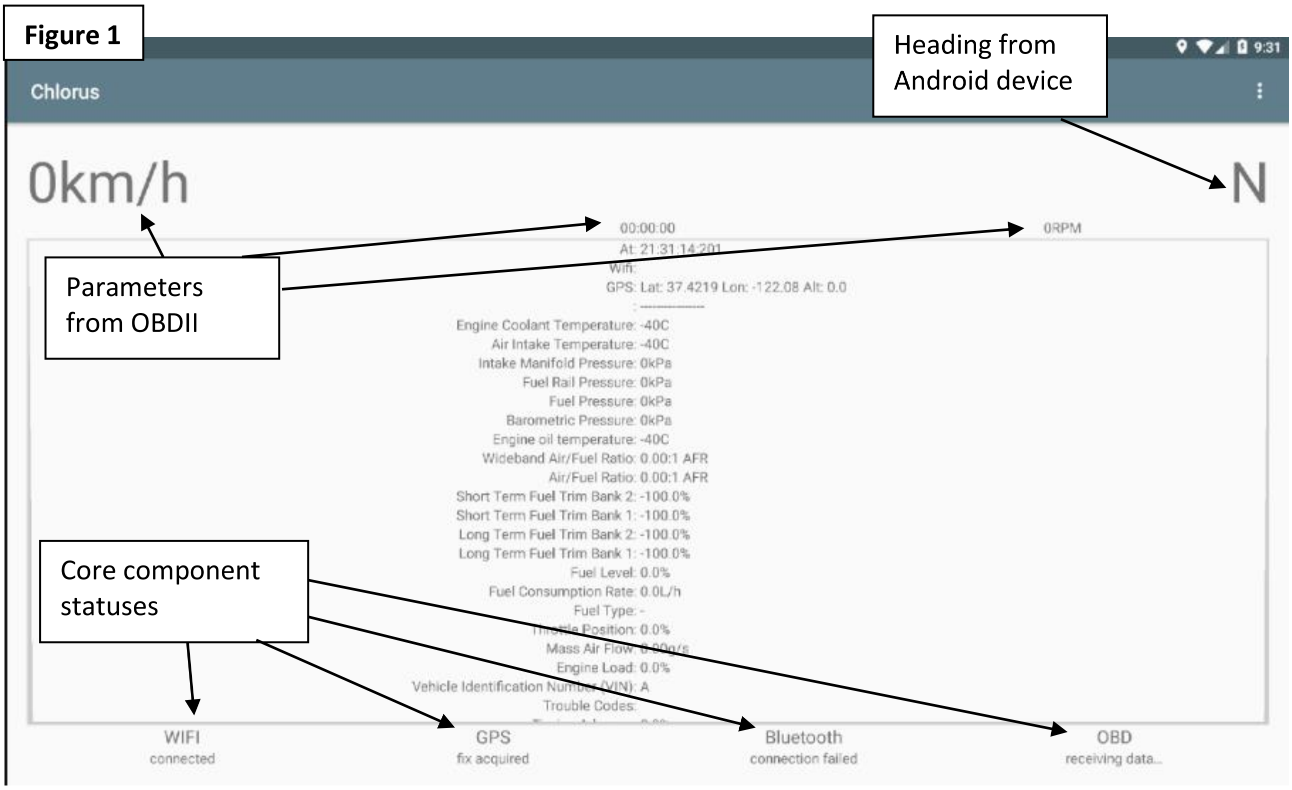


Figure 2

