< CONTAINER MONITORING SYSTEM >

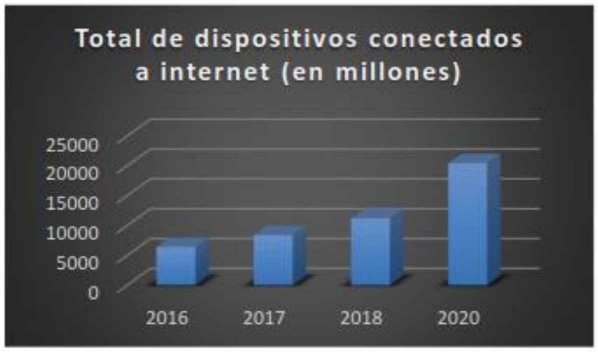
WWW.PARADIGMA.CL
INFO@PARADIGMA.CL

OCTOBER 21, 2018

Contenido

1. ABSTRACT

In 2017 a total of 8,400 million devices were connected to the internet, which represents an increase of 31.3% over the previous year. It is estimated that in 2020 a total of 20,415 million devices will be connected to the network, which shows a trend towards connectivity of an increasingly large range of devices. The use of connected devices in companies in 2017 stands at 1.6 million units, some 300,000 more units than in 2016. Likewise, the total expenditure related to connectivity has reached more than 2 billion US dollars in 2017 [1].



Evolution of the total number of devices connected to the internet

In this area of connectivity, highlights the so-called Internet of Things (IoT), a concept that refers to the connection of everyday elements to the network to have more control over them and move towards a type of intelligent devices.

One of the fields in which IoT has been used with greater interest and success, is in the domain of application of logistics and transport, because it is intended to take advantage of this higher level of interconnectivity of elements to achieve safe, reliable and in real time of the processes of freight transport. In addition, these advances will allow a greater level of rapport with the clients of these services, since they will be able to know details of their products transported in real time and without the direct intermediation of the logistics company. It is necessary to emphasize that maritime transport represents 80% of world trade, or what is the same, 80% of world trade is done through maritime containers. These data make interesting the implementation of an IoT project for the remote management of the characteristics of the containers, minimizing possible logistical errors and increasing control over the transported products, especially for refrigerated containers or reefers, since they need to maintain the values of certain parameters (such as the temperature inside them) between thresholds.

The work that is going to be done takes as its objective the creation of a container management system, using the tools of the IoT platform, LPWAN Networks, Smart Contract for the Bill of Lading, and Blokchain and Cloud storage, where you can consult all the data stored in real time related to each container. Additionally, the data should be provided to users in an easily supported format for Big Data Analytics tools.

For the realization of the Global Connect project, it has been necessary to carry out a study of the current state of the art of existing technologies, such as the communications standards for networks of type LPWAN (Sigfox, LoRa, WeightLess, NB-IoT, ...), existing IoT platforms in the market (FIWARE, OpenIoT, ...), Blockchain and Smart Contract platforms (Blockstack, Kadena, ...) and their adaptability to the public cloud (Google Drive, AWS, Dropbox, ...), as well as the sensors that they could be placed in a container or existing container monitoring technologies.

2. STATE OF THE ART

2.1 Sensors

To measure the different characteristics of a cargo container, it is necessary to place several sensors. There are several types of sensors in the market, but by carrying out a research work in the field, it have been selected the most appropriate types of sensors for monitoring the characteristics of a container, which a brief description is presented.

2.1.1 Position and deviation of the route

One of the most important aspects in the monitoring of cargo containers is undoubtedly the control of the exact position in real time of the container. This allows the user to know where the merchandise is at each moment and to control the path of said container. Another important aspect that allows the monitoring of the position of the container is the control of the route of the vessel that transports the container, since it is possible to detect if there has been a deviation of the initially planned route that could affect the estimated time of arrival at the port (ETA). The most used system for position monitoring is the GPS (Global Positioning System), since it allows to know the position of an object on Earth with a precision of a few meters by communicating with a network of 24 satellites using the method of trilateration [2].

There are other systems for position determination, such as the European navigation system Galileo or the Russian GLONASS system, but GPS technology is currently the most used. Another possibility would be the determination of the position through the internet, but it is a less precise option than the technology of position determination based on satellite constellations. In summary, the best option is the placement of a GPS receiver for the monitoring of the position in the containers. A valid GPS module could be the Adafruit Ultimate GPS HAT, which has a sensitivity of -165 dBm, 66 channels, a consumption of 20mA, real-time clock (RTC) and connector for an external antenna. This GPS module is designed to be placed in an embedded system, such as a Raspberry Pi [3].

2.1.2 Temperature

The control of the temperature inside a cargo container is a very important aspect, especially if the container transports goods sensitive to changes in temperature, such as food or pharmaceutical products.

For temperature monitoring it is necessary to place a temperature sensor that allows to know at all times the temperature inside a container. Currently, the most commonly used types of temperature sensors are the thermoresistor, the PN junction sensor and the thermocouple, the latter being the most accurate. A thermocouple consists of two different conductors that contact one or more points, which produces a voltage when the temperature at one of these points differs from the reference temperature in other parts of the circuit.

There are different types of thermocouples, depending on the combinations of metals that have been used for their manufacture. The most common thermocouple is the K type, because it has a wide range of temperatures and a low cost [4]. The type K thermocouple is formed by a positive conductor of nickel-chromium and a negative conductor of nickel-aluminum. According to the reference tables of a K-type thermocouple, its temperature range covers from -200 to 1250ºC, enough for the range of temperatures to be measured inside a container [5].

2.1.3 Humidity

In addition to temperature, another of the most influential environmental factors in the state of merchandise that transports a container is humidity. The important parameter to measure is the relative humidity in the interior, which measures the amount of water in the air inside the container in the form of vapor compared to the maximum amount of water that can be maintained at the temperature at which this has been done. The measurement results in a numerical value in percent. Currently there are several types of humidity sensors: mechanical, conductivity, condensation, capacitive, infrared and resistive polymer block. Capacitive sensors are the most commonly used, since they are suitable for environments with high temperatures and for applications that require a high degree of sensitivity to low humidity levels. These sensors are designed by placing dielectric material between two electrodes, so that a small capacitor is formed. The variations due to the presence of water vapor in the dielectric constant of the dielectric material produce a variation in the capacity value of the capacitor. The sensor SHT7x (RH/T) of Sensirion is a sensor capacitive of type pin that measures the relative humidity in the atmosphere, with a temperature of operation of -40 to 125ºC, a time of response of 8 seconds and a precision with an error ±3% in the relative humidity measurement [6].

2.1.4 Concentration of Gases

The level of concentration of certain gases, such as oxygen (O2), carbon dioxide (CO2) or ethylene (C2H4), can have a great impact on the merchandise inside a container, especially if the container transports fruits and vegetables. These foods continue to perform the process of breathing even after being cut from the plant, consuming oxygen and expelling carbon dioxide, this means that by monitoring the levels of O2 and CO2 inside the container, you can know how the process is progressing of maturation of the transported food. The use of sensors capable of measuring the levels of O2 and CO2 in the environment (in parts per million -ppm-) are necessary to perform this type of monitoring. Another interesting aspect related to this type of sensors is the possible automation of the container to maintain optimal levels of O2 and CO2 to slow down the ripening process of the fruits and vegetables transported. This would be achieved by the expulsion of CO2 and the introduction of O2 automatically into the container.

2.1.5 Vibrations

The detection of strong vibrations is an important aspect in the monitoring of containers because a strong vibration or prolonged exposure to these vibrations can directly affect the integrity of the content stored by the container. These phenomena become especially sensitive if the contents transported are dangerous goods. The vibration sensors are piezoelectric accelerometers that allow working at high frequencies, since vibration is produced at these frequencies. Accelerometers are devices that measure the acceleration or vibration of an object. The force generated by the vibration causes the mass to compress the piezoelectric material, which generates an electric charge that is proportional to the force that has been exerted on the material [7]. The unit of measure for the vibrations are meters per second squared (m/s2).

2.1.6 Impacts

In addition to vibrations, another factor to consider for greater protection of the integrity of the merchandise transported in a container is the impact control. The collision between two containers or with any element that can be potentially dangerous if dangerous goods are transported, so it is convenient to record any impact to prevent them more efficiently. The most used impact sensors are formed by a simple electrical circuit with a switch, in this way, the circuit will remain open and will only close when an impact is detected, which will produce an electrical signal and the collision will be recorded. By using simple impact sensors, it is possible to detect these shocks easily to obtain a record of collisions during a journey.

2.1.7 Proximity

In the same order as the sensors of vibrations and impacts, in the containers that transport dangerous goods it is convenient to install proximity sensors, being able to detect the presence of elements in a certain radius of the container and avoiding possible impacts with these elements, which it could be the case that it was another container. A proximity sensor is a transducer that detects objects located at a certain distance from the transducer. There are several types of proximity sensors: inductive, capacitive, position switches, photoelectric and ultrasonic sensors. Ultrasonic proximity sensors are a low cost and quite reliable alternative, since they can detect objects at a distance of up to 8 meters. Its operation is as follows: the sensor emits a sound and measures the time it takes to return. In addition, they can detect objects of different shapes, surfaces and materials. An example is the RS Pro ultrasonic sensor, a low-cost sensor, easy to place in an embedded system or a plate, and with a range of detection ranging from 0.2 to 4 meters away [8].

2.1.8 Door opening and seal integrity

In order to guarantee the integrity of the merchandise transported by a container, it is critical to ensure that the merchandise has not been handled during transport of the container. This can be achieved through the use of sensors to detect the opening of the doors of the container, allowing to know the exact moments in which the container has been opened. Currently, there are numerous magnetic sensors on the market that detect the opening of a door. Its operation is simple, the sensor consists of two components and, when the door is closed, a magnet creates a magnetic field on the other part of the sensor. This magnetic field will disappear when the door is opened. Another possible alternative would be the placement of a luminosity sensor. When a container is closed, inside there is hardly any light, so that opening its doors would cause a sudden change in the luminosity values (measured in lux) that would allow detecting the opening of the doors.

2.1.9 Weight

Among some of the objectives, its implementation responds to an international measure that protects the safety of navigation on ships, associated with the stowage or distribution of the cargo they transport.

In addition to the weight of the container it is interesting to measure the eccentricity, that is, to detect the location of the center of gravity of the loaded container. The warning and alarm levels can be set based on the eccentricity [32].

2.2 Communications technologies for LPWAN networks

One of the most important parts in the development of a system based on the Internet of Things is the communication between the different devices of the platform, especially the communication between the sensor networks of the containers and the final devices or backends that will establish communication with the cloud platform (TMS).

To meet the needs of IoT, several Low-Power Wide Area Networks (LPWANs) technologies have been developed, which are characterized by offering long-range wireless connectivity to a large number of devices, lower energy consumption and lower cost than mobile networks. being its main limiting the reduced bandwidth.

2.2.1 Sub 1GHZ

Sub 1Ghz RF operates in the ISM spectrum bands below Sub 1Ghz – typically in the 769 – 935 MHz, 315 Mhz and the 468 Mhz frequency range. This spectrum band below 1Ghz is particularly useful for RF IOT applications.

- ❖ *Why is the Sub 1Ghz RF preferred to the 2.4Ghz RF for IOT applications?*

Applications for IOT or the internet of things, are typically low power. The Sub 1Ghz offers several advantages over the 2.4Ghz spectrum for IOT applications.

Range of Sub 1Ghz wireless: Sub 1Ghz offers more range than the 2.4Ghz. If range is an important criteria for an IOT product, then the Sub 1Ghz is a better choice. Sub 1Ghz wireless transmission offers 1.5 to 2 times more distance coverage than the 2.4Ghz spectrum. Also, the Sub 1Ghz wireless spectrum has a long-range mode that can have more than 100 km coverage for a IOT application.

Lower Power Consumption: Wireless Sub 1 Ghz RF needs a lower power signal from the transceiver compared to the 2.4Ghz spectrum to get the same output power signal at the receiver. This makes the sub 1Ghz RF a great choice for battery operated IOT sensor devices.

