ISSN 1870-4069

Proposal of an Interconnection Management Model and Availability of Internet Access for Things (MGID-IoT)

Chadwick Carreto, Jhovanny Pasaran, Alfonso Fernandez

National Polytechnic Institute, School of Computer Science, Mexico
ccarretoa@ipn.mx, jepasarang@gmail.com,
alfonso.fernandez.v@gmail.com

Abstract. This paper proposes a Model for the Management and Strengthening of Quality of Service (QoS) in the aspects of Interconnection and Availability of Access for networks of nodes and their application in the Internet of Things, it is a Model of Connectivity that will allow the services provided by a network of Mobile Nodes to be most of the time available to users when they require it. This is done through a layered model that implements between each layer a Quality of Service mechanism that allows in the first instance to measure the performance in that layer and secondly to establish an improvement or correction action according to the layer. The proposed Model aims to be endowed with a standard character to be implemented in different Services Architectures, which will allow optimizing the aspects of availability, connectivity, reconnaissance and monitoring of the state of the network, link between nodes and user connectivity.

Keywords: quality of service (QoS), internet of things (IoT), interconnection, availability, node networks.

1 Introduction

Mobile networks are evolving to different types of applications, providing new services to users and supporting new communications protocols, it is necessary to have Quality of Service (QoS) architectures that guarantee reliable data delivery, in time and form [1].

The subject of QoS research in mobile networks has been of great interest and models and architectures have been proposed that aim to provide a solution to this problem, however, there are still few model options that are compatible for the Internet of the Things (IoT). It is for this reason that the present work proposes a new layered QoS model designed to be compatible with any type of new generation mobile network and applicable to IoT schemes.

This QoS model is based on different metrics for the interconnection and network model according to each layer. It proposes to analyze and review the layers of: Applications, services and communications. Metrics will be done with existing and free tools in order not to generate a consumption of extra resources in the network. This model is implemented in a mobile node architecture in order to validate the model by analyzing the information and data generated.

The following sections will describe the state of the art of the networks and the aspects of QoS and from Section 4 will describe the proposed model. Finally, a series of conclusions and future work will be made.

2 New Generation Networks

New generation networks have become a topic of great interest not only academically but also in the industry. These networks are characterized by high transfer rates to support various applications [2]. As time has passed the mobile networks have undergone changes in their architectures and also in the way they process information, these changes are called: generations of mobile communications.

One of the main characteristics of each generation is the way in which the mobile device accesses the channel or medium in the network, for example for a third generation network (3G) a code division medium access scheme is used (CDMA), while the fourth-generation (4G) network uses the Orthogonal Frequency Division Multiple Access (OFDMA) technique. For the fifth generation (5G) networks there is still no definite, so far the research done has simulated networks that could be considered as candidates for a 5G network. Another aspect that is important to consider is how QoS is performed in each of these generations [3].

3 QoS in Next Generation Networks and IoT

Mobile communication systems base their QoS architectures on three main focuses [4]:

1. **Best Effort:** It is the model applied to any network that does not have policies explicitly defined as the Internet. It does not guarantee any treatment or specific resource to any information flow. Every package is treated equally; there is no preferential treatment.
2. **Integrated Services:** Implementation model of low-demand service that aims to guarantee resources available along a route for a specific application. Before starting the application session properly, the route is signaled to verify the availability of the necessary resources for an adequate development of the same. It allows to guarantee the operating conditions of critical applications.
3. **Differentiated Services:** It is a model of implementation of resources guaranteed in generic way and not by flows or sessions. It allows to guarantee different service conditions for different types of traffic, in a much more scalable and effective way, throughout the network.

The resource allocation is done jump-by-jump on each device in the network and not for a specific route. However, the implementation mechanism is relatively complex.

QoS services in IoT are considered to be end-to-end, meaning that they go from one end-to-end device, for example a mobile device connected to another through a network wireless [6].

Each end-to-end service has its own QoS which is provided to each user within the network, therefore the user is the one who can decide whether or not he is with the assigned QoS. To guarantee the QoS in a network, a bearer service with defined characteristics and functionalities is adjusted in the network from the source of the service to the final destination [7].

Each service carrier includes all aspects to enable the provision of contracted QoS. The QoS architecture in IoT is a layered architecture based on service bearers, where each bearer in a specific layer offers an individual service provided by the lower layers.