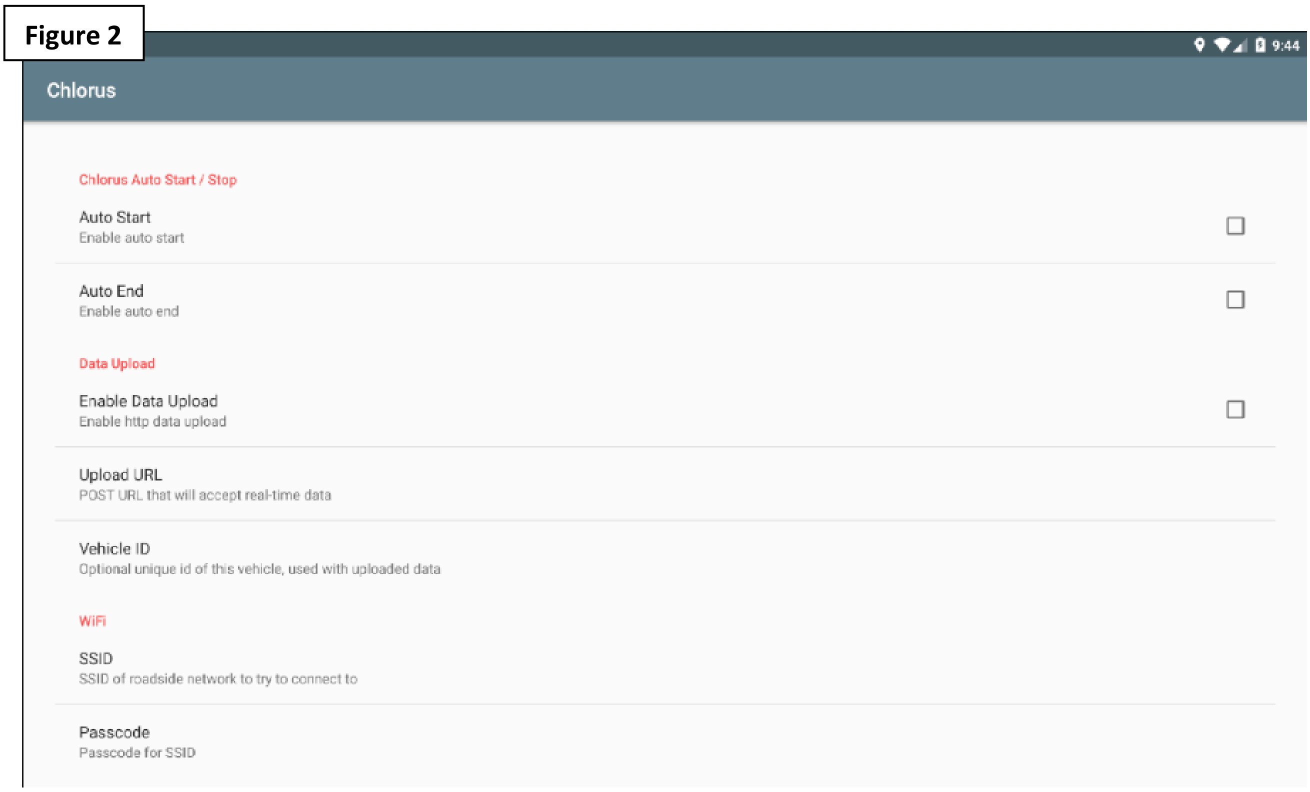
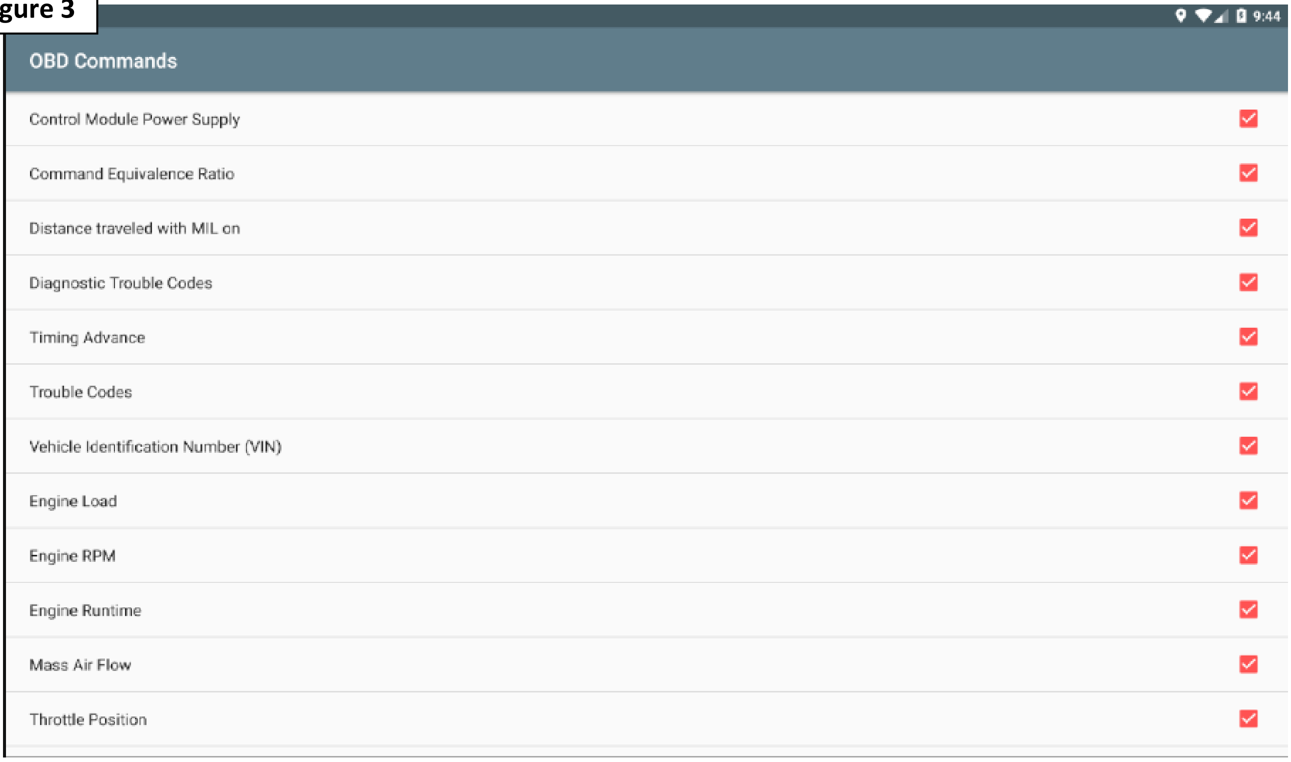