Interference: IOT Sensor products using the wireless Sub 1Ghz spectrum can handle interference better. This is because they operate on a lower frequency with fewer existing applications using that spectrum. Also, the lower frequency ISM bands enable the Sub 1Ghz transmissions to weave between buildings in an urban environment better.

❖ *Sub 1Ghz Range: The over-reaching feature of the Sub 1Ghz spectrum band*

The superior range of Sub 1Ghz is probably the best feature of this spectrum band. Typically, many wireless applications fail only because all the nodes cannot be reached by the existing protocol. By spanning a larger range, especially at lower baud rates, Sub 1Ghz is becoming a preferred alternative vs 2.4Ghz

❖ *Wireless RF IOT applications for Sub 1Ghz*

smartmetering-sub1ghz: Ideal RF IOT wireless applications for Sub 1Ghz include smart metering and industrial lighting. These IOT applications typically cover harsh environments. The long range and the lower interference features of the Sub 1Ghz spectrum band make it a perfect fit for these IOT applications.

❖ *Sub 1Ghz Wi-Fi:*

Often, Wi-Fi over Sub 1Ghz is becoming a popular option for several wireless applications. Industrial lighting is of course a popular application for Sub 1Ghz Wi-Fi, but recently, a lot of hardware RF designers are also using Sub 1Ghz Wi-Fi for home automation. One reason for using WiFi over Sub 1Ghz is the effective handling of interference by Sub 1Ghz.

❖ *Wireless Standards & protocols that fit into Sub 1Ghz RF- LoraWan & Sigfox*

Sigfox is a technology that has a higher range than Wi-Fi Networks and a lower range than cellular networks. It typically uses the 900 Mhz (sub 1Ghz) spectrum band to provide a range upto a few miles in the urban environment. LoraWan is a technology that provides Wide Area Networks (WAN) at a lower cost. Lorawan works in various Sub 1Ghz frequency bands. These include the 902-928 Mhz band in the US and the 779-787 MHz & 470-510 MHz bands in China.

❖ *Wireless Sub 1Ghz Modules to prototype your IOT application*

The Sub 1Ghz ISM frequency band is becoming very popular for IOT applications needing long range and low power. As a result, companies like Texas Instruments, Atmel,Semtech and Silicon Labs have developed interesting modules to kick-start your RF Sub 1Ghz IOT application.These sub 1Ghz modules typically have low power transceivers and receivers.

The makers of the CC1310 [36] Launchpad Evaluation Module have a special claim to fame. CC1310 has been proved to support a range of over 100 km using Sub 1Ghz RF and LoraWan technology.

In summary, Sub 1Ghz RF is getting very popular for interesting low power IOT applications.

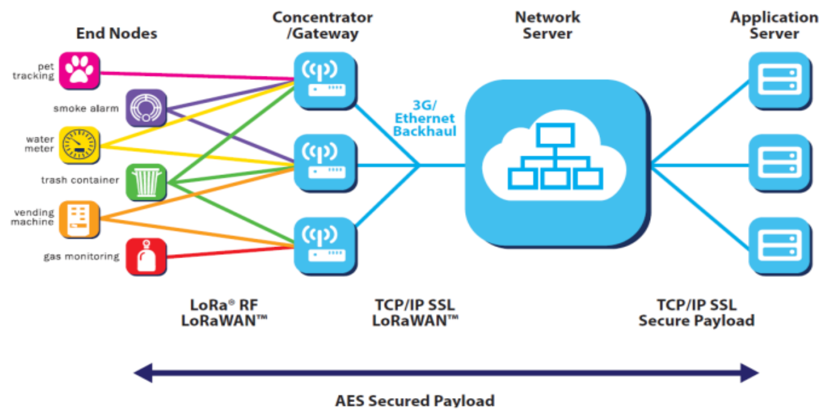
2.2.2 LoRaWAN

LoRaWAN is a communications standard designed for wide area networks of elements with low power consumption (LPWAN) developed by the LoRaWAN Alliance, an alliance that has outstanding members in the field of communications such as Cisco, Orange, IBM and ZTE. LoRaWAN defines the communications protocol and the system architecture for the network, while the LoRa physical layer is the one that enables the long-range communications link.

This specification is clearly intended for use in the field of the Internet of Things, since it is optimized for battery-operated low-cost sensors and provides interoperability between different intelligent devices without the need to design and implement complex networks, if not It gives the user freedom to carry out his own network. Other important features offered by LoRaWAN are a secure two-way communication, in addition to being oriented to the mobility of the devices and the location services.

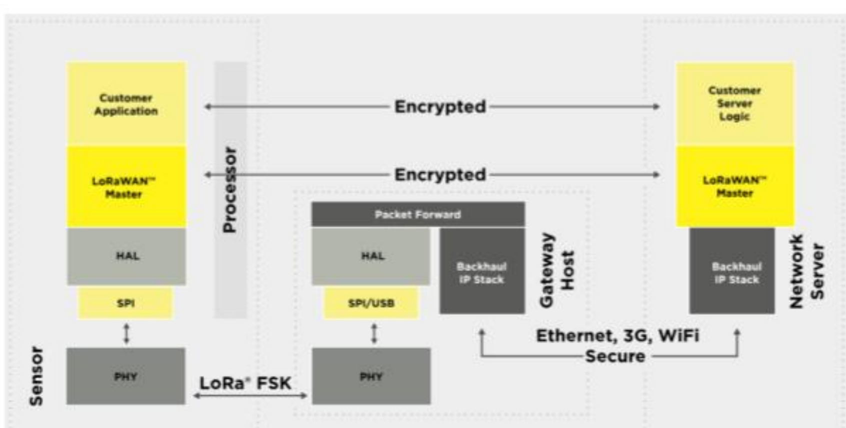
❖ *Architecture*

LoRaWAN uses a star-shaped network architecture in which the gateways act as transparent gateways between the end devices of the network (data capturers of the sensors, ...) and the backend network servers. The final devices communicate directly with the gateways by means of wireless communications in a single jump, using the technology of the widened spectrum to avoid the interferences between the communications of different devices, besides using an adaptive bandwidth (ABR) to increase the capacity of the gateways. The intelligence and complexity of the system is found in the network server, responsible for filtering redundant packets, performing security checks and managing the data rate and RF output for each final device individually by means of an adaptive data rate scheme (ADR). The gateways communicate with the network servers using standard IP communications.



Network Architecture of LoRaWan

Each gateway will deliver the received packet from a node to the network server, transforming it into Backhaul network protocols (cellular, Ethernet, satellite, Wifi, ...). In addition, the server is responsible for the upper layer protocols exchanging MAC / L2 / L3 control signaling messages with the nodes.
Note: An indispensable condition for a long-range star network to be viable is that gateways must have a large capacity to receive messages from a large number of nodes. In LoRaWAN this is achieved through the use of an adaptive data rate and through the use of a multichannel transceiver in the gateway, resulting in the correct reception of simultaneous messages on multiple channels.

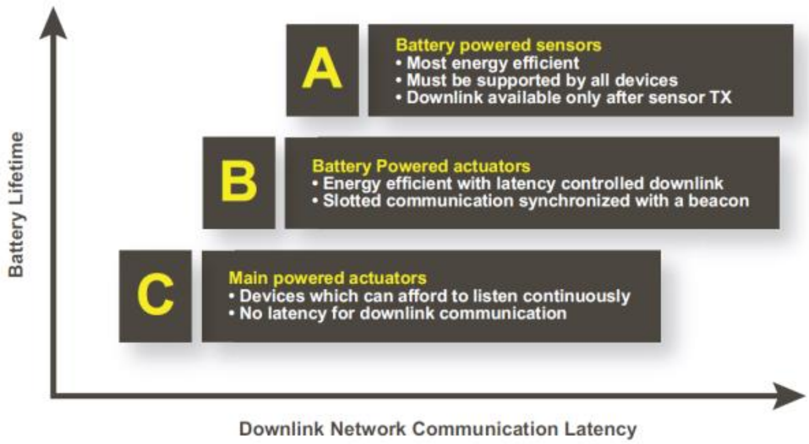


Stacks of communications protocols in LoRaWAN [9]

The network servers in the cloud will be in charge of communicating with the user's IoT platform through secure TCP / IP communications using SSL [10 11].

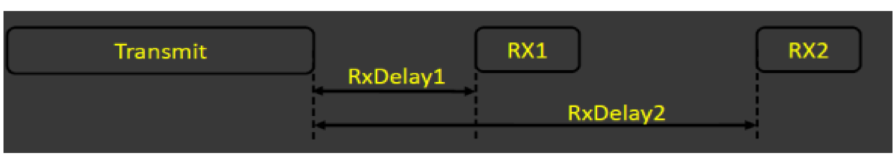
❖ *Classes of LoRaWan devices*

The final devices of the network do not have the same characteristics and will not be used for the same purpose or for the same applications, so LoRaWAN has designed three different kinds of devices that will be described in this section.



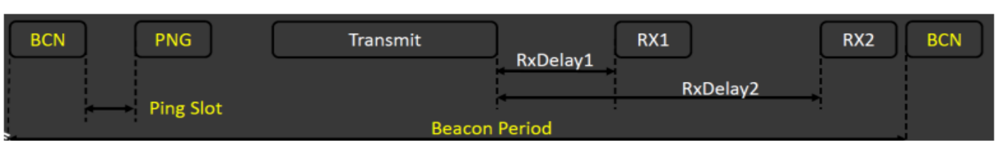
Classes of LoRaWAN devices [10]

Class A devices: the class A end devices allow bidirectional communications, where a transmission is followed by two small reception windows, the first one second after the transmission and the second one after the first window. The transmission slot programmed by the final device is based on its own communication needs, with a small variation based on a random time base (Aloha protocol). This type of operation of class A is the one with the lowest power consumption, because it only expects a communication from the server if the device has transmitted before.



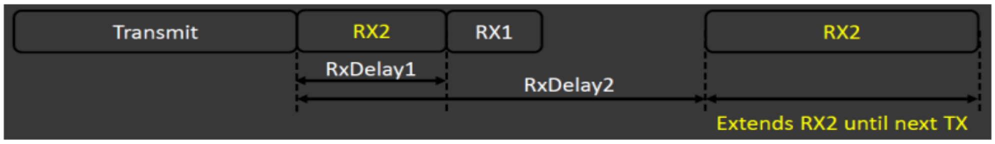
Transmission and reception windows of a class A device [11]

Class B devices: they allow the reception of data without the need to transmit in advance, functionality that is achieved by opening extra reception windows at scheduled times. In order for the final device to open its reception window at the scheduled time, the device receives a beacon synchronized at the time of the Gateway, telling it to open its reception window. Finally, this allows the server to know when the final device is listening. These devices have a higher power consumption than class A devices.



Transmission and reception windows of a class B device [11]

Class C devices: Class C devices are permanently listening, that is, their reception windows are always open, unless the device is transmitting. These devices provide the best response times and ability to send from the server to the nodes, but consume much more power compared to devices of class A and B [10 11].



Transmission and reception windows of a class C device [11]

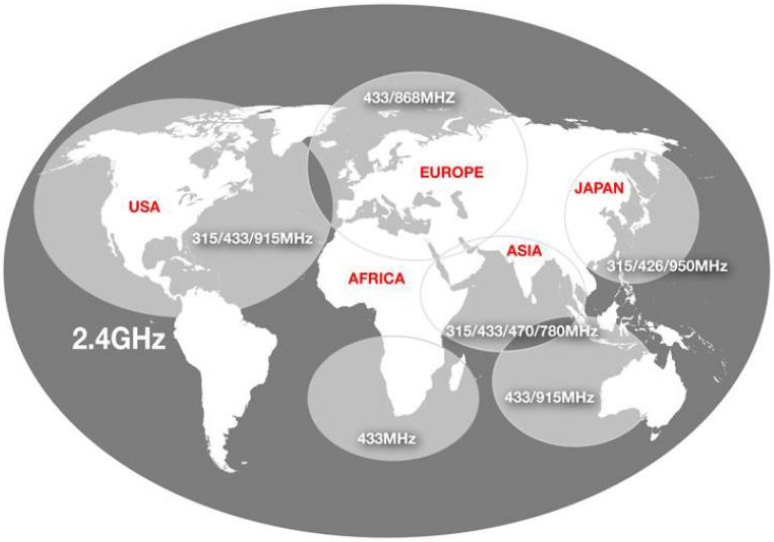
❖ *Transmission Channels*

In LoRaWAN the nodes can transmit by any available channel and using any binary rate, as long as the node complies with the following conditions: it must change the channel in a pseudo-random manner in each transmission and must exceed the maximum work cycle or the maximum transmission time, all this in relation to the subband used and local regulations.

