

An IoT Based Predictive Connected Car Maintenance Approach

Rohit Dhall¹, Vijender Solanki²

¹Enterprise Architect, HCL Technologies, Noida, India

²Vijender Solanki, Research Scholar, Anna University, Chennai, India

Abstract — Internet of Things (IoT) is fast emerging and becoming an almost basic necessity in general life. The concepts of using technology in our daily life is not new, but with the advancements in technology, the impact of technology in daily activities of a person can be seen in almost all the aspects of life. Today, all aspects of our daily life, be it health of a person, his location, movement, etc. can be monitored and analyzed using information captured from various connected devices. This paper discusses one such use case, which can be implemented by the automobile industry, using technological advancements in the areas of IoT and Analytics. ‘Connected Car’ is a terminology, often associated with cars and other passenger vehicles, which are capable of internet connectivity and sharing of various kinds of data with backend applications. The data being shared can be about the location and speed of the car, status of various parts/lubricants of the car, and if the car needs urgent service or not. Once data are transmitted to the backend services, various workflows can be created to take necessary actions, e.g. scheduling a service with the car service provider, or if large numbers of cars are in the same location, then the traffic management system can take necessary action. ‘Connected cars’ can also communicate with each other, and can send alerts to each other in certain scenarios like possible crash etc. This paper talks about how the concept of ‘connected cars’ can be used to perform ‘predictive car maintenance’. It also discusses how certain technology components, i.e., Eclipse Mosquito and Eclipse Paho can be used to implement a predictive connected car use case.

Keywords — Internet of Things, Connected Cars, Predictive Maintenance, MQTT, Eclipse Mosquito, Eclipse Paho, Smart City.

I. INTRODUCTION

THE automobile and fleet management industries, majority of the consumers and the car service companies are following the ‘periodic maintenance’ for their automobiles. In periodic maintenance, car owners are advised to take their cars for regular service and maintenance either after certain specified time period or distance covered. For example, it is generally advised to get car serviced within three months of the last service date or after travelling 10000 kilometers, whichever comes first. Another instance where the car can be taken out for emergency service/maintenance is after some breakdown or malfunctioning of any part in the vehicle.

The way periodic maintenance works, is depicted in Fig. 1. [15]

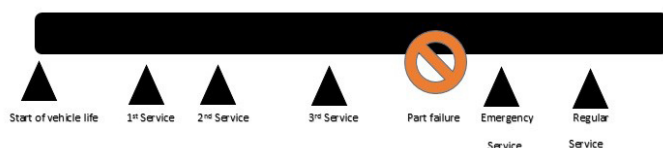


Fig. 1 Periodic Maintenance of Cars

A. Periodic Car Maintenance

Fig. 1 summarizes Periodic car maintenance. It can be explained as a service/maintenance model, where a car undergoes a service/maintenance either after a certain specified time period or on the basis of distance covered, e.g. as shown in Fig. 1, during the lifetime of a vehicle, regular services are carried out, as advised by the car manufacturer. Similarly, a car can be serviced/repaired, if there is any of the part gets faulty.

B. Drawbacks of Periodic Car Maintenance [15]

Some of the major drawbacks of periodic car maintenance are listed below:

- Higher cost of service, as vehicles are required to be get serviced as per the schedule
- Even if vehicle/parts are in perfect health, still service needs to be done and parts to be replaced
- No way of knowing, if a part needs immediate attention, and can result in breakdown of the vehicle
- This breakdown could cost significant charges for the car owners

II. ALTERNATE APPROACH TO PERIODIC CAR MAINTENANCE

Instead of getting a car serviced periodically, if a system developed using sensors and IoT [9] technology stack is used, which collect and analyze fitness and running condition of different parts of the car, and send this data to a centralized system. In this centralized system, data received from these connected cars, can be analyzed further and if any service is needed, a service request can be raised. This proposed system can also generate emergency alerts, in case any part is about to break down, thus avoiding car/part failure [15][17]

A. Advantage of Proposed system [18]

- Reduction in service and maintenance costs, as only parts which needs to be replaced or serviced
- Real time alerts of possible part failure, thus avoiding breakdown and costs associated with outages
- Analytics and reporting dashboards can be used to view how the car is performing over different periods of time and in different locations
- Driver’s driving habits can be analyzed and appropriate action can be taken
- Tour and cab providers can manage their fleet better, thus maximizing profits
- Target advertisements for monetization of data received from connected cars (e.g. offering service discounts for car which needs to be serviced etc.)

III. WHAT IS PREDICTIVE CAR MAINTENANCE AND WHY WE NEED IT?

A common practice, generally followed in automobile world is 'periodic car maintenance'. In this, the car is supposed to undergo periodic service and maintenance routine. When to get a car serviced, is generally decided by either a specified time period or distance covered. For example, it is generally advised to get the car serviced within six months of the last service date or after travelling 10000 kilometers, whichever happens first.

