International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

Y.4460

(06/2019)

SERIES Y: GLOBAL INFORMATION INFRASTRUCTURE, INTERNET PROTOCOL ASPECTS, NEXT-GENERATION NETWORKS, INTERNET OF THINGS AND SMART CITIES

Internet of things and smart cities and communities – Frameworks, architectures and protocols

Architectural reference models of devices for Internet of things applications

Recommendation ITU-T Y.4460

ITU-T

International Telecommunication Union

ITU-T Y-SERIES RECOMMENDATIONS

GLOBAL INFORMATION INFRASTRUCTURE, INTERNET PROTOCOL ASPECTS, NEXT-GENERATION NETWORKS, INTERNET OF THINGS AND SMART CITIES

GLOBAL INFORMATION INFRASTRUCTURE	
General	Y.100–Y.199
Services, applications and middleware	Y.200–Y.299
Network aspects	Y.300–Y.399
Interfaces and protocols	Y.400–Y.499
Numbering, addressing and naming	Y.500–Y.599
Operation, administration and maintenance	Y.600–Y.699
Security	Y.700–Y.799
Performances	Y.800–Y.899
INTERNET PROTOCOL ASPECTS	
General	Y.1000–Y.1099
Services and applications	Y.1100–Y.1199
Architecture, access, network capabilities and resource management	Y.1200–Y.1299
Transport	Y.1300–Y.1399
Interworking	Y.1400–Y.1499
Quality of service and network performance	Y.1500–Y.1599
Signalling	Y.1600–Y.1699
Operation, administration and maintenance	Y.1700–Y.1799
Charging	Y.1800–Y.1899
IPTV over NGN	Y.1900–Y.1999
NEXT GENERATION NETWORKS	
Frameworks and functional architecture models	Y.2000–Y.2099
Quality of Service and performance	Y.2100–Y.2199
Service aspects: Service capabilities and service architecture	Y.2200–Y.2249
Service aspects: Interoperability of services and networks in NGN	Y.2250–Y.2299
Enhancements to NGN	Y.2300–Y.2399
Network management	Y.2400–Y.2499
Network control architectures and protocols	Y.2500–Y.2599
Packet-based Networks	Y.2600–Y.2699
Security	Y.2700–Y.2799
Generalized mobility	Y.2800–Y.2899
Carrier grade open environment	Y.2900–Y.2999
FUTURE NETWORKS	Y.3000–Y.3499
CLOUD COMPUTING	Y.3500–Y.3999
INTERNET OF THINGS AND SMART CITIES AND COMMUNITIES	
General	Y.4000–Y.4049
Definitions and terminologies	Y.4050–Y.4099
Requirements and use cases	Y.4100–Y.4249
Infrastructure, connectivity and networks	Y.4250–Y.4399
Frameworks, architectures and protocols	**Y.4400–Y.4549**
Services, applications, computation and data processing	Y.4550–Y.4699
Management, control and performance	Y.4700–Y.4799
Identification and security	Y.4800–Y.4899
Evaluation and assessment	Y.4900–Y.4999

For further details, please refer to the list of ITU-T Recommendations.

Recommendation ITU-T Y.4460

Architectural reference models of devices for Internet of things applications

Summary

Recommendation ITU-T 4460 describes the architectural reference models of devices for Internet of things (IoT) applications, based on a classification of devices defined by processing power and communication capabilities. The architectural reference models described also includes the device's functional entities and the functional entities interaction for each device's architectural reference model.

Processing power and communication capabilities define how the device communicates and interacts with other entities in an IoT solution. By correlating the processing and communication capabilities classifications, it is possible to enumerate three types of devices:

1) low processing and low connectivity device – LPLC device;

2) low processing and high connectivity device – LPHC device;

3) high processing and high connectivity device – HPHC device.

NOTE – Devices with no processing capabilities are also not considered on this Recommendation because they are simple devices (ID tags) that were defined in Recommendation ITU-T Y.4108/Y.2213.

History

Edition	Recommendation	Approval	Study Group	Unique ID*
1.0	ITU-T Y.4460	2019-06-13	20	11.1002/1000/13921

Keywords

Architectural reference models, devices, interoperability, IoT.