The LoRaWAN specification varies slightly between regions due to regional spectrum allocations and regulatory requirements. In Europe and North America, the LoRaWAN specification is defined, but in other areas such as Asia it is still being defined.

In Europe, LoRaWAN defines ten channels, eight of which are multi-data rate from 250 bps up to 5.5 kbps, a single channel with a high data rate of 11kbps and a single FSK channel at a speed of 50 kbps [10].

In North America, the ISM band goes from 902 to 928 MHz, where LoRaWAN defines 64 channels of 125 kHz transmission from 902.3 to 914.9 MHz, with 200 Hz increments. In addition, there are eight 500 kHz reception channels of width ranging from 923.3 MHz to 927.5 MHz.



Regulations: ISM/SRD frequency bands

❖ *Security*

In security protocols, security is very important, since data and information shared in communications can be sensitive. LoRaWAN uses two security layers: one for the network and one for the application. The security layer of the network ensures the authenticity of the node in the network while the security layer of the application ensures that the operator of the network does not have access to the user data of the application. To achieve these security levels, AES encryption is used with the exchange of keys using an IEEE EUI64 identifier. In order for a final device to communicate with the LoRaWAN network, it must be activated and the following information is needed [10] [11]:

Device address: 32-bit identifier, unique in the network and present in each data frame. It is shared between the final device, the network server and the application server.

Network Session Key: 128-bit key, unique for each final device and shared between the final device and the network server. It provides security in the communication between the final device and the network server.

Application Session Key: 128-bit key, unique for each final device and shared between the final device and the application server. It is used to encrypt and decrypt application data messages. Guarantees end-to-end security at the application level.

2.2.3 SigFox

The French company Sigfox has developed a technology for use in the field of Internet of Things, its main feature being communication between different devices without the need to establish and maintain network connections. This is achieved with a software-based solution, where the complexity of the network and computing is not managed in the devices, but it is managed from the cloud, which means a great saving of energy and computing in the devices. The use of Ultra Narrow Band coupled with the exchange of messages with short frames allows the devices to drastically reduce energy consumption and increase battery life, a very important aspect and to improve in IoT systems.

❖ *Network Architecture*

The network architecture of Sigfox is of the star type, where four elements are included:

- The different devices or objects that you want to connect. They need to include a module verified by Sigfox to be able to connect to the network.
- Sigfox stations.
- The Sigfox cloud: Sigfox CLOUD.
- The network servers or client platform.

Elements of Sigfox architecture [12]

The objects emit Sigfox radio messages that are received by the Sigfox stations that are within the range. All stations that receive these messages will transmit it to the Sigfox cloud through a point-to-point link between each station and the cloud. Finally, the cloud duplicates these messages and introduces a single message on the client's platform. These transmission or uplink messages can contain from 0 to 12 bytes of data, use D-BPSK modulation, their bitrate is 100 or 600 bps depending on the region of operation and a transmission power that can reach 22 dBm.
The reception of data by objects is optimized to minimize energy consumption, so that messages can not be transmitted to objects at any time. The object must ask the network to receive a message, transmitting a message containing an uplink request flag. This message is detected by all stations in the same area, demodulated and transmitted to the cloud. The cloud transmits the message to the client platform, which detects that not only is an uplink message being received, but the object is asking if it has to receive a downlink message. At this moment, the client's platform decides whether or not to transmit a downlink message, if it decides to transmit a message, it sends

it to the cloud and it selects a station to transmit it to the object in question. These messages of reception of objects or downlink can contain 8 bytes of data, use GFSK modulation, its bitrate is 600 bps and the minimum sensitivity that the receiver must have (in this case the object) is -132 dBm [12].

❖ *Radio Technology*

The devices communicate with Sigfox stations by radio communication, operating in the non-commercial frequency band (ISM, from 868 MHz to 869 MHz and from 902 to 928 MHz, depending on regions), using Ultra Narrow Band technology (UNB) combined with the D-BPSK and GFSK modulations, which makes possible the existence of more simultaneous signals in the operation band. The D-BPSK modulation allows to have a high spectral efficiency, since it makes it possible to take a Hertz of the operation band to transmit at one bit per second (1bps = 1Hz). The Sigfox protocol uses D-BPSK modulation at a rate of 100 or 600 bps (depending on the region of operation) for three main reasons: the D-BPSK modulation is easy to implement, a low bit rate allows the use of low cost components and , finally, the receiving base station has a high sensitivity, so it can demodulate signals very close to the Noise Floor (NF). The sensitivity of the receivers can reach -134 dBm for a transmission at 600 bps and -142 dBm for a transmission at 100 bps.

A device can transmit anywhere within the operation band, since there is no synchronization between the device and the base station, and, on the other hand, a base station is capable of demodulating any Sigfox signal within the band of operation. For these two reasons, there are no restrictions of low frequency accuracy in the devices.

The Sigfox protocol is a light protocol with the aim of delivering messages of reduced size. An uplink message contains a maximum of 12 bytes of data and a downlink message contains only 8 bytes of data. For a message with 12 bytes of data, a frame of Sigfox will have a size of 26 bytes in total, which makes it a protocol with frames of reduced size [12].

Size of a Sigfox frame [12]

❖ *Security*

Security is a very important aspect in Sigfox and is applied first to the devices. There is an end-to-end authentication method between the devices and the Sigfox cloud based on a secret key. This secret key is stored in a non-accessible memory associated with the visible and specific ID in the read-only memory. The secret key is used in messages sent by the device to create a unique signature for each message that will authenticate the sender. This signature incorporates a sequence number that is added to the radio frames to prevent re-transmission or reproduction.

The downlink process provides an additional security robustness, since the devices choose when they want to communicate and at what frequency, which does not allow them to receive malicious messages from an attacker.

The base stations are connected to the cloud through a point-to-point link, using a VPN encrypted via SSL, so these communications between these elements are secure, robust and reliable. On the other hand, the cloud of Sigfox is virtualized in private servers hosted in data centers distributed in multiple physical locations and, in addition, it is secure, robust and scalable. Finally, the users' platforms connect to the cloud through the use of HTTPS encrypted interfaces for web pages, APIs and callbacks. Finally, it should be noted that Sigfox has a high resistance to interference, since it implements three diversity schemes to make robust transmissions:

- **Temporal diversity**: the data is transmitted in three consecutive radio frames.
- **Frequency diversity**: each of these radio frames are transmitted on a different set of random subcarriers.
- **Spatial diversity**: when these radio frames are transmitted there is no link or path established between the device and the network, but there are several propagation paths between one device and multiple base stations in the range [12].

2.2.4 Machine-to-Machine communications provided by mobile operators

The concept of machine-to-machine (M2M) refers to the exchange of information between two remote machines in an autonomous, reliable and efficient way. With the popularization of the Internet of Things and the need to connect different devices remotely, various M2M solutions have been adapted to create IoT infrastructures. Currently, large mobile operators, such as Telefónica or Vodafone, offer their customers various M2M solutions for their customers.

The main idea is the interconnection of the different devices with the client's platform using the network infrastructures (mostly the LTE network) of the mobile operators, with direct connections from machine to machine, and without the need for the clients to deploy and manage your own network Each device needs an M2M SIM card from the operator to be able to communicate in the LTE network [13].

2.2.5 NarrowBand IoT

NarrowBand IoT (NB-IoT) [14] is a radio technology based on cellular communications (LTE) for LPWAN networks standardized in 2016 by the 3rd Generation Partnership Project (3GPP) -participating in the development of GSM, UMTS and LTE standards- within the framework of its projects in the field of the Internet of Things. NB-IoT can operate in free frequency bands, in the currently unused 200kHz band (previously occupied by GSM) and in LTE base stations, being necessary to assign them a resource block (RB) for the NB-IoT operations in his guard band. In addition, it allows to connect sensors directly to the base stations.

NB-IoT stands out because it allows connecting a large number of devices, specifically more than 50000 for each NB-IoT cell, while minimizing energy consumption and increasing the range of coverage in locations that are not covered by conventional cellular communications technologies. On the other hand, most LTE-Advanced features such as carrier aggregation, dual connectivity or device-to-device services are not supported by NB-IoT. Also, in this technology is not present the concept of quality of service (QoS), since this technology is not designed to deliver data packets sensitive to delay.

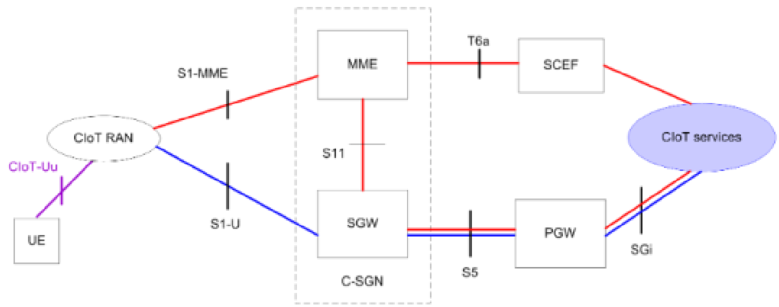
❖ *Kernel network architecture*

To deliver data to an application, two optimizations have been designed for the cellular Internet of Things (CIoT) in the Evolved Packet System (EPS): control plane optimization (CP) and user plane optimization (UP).

In optimizing the CP, the uplink data is transferred from the eNB (CIoT RAN) to the MME (Mobility Management Entity). At this time, the data can be transferred from the Serving Gateway (SGW) to the Packet Data Network Gateway (PGW), or to the Service Capability Exposure Function (SCEF), which is the only possible destination for non-IP packets. From these nodes they are finally delivered to the application server (CIoT Services). The downlink data is transmitted in the same way, but in the opposite direction.

The SCEF is a new node designed especially for machine-type data. It is used to deliver non-IP packets across the control plane, and provides an abstract interface for network services, such as authentication and authorization.

Regarding UP optimization, the data is transferred over radio carriers through the SGW and the PGW to the application server. This path supports IP packets and not IP.



Network architecture of the NB-IoT core. The control plane is shown in red and the user's plane is displayed in blue [15]

❖ *Access network architecture*

The architecture of the access network does not differ from the LTE architecture. The eNBs (evolved Node B) are connected to the MME and the S-GW using the S1 interface. Although no handover has been defined, there is still an X2 interface between two eNBs [15].

Architecture of the access network of NB-IoT [15]

❖ *Radio Design*

The NarrowBand IoT radio interface design is derived from LTE. The carrier of NB-IoT has a bandwidth of 180 kHz with support to operate with multi-carrier. In the downlink, orthogonal frequency division multiple access (OFDMA) is used with a 15 kHz subcarrier spacing on 12 subcarriers with 14 symbols used to span a 1ms subframe. In the uplink, the single-carrier frequency division multiple access (SC-FDMA) is applied, with a subcarrier spacing of 3.75 or 15 kHz.

NB-IoT defines three modes of operation to provide flexibility:

- **Autonomous**: using, for example, one or more GSM carriers.
- **Guard band**: using unused resource blocks (RB) with an LTE carrier guard band.
- **In band**: using resource blocks with an LTE carrier.

Additionally, NB-IoT uses the concept of repetitions and signal combination techniques to improve the coverage area. To serve the user equipment (UE) in different coverage conditions that have different ranges of path losses, there can be up to three coverage improvement (CE) configurations in the random access with their specific configurations. After this, the eNodeB selects the configuration of the radio resources, the modulation and coding scheme (MCS) and the repetitions depending on the coverage of the UEs [14].