Now, the problem with 'periodic car maintenance' is that nobody is sure if any part or lubricants really needs to be serviced/replaced. This normally leads to parts/lubricants, which are in good condition, getting changed/serviced. Another problem, which is generally faced is that though scheduled 'periodic car maintenance' is still some time away, there is some problem with a part, which needs immediate attention. 'Periodic car maintenance' cannot solve this problem, and only way to know about this is after break down. So, two problems, associated with this model of car servicing can be summarized as:

- Higher service costs, as parts which are fine, will also be replaced
- Unable to generate any alert, if any part needs immediate attention/service, resulting in breakdown/outages

This is where 'Connected Cars' and 'Predictive car Maintenance' can help [15][16]. Connected cars can collect data, from different sensors installed in the car, related to the health status of different parts, and send it over the internet to backend applications, for analytical and decision making purposes. One of the backend analytical application, based on the health status of different parts, can invoke a workflow and schedule an appointment with the service provider, if some part needs immediate attention. Similarly, real time alerts can be sent to concerned parties, in case something need immediate attention.

This can result in considerable savings in terms of service and maintenance charges of the car [18]. Now, only the parts which actually need replacement, will be serviced. This data will be collected and transmitted by different sensors fitted on the car for performing health check of different parts oil health check, tire and pressure health check, filters health check and so on.

IV. HOW THIS WORK DIFFERS FROM OTHER WORK DONE IN THIS AREA

Good amount of research work is done on the Predictive maintenance topic [1][4][6]. Some of the work talk about how to collect or read sensor data (from cars, from manufacturing industrial machines etc.), or propose a model to perform predictive analysis and so on [2][3][5][7]. This work proposes an IoT based approach [15] to collect this data, send it to the cloud and perform predictive analytics on this huge amount of data. The proposed approach is based on industry proven protocols and products, some of which are Eclipse IoT's [11] Eclipse Mosquitto [13], Eclipse Paho[12] and MQTT[10] protocol. High level IT architecture is also provided, so that same can be referenced by people to build, extend and further improve the system based on this architecture.

Finally, a simulation of the proposed architecture model is also given, where a client GUI utility simulates the car sensor, and sends data to the cloud.

V. PROPOSED TECHNOLOGY FOR IMPLEMENTING PREDICTIVE CONNECTED CAR MAINTENANCE

This section introduces some of the important technology components, which will be used in the proposed implementation of 'Connected car' use case for predictive maintenance.

A. MQTT Protocol

Message Queue Telemetry Transport (MQTT) is a light-weight messaging protocol based on publish-subscribe model. MQTT uses a client-server architecture where the client (such as a sensor device on cars) connects to the MQTT server (called a broker) and publishes messages to server topics. The broker forwards the messages to the clients subscribed to topics. MQTT is well suited for constrained environments where the devices have limited processing and memory resources and the network bandwidth is low.

B. Deeper look into MQTT

MQTT is an extremely lightweight messaging protocol. Its publish/subscribe architecture is designed to be open and easy to implement. Single MQTT server can support up to thousands of remote clients. These characteristics make MQTT ideal for use in constrained environments where network bandwidth is low or remote devices that might have limited processing capabilities and memory, need to be supported. The MQTT protocol is based on publish/subscribe model. Publishers can send the messages to the topics, configured on the MQTT server (also called MQTT broker). Clients can subscribe to these topics and receive whatever messages are published on those topics.

Fig. 2 depicts the publish/subscribe model of MQTT

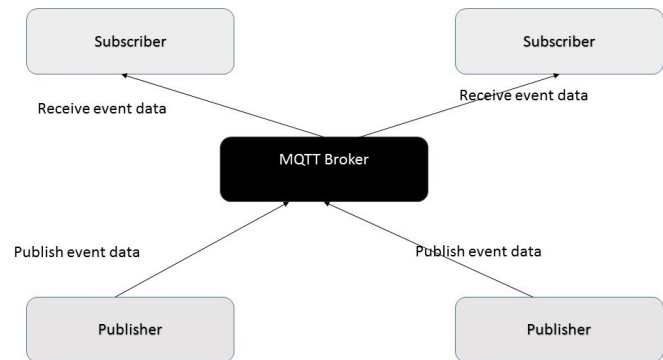


Fig. 2 Publish/Subscribe model of MQTT.

Though MQTT's publish-subscribe model is identical to any existing enterprise messaging systems, the main advantage of MQTT has over fully blown "enterprise messaging" systems are that its low footprint makes it ideal for developing IoT applications with small sensors, devices and other low-capacity things. For example, Facebook uses MQTT for its messenger product on the mobile platform, to ensure that battery usage of this application is minimized.