* To access the Recommendation, type the URL http://handle.itu.int/ in the address field of your web browser, followed by the Recommendation's unique ID. For example, http://handle.itu.int/11.1002/1000/11830-en.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure, e.g., interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had not received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database at http://www.itu.int/ITU-T/ipr/.

Table of Contents

Recommendation ITU-T Y.4460

Architectural reference models of devices for Internet of thingsapplications

1 Scope

This Recommendation describes the architectural reference models of devices for Internet of things (IoT) applications, by providing:

- a classification of devices for IoT applications, regarding their processing and connectivity capabilities;
- the architectural reference models of devices for IoT applications based on the classification above;
- the functional entities for each architectural reference model;
- the functional entities interaction for each architectural reference model.

2 References

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

[ITU-T Y.4000] Recommendation ITU-T Y.4000 (2012), *Overview of Internet of things*.

[ITU-T Y.4108] Recommendation ITU-T Y.4108/Y.2213 (2008), *NGN service requirements and capabilities for network aspects of applications and services using tag-based identification*.

3 Definitions

3.1 Terms defined elsewhere

This Recommendation uses the following terms defined elsewhere:

3.1.1 application [b-ITU-T Y.2091]: A structured set of capabilities, which provide value-added functionality supported by one or more services, which may be supported by an API interface.

3.1.2 cloud service [b-ITU-T Y.3500]: One or more capabilities offered via cloud computing invoked using a defined interface.

3.1.3 device [ITU-T Y.4000]: With regard to the Internet of things, this is a piece of equipment with the mandatory capabilities of communication and the optional capabilities of sensing, actuation, data capture, data storage and data processing.

3.1.4 gateway [b-ITU-T Y.4101]: A unit in the Internet of things which interconnects the devices with the communication networks. It performs the necessary translation between the protocols used in the communication networks and those used by devices.

3.1.5 ID tag [ITU-T Y.4108]: A physical object which stores one or more identifiers and optionally application data such as name, title, price, address, etc.

NOTE – It may have a communication capability with an ID terminal depending on its implementation.

3.1.6 ID terminal [ITU-T Y.4108]: A device with a data reading and optional writing capability which reads (and optionally writes) identifier(s) and optionally application data from/into an ID tag.

NOTE – The data reading (and optionally writing) capability depends on its implementation.

3.1.7 Internet of Things (IoT) [ITU-T Y.4000]: A global infrastructure for the information society, enabling advanced services by interconnecting (physical and virtual) things based on existing and evolving interoperable information and communication technologies.

NOTE 1 – Through the exploitation of identification, data capture, processing, and communication capabilities, the IoT makes full use of things to offer services to all kinds of applications, while ensuring that security and privacy requirements are fulfilled.

NOTE 2 – From a broader perspective, the IoT can be perceived as a vision with technological and societal implications.

3.1.8 service [b-ITU-T Y.2091]: A set of functions and facilities offered to a user by a provider.

3.1.9 tag-based identification [ITU-T Y.4108]: The process of specifically identifying a physical or logical object from other physical or logical objects by using identifiers stored on an ID tag.

3.1.10 thing [ITU-T Y.4000]: With regard to the Internet of things, this is an object of the physical world (physical things) or of the information world (virtual things), which is capable of being identified and integrated into communication networks.

3.2 Terms defined in this Recommendation

This Recommendation defines the following terms:

3.2.1 HPHC device: An IoT device that not only has high connectivity capabilities, allowing it to directly connect to applications and cloud services, but also has high enough processing capabilities to make decisions and run complex algorithms (e.g., artificial intelligence (AI) related algorithms). Devices are autonomous. They make decisions about their own functions and can also coordinate other devices.

3.2.2 LPHC device: An IoT device that only acts as an interface for data collection from physical things or the surrounding environment, and/or performs operations on physical things or the surrounding environment. This device has sufficient connectivity capabilities to directly connect to the communication networks.

3.2.3 LPLC device: An IoT device that only acts as an interface for data collection from physical things or the surrounding environment, and/or performs operations on physical things or the surrounding environment. This device does not have sufficient processing capabilities to make decisions or run complex algorithms; it also does not have sufficient connectivity capabilities to directly connect to the communication networks.