❖ *Energy Efficiency*

NB-IoT is designed so that the devices have a long useful life, whose battery can go beyond 10 years of life. To achieve this end, NB-IoT reuses LTE's energy saving mechanisms, but understands the timers involved to achieve a longer battery life. In LTE there are two key mechanisms for energy saving: Discontinuous Reception (DRX) and Energy Saving Mode (PSM). Both mechanisms vary the way in which the UE communicates with the network. This communication requires a Radio Resource Connection (RRC) established between the UE and the eNB. There are two possible states: connected and at rest, likewise, a UE in a connected RRC state has an active RRC connection. The UE can go from being in RRC mode connected to being in RRC mode at rest for different reasons, such as the inactivity of the UE or that this UE is disconnected. To detect inactivity in the UEs, the eNB uses inactivity timers that are reset after sending a data packet [14].

- *LPWA with License*

The mobile network operators (MNOs) are implementing licensed cellular LPWA networks, such as NB-IoT (LTE Cat NB1) and LTE Cat M1, and the required hardware is following their example. By the end of 2018, licensed LPWA networks are likely to serve most markets in the United States, Europe and Asia Pacific.

Because they are standardized, licensed LPWA technologies offer the most reliable communication solution for global IoT applications. The widespread availability of NB-IoT, for example, means that you can use the same underlying technology for applications targeting markets in the four corners of the world. The introduction of roaming agreements between operators and regions will simplify implementations for applications that cross national or even continental borders. And because operators must comply with service level agreements, they have a strong incentive to ensure that networks are reliable.

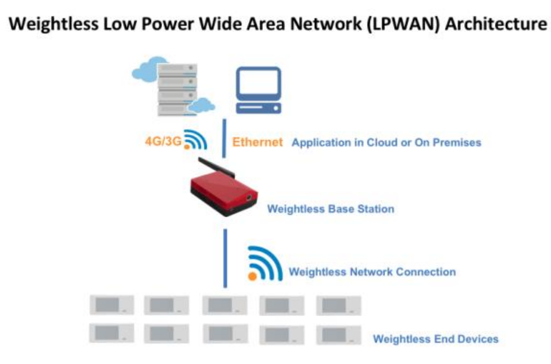
2.2.6 WeightLess

Weightless is a set of wireless technology standards for the exchange of data between a base station and many machines around it. These technologies allow developers to build low-power wide area networks (LPWAN).

Originally, there were three published Weightless connectivity standards: Weightless-P, Weightless-N and Weightless-W. Weightless-N was an uplink LPWAN technology. Weightless-W was designed to operate in the blank space of the TV. Weightless (Weightless-P) was the true winner with its true bi-directional narrow band technology designed to be operated on ISM frequencies with a global license and without a license.

Weightless is managed by Weightless SIG, or Special Interest Group, which began operations on December 7, 2012.

The intention is that the devices must be qualified by the Weightless special interest group according to the standards defined by the SIG. Patents would only be licensed for those qualified devices; therefore, the protocol, although open, can be considered as proprietary.



Weightless architecture [31]

Weightless technology provides wireless connectivity for low-power wide area networks (LPWAN) designed specifically for the Internet of Things and can operate in both the sub-1GHz and license-exempt spectrum.

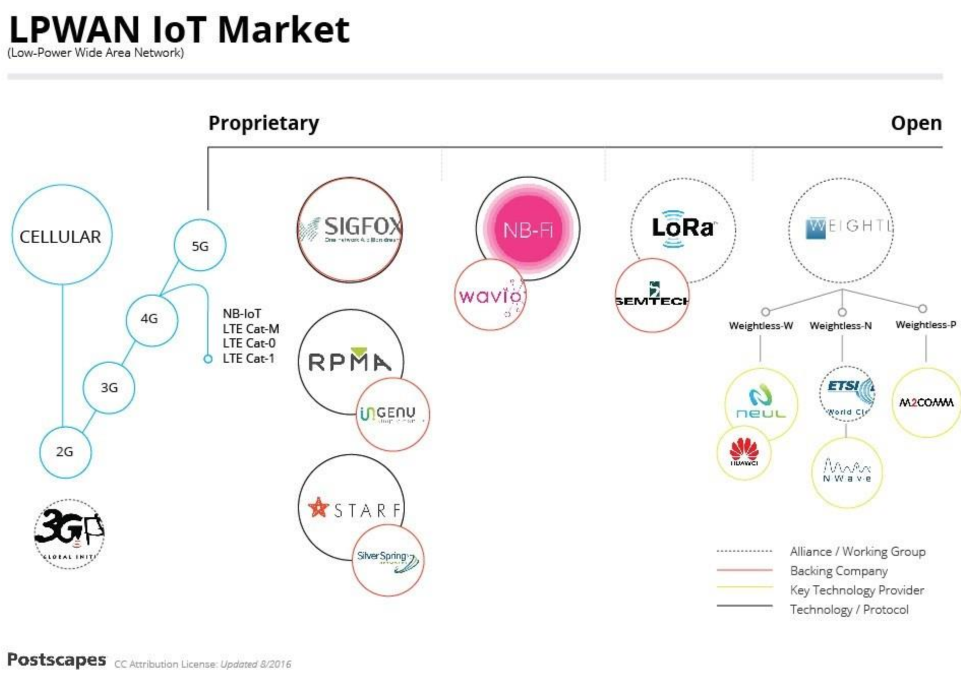
Conventionally, IoT / M2M products have been developed around GSM-based connectivity technologies, but for most applications they are not optimal. GPRS, 3G and LTE technologies provide sufficient coverage for most applications, but the hardware costs of the terminal are high. The 3GPP technologies and the variants based on LTE, NB-IoT / CIoT, which will emerge over the next few years, are legacy standards that admit completely different use cases and technical requirements. They are exceptionally sophisticated and powerful, but that sophistication ultimately makes NB-IoT / CIoT not optimal in terms of cost and power consumption for most IoT applications. GSM-based networks are often not suitable for the short message sizes typical of IoT use cases. Significant overhead associated with signage to move terminals from passive states to assets, report on status and more lead to highly inefficient networks. Legacy telephony technologies will satisfy the requirements of the user who can tolerate higher costs and where the devices have external power; they can never meet the low cost and energy consumption characteristics of LPWAN alternatives of efficient 'clean slate' design.

Several short-range technologies, especially Wi-Fi, Bluetooth and Zigbee, offer endpoints at low prices, around US $ 1-2. However, being short-range, they can not provide the necessary coverage for applications such as automobiles, sensors, asset tracking, medical care and many more. Instead, they are restricted to connected machines within home or office environments. Nor do they allow economies provided by much larger cells, with few base stations that cover large areas, such as entire cities.

Low-power wide area network (LPWAN) connectivity offers the long-range capability of GSM-based technologies with the low power and power consumption characteristics of LAN / PAN.

It is essential that technology is an open global standard rather than a patented technology to ensure low cost and low risk, and maximize user choice and continuous innovation. A vibrant ecosystem that delivers terminals, base stations, networks and applications supports a wide range of IoT use cases. Weightless is an open standard of LPWAN technology that enables rapid and cost-effective deployment of long-range IoT products, long battery life and ultra low cost worldwide.

2.2.7 Comparisons of LPWAN technologies



The market for LPWAN technologies

	NB-IoT	EC-GSM-IoT	LTE Cat M1	LoRa	SigFox	IQRF	RPMA	Telensa	DASH7	Weightless-N	Weightless-P	SNOW
Modulation	QPSK, OFDMA (UL), SC-FDMA (DL)	GMSK, 8PSK	QPSK	CSS	DBPSK, GFSK	GFSK	DSSS, CDMA	FSK	GFSK	DBPSK	GMSK, OQPSK	BPSK
Band	Licensed, Sub-GHz	Licensed, Sub-GHz	Licensed, Sub-GHz	Unlicensed, Sub-GHz	Unlicensed, Sub-GHz	Unlicensed, Sub-GHz	Unlicensed, 2.4 GHz	Unlicensed, Sub-GHz	Unlicensed, Sub-GHz	Unlicensed, Sub-GHz	Unlicensed, Licensed, Sub-GHz	Unlicensed, TV white spaces
Max Range (Km)	15	15	15	15	10	0 - 5	15	1 - 10	0 - 5	0 - 3	0 - 2	5
Peak data rate (kbps)	250 kbps (UL), 170 kbps (DL)	10	375	27	1	20	80	65	9.6, 55.666, 166.766	100	100	50kbps per node
Security	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	N/A
Indoor	Yes	Yes	Yes	Yes	No	Yes	Yes	No	No	No	Yes	Yes
Link budget (dB)	164	164	164	164	N/A	N/A	177	N/A	N/A	N/A	N/A	N/A
Mobility	No	Yes	Yes	Yes	No	Yes	Limited	No	N/A	No	No	N/A
Battery lifetime (Years)	10	10	10	10	5	N/A	15	10	N/A	N/A	N/A	N/A

Summary of LPWAN technologies

2.3 6LoWPAN

6loWPAN is a standard that allows the use of IPv6 over networks based on the IEEE 802.15.4 standard. Enables devices such as nodes in a wireless network to communicate directly with other IP devices.

Thread and Wi-SUN IPv6 stack implement IPv6 on top of IEEE 802.15.4 radios. There are, however, attractive alternatives depending on the use case. Thread is designed for home environments with relatively confined spatial coverage. For a lot of such smart devices Bluetooth Low Energy, a widely supported wireless communication protocol, is a possible alternative.

The successor, Bluetooth 5, adds new features that are tailored for the IoT.

Powered devices in homes can also connect to existing but more power hungry IEEE 802.11 WiFi networks. The recently published IEEE 802.11ah protocol focuses, like Bluetooth 5, on the IoT and provides low-power communication for distances of up to one kilometer.

The LoRa Alliance specifies another long range communication network, which is feasible for distances up to some kilometers, depending on environmental conditions like line-of-sight between the nodes. This high transmission range requires radios with a higher power consumption compared to IEEE 802.15.4 radios. The nodes and the gateway, however, are only one hop apart and there is no need to maintain a network topology.

Another option are backscatters; i.e., radios that do not actively transmit but modulate ambient electromagnetic waves. Recent research shows that such devices can communicate over distances of up to one kilometer while consuming only microwatts instead of milliwatts like traditional radios in the field.

2.3.1 OpenThread (IPV6)

The initial research of wireless sensor networks was motivated by military considerations; most notably the development of surveillance systems was of interest. Nowadays wireless sensor networks find use in a wide variety of applications. The number of devices, also called nodes, ranges from a few to several thousands. The nodes typically perform only a limited set of tasks like sensing and controlling the surrounding environment. Therefore, the nodes are often low-cost embedded devices with constrained computational resources, equipped with a radio transceiver. They are optimized for low energy consumption and can run on batteries for years, which makes them suitable to be placed at remote or non-fixed positions.

Due to the nodes' low-price tag and low energy consumption, low-power wireless networks play an important role in the emerging Internet of Things (IoT). It is expected that the IoT grows from approximately 18 billion devices in 2022. In this huge market the development of industry standards for communication in low power wireless networks is essential; a common, well-tested, and established standard speeds up application development and enables interoperability between nodes running different operating systems. Different standardization bodies and industry alliances, however, try to establish their own standards.

With the advent and spread of IPv6 in recent years there is a trend to directly implement this protocol for low-power wireless networks. Contiki is a well-known operating system tailored for such networks, which runs on a variety of platforms. Following this trend, its current communication stack implements a variety of open standards for low-power wireless IPv6 networking. The performance of Contiki's low-power IPv6 stack has been thoroughly tested and it has been awarded the IPv6 Ready Silver logo.

During the last years, an industry alliance, the Thread group, developed a new IPv6-based network stack called Thread. It is specifically designed for smart IoT applications in home environments, and tries to overcome the segregation in this market. It is partially based on open standards, but until recently only members of the Thread group had access to the complete Thread specification. An open source implementation, however, is available: OpenThread.

2.3.2 Contiki

Contiki is a portable operating system designed for resource-constrained IoT devices. It includes a small kernel, reusable application modules and hardware drivers for a variety of supported platforms.

Applications running on top of Contiki are implemented using processes which are scheduled and executed by Contiki's kernel. The kernel uses a cooperative scheduling strategy, i.e., the processes either run to completion or return control to the scheduler on a voluntary basis. A process can be preempted by interrupts, the preempted process however resumes directly when the interrupt handler returns. In order to obtain a responsive system, the processes need to return control frequently.