Some of the major advantages of MQTT are listed below:

- Publish Subscribe model provides one-to-many message delivery
- Uses TCP/IP for network connectivity
- Can work with SSL/TLS for security
- MQTT offers three message delivery QoS: 1) at most once ,2) at least once and 3) exactly once
- These QoS are met even in case of network, publisher or client failures
- Very simple specification and APIs, making it easier for developers to work with MQTT based products
- Most important APIs are CONNECT, PUBLISH, SUBSCRIBE, UNSUBSCRIBE, and DISCONNECT
- As MQTT is specifically designed for constrained device, it provides only the bare minimum features to support them.
- The message header is short in MQTT and smallest packet size in 2 bytes, making it ideal for small and constrained devices

- As MQTT is a publisher/subscribe model, sender and receivers are decoupled from each other
- Doesn't restrict the format of data to be in any particular format, thus allowing flexibility
- 'Last Will' feature, which allows abnormal client/sensor termination to be notified to all interested parties
- Both commercial and open sources MQTT based broker products are available. These include IBM WebSphere MQ v 7.1 onwards, EclipseIoT Mosquitto, ActiveMQ and HiveMQ.

C. Eclipse Mosquitto

Eclipse Mosquitto is an open source MQTT broker/server. Based on the lightweight MQTT protocol, Mosquitto is ideal for devices, sensors and other 'Internet of Things' devices, with low processing capacity. MQTT clients can connect to a given Mosquitto broker and publish/subscribe the messages from a topic.

Eclipse Mosquitto's main responsibility is to provide a communication channel between publishers/senders and subscribers/receivers. Any publisher, using the Eclipse Paho MQTT Client API can publish the messages to an MQTT Broker. These MQTT clients should specify the topic, on which they want to publish the message. These topics are configured on MQTT broker. Any subscriber or receiver, that want to receive the message, subscribe to that particular topic. It is the responsibility of the broker to deliver all the messages arriving on a topic to all interested clients. As different clients (both publishers as well as subscribers) need to know only broker/topic details, both are decoupled from each other. This architecture pattern has many advantages, e.g. highly scalable solution, where subscribers needn't to be overwhelmed by publishers sending messages at a rate faster than what a subscriber can process.

D. Eclipse Paho

Eclipse Paho is an EclipseIoT project and is implementation of MQTT protocols. Eclipse Paho provides MQTT client libraries in multiple languages including Java/C++, C#, .NET and Python. Eclipse Paho also has utilities for MQTT-SN (sensor networks). Both publishers and subscribers (as shown in Fig. 2) can use API's provided by Eclipse Paho MQTT Client library, and send/receive messages to/from MQTT broker (e.g. Eclipse Mosquitto).

VI. WHY MQTT AND OTHER PROPOSED TECHNOLOGY COMPONENTS IN CONNECTED CAR IMPLEMENTATION

- Suitable for low capacity devices like sensors fitted on connected cars
- Provides Quality of services to handle connectivity and other errors, which can be quite common in the case of cars and automobiles, which are on the move, and n/w connectivity can be an issue
- Supports wide variety of languages, so compatibility will not be an issue for any existing technology platform of a car manufacturer
- Also integrated with proven and well adopted industry leading messaging systems like WebSphere MQ and ActiveMQ
- Message formats can be customized, allowing manufacturer to customize and innovate the solutions

Details of MQTT protocol and its specification can be found on the MQTT site, given in the reference section.

VII. PROPOSED ARCHITECTURE OF 'PREDICTIVE CAR MAINTENANCE' USING ECLIPSE MOSQUITTO AND ECLIPSE PAHO

Fig. 3 shows the simplistic high level architecture context diagram

of a system implementation of 'predictive car maintenance' using Eclipse Mosquitto and Eclipse Paho.

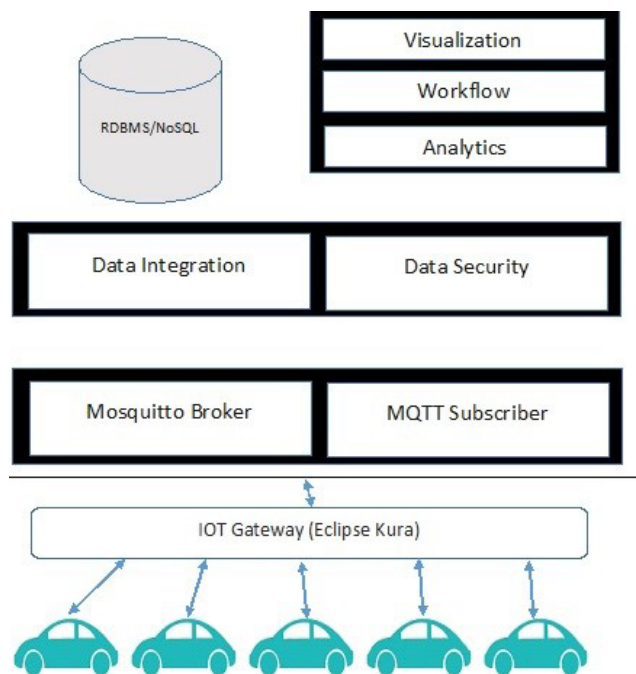


Fig. 3 Architecture diagram of 'Connected car' ecosystem.