Figure 3



Roadside Beacons

The roadside beacons were IDT Mesh units (also used for journey time collection) that were installed at traffic signals, CCTV and lighting columns as required and then connected onto the CYC fibre network which then in turn routed out to the internet (to a limited subset of IP addresses).

All data from the app was transmitted as HTTP requests to the Amazon Web Services (AWS) infrastructure that has been used as the backbone of the project, primarily:

a. A micro EC2 instance running Ubuntu to process the received data; and
b. An RDS instance running PostgreSQL.

Data

Table 1 and Figure 4 show a single trip taken from West of the outer most node to the traffic signal controlled junction of the A59 / Beckfield Lane (YK-2262) a distance of 2km. The trip was undertaken on 13/07/2018 from 08:42 to 08:46 utilising a 2015 2.0L diesel Mercedes CLA.

Table 1 shows data transmitted at a given point (visualised in Figure 4) including:

a. Timestamp;
b. Connected node (and associated signal strength);
c. Location; and
d. Parameters retrieved over OBDII.

It is possible to extrapolate queueing from the data as can be seen from rows 617 to 632 in which the speed slows to a stop, dwells and then accelerates once the queue begins to discharge at green.

Note: Blank rows are present in the data due to the resolution that commands were being passed over the OBDII interface and the associated queueing of both the requests and responses.