The processes are implemented as Protothreads. They share the stack with Contiki's kernel, which leads to a very small memory overhead and fast context switches. They were specifically designed to facilitate an event-driven execution model and provide a variety of tools to communicate with other processes via events. Synchronous events are immediately delivered to the target process, and the sending process continues when the receiving process returns control. Asynchronous events are inserted in an event queue and dispatched by the kernel.

Processes often return control to the scheduler by blocking waiting for an event.

There are three wireless network stacks included in Contiki: Rime, an IPv4 stack, and an IPv6 stack. The IPv4 stack, uIP, was the first IP-based network stack for low-power wireless networks. It uses on-demand routing similar to the Ad-hoc On-demand Distance Vector protocol: a node broadcasts the desired destination address and neighboring nodes then advertise if they have a route to the destination available. The Rime network stack uses a similar routing strategy and is more lightweight compared to the IPv4 stack. Another ondemand routing protocol is LOADng, which is still being actively developed.

For many platforms, however, the default network stack is the IPv6 stack, which is used for the comparison with OpenThread. It uses a different routing strategy than the other two stacks and is described in more detail in the next sections.

Before the advent of the IoT, most applications based on low-power wireless networks were deployed at remote locations, which might be difficult to reach. In order to minimize the maintenance cost and to maximize the expected lifetime of the network, energy efficiency has been a main objective of research. The energy consumption of the nodes is dominated by the radio idle-listening for incoming packets. This led to the development of duty-cycled link layer protocols which strive to minimize idle listening. Examples of such protocols are B-MAC, X-MAC and ContikiMAC.

Even though Contiki supported IPv4 from the beginning, the Internet Protocol was often considered too complex and not appropriate for the characteristics of low-power wireless networks, which are primarily given by constrained access to energy and constrained computational resources.

The active research in the field produced many lightweight ad-hoc network protocols like Contiki's Rime stack. Even industry-driven standardization attempts like ZigBee and Z-Wave were not based on the Internet Protocol. The lack of a common network layer with the Internet, however, introduces several issues like the need for an application-layer gateway that translates between the two protocols. This is often difficult and packets might be dropped if no mapping exists. Furthermore, such mappings are often stateful, which implies that only one gateway can be used to route the traffic between the sensor network and adjacent networks.

This leads to a single point of failure in the application. In 2008, Hui and Culler demonstrated the feasibility of the Internet Protocol version 6 for low power wireless networks. IPv6 has several advantages over IPv4, like the Stateless Address Autoconfiguration (SLAAC) and a huge address space. This way every node in a low-power network is directly addressable using a widespread technology, which simplifies development and deployment of IoT applications. Since then several working groups within the Internet Engineering Task Force (IETF) worked on IPv6 related standards for low-power wireless networks, amongst others the IPv6 over Low power Wireless Personal Area Networks (6LoWPAN) group, the Routing Over Low power and Lossy networks (ROLL) group, and the Constrained RESTful Environments (CoRE) group.

Low-power wireless networks often use the widespread IEEE 802.15.4 wireless communication technology. The 6LoWPAN group standardizes the transmission of IPv6 packets using IEEE 802.15.4 radios by means of the 6LoWPAN adaption layer. This layer compresses the comparatively large IPv6 and UDP headers and handles packet fragmentation and reassembly.

The ROLL group focuses on routing solutions for lossy, low-power networks, and released the RPL routing protocol in RFC6550. RPL is a major component in Contiki's IPv6 stack. The group also standardized the Trickle algorithm, a well-known mechanism to disseminate information in a network in an energy efficient manner.

Changes or inconsistencies in the network topology lead to a fast communication rate, which then decreases exponentially such that only a few routing protocol packets per hour are sent.

The Trickle algorithm is used in Thread and Contiki's IPv6 stack to spread link quality and routing advertisements throughout the network.

The CoRE group focuses on the application layer of the network stack and specified the Constrained Application Protocol (CoAP) in RFC7252. CoAP provides functionality similar to the HTTP application layer. It is based on UDP instead of TCP to account for the characteristics of lossy low-power networks. Both stacks, Contiki (see the er-coap application) and OpenThread, provide a CoAP implementation.

2.3.3 WI-SUN

Wi-SUN [37] (Wireless Smart Ubiquitous Network) is a wireless communication technology designed for Utilities, Smart Cities and IoT. Wi-SUN is based on various IEEE, IETF, and ANSI/TIA standards supporting low power and lossy networks.

WiSUN is an established suite of IoT technologies that is based on IEEE 802.15.4, TCP/IP, and related standard protocols as detailed in the following sections. Important characteristics of WiSUN include the following:

- Coverage
- Range measured in kilometers
- Development Ecosystem
- WiSUN Alliance with task groups for targeted use cases and assured interoperability
- High Bandwidth
- Up to 300 kbps
- Low Latency
- 0.02 seconds
- Mesh Routing
- Resilient and scalable
- Power Efficiency
- less than 2 uA when resting; 8 mA when listening

- Scalability
- Networks to 5,000 devices; 10 million endpoints worldwide
- Security
- Public key certificates, AES, HMAC, dynamic key refresh, hardened crypto
- Taken as a whole, these characteristics support consideration of WiSUN protocol solution as an implementation choice for LPWAN.

The Wi-SUN Alliance, a global ecosystem of member companies that seeks to accelerate the implementation of field area networks (FAN) based on open standards and the Internet of Things (IoT). The FAN is a communications infrastructure for large-scale networks. In the same way that the Internet is used on a smartphone or a computer, a field-area network allows the interconnection of devices in a common network.

Exclusive systems are no longer flexible enough, or as profitable as solutions based on open standards, and will lose relevance in a few years. Adhering to the Wi-SUN FAN profile, based on IEEE and IETF open standards, enables service providers, smart cities and utilities to deploy customizable multiservice networks and help ensure interoperability, not only currently, but also also in the years to come.

Wi-SUN continues to be at the forefront of IoT innovation, and as cities, utilities and industry expand their IoT networks, they increasingly recognize that open standards for the entire industry they are essential for interoperability and scalability.

- It offers public service companies, cities and service providers adaptable multiservice networks that will help ensure current interoperability and for future generations.
- It reduces the time needed to evaluate new products, since the behavior, performance and interoperability are well defined.
- Remove the link with a single provider.
- Encourages the development of a global ecosystem of standards-based products, thus reducing the risk and costly impact of stranded assets.

The stack overview of Wi-SUN FAN. Based on IEEE 802.15.4g, which provides bi-directional communication with high data rate (up to 300kbps) and low latency. The low power consumption permits a battery-powered FAN device to listen frequently while maintaining a lifetime measured in years. The MAC sub-layer is based on IEEE 802.15.4e along with Wi-SUN defined Information Extensions (IEs). The network layer is IPv6 with 6LoWPAN adaptation. The Wi-SUN FAN supports star and mesh topologies, as well as hybrid star/mesh deployments. Two methods are available for packet routing: RPL non-storing mode is mandatory at the network layer, and MHDS is optional at the Logical Link Control (LLC) sub-layer. The transport layer provides both UDP and TCP services.

A Wi-SUN FAN node can operate in any of the regional frequency bands, i.e. 470-510MHz, 779-787MHz and 920.5-924.5MHz in China, 863-870MHz and 870-876MHz in Europe, or 920-928MHz in USA, Canada and Japan. The radio interface is also compliant with local regulations of India, Mexico, Brazil, Australia, New Zealand, Korea, Philippines, Malaysia, Hong Kong, Singapore, Thailand, and Vietnam.

The MAC layer supports channel hopping for both unicast and broadcast frame transmission. The total number of channels available is determined by the regional band and the channel spacing. A node can also choose to exclude a set of channels from its hopping sequence. A channel function defines a method used to determine, from the list of available PHY channels, the specific channel upon which a node is operating at a given time. The resulting hopping schedule is advertised to the neighbors. A variety of channel functions can be implemented, including TR51, direct hash, fixed channel and vendor defined channel functions. Related information is encapsulated in the unicast/broadcast schedule IE.

Wi-SUN FAN's PHY layer supports data rates ranging from 50-300kbps. Wi-SUN devices support low latency (0.02s) and frequent (as often as every 10 seconds) communications.

2.3.3 Comparing IoT technologies: WI-SUN, LoRaWan and NB-IoT

The network architecture define our IoT deployment's:

- Performance
- Longevity
- Reliability
- Security
- Interoperability
- Scalability

Wireless Smart Ubiquitous Network (Wi-SUN) is a technology based on the IEEE 802.15.4g standard. It is backed by the Wi-SUN Alliance™, a global industry alliance that promotes the Wi-SUN specification. Wi-SUN has a 3rd party organization that develops tests to certify that IEEE 802.15.4g based IoT equipment is both conformant to the standard and interoperable with other certified equipment. Wi-SUN networks support star and mesh topologies, as well as hybrid star/mesh deployments, but are typically laid out in a mesh topology where each node relays data for the network to provide network connectivity. Wi-SUN networks are deployed on both powered and battery-operated devices.

LoRaWAN is a Low Power Wide Area Network (LPWAN) specification based on chirp spread spectrum radios from Semtech, a supplier of proprietary semiconductor solutions. LoRaWAN is supported by the LoRa Alliance™, which provides certification for vendor interoperability. LoRaWAN radios are commonly used in low power devices with infrequent data transmissions. LoRaWAN is typically laid out in star topology with gateways relaying messages between end-devices and a central network server.

NB-IoT is a new mobile technology specification by the 3GPP standards body, and is expected to be used for low power and infrequent data transmission devices. NB-IoT can operate in the GSM spectrum or utilize an un-used resource block within a LTE's carrier's guard-band. NB-IoT compliant chipsets are now becoming available in the prototype stage. Most network topologies are star-based. 3GPP has not announced plans for a certification program yet. Certification programs with cellular companies have included high fees in the past, but no details have been announced as of the writing of this paper.

The three wireless networking technologies differ in several key areas. Following is a quick overview of those differences.

❖ *Network Topology and Coverage*

IoT networks typically are deployed in one of two topologies: star or mesh.

Star: LoRaWAN and NB-IoT networks are usually connected via a tower- or star-based topology. Commonly used in cellular networks, a star topology uses gateways (cellular towers, for example) that must have a direct connection to every device for that device to communicate. However, anything that obstructs the path from device to gateway (weather, construction or temporary obstacles) can impact a device's ability to remain connected to the network, which in turn compromises the reliability of the network and the services it carries. These obstructions are known as "black spots" or "shadows". In urban areas, the emergence of black spots could mean having to install more gateways to ensure a reliable connection for all devices, as many IoT devices are fixed in place and can't be moved to a better coverage location.

Mesh: Wi-SUN networks are usually mesh networks. In a mesh network, IoT devices communicate with neighboring devices on their network, all of which can serve as conduits to the network base station. This allows for multiple, redundant connection paths, so unlike star-based dramatically reduced, or even eliminated entirely. Infact, as mesh networks scale, their reliability and performance improves because the possible communication paths multiply. Mesh networks devices tend transmit short distances which enable them to be power efficient, deliver long battery life and have more consistent data rates.

A mesh architecture is essential for applications which require distributed computing or some local intelligence. These applications go beyond simple data acquisition and process locally available information from in field devices to make quick, intelligent decisions. Wi-SUN mesh networks are self-forming (when a new device is added, It automatically finds peers to communicate with) and self-healing (if obstacles emerge, devices will automatically reroute to the nearest available peers). Wi-SUN-based mesh networks have proven themselves for years in an array of harsh and remote environments across the globe. Tens of millions of reliably connected endpoints demonstrate that a Wi-SUN-based IoT mesh network can achieve the ubiquity and scalability many IoT customers demand.

❖ *Communication Performance: Bandwidth, Latency and Bi-Directional Communication*

The bandwidth, latency and bi-directional communication performance of a technology can have significant impact on the usable lifetime and functionality of the network.