Flow of context diagram (Fig. 3) can be explained as follow:

- 'Connected Cars' send data in predefined format to IoT Gateways like Eclipse Kura.
- Cars can use any possible way to send the data, i.e. via Wi-Fi, Telco services etc.
- IoT Gateway would send this data to MQTT based Eclipse Mosquitto Broker hosted in a cloud environment
- In many scenarios, there can be additional components like coordinator/controller nodes in the architecture, which do some kind of pre-processing/aggregation of data collected from various devices, before sending it to the cloud
- Once data is received by Eclipse Mosquitto, subscribers will receive this message, using Eclipse Paho API
- After doing basic validations and any data conversion, subscribers can send this message to downstream systems, using services exposed by the Data Integration component of the architecture
- It could be that these messages are sent to a messaging system, e.g. Apache Kafka, where these messages can be consumed by the workers
- These workers can push these messages to Apache Hadoop or other such data processing service, for analytical purpose
- These messages can also be consumed by data processing services, handling real time stream of messages e.g. Apache Storm
- These real time stream of messages can be used for real time analytics purpose e.g. sending alerts in case something needs immediate attention

'Workflow' component can be used to define and execute different workflows, based on some conditions

For example, to schedule a service appointment, invoke a REST based service of the car service provider, in case some parts need to be replaced.

There can be various visualization tools to view reports of summarized data and perform analytical queries.

Note that, in real life complex scenarios, there can be many more components involved in the architecture (e.g. configuration services, policy manager, rule engine and so on), but for simplicity's sake, those have not been included in this paper.

Fig. 4 summarize the high level data flow of the proposed system.

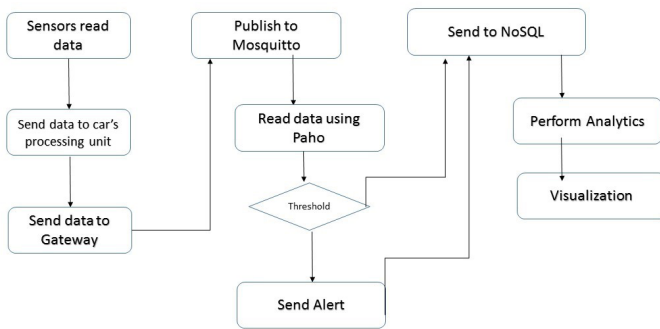


Fig. 4 Sample data flow for connected car.

VIII. SAMPLE IMPLEMENTATION OF PREDICTIVE CAR MAINTENANCE USE CASE WITH ECLIPSE MOSQUITTO AND ECLIPSE PAHO CLIENT UTILITY

In this section will use Eclipse Paho Client utility to simulate a connected car, which will send city where the car is (can send exact location also), speed and current car health check, including if any part needs to be replaced or not.

In the real world, devices like connected cars might be sending data first to a IoT gateway, as shown in architecture diagram in Fig. 3 in last section, where this data will be processed and aggregated, before being fed to further downstream applications. Depending upon the actual requirement, applications can be designed to take appropriate actions based on the data being received e.g. Send alerts if speed is too high or schedule an appointment with the car service agency, if some part/lubricants needs to be replaced.

Sample format of the MQTT message, transmitted by sensor fitted on the car is shown on Fig. 5.

```
CarId=xxxxxx;Location=XXXX;ItemName=<<Value>>;currentStatus=<<value>>
```

Fig. 5 Sample MQTT message format, sent by connected car

Some of the information, which can be sent by the connected car is shown in Table 1.

TABLE I.
LIST OF SAMPLE PARAMETERS WHICH ARE MONITORED

Parameters
Location
Mileage per litre
Quantity of fuel consumed
Total Distance Covered
Trip Distance
Distance covered in top gear
Top Speed
Fuel Level
Current temperature
Coolant Level
Engine Oil level
Fuel Level

In this paper, we will be simulating the behavior of connected car, using Eclipse Paho MQTT Utility, a Java Swing based GUI application, to connect to a Mosquitto server, publishing message to a topic on Mosquitto server. The client will receive this message and for simplicity, will display this message content in the GUI.