Table 1

								Engine					Temperature				Pressure			Faults		
rec_id	user_id	Timetamp	BSSID	Signal Strength	Trip Ref	Lat	Lng	SPEED	THROTTLE POSITION	ENGINE RPM	ENGINE LOAD	Mass Air Flow Sensor	ENGINE OIL TEMP	ENGINE COOLANT TEMP	AMBIENT AIR TEMP	AIR INTAKE TEMP	FUEL RAIL PRESSURE	INTAKE MANIFOLD	BAROMETRIC PRESSURE	MIL STATUS	Distance Travelled MIL on	Trouble Codes
612	vy924	2018-07-13 08:42:14+00	06:15:6d:86:d6:0f	66	6512bd4	53.97511	-1.15919	67km/h					21C	27C	18C	21C	108500kPa	155kPa	101kPa			
613	vy924	2018-07-13 08:42:20+00	06:15:6d:86:d6:0f	37	6512bd4	53.97495	-1.15747		46.70%	1520RPM	48.60%	30.40g/s								DTC_NUMBER=MIL is OFF0 codes	0km	C0300
614	vy924	2018-07-13 08:43:58+00	02:15:6d:86:d6:43	46	6512bd4	53.97274	-1.14378		86.30%	0RPM	0.00%	3.15g/s								DTC_NUMBER=MIL is OFF0 codes	0km	C0300
615	vy924	2018-07-13 08:44:20+00	06:15:6d:86:ec:c1	59	6512bd4	53.97245	-1.14256	3km/h					31C	43C	18C	21C	27800kPa	101kPa	101kPa			
616	vy924	2018-07-13 08:44:27+00	06:15:6d:86:ec:c1	63	6512bd4	53.97245	-1.14256		45.10%	350RPM	0.00%	15.18g/s								DTC_NUMBER=MIL is OFF0 codes	0km	C0300
617	vy924	2018-07-13 08:44:47+00	02:15:6d:86:d6:43	52	6512bd4	53.97211	-1.14034	42km/h					32C	44C	18C	21C	69000kPa	116kPa	101kPa			
618	vy924	2018-07-13 08:45:14+00	06:15:6d:86:d6:06	41	6512bd4	53.97055	-1.13521	45km/h					34C	47C	18C	21C	35400kPa	101kPa	101kPa			
619	vy924	2018-07-13 08:45:21+00	06:15:6d:86:d6:06	41	6512bd4	53.97041	-1.13485		44.70%	781RPM	16.10%	5.56g/s								DTC_NUMBER=MIL is OFF0 codes	0km	C0300
620	vy924	2018-07-13 08:45:23+00	06:15:6d:86:d6:06	37	6512bd4	53.97037	-1.13475	8km/h					35C	47C	18C	21C	28400kPa	101kPa	101kPa			
621	vy924	2018-07-13 08:45:30+00	06:15:6d:86:d6:06	39	6512bd4	53.97035	-1.13467		10.60%	0RPM	0.00%	0.75g/s								DTC_NUMBER=MIL is OFF0 codes	0km	C0300
622	vy924	2018-07-13 08:45:35+00	06:15:6d:86:d6:06	28	6512bd4	53.97035	-1.13467	0km/h					35C	47C	18C	21C	20300kPa	101kPa	101kPa			
623	vy924	2018-07-13 08:45:38+00	06:15:6d:86:d6:06	28	6512bd4	53.97035	-1.13467		87.80%	0RPM	0.00%	0.00g/s								DTC_NUMBER=MIL is OFF0 codes	0km	C0300
624	vy924	2018-07-13 08:45:43+00	06:15:6d:86:d6:06	33	6512bd4	53.97035	-1.13467	0km/h					35C	47C	19C	22C	14400kPa	101kPa	101kPa			
625	vy924	2018-07-13 08:45:48+00	06:15:6d:86:d6:06	33	6512bd4	53.97035	-1.13467		87.80%	0RPM	0.00%	0.00g/s								DTC_NUMBER=MIL is OFF0 codes	0km	C0300
626	vy924	2018-07-13 08:45:52+00	06:15:6d:86:d6:06	33	6512bd4	53.97035	-1.13467	0km/h					36C	45C	19C	22C	10900kPa	101kPa	101kPa			
627	vy924	2018-07-13 08:45:56+00	06:15:6d:86:d6:06	37	6512bd4	53.97035	-1.13467		87.80%	0RPM	0.00%	0.00g/s								DTC_NUMBER=MIL is OFF0 codes	0km	C0300
628	vy924	2018-07-13 08:46:00+00	06:15:6d:86:d6:06	37	6512bd4	53.97035	-1.13467	0km/h					36C	46C	19C	22C	8700kPa	102kPa	101kPa			
629	vy924	2018-07-13 08:46:04+00	06:15:6d:86:d6:06	37	6512bd4	53.97035	-1.13467		45.10%	938RPM	59.20%	19.75g/s								DTC_NUMBER=MIL is OFF0 codes	0km	C0300
630	vy924	2018-07-13 08:46:08+00	06:15:6d:86:d6:06	50	6512bd4	53.97028	-1.13443	22km/h					36C	47C	19C	21C	94500kPa	123kPa	101kPa			
631	vy924	2018-07-13 08:46:14+00	06:15:6d:86:d6:06	50	6512bd4	53.97006	-1.13381		47.50%	1684RPM	9.80%	10.28g/s								DTC_NUMBER=MIL is OFF0 codes	0km	C0300
632	vy924	2018-07-13 08:46:20+00	06:15:6d:86:d6:06	68	6512bd4	53.97016	-1.13342	18km/h					37C	48C	18C	21C	37000kPa	101kPa	101kPa			