IoT applications are rapidly developing with new uses and benefits being discovered and created. Having a network with the capacity to handle new applications and data streams will avoid early obsolescence. Additionally, the ability to update and upgrade deployed devices remotely can significantly extend the working life of those devices, while minimizing the need to conduct expensive field visits. Most of the current IoT applications today require on-demand control. On-demand commands require a network that can listen frequently and respond in a timely manner to on-demand commands. Not all IoT technologies offer the necessary bandwidth, latency and bi-directional communication to ensure long usable life and full-featured functionality.

IoT NETWORK	FEATURE	
	Bandwidth	Latency
Wi SUN Alliance	Up to 300 Kbps	0.02 sec
LoRa	Up to 50 Kbps[2]	1-2 sec[4]
Lte Advanced Pro	Up to 60 Kbps[3]	2 – 8 sec[5]

Wi-SUN provides high data rates that are consistent throughout the network and low latency. Additionally, Wi-SUN modules use less power for listening which enable customers to configure devices to listen frequently and still maintain a long-life.

LoRaWAN & NB-IoT In an attempt to drive down device costs and extend battery life, these devices are designed for low bandwidth (bandwidth also drops the further a device is from the base station) and infrequent communication. This may compromise the ability for the device to do future applications, firmware upgrades, security updates, and receive on-demand commands. Transmissions can be longer, too, because communication speeds can at maximum be only 50-60 Kbps—up to ~5-6 times slower than Wi-SUN—which results in more power consumption.

❖ *Security*

Security is a core concern for any network as compromised devices can be used to mount attacks on other networks, result in costly technology replacements or, even worse, disrupt essential services or public safety as would be the case for critical IoT networks. Military-grade security is thenetworks have achieved military-grade security by offering the security capabilities described below.

ELEMENT OF SECURITY	IoT NETWORK		
	Wi-SUN	LoRaWAN	NB-IoT
Authentication Chain of Trust (for mutual authentication of all devices and applications across the platform)	Public key infrastructure with x.509 certificates	Not Specified	Challenge/response authentication using pre-shared secret
Key Exchange and Distribution	Automated, symmetric key exchange and rotation using 802.11i	Static key exchange; no rotation	Static key exchange; no rotation
Key Generation and Storage	Hardware security modules and hardened crypto processors	Not specified	Not specified
Data Transmission	AES, HMAC and certificates	AES, CMAC and pre-shared secret	LTE data transmission encryption: AES
Network Access Control	Authentication via certificate prior to network enrollment	Authentication via pre-shared secret prior to network enrollment	Challenge/response authentication using pre-shared secret

❖ *Provider Ecosystem*

When an IoT network standard has a broad ecosystem of providers and developers that are interoperable, you have a broader set of product options and features supported by stronger pricing competition. Having access to a wide variety of options enables you to deploy a more cost-effective and optimized IoT network solutions.

Wi-SUN – The Wi-SUN Alliance consists of more than 130 member companies including product vendors, silicon vendors, software companies, utilities, government institutions and universities. Each member company contributes to the Wi-SUN ecosystem as the Wi-SUN Alliance has defined testing and certification programs for multi-vendor interoperability. Wi-SUN networks have been deployed for more than 10 years in mixed-vendor environments, demonstrating ongoing commitments from a wide variety of organizations.

LoRaWAN – The LoRa Alliance™ has more than 400 members, however it is unclear how interoperable each member's provider ecosystem is.

NB-IoT – Because NB-IoT is based on 3GPP, an alliance of multiple standards organizations, NB-IoT has the potential to have a healthy ecosystem of partners. However, due to the developmental nature of NB-IoT, there are no ecosystems built yet and it is unclear how the interoperability programs will work and at what cost. In fact, questions have been raised about the interoperability of two different implementations of NB-IoT.

❖ *Power Efficiency*

Many IoT devices do not have access to direct power and thus require battery power. As such, low-power devices are required to ensure long service life. But consuming less power can come with trading off functionality. To preserve battery life, some IoT devices will listen less frequently—a problem for applications where on-demand commands are needed, or when it's time to configure and update firmware. As explained below, these trade-offs differ based on the type of network technology that connect those devices.

Wi-SUN – Wi-SUN devices can be designed for frequent (up to 10 seconds), low-latency communication that draws less than 2 µA when resting and, more importantly, only 8 mA when listening.

LoRaWAN – LoRaWaN devices are designed for infrequent communication (up to 128 seconds) and draws 2 µA when resting and 12 mA when listening. The longer latency compromises the ability for the device to receive on-demand commands.

NB-IoT – NB-IoT devices are designed for infrequent communications (up to 600+ seconds). Power consumption during resting and listening have not been published as of this printing.

❖ *Scalability*

Today's largest IoT networks will look small in five years. The ability to grow a network—while adding new applications —is a crucial consideration when choosing a network platform standard.

Wi-SUN – Wi-SUN-based mesh networks can scale in both capacity and size. Due to the higher bandwidth capabilities noted above, Wi-SUN-based networks provide the ability to add new applications as operational needs require. Due to the mesh topology, the network reliability improves as more devices participate in the network. Additionally, with 10s of millions of endpoints deployed across the world, including several deployments with more than 1 million devices, Wi-SUN-based IoT networks are the most proven in the world today. Successful deployments on five continents demonstrate that Wi-SUN-based networks perform well at scale in urban, suburban and rural environments.

LoRaWAN – LoRaWAN networks require the installation of additional gateways to provide the direct connections needed for adding devices to star-based networks. In real world deployments, LoRaWAN is yet to be proven at very large scale and because LoRaWAN networks do repeated transmissions to ensure reliability. It is unclear if the network will have enough capacity at large scale.

NB-IoT – Scalability is largely undetermined as NB-IoT have not yet been deployed at scale yet.

❖ *Network and Device Longevity*

A general rule of IoT networks is that longer-lasting devices connected via higher-performing networks require fewer field visits and swap-outs, resulting in lower total cost of ownership (TCO) overall. Another factor influencing device longevity is backward compatibility. When choosing a network, evaluate each platform's ability to accommodate multiple generations of hardware on a single network; this too will lower TCO by allowing you to use older devices longer.

Wi-SUN – To date, higher performing networks and backward compatibility are more prevalent among Wi-SUN-based networks. One leading provider of Wi-SUN-based networks deploys hardware designed to last 15 years, battery included. That same provider has customers who operate multiple different generations of hardware within the same networks, proving both backward and forward compatibility.

LoRaWAN – LoRaWAN Alliance members claim to offer network and devices that will last up to 10 years, however these claims will need to be backed by commercial terms and proof points that the technology will be supported for an extended period of time.

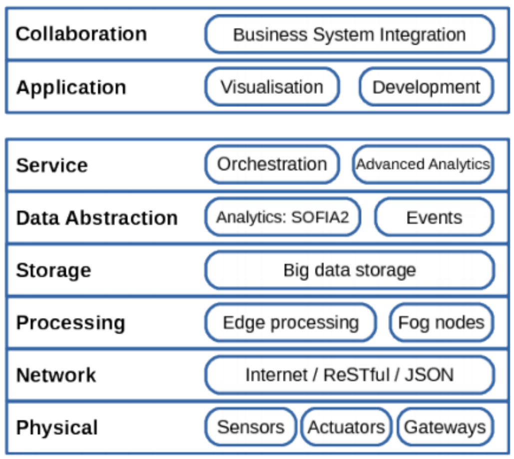
NB-IoT – Since it's too early to assess device and network longevity for NB-IoT devices, we can only evaluate how the infrastructure supporting NB-IoT devices have evolved over the past several years. Cellular carrier networks are primarily designed to support the increasingly data-intensive consumer mobile device market. Addressing those needs has resulted in an acceleration of changes to generation platforms in carrier networks. Not all of those generations have been supported on the network—a trend that may impact backward-compatibility in the future and potentially leave stranded assets. Additionally, there are a few different bands involved in 3GPP based technologies (e.g. others are pushing Cat-M1). The question becomes for a potential user is how can I guarantee my technology choice will endure for the 10-15 years required for an IoT device.

2.4 IoT Platforms

Currently, according to the Internet of Things (IoT) Platform Market Report 2017-2022 [16] there are more than 360 IoT platforms in the market, so it is impossible to describe each of these platforms.

Many of the platforms are proprietary and either fully or partially closed source, e.g., Axeda, Carriots, and Niagara. Therefore, only some of the most widely used open source IoT platforms will be described in this section.

Eight-layer IoT architecture, adopted from the UNIFY-IoT project report: At the bottom, we have the usual physical and network layers, followed by processing, storage, data abstraction and service layers, roughly corresponding to the layers typical in the OSI 7-layer model and many other architectures. On the top there are application and collaboration layers.



Eight-layer IoT architecture, adopted from the UNIFY-IoT project report

2.4.1 SOFIA2

SOFIA (Smart Objects For Intelligent Application) is a middleware architecture that allows interoperability between various devices and systems developed in a project involving companies such as Nokia, Philips, Fiat, Acciona and Indra. SOFIA is an open source platform, multiplatform (Windows, Linux, Android, ...), multilanguage (Java, Javascript, Arduino, ...) and has implementations for various communication protocols such as TCP, HTTP (Rest and web services), Ajax Push, etc.

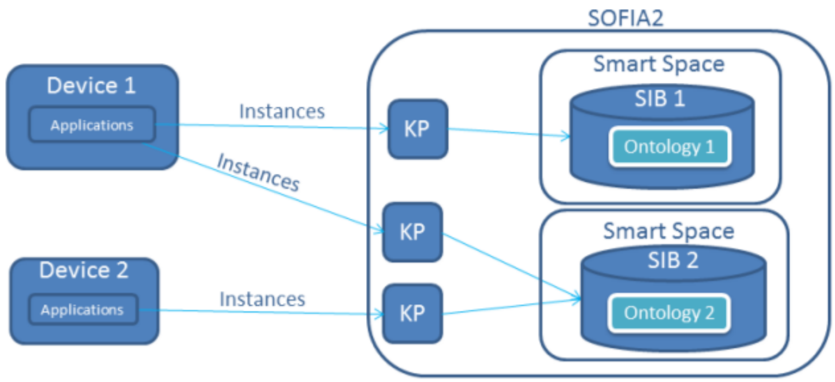
Later, Indra improved this project by creating a platform called SOFIA2 that focused on business use, adding features such as Big Data interfaces, back-end integration capabilities through standard protocols or REST interfaces to be able to connect from smartphones and other devices.

The architecture of SOFIA2 is formed by various elements that are going to be described below. The Smart Space is the virtual environment where different applications interact with other applications to provide complex functionality, and can also communicate with other Smart Space by establishing trust relationships. The core of a Smart Space is the Semantic Broker Information (there is usually one SIB in each Smart Space, but it can have several to communicate different ontologies, which are semantic descriptions of a set of classes and attributes in order to represent objects), which is in charge of receiving, processing and storing all the information of the applications connected to the platform.

Each SIB is designed for a single ontology, defined by a JSON structure.

On the other hand, the Knowledge Processor (KP) allows the connection of each application with the Smart Space through the SIB, allowing the execution of operations with the SIB. There are three types of KP:

- **Producer**: enter information in the SIB.
- **Consumer**: extracts information from the SIB.
- **Prosumer**: enter information in the SIB and extract it from it.

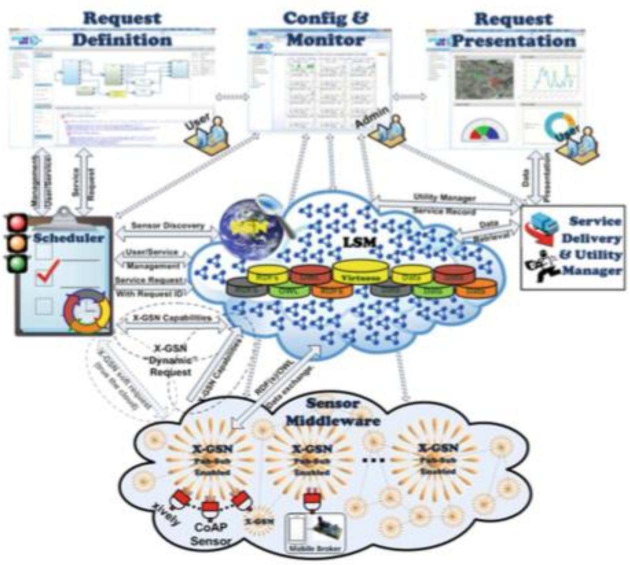


Main concepts of SOFIA2 [16]

To establish a connection from a client you need a KP name, a KP identifier and a token. The Smart Space Access Protocol (SSAP) is the standard language used in communication between SIBs and KPs, which in turn uses a JSON format. In addition, SOFIA2 includes tools for the analysis of Big Data, such as the storage of data in real time and a history of this data, a visual tool (SOFIA2 DATAFLOW) for the control of the data that is introduced in the platform or tools of machine learning for the treatment of data.