Once the utility is launched, screen as shown in Fig. 6 will appear. Specify the address of the MQTT Mosquitto server and port and connect to the server. We are using a cloud based test Mosquitto server, available for public, at the address “test.mosquitto.org”, port 1883.

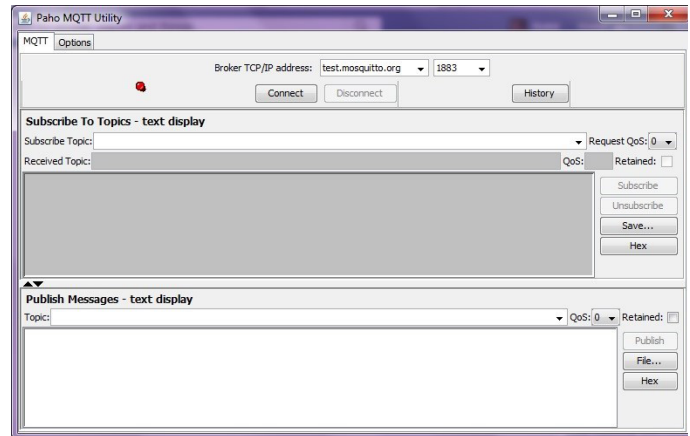


Fig. 6 Connecting to MQTT server using Paho MQTT utility.

After specifying the address, click “Connect”. If everything goes right, you will be connected to server, else you will get error message.

Once connected, enter the name of the topic, and click subscribe. Now, any message, published on this topic will be displayed in the text area of the subscriber. Fig. 7 shows a subscriber, connected to a topic named ‘CarHealth’.

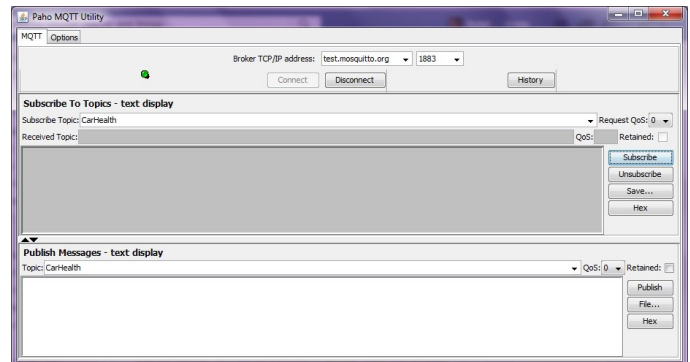


Fig. 7 Subscribing to a MQTT topic.

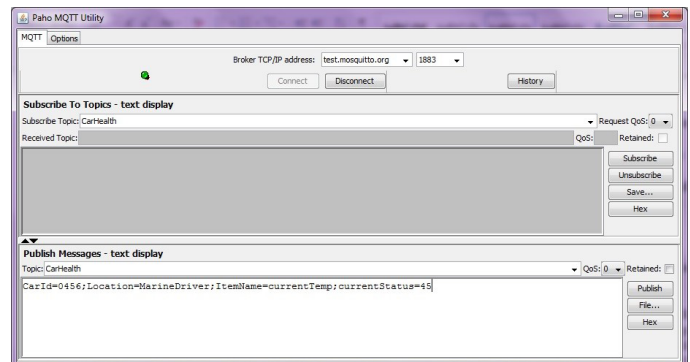


Fig. 8 Publishing a message to MQTT topic.

Enter the message payload in the ‘Publish Messages – text display’ area and click publish. Fig. 8 shows the step to publish a message onto MQTT topic.

Once the message is published, all clients who have subscribed to this topic, will receive this message. In our scenario, this will be displayed in the utility GUI. Fig. 9 shows the scenario of a subscriber receiving the message.

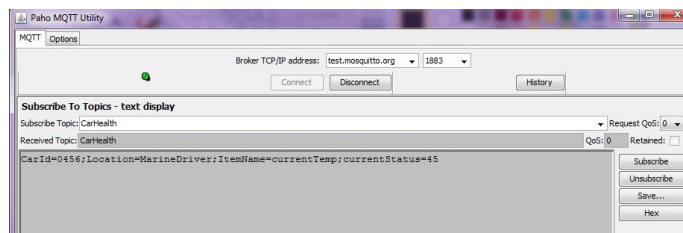


Fig. 9 Subscriber receiving the message

IX. COST BENEFIT OF PREDICTIVE CAR MAINTENANCE

Predictive car maintenance can help address the issues of traditional periodic car maintenance approach. Some of the advantages it provides are

- Reduction in service and maintenance costs, as only parts which needs to be replaced are serviced
- Real time alerts of possible part failure, thus avoiding breakdown and costs associated with outages
- Analytics and reporting dashboards can be used to view how the car is performing over different periods of time and in different locations
- Driver’s driving habits can be analyzed and appropriate action can be taken
- Tour and cab providers can manage their fleet better, thus maximizing profits
- Target advertisements for monetization of data received from connected cars (e.g. offering service discounts for car which needs to be serviced etc.)