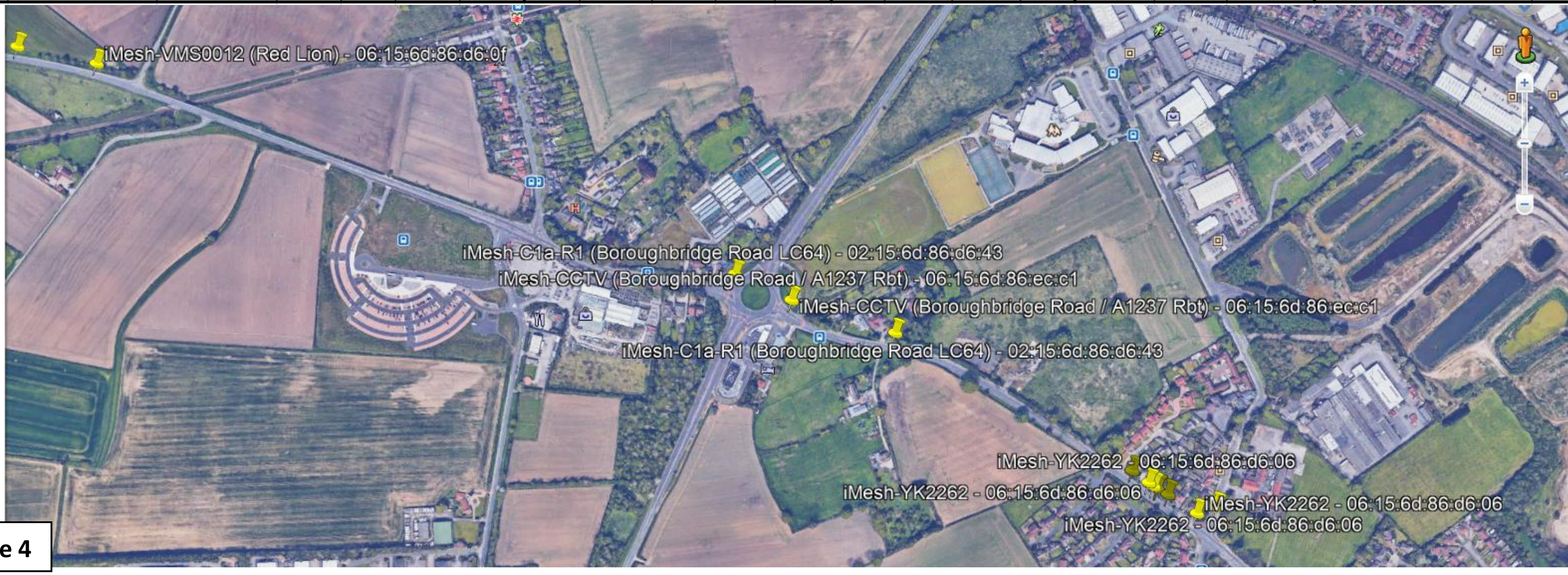


Figure 4

3 Lessons learned

The project has provided a unique opportunity to explore utilising existing technologies in an effort to provide low cost probe vehicles that in theory could be retrofitted into existing fleets. Some of the key lessons learned were as follows:

a. It is possible to utilise existing low cost technologies to collect data from probe vehicles;
b. Utilising off the shelf devices for connection to wireless nodes is not without challenges, particularly the relatively low gain antennae used in off the shelf hardware;
c. When travelling at speed the speed of vehicle combined with the time required for the Android device to connect and negotiate a connection can limit the number of successful connections
d. By default WiFi devices want to 'hang on' to an existing node as long as possible, additional configuration based on signal strength as well as other parameters needs to be employed to ensure optimum connections from a moving vehicle; and
e. Certain OBDII codes such as fuel type can be queried less frequently as they will not change.

In addition, whilst developing the application it was noted that Android ecosystem evolution may potentially limit feasibility for future deployment as it appears to be Google's intent to limit the level of access applications have to underlying functions such as WiFi.

4 What Next

Going forward It is hoped that there will be the opportunity to:

a. Explore the potential of greater COYC fleet penetration;
b. Expand the OBDII codes collected to include PIDs such as 8F Particulate Sensor Banks 1 & 2 etc;
c. Investigate the deployment of more beacons as part of York STEP project; and
d. Assess the potential of utilising the BSSID (its unique reference) of mesh node to pass data as a parameter to facilitate passing dedicated messages to road side intelligence unit (in conjunction with additional OBDII codes).

5 Acknowledgements

The author wishes to thank the following for their support:

- COYC Signals Team
- DfT (Graham Hanson and Darren Capes)
- IDT Ltd (Richard Gibson and Robin Jefferson)
- White Willow Consulting (Andy Graham, Jon Wade and Charles Hewson)

The views expressed in this paper are that of the Author and do not necessarily represent other parties within the Eboracum project.