4 Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

ADC	Analogue to Digital Converter
AI	Artificial Intelligence
API	Application Programming Interface
D2D	Device to Device
DAC	Digital to Analogue Converter
GPIO	General Purpose Input/Output
GPRS	General Packet Radio Service

HPHC	High Processing High Connectivity
I2C	Inter-Integrated Circuit
IoT	Internet of Things
IP	Internet Protocol
IR	Infrared
LPG	Liquefied Petroleum Gas
LPHC	Low Processing High Connectivity
LPLC	Low Processing Low Connectivity
LTE	Long Term Evolution
NFC	Near Field Communication
NGN	Next Generation Network
PWM	Pulse-Width Modulation
QoS	Quality of Service
RFID	Radio Frequency Identification
SPI	Serial Peripheral Interface
WCDMA	Wide-band Code Division Multiple Access
WECA	Wireless Ethernet Compatibility Alliance
Wi-Fi	Wireless Fidelity

5 Conventions

In this Recommendation:

The keywords **"is required to"** indicate a requirement which must be strictly followed and from which no deviation is permitted if conformance to this document is to be claimed.

The keywords **"is recommended"** indicate a requirement which is recommended but which is not absolutely required. Thus, this requirement need not be present to claim conformance.

The keywords **"can optionally"** and **"may"** indicate an optional requirement which is permissible, without implying any sense of being recommended. These terms are not intended to imply that the vendor's implementation must provide the option and the feature can be optionally enabled by the network operator/service provider. Rather, it means the vendor may optionally provide the feature and still claim conformance with the specification.

6 Introduction

As described in [ITU-T Y.4000], "A device is a piece of equipment with the mandatory capabilities of communication and optional capabilities of sensing, actuation, data capture, data storage, and data processing. The devices collect various kinds of information and provide it to the information and communication networks for further processing. Some devices also execute operations based on information received from the information and communication networks".

From the perspective of Internet of things, a device can be anything with communication capabilities. This heterogeneity of IoT devices is one of the characteristics that make the development of IoT solutions quite complex.

IoT devices can be categorized or classified according to some similarities so that it is possible to describe some classes of devices that share similar characteristics. From this classification, it is possible to design architectural reference models that define the main functional entities of the IoT devices and serve as a reference design for the implementation of solutions.

This Recommendation, therefore, defines a classification of IoT devices and presents the architectural reference model for each class of IoT device.

6.1 IoT devices classification

It is well known that IoT comprises an enormous number of devices with different capabilities. Supporting interoperability of such heterogeneous devices is one of the important challenges to operationalize IoT.

[ITU-T Y.4000] categorizes IoT devices as: data-carrying devices, data-capturing devices, sensing and actuating devices and general devices. This categorization is regarding the way in which a device interacts with physical things.

Data-carrying and data-capturing devices are responsible for reading/writing data from/to physical things. Examples of these devices include infrared (IR) readers, card readers, barcode scanners, etc.

According to [ITU-T Y.4000], "Sensing and actuating devices may detect or measure information related to the surrounding environment and convert it into digital electronic signals. It may also convert digital electronic signals from the information networks into operations".

A general device, as defined in [ITU-T Y.4000], has embedded processing and communication capabilities, and may include equipment and appliances for different IoT application domains. A general device is also a physical thing or a set of physical things.

From an architectural point of view and based on this categorization, it is possible to classify the devices accordingly to two capabilities that are among the most essentials: processing power and communication. Those capabilities were chosen because they define how the device communicates and interacts with other entities in an IoT solution, they are the ones that limit or potentialize devices functioning.

6.1.1 Regarding processing capabilities

The processing capability defines how the devices can perform computational tasks and execute algorithms. It is possible to classify a device as:

- No processing capability;
- Low processing capability;
- High processing capability.

6.1.1.1 Devices with no processing capabilities

These devices have no processing capabilities to execute any behavior. With regard to the Internet of Things, they are passive devices. They are low-cost devices with no microcontrollers. For instance, ID tags using identification technologies such as radio frequency identification (RFID) or near field communication (NFC), applied to a disposable package. It is not reasonable to embed microcontrollers that are more expensive than the package itself.