Table 2 shows the cost comparisons of periodic vs predictive maintenance for a medium sedan car. Periodic service cost figures are taken from a leading automobile web site (see reference). As most of the car vendors provide first two services as free services, zero cost is taken for these two services. For predictive maintenance, it is assumed that service cost will go down by 30%.

TABLE II. SERVICE COST COMPARISON OF PERIODIC VS PREDICTIVE CAR MAINTENANCE

Service	Cost (Periodic Maintenance) in INR	Cost (Predictive Maintenance) in INR
1st Service	0	0
2nd service	0	0
3rd service	2465	1726
4th service	6455	4519
5th service	3835	2685
6th service	6455	4519
7th service	3835	2685
8th service	6455	4519
Total	29500	20653

Sample calculation for 3rd service is as follow

Cost of 3rd periodic service – 2465 INR [21]

Cost with predictive maintenance with 30% saving = 2465 * ((100-30)/100) = 1726 INR

Fig. 10 represents another view of the service cost comparison data of Table 2. During the first eight services (scattered across 5 years) of the car, total costs incurred on periodic maintenance is 29500 INR. This cost will come down to 20653 INR (assuming 30% reduction in service cost), with the help of predictive car maintenance.

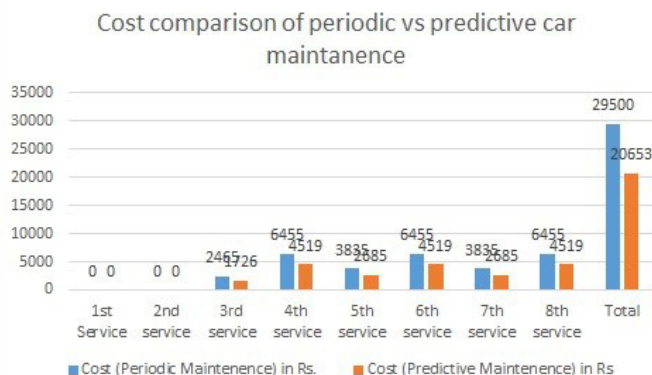


Fig. 10 Service cost comparison of periodic vs predictive car maintenance.

To understand the outage costs of a fleet/transport organization, let us take an example of a company with a fleet strength of 100 cars. If a car of such commercial organization is out of service because of breakdown or faulty part, there will be various costs associated with it e.g. pay for an unutilized driver and support staff, rental of the car for that day, fixing and service cost, need to ensure alternate vehicle for the customer to ensure company’s commitment, else loses on reputation part in the consumer market and so on.

Assume that the total outage cost for one vehicle going out of service is Rs. 5000 per day. So, for a company with one vehicle out of service, associated costs can be calculated as follows

Outage cost of one vehicle going out of service – 5000 INR (A)

Annual cost of one vehicle going out of service – 5000(A) * 365 = 1825000 INR (B)

Table 3 shows the outage cost per year for multiple number of vehicles going out of service on a given day. For a company with a fleet size of 100, number of vehicles going out of service can be much higher, but for simplicity, table 3 shows costs for maximum three vehicles going out of service.

TABLE III. OUTAGE/BREAKDOWN COST FOR A FLEET/TRANSPORT COMPANY

No of Vehicles	No. of out of service vehicles	Outage Cost per year(in INR)
100	1	1825000
100	2	3650000
100	3	5475000

Fig. 11 represents another view of the outage/breakdown cost data of Table 3. Assuming that per day cost of a vehicle going out of service is 5000 INR. As shown, on average with only one vehicle out of service, cost per year is 1.8 million INR and with three vehicles going out of service, this will increase to 5.4 million INR per annum.

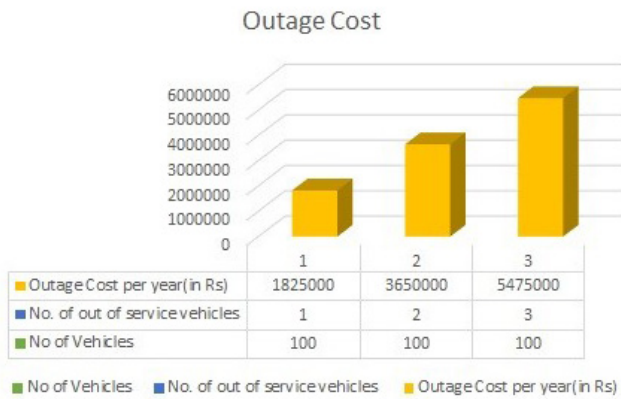


Fig. 11 Outage/Breakdown cost for a fleet/transport company.