4 QoS Model for IoT

One way to solve this problem is through a layered QoS model, the metrics will be made based on the parameters that each layer has. Fig. 1 shows the proposed model of this research work. The main characteristic of this model is that it will be developed in three layers:

As can be seen in the figure this model consists of three layers where QoS metrics will be made. In the application layer will measure which applications are most used. For the services layer, the types of services offered on the network will be obtained, ie whether they are real-time or not. Finally, in the communications layer, the metrics will be performed for: jitter, latency, bandwidth, packet loss, etc.

All metrics will be performed using existing tools in a network architecture of mobile nodes that will validate the model.

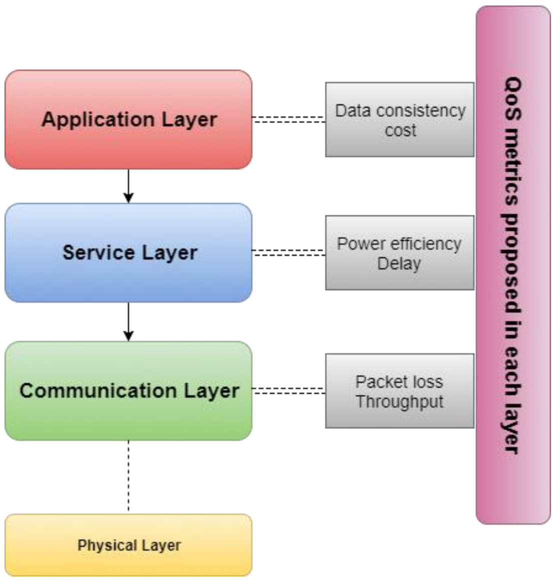

Fig. 1 Layer QoS model proposal for new generation networks

Applications Layer. It refers to the metrics of the application levels. Some classification schemes for applications that require QoS levels [5] can be presented as follows:

- Elastic Applications: These are the ones that can adapt their operation and quality parameters of the services on the "best effort" scheme. In these applications we do not see a very close human interaction with communication. Examples are the transfer of archives, or sending mail. Technologically, these applications do not require special parameters such as a specific bandwidth or a delays limit. Since in the case that they have a small bandwidth, it will only take more time to be sent, the transfer time will be longer or even exist mechanisms of data recovery and lost packets, correction of errors that are characteristic (but insufficient for other services) of the networks.

– Non-Elastic Applications: In contrast to previous applications these applications do require that they be guaranteed some QoS parameters. Real-time applications, for example, require bandwidth and a minimum level of delays for transmission stability. Although some of these applications show some ability to adapt to changes in quality parameters of services. (Without fail to ensure the same).
– Interactive Applications: Typically, a human interacts in this communication scheme, usually on an endpoint device that requires a fast response so that another user, device or host node can perform well. The parameters that are generally required to guarantee a high quality of services are strict limits for a low delay time, minimizing the error rate etc. Examples of these applications are: Voice over IP (VoIP), videoconferencing, online collaborative applications, online games, and so on.
– Non-Interactive Applications: Human direct collaboration is not required here to perform successful transmission. Nor are strict quality parameters required in the service. For example, you can minimize the low bandwidth with previous storage of the files or components. If audio or video streaming is required, it can be done with a slight delay (without storage) without this meaningful problem. Examples of these applications are: WEB browsing, file transfer, chats or preloading of audio or video content (streaming).

Services Layer. It refers to the metrics of the services offered by the network, see Table 1.

Table 1. Services can be classified according to [6].

SERVICES
Real Time
Signaling and Control
Data
Best Effort

– **Signaling and Control Services.-** This type of service is intended for traffic that supports network control, such as routing protocol traffic, and traffic signaling for voice and video , in particular this class must have a guaranteed bandwidth for its correct operation.
– **Data Service.-** This class of service will be designed for the highest priority data such as data bank transfers, FTP / SFTP services, etc.
– **Best Effort Service -** This kind of service will be devoted to data or video traffic that is for entertainment or game-oriented. Some applications for example are: mail servers, online games, music and video on the web, etc.

Communications Layer. It refers to the metrics in the communications channel as: latency, jitter, [6], etc.