6.1.1.2 Devices with low processing capabilities

These devices have processing capabilities just sufficient for reading/writing data from/to sensors/actuators and sending/receiving those data as messages to IoT applications. They have not sufficient processing capabilities to make decisions or run complex algorithms. For this reason, they can rely on other architectural elements like cloud services, for data storage or data processing. Usually, they are low-cost devices with very limited microcontrollers to make the product economically viable. For instance, smart lights or door sensors. It is not reasonable to embed powerful microcontrollers that are more expensive than the product itself.

6.1.1.3 Devices with high processing capabilities

These devices have enough processing capabilities to make decisions and run complex algorithms. They also can directly coordinate other devices. Usually, they are high-cost devices in the first place. Thus, embedding a powerful microcontroller running an embedded operating system is economically viable since its cost can be diluted within the product cost. An example of this type of device would be a cooling/heating system with a smart thermostat.

6.1.2 Regarding communication capabilities

Communication capabilities defines how the devices can connect to the communication networks. As communication is a mandatory capability for an IoT device, it is possible to classify a device as having:

- low connectivity capability;
- high connectivity capability.

6.1.2.1 Devices with low connectivity capabilities

This classification considers a device with low connectivity capability as one that does not directly connect to a communication networks (i.e., does not implement an IP stack or any other NGN stack). These devices can not directly communicate with the applications or cloud services through the Internet. For this reason, they must rely on other architectural elements such as gateways for protocol translation and Internet connectivity.

6.1.2.2 Devices with high connectivity capabilities

This classification considers a device with high connectivity capability as one that can directly connect to a communication networks (i.e., it has implemented an IP stack or any other NGN stack). Those devices can directly communicate with applications or cloud services through the Internet. For this reason, they do not need to rely on other architectural elements such as gateways for protocol translation nor Internet connectivity.

7 Architectural reference models

This clause describes the architectural reference models of devices for IoT applications, considering the classification presented in clause 6.1. By correlating the processing and communication capabilities classifications, it is possible to enumerate three types of devices:

1) low processing and low connectivity (LPLC) device;
2) low processing and high connectivity (LPHC) device;
3) high processing and high connectivity (HPHC) device.

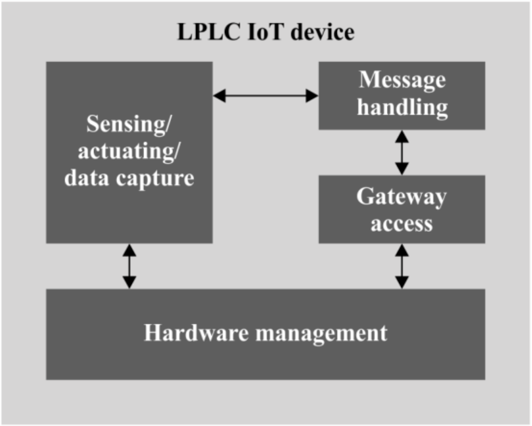
As the connectivity capability also depends on processing capability, the combination of high processing and low connectivity is not usual (because a device that already has high processing power will also have high connectivity capabilities) and will not be considered on this Recommendation.

Devices with no processing capabilities are also not considered on this Recommendation because they are simple devices (e.g., ID tags); these are defined in [ITU-T Y.4108].

7.1 Architectural reference model for LPLC devices

In some cases, IoT devices simply act as an interface to collect data from physical things or the surrounding environment, and/or perform operations on physical things or the surrounding environment. These devices do not have sufficient processing capabilities to make decisions or run complex algorithms; they also do not have sufficient connectivity capabilities to directly connect to the communication networks (i.e., they do not implement an IP stack). For these reasons, a gateway is needed to act as an intermediary between these devices and the IoT (e.g., cloud services and applications).

Figure 1 shows the architectural reference model for an LPLC device. In this reference model, the message handling and gateway access functional entities are the core functional entities. Besides the core functional entities, other functional entities in the architectural reference model are often commonly used.



Y.4460(19)_F01

Figure 1 – Architectural reference model for LPLC devices