Table 4 shows the outage cost comparison with 30% improvement in outage scenarios.

Original Outage cost of 1 vehicle out of service = 1825000 Rs (B)

Cost after 30% improvement in outage situations = 1825000 * ((100-30)/100) = 1277500 Rs.

TABLE IV BREAKDOWN/OUTAGE COST COMPARISON

No of Vehicles	No. of out of service vehicles	Outage Cost per year(in INR)	Outage cost @30% improvement
100	1	1825000	1277500
100	2	3650000	2555000
100	3	5475000	3832500

Fig. 12 represents another view of the outage/breakdown cost comparison data of Table 4. Even 30% assumed reduction in downtime will provide considerable savings for the organization. For three vehicles out of service, cost of breakdown will come down from 5.4 million INR to 3.8 million INR.

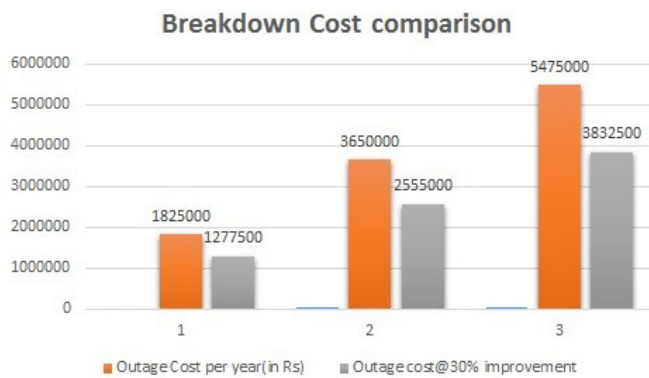


Fig. 12 Breakdown/outage cost comparison.

Fig. 13 depicts a sample portal/dashboard to view the status of a given car. Any authorized person can view, whether any part needs replacement or not, current status of different parts, historical information, including details of any alert that was sent.

For example, the second row in this dashboard shows that an alert was raised for a particular part. Though the current value of this part is within a valid range (between 0-1), but as it is almost on the threshold to breach the value, a pro-active alert was sent, thus avoiding any possible outage and associated costs



Fig. 13 Sample dashboard for viewing parts/items health.

This dashboard can also be used to trigger additional workflows. For example, a request to book a service appointment with the car service provider can be raised using this portal.

X. CHALLENGES IN THE PREDICTIVE MAINTENANCE OF CONNECTED CARS

Some of the challenges in successful implementation of predictive maintenance and connected cars are as follows [8]:

- Govt. and Regulatory policies, restricting on what kind of sensitive data can be transferred
- Security concerns related to location and other sensitive data being shared and transmitted
- Lack of industry standards. Right now, most of the work done is vendor specific/proprietary
- Need to have a proper IT Analytics system in place. Can involve huge costs upfront
- Need better connectivity in terms of telecom, Bluetooth, Wi-Fi and other such networks for transmission of real time data from sensors
- Associated business use cases are still evolving, so justifying initial costs can be difficult
- With higher number of sensors needed on the car, cost of buying a new car can go up

XI. CONCLUSION

‘Connected car’ concept is getting lots of traction with automobile companies these days. There are multiple benefits of ‘Connected Car’ ecosystem, and one such benefit is Predictive Car Maintenance. This paper talked about what predictive car maintenance is all about, which problems it could solve. MQTT, a popular protocol for IoT is also discussed, followed by an introduction to Eclipse Mosquitto and Eclipse Paho, an implementation of MQTT.

This paper also presented a high level architecture of how Connected car use can be implemented, using Eclipse Paho and Eclipse Mosquitto. A simulation of ‘connected car’ sending sensor’s data to the cloud is also discussed. Finally, cost saving of a predictive car maintenance system over a traditional periodic car maintenance system is shown. This paper concluded by sharing of some of the challenges in implementing predictive maintenance of connected cars.

REFERENCES

- [1] Kevin A. Kaiser ; Cerner Corp., Kansas City, MO ; Nagi Z. Gebrael ,Predictive Maintenance Management Using Sensor-Based Degradation Models, IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans (Volume:39,Issue: 4),pp 840-849 ,2009
- [2] D. C. Swanson ; Appl. Res. Lab., Pennsylvania State Univ., University Park, PA, USA ,A general prognostic tracking algorithm for predictive maintenance , Aerospace Conference, 2001, IEEE Proceedings. (Volume:6) pp 2971 - 2977 vol.6 ,(2001)
- [3] A. Grall ; Lab. de Modelization et Surete des Systemes, Univ. de

Technologie de Troyes, France ,L. Dieulle , C. Berenguer and M. Roussignol ,Continuous-time predictive-maintenance scheduling for a deteriorating system,IEEE Transactions on Reliability(Volume:51 ,Issue: 2),pp 141 - 150 , 2002