- **Latency.-** Refers to the total time that elapses since a data packet is transmitted from a source node until it is received by the destination node, this parameter is measured in units of time. In real time applications such as video and voice it is necessary that there is a minimum level of delay in order to obtain a good quality of the application.
- **Jitter.-** In telecommunications jitter is called the variability of the execution time of the packages. This effect causes some packets to arrive too soon or too late to be delivered on time.
  - The jitter or delay variation can be caused by different factors such as:
  - Different packets may have different deadlines in queues on the same network device.
  - Different packets may have different processing times on the same network device with different delay times.
  - Different packets can travel through different network paths and delays would accumulate at different queue times and propagation delays.
  - From the perspective of users, jitter or lack of it, has repercussions on the consistency or consistency of the applications. However, while it is a low and constant level, applications can be adaptive.
- **Packet loss.-** It is the measurement of packets that have not been transmitted successfully on the network in relation to all packets sent on the network. Usually detected via ARQ (Automatic Repeat-reQuest) methods, there are four main causes of losing packets on the network.
  - Due to poor media quality either through physical or electromagnetic interference (often on wireless media).
  - Due to the congestion of links causing buffer overflow on the used network devices.
  - Faults in network devices.
  - Changes in the routing scheme or network protocols causing loss and damage to packets.

The loss of packages is reflected in the quality of presentation of the applications, such as: a good sound in an audio or sharpness if it is a video image.

5 QoS Model Test Architecture

The Quality of Service model was tested in the following non-two mobile architecture that aims to provide security monitoring in the halls, laboratories, offices, etc. of the National Polytechnic Institute. See Fig. 2.

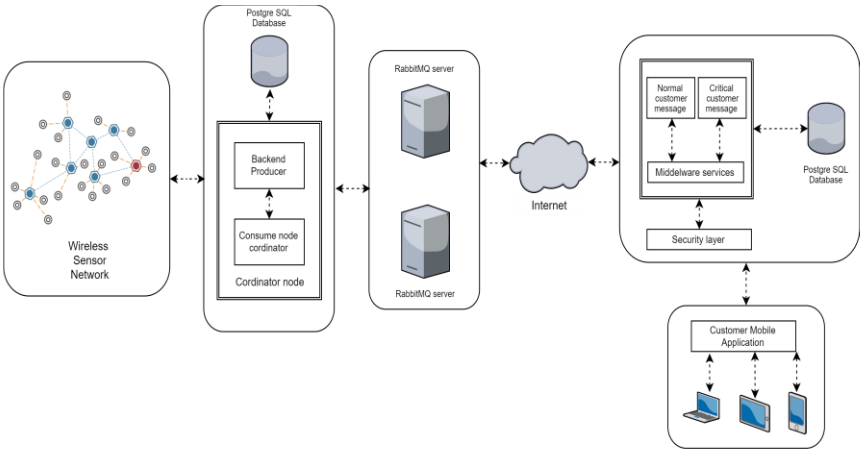

Fig. 2. Mobile node architecture proposed.

Architecture is divided into five important parts: 1) Xbee Network; 2) The coordinating node; 3) RabbitMQ Servers; 4) The middleware, and 5) The Web application (Client).

Xbee Network. The xbee nodes will be located at the doors of each school and through a proximity sensor they will be monitoring if any doors open at an inappropriate time. As shown in Fig. 3 there must be a coordinating node to which the other nodes are connected, and this node will be the gateway that will distribute network traffic to the next part of the architecture.

Node Coordinator. The coordinating node consists of a small server that will be in charge of receiving all the information that is being censored by the whole network of Nodes, so there will be a module called "Consumer Node Coordinator" that will have two important functions the first to perform the register of all the information of each one of the nodes, this will be done by means of a PostgreSQL database where the detailed information of the node will be stored (serial number, variable to be census, Possibly the frequency in which it is located, location, active) so that it will serve to receive each of the configurations by the user (Customer), the second function will be to receive all that data and then send them through the "Ba-ckend / Producer / Consumer". In the case of "Backend / Producer / Consumer" will be responsible for establishing two types of communications:

- **Producers:** It will establish two communication channels, one to send the data that are sent by the nodes and another channel to be able to send messages of high priority which can be sent given several situations (one node was decompressed, the base of data stops working).
- **Consumers:** In this case, a single communication channel will be established, which will be responsible for receiving the configuration data of the nodes, that is to say that some of its operation can be altered, such as to disable it, give it even priority or simply to return location).