2.4.2 OpenIoT

OpenIoT is an open source middleware platform for the integration of IoT and the cloud, its main objective being to allow a flexible configuration and the deployment of algorithms for the collection and filtering of data transmitted by devices connected to the Internet. The platform allows the integration of IoT services dispersed administratively and geographically. It is also necessary to emphasize that Open IoT allows to provide IoT services with immediate payment on demand such as "Sensing-as-a-Service" or sensorization as a service.



Main components of the OpenIoT platform [16]

The Extended Global Sensor Network (X-GSN) acts as a hub between the IoT platform and physical devices: it collects, filters and combines the data flows of physical devices and virtual sensors. This data is stored in a database in the cloud through a middleware semantic platform, where annotated data is used to store the data and sensor metadata as triples in RDF format. In addition, a classifier (Scheduler) is used to ensure that each requested service uses the resources it needs.

The Service Delivery & Utility Manager has two functionalities. On the one hand, it combines the data flows as indicated by the workflows of the service within the OpenIoT system to deliver the demanded service, making use of the information provided by the Scheduler. On the other hand, this component acts as a measurement service that keeps track of measurements for each individual service. This functionality can help in charges for the use of the service and in the optimization of resources.

The platform has graphical user interfaces to perform service requests (Request Definition) and to view the results of the services used (Request Presentation).

2.4.3 UniversAAL IoT

UniversAAL IoT is an open source platform that enables the continuous integration of devices, services and applications on a large scale, implementing semantic interoperability for service-oriented architectures at the communication protocol level. The core of the platform is a middleware that understands data and functionality only through the definition of ontologies. This middleware is basically a facilitator of the exchange of messages between autonomous components.

One of the most important processes of UniversAAL IoT is the discovery of semantics. The first level of discovery is related to the mutual discovery of middleware instances in a subnet, which is based on the broadcast of a standard network. When a node is discovered, the pairing phase begins to check compatibility and authorization. If pairing occurs, communication is established based on the concept of a connector, which enables the creation of a logical network called uSpace, whose purpose is the distribution of messages in a secure manner between nodes of that same network.

The messages are classified according to communication patterns, which translates into the existence of different buses, each one to serve a specific purpose. The second level of discovery is the discovery of semantics, provided by middleware buses. The modules that connect to the buses must semantically describe their capabilities and requirements, and then register them with the buses. The buses are based on a semantic pairing to route messages with semantically formulated content, resulting in the discovery of the right provider (of functionality) or of the consumer (of data).

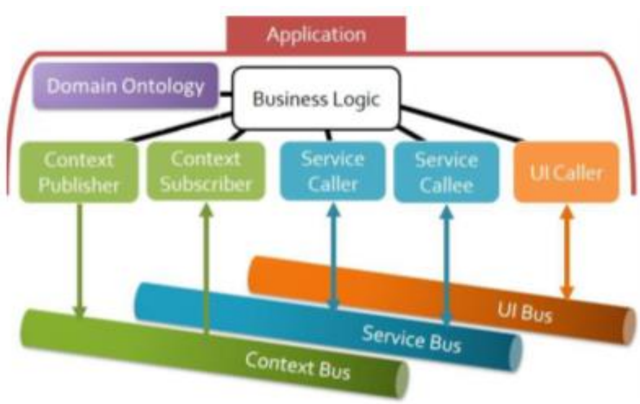


Diagram of operation of the UniversAAL IoT platform [16]

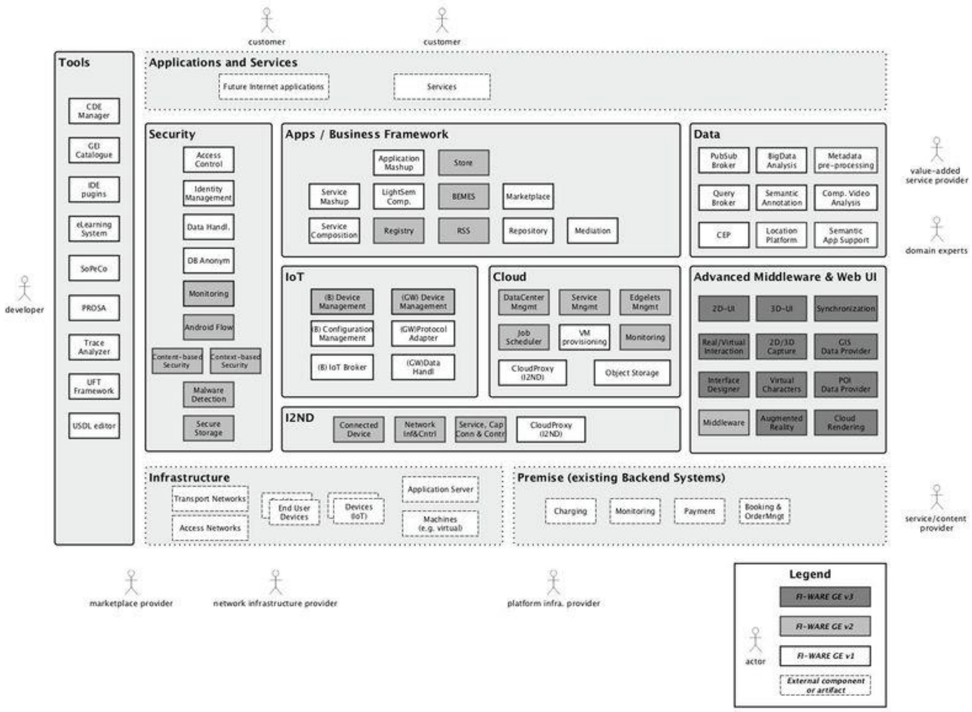
La abstracción del bus es el concepto principal detrás del paradigma de la interoperabilidad semántica tal como lo realiza UniversAAL.

2.4.4 FI-WARE

FI-WARE is an open middleware platform for the Internet of Things, supported by the European Commission within the Future Internet Public Private Partnership Program. This platform provides public API specifications and interoperable protocols for the creation of new services and internet applications. FI-WARE provides generic enablers (Generic Enablers -GE) that cover different areas such as IoT, data processing, security, advanced user interface or the cloud.

This platform allows the portability of data for different applications and provides a set of GEs that allows access to relevant information through the NGSI API, which translates into a fast and efficient way to collect, publish, exchange and process large quantities. of data. The FI-WARE data management allows the generation of context information, the management of context changes through events and the processing of large amounts of information using Big Data techniques.

In addition, FI-WARE provides support for a cloud hosting infrastructure through GEs that are accessible through REST APIs. The FI-WARE cloud architecture includes various services, such as IaaS (Infrastructure as a Service) services: cloud computing, cloud networking, ...; data storage services and metadata in the cloud, and management services and monitoring of applications and devices.



FI-WARE architecture with the Generic Enablers [16]

In FI-WARE, the data processing GE is divided into two functional blocks. On the one hand, the batch processing block provides on-demand infrastructures that support Big Data analysis software, such as Apache Hadoop clusters, since in the FI-WARE architecture the data processing infrastructure is independent of the data storage service. On the other hand, the flow processing block provides a modular interface for real-time data processing using Apache Storm [16].

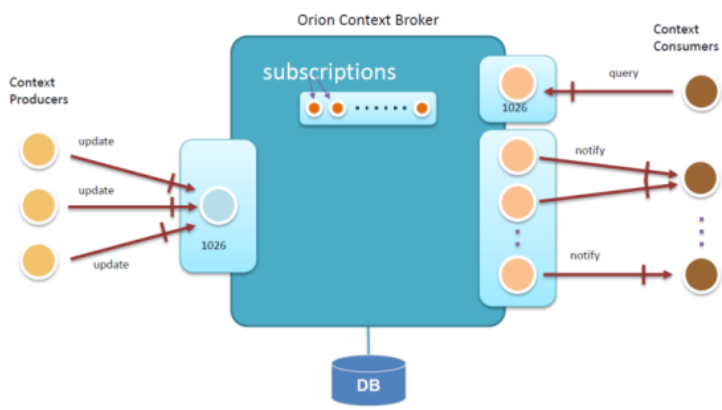
❖ *FI-WARE Orion Context Broker*

The Context Broker is responsible for managing communication between applications and devices or gateways in order to decouple IoT applications from the installations of the devices. The Context Broker communicates with IoT devices and applications through the NGSI protocol (currently using the NGSIv2 version) and allows the generation and reception of notifications of contextual events.

The data is stored in units called entities. Each entity is identified by a unique id and belongs to a type (type), to which several entities can belong. In addition, an entity can have attributes (attributes), which describe information about the entity, through a value (value), a type of data (type) and optional metadata (metadata) to be entered by the user. It is necessary to emphasize that only the last value of each attribute of the entities is stored. This data is stored in a NoSQL database of the MongoDB type.

Entities can be classified by a hierarchical system, where the main level is a Fiware-Service, which can contain lower levels in the form of a tree, whose route is indicated by the so-called Fiware-ServicePath.

In Orion there are two different roles: producers and consumers. The producers are responsible for creating and updating the data of the entities, while consumers subscribe to Orion so that it notifies them of the data of the entities to which they have subscribed previously [17].



Different roles in Orion Context Broker [17]

❖ *FI-WARE Short Time Historic - Comet*

The FI-WARE STH is in charge of storing and managing the historical information of the entities previously registered by Orion in the database. It is necessary because in the Orion database only the last value of the attributes of each entity is stored, so it is not possible to consult historical data by consulting Orion.

In order to store the historical data of an entity, the STH must subscribe to that same Orion entity. In this way, when there is a change in the values of the attributes of this entity, by means of the subscription made by the STH, the data of said entity will be notified and stored in its own MongoDB database.

When a user wants to obtain a series of historical data, there are several filters (by type, during a specific period of time, ...) so that the user can better manage the data provided by the STH [18].

❖ *FI-WARE Complex Event Processing (CEP): Perseo*

CEP Perseo is a "distributed rule engine" [19], which analyzes the data of the entities of the Orion Context Broker in order to check whether specific rules have been met, previously indicated by the user. If Perseo detects that a value of an attribute of an entity meets a certain rule, it will generate an event such as sending an email or sending an HTTP request (GET, POST, ...) to a url with a body of the indicated request by the user.

The user must create their own rules through a POST request, including in the body of the request the rule in the rules language of Esper, which has a format very similar to the language of SQL [19].

2.4.5 Other Cloud Computing technologies

Beyond the Fi-Ware, OpenIoT and Sofia2 platforms, there are other platforms on the market for the definition and start-up of new products and services within the framework of IoT, which have a high potential for development in the coming years.

ThingSpeak, a free Web Service hosted by ioBridge that allows you to collect and store data from devices / sensors (such as Arduino or Raspberry Pi, among others), to develop IoT applications. ThingSpeak also allows to analyze and visualize the stored data, using MATLAB for it.

Nimbits, a platform to interconnect people, sensors, and software to the cloud. It has a server, Nimbits Server, that stores the data. Nimbits Public Cloud is the free version of the server in the cloud, which can be accessed to develop IoT applications.

Carriots is a cloud platform designed for IoT and M2M projects. It allows to collect and store any type of data from the sensors, build solutions thanks to its application engine and deploy a solution that includes multiple devices.

OpenPicus markets programmable modules (System on Module, SoM) with Wi-Fi, GPRS or Ethernet connectivity. These modules act as receivers and processors for the development of applications in IoT, speeding up development times thanks to the battery of libraries offered.

Xively, developed by LogMeIn, is the corporate platform for the development of IoT solutions. It simplifies the way companies securely connect their products with their customers, managing the data within the IoT framework.