- [4] Stabler, L,Hawthorne, K(2004) FIX IT BEFORE IT'S BROKE, Railway Age , Volume: 205, Issue Number: 9 ,pp. 83-84,2004-9
- [5] J. Endrenyi , Ontario Power Technol., S. Aboresheid , R. N. Allan , G. J. Anders , The present status of maintenance strategies and the impact of maintenance on reliability,IEEE Transactions on Power Systems(Volume:16 ,Issue: 4),pp 638 - 646 ,2001
- [6] Joel Levitt,Complete Guide to Predictive and Preventive Maintenance ,Industrial Press, Inc.; 2 edition (June 15, 2011)
- [7] Joseph S. Ng,Automated wireless preventive maintenance monitoring system for magnetic levitation (MAGLEV) trains and other vehicles <http://www.google.com/patents/US5445347>, 1995
- [8] Why Do Predictive Maintenance Programs Fail? - http://reliabilityweb.com/articles/entry/why_do_predictive_maintenance_programs_fail/
- [9] What is Internet of things - <http://whatis.techtarget.com/definition/Internet-of-Things>
- [10] Details about MQTT protocol - <http://mqtt.org/>
- [11] Eclipse IoT - <http://iot.eclipse.org/>
- [12] Eclipse Paho - <http://www.eclipse.org/paho/>
- [13] Eclipse Mosquitto - <http://projects.eclipse.org/projects/technology.mosquitto>
- [14] Eclipse Paho MQTT Client API - <http://www.eclipse.org/paho/files/javadoc/index.html>
- [15] IoT, Analytics & Cars – Joe Speed - <https://mobilebit.wordpress.com/>
- [16] Practical MQTT with Paho- <http://www.infoq.com/articles/practical-mqtt-with-paho>
- [17] IoT and Predictive Maintenance- <http://blog.bosch-si.com/categories/manufacturing/2013/02/iot-and-predictive-maintenance/>
- [18] The Smart and Connected Vehicle and the Internet of Things - http://tf.nist.gov/seminars/WSTS/PDFs/1-0_Cisco_FBonomi_ConnectedVehicles.pdf
- [19] IBM Predictive Maintenance and Quality for automotive - <http://www.novemba.de/wp-content/uploads/IBM-Predictive-maintenance-and-Auality-for-automotive.pdf>
- [20] The advantages of implementing a predictive management within the maintaining and equipment repair at an enterprise which produces components for the automotive including technical assistance and services - <http://eccsf.ulbsibiu.ro/repec/blg/journl/5317sima.pdf>connected cars - use cases for Indian scenario - <http://www.hcltech.com/white-papers/engineering-and-rd-services/connected-cars-use-cases-indian-scenario>
- [21] Maruti Swift Diesel Estimated Maintenance Cost - <https://www.cardekho.com/maruti-swift/service-cost.htm>



Rohit Dhall is working as an Enterprise Architect with Engineering and R & D Services,HCL Technologies,India. He has over 19 years of software industry experience. He helps global clients build technical solutions to solve their complex business problems. His main area of expertise is architecting, designing and implementing high performance, fault tolerant and highly available solutions for leading Telco and BFSI organizations. He has worked on diverse technologies like java/J2ee, client-server,P2P ,DWH,SOA, BigData and IoT etc. He regularly writes articles, blogs and white papers for various IT forums, portals and events. He is also a coauthor of IBM Redbook and Redpaper on 'ITCAM for WebSphere'.



Vijender Kr. Solanki, Ph.D is a research scholar in the department of Computer Science and Engineering at Anna University, Chennai. He has completed his graduation and postgraduation (B.Sc., M.C.A and M.E) from institution affiliated with Maharishi Dayanand University, Rohtak (MDU) Haryana, India in 2001, 2004 and 2007 respectively. He has attended an orientation program at UGC-Academic Staff College, University of Kerala and a refresher course at IIIT-Allahabad. He has participated in more than 15 seminars, summits and conferences at various national & international levels, including IIT-Delhi, Bharathiar University, Coimbatore and Anna University, Chennai. He has published more than 10 technical papers with IEEE, Springer and Elsevier- Science-Direct library. His research interest includes smart city, network security and network management. He is having 08 Years of rich academic experience. He has delivered many technical lectures in various institutions including AICTE Sponsored SDP-FDP Lectures at SKNCOE, Pune, SNS College, Coimbatore, ITS Ghaziabad and DAVIM, Faridabad. He was an invitee as key note speaker in DST Sponsored seminar at RCEW, Jaipur. He has chaired session in many conferences. He is reviewer of some of the IEEE, Springer and Elsevier Journals and Conferences which are indexed in Scopus, DBLP, ACM Digital library. He is also a book editor with Universities Press.