RabbitMQ Servers. This part of the system consists essentially of the servers that will be raised by RabbitMQ which are mainly the "Tracker" that will be in charge of redirecting each one of the messages to its different destinations, in this case two or more are proposed (depends on the number of schools required) that will have two main functions:

- **Server 1 (Important):** This server is very important because it will be responsible for redirecting those messages that are very important, when there are already high-impact situations (a node was shut down / decomposed, the data base left to operate).
- **Server 2 (Data):** This server will mainly be used to manage all the information that will be registered by the nodes.

Middleware. This middleware is a fundamental part because it will take care of receiving / sending all the messages that are necessary for the system, this middleware is composed by 5 parts:

- **Consumers:** There are 3 connections that will remain connected to the servers in order to be able to receive all the data that is sent from "Backend", specifically two are required so that the work can be distributed in both "Consumers" and thus any of the two stacks are saturated, although the connection is more focused to the channel where those alerts or problems will be sent by the coordinating node.
- **Producers:** This will be in charge of sending all necessary information on the configuration of the nodes, so it is necessary only a single Producers since it is a task that will not be used continuously.
- **Saisis Services:** This is an MVC framework made for Node.Js that will help us primarily to provide all necessary API Rest services to be able to interact with the client, but in turn to empty it into a NoSQL database, as in the case of the CassandraDB database, so you will be interacting directly with the 3 Consumers and the Producer in order to send / receive information.
- **Security (Authentication):** It is a layer that will be responsible for establishing a security layer between the SailsJS services and the Client, this will be done through an integration that has the same SailsJS Framework in this case the pro-tocolo that will be used , OAuth 2.0 that consists of the handling of tokens that are valid for a certain time and that are used to be able to accede to the SailsJS API, reason why only valid users will be able to request to the server a valid token and with only certain privileges.
- **CassandraDB database:** This database will store all information related to the system, ie profiles, users, tokens, etc., including the same information of the nodes, this CassandraDB case is considered for solve problems of scalable systems, so it will be useful when handling a large amount of information.

The Web application (Client). This application will be made using Angular 2, considering two important aspects, the first is the robustness that will be necessary for the system so Angular 2 along with TypeScript will be a great help to be part and the second is that it allows us to have greater benefits for the case of mobile devices, even give an experience to the user that it is a mobile application when it is a web application.

Here are the possible profiles that the client application will have:

1. User Manager.
2. Node Manager.
3. Monitoring User.

Where the same system will contain different modules:

1. Module for User Administration.
2. Node Management Module.
3. Notifications module (to warn users of major incidents).
4. Module for generating reports.
5. Module for monitoring the different nodes and variables.

6 Conclusions

The main reason for proposing and implementing this model is that it can become a flexible and robust medium or method that, at the moment of being implemented within the context of a mobile network or an IoT system, allows users to have an "Acceso "to the corresponding information, services and / or applications wherever and whenever they need it, in a more efficient and reliable way. The Model is not intended to work with a specific access technology, however for our case study we will choose to work with a single IoT Access Technology. At the moment and with what it has mounted has proved to be a viable option with multiple application options.

References

1. Viswanathan, H., Weldon, M.: The Past, Present, and Future of Mobile Communications. Proc. Bell Labs Technical Journal 19, 8–21 (2014)
2. Wunder, G., Boche, H., Strohmer, T., Jung, P.: Sparse Signal Processing Concepts for Efficient 5G System Design. IEEE Access 3, 195-208 (2015)
3. Andrews J. *et al*.: What Will 5G Be? IEEE Journal on Selected Areas, Communications 32(6), 1065–1082 (2014)
4. Agiwal, M., Roy, A., Saxena, N.: Next Generation 5G Wireless Networks: A Comprehensive Survey. IEEE Communications Surveys & Tutorials 18(3), 1617–1655 (2016)
5. GSMA Intelligence: Understanding 5G: Perspectives on future technological advancements in mobile. White paper (2014)
6. Solomitskii, M.: About Quality of Service concept's usage at delivery of messages streams circulating in convergent telecommunication network. In: Microwave and Telecommunication Technology (CriMiCo), 23rd International Crimean Conference, Sevastopol, pp. 430–431 (2013)
7. Rajveer, H., Hemant, S.: Comparative Study of Different Mobile Generation Technologies and Various Challenges in Integrating 4G and 5G. IEEE Communications Magazine 42, 165–187 (2016)