WikiSensing is a platform for the management of data from sensors. It has been developed by the computer department of Imperial College London, using the cloud computing infrastructure of the College itself.

Open.sen.se, an open and free platform for any type of user (professional or amateur), to develop and test applications in IoE. At the moment, open.sen.se is in beta, so the number of users who can access to try the platform remains limited.

Lhings is based on 3 elements to define applications in IoT: devices, applications and rules that cause a specific action to be triggered before a certain event. Each device has an owner, with whom you can interact. You can also invite other users to interact with the devices.

WISE IoT is a collaboration project between Europe and Korea funded under the H2020 framework program at the EU side. It aims at deepening the interoperability and interworking of IoT existing systems. The Wise-IoT55 vision is creating a reference architecture for semantic interoperability of worldwide IoT systems. Wise-IoT is built upon existing systems, standards, and reference implementations, as far as they are publicly available, e.g. as open source.

IoT-EPI is a European initiative for IoT platform development aiming at creating opportunities for platform development, interoperability and information sharing. Currently, IoT-EPI acts as an umbrella of seven projects, namely symbIoTe, bIoTope, BIG-IoT, Agile, Vicinity, TagItSmart!, and INTER-IoT. The symbIoTe, INTER-IoT, and bIoTope projects aim at providing interconnection and interoperability of heterogeneous IoT platforms through a "super platform" that will enable a unified view of the available resources, IoT platform federation, and it will facilitate discovery, sharing of resources, and new applications.

The BIG-IoT project targets to define a unified Web API for IoT platforms, aligned with the standards currently developed by the W3C Web of Things (WoT) group. The goal of the Agile project is to build a modular and adaptive gateway for the IoT; this gateway will support multiple hardware components and various protocols, providing at the same time a unified API to applications.

At this point, we would like to present two well-established, open-source IoT platforms, that are being used both commercially and research-wise by companies and other EU projects; Kaa and DeviceHive:

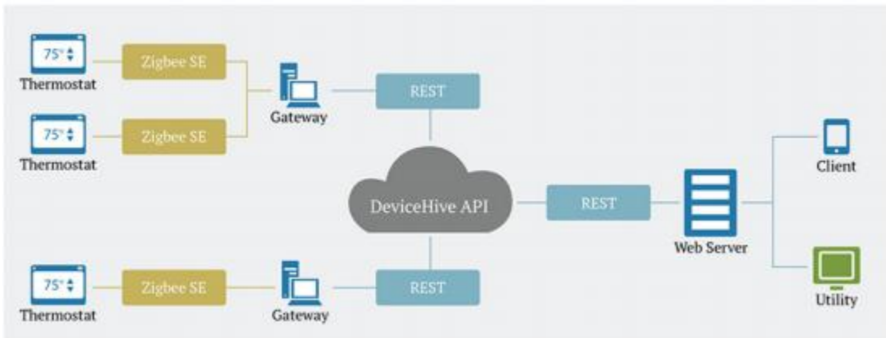
The Kaa IoT platform provides an open-source, Cloud-based, scalable IoT framework which is feature-rich and flexible in terms of data management, device integration and protection. Kaa enables data management of connected things and back-end infrastructure by providing the Kaa server (or Kaa node) and the endpoint Service Development Kit (SDK) components52. The Kaa server provides all the back-end functionality needed to operate the IoT solution and offers administrative capabilities. It handles all the communication across connected devices, including data consistency and security, device interoperability, and failure-proof connectivity and features well-established interfaces for integration with internal and external data management and analytics systems. In Kaa terminology, an endpoint SDK is a library which provides communication, data marshalling, persistence, and other functions available in Kaa for specific type of an endpoint (e.g. Java-based, C++-based, C-based, Objective-C-based). The SDKs get embedded into the connected devices and implement real-time bi-directional data exchange with the server. These SDKs can be used to create Kaa clients, which are any pieces of software that utilize Kaa functionality and are installed on the connected devices. It is the responsibility of the Kaa client to process structured data provided by the Kaa server side (configuration, notifications, etc.) and to supply data to the return path interfaces (profiles, logs, etc.). A Kaa endpoint is a particular application which uses the Kaa client SDK and resides on a particular connected device. The Kaa endpoint SDK provides functionality for communicating with the Kaa cluster, managing data locally in the client application, as well as provides integration APIs. The client SDK abstracts the communication protocol, data persistence, and other implementation details that may be specific for any concrete solution based on Kaa. A Kaa cluster represents a number of interconnected Kaa servers delivering three types of services, i.e. control, operations, and bootstrap services, that can be enabled or disabled by the Kaa administrator in an individual manner. Specifically,

- A Kaa control service is responsible for managing overall system data, processing API calls from the Web UI and external integrated systems, and delivering notifications to operations servers.
- A Kaa operations service is responsible for concurrently handling multiple requests such as endpoint registration, processing endpoint profile updates, configuration updates distribution, and notifications delivery. In the case of multiple nodes (and thus operations services), Kaa cluster provides tools for workload re-balancing at run time to effectively routing endpoints to the less loaded nodes.
- A Kaa bootstrap service is responsible for directing endpoints to operations services. Kaa endpoints have a built-in list of set bootstrap services to query and retrieve a list of currently available operations services from them, as well as security credentials.

Apart from Kaa endpoints and cluster, the Kaa instance also makes use of the following:

- an Apache ZooKeeper for services coordination, continuous monitoring and information sharing about servers' connection parameters and workload status,
- an SQL database instance to store metadata about endpoints, applications, endpoint groups, etc.,
- a NoSQL database instance (Apache Cassandra or MongoDB up to Kaa 0.8.x), with volume scaling linearly with the number of endpoints, to store measurements of devices and also information about endpoint profiles, notifications, configurations, etc.

The DeviceHive IoT platform is a customizable, feature-rich and open-source platform with a programming framework. Further, DeviceHive is a well-maintained, up-to-date solution that integrates several cutting-edge technologies in IoT communication and offers a series of powerful services and plugins supporting the development of more efficient, secure and highly performing smart applications. DeviceHive exposes an API as the central part of its framework allowing the different components to interact with each other and exchange messages in real time. The platform has its own GUI - admin console which allows end user to create and register devices, connect them to networks, and manage users' authentication and authorization services. One of the advantages of DeviceHive is that it empowers online processing and machine learning over the collected data (which are stored in a time-series database) with Apache Spark to enable business intelligence and analytics. In addition to Apache Spark, the DeviceHive platform is integrated with other state-of-the-art technologies and frameworks in big-data stacks, such as ElasticSearch, Cassandra, Kafka, etc. An additional attractive feature is the integration of the platform platform with an out-of-box gateway that supports the LoRa54 protocol for data transmission over long distances with low power consumption.



High-level architecture of DeviceHive IoT

3. Glossary of Térms

6loWPAN. IPv6 over Low Power Wireless Personal Area Networks
API. Application Programming Interface
ATA. Actual Time of Arrival (at Port of Destination)
ATD. Actual Time of Departure (from Port of Origin)
BLE. Bluetooh Low Energy
CEP. Complex Event Processing
CO2. Carbon Dioxide
CORS. Cross-Origin Resource Sharing
CPS. Cyber Physical System
CTS. Container Tracking Service
DAG. Directed Acyclic Graph
DLT. Distributed Ledger Technologies
DSNS. Domain Sensor Name Server
EPL. Event Processing Language
ETA. Estimated Time of Arrival (at Port of Destination)
ETD. Estimated Time of Departure (from Port of Origin)
GPS. Global Positioning System
HMI. Human Machine Interface
HTTP. Hypertext Transfer Protocol
HTML. HyperText Markup Language
IANA. Internet Assigned Numbers Authority
ICO. Initial Coin Offering
IDE. Integrated Development Environment
IEEE. Institute of Electrical and Electronics Engineers
IETF. Internet Engineering Task Force
IoE. Internet of Everythings
IoT. Internet of Things
ISO. International Organization for Standardization
JSON. JavaScript Object Notation
LPWAN. Low-Power Wide-Area Network
M2M. Machine to Machine
MNO. Mobile Network Operator
O2. Oxygen
OTT. Over The Top
REST. Representational State Transfer
RFC. Request for Comments
Router ABC. Always Best Connected
SATDR. Group of Distributed Real Time Systems and Applications
STH. Short Time Historic
TEU. Twenty-foot Equivalent Unit
TMS. Traveling Messaging System
WI-SUN. Wireless Smart Ubiquitous Network

4. BIBLIOGRAPHY

[1] https://blogthinkbig.com/8-400-millones-de-dispositivos-estaran-conectados-a-internet-a-finales-de-2017
[2] Government of the United States, Official information regarding the Global Positioning System, https://www.gps.gov/
[3] Adafruit Inudstries, Adafruit Ultimate GPS HAT for Raspberry Pi https://www.adafruit.com/product/2324
[4] Omega engineering, ¿Qué es un sensor termopar? https://es.omega.com/prodinfo/termopares.html
[5] K-type thermocouple reference tables: https://es.omega.com/temperature/pdf/Type_K_Thermocouple_Reference_Table.pdf
[6] Sensirion sensors, Digital Humidity Sensor SHT7x, https://www.sensirion.com/en/environmental-sensors/humidity-sensors/pintype-digital-humidity-sensors/
[7] Omega engineering, ¿Qué es un acelerómetro? https://es.omega.com/prodinfo/acelerometro.html
[8] Rs store, Sensor ultrasónico RS Pro, https://es.rs-online.com/web/p/sensores-de-proximidad-ultrasonicos/2370799/
[9] LoRa Alliance, What is the LoRaWAN Specification? https://www.lora-alliance.org/about-lorawan
[10] LoRa Alliance, A technical overview of LoRa and LoRaWAN, November 2015
[11] LoRa Alliance, LoRaWAN 101A Technical Introduction, 2015
[12] Sigfox, Sigfox technology overview https://www.sigfox.com/en/sigfox-iot-technology-overview
[13] Carles Antón-Haro and Mischa Doler, Machine-to-machine (M2M) Communications, 2015
[14] Pilar Andrés-Maldonado y PabloAmeigeras, NarrowBand IoT Data Transmission Procedures for Massive Machine Type Communications¸ Universidad de Granada
[15] Rohde&Schwarz, NarrowBand Internet of Things Whitepaper, https://www.rohde-schwarz.com/es/aplicaciones/internet-de-las-cosas-en-banda-estrecha-white-paper_230854-314242.html
[16] Activage Project, Report on IoT European Platforms, September 2017, https://iot-epi.eu/
[17] Documentación oficial de FIWARE Orion: https://fiware-orion.readthedocs.io
[18] Official documentation of FIWARE STH Comet https://fiware-sth-comet.readthedocs.io
[19] Official documentation of FIWARE Perseo (CEP) http://fiware-iot-stack.readthedocs.io/en/latest/cep/index.html
[20] MongoDB Inc, What is MongoDB? https://www.mongodb.com/what-is-mongodb
[21] University of Berkeley, Cyber-physical Systems https://ptolemy.berkeley.edu/projects/cps/
[22] Docker Inc., What is Docker, https://www.docker.com/what-docker
[23] Docker Inc., What is a container, https://www.docker.com/what-container
[24] Canonical Group, LXD Introduction, https://linuxcontainers.org/lxd/#
[25] Canonical Group, The no-nonsense way to accelerate your business with containers, February 2017
[26] Maersk Line, What is Remote Container Management, https://www.maerskline.com/shipping/remote-container-management
[27] Loginno, Smart container https://loginno.com/
[28] Traxens, TRAXENS Technology, http://www.traxens.com/en/technology
[29] Pointer Telocation LTD, Cellocator, https://www.cellocator.com/products/cellotrack/
[30] Tritón of GlobeArea, http://www.globearea.com/triton
[31] Weightless, http://www.weightless.org/
[32] Weight Sensor of Nanolike https://www.nanolike.com/fill-level-monitoring-solution/
[33] https://www.u-blox.com/en
[34] Blockstack https://bitcoinexchangeguide.com/kadena/
[35] Kadena https://kadena.io/
[36] https://www.plugintoiot.com/sub-1ghz-modules/#CC1310-module
[37] https://www.wi-sun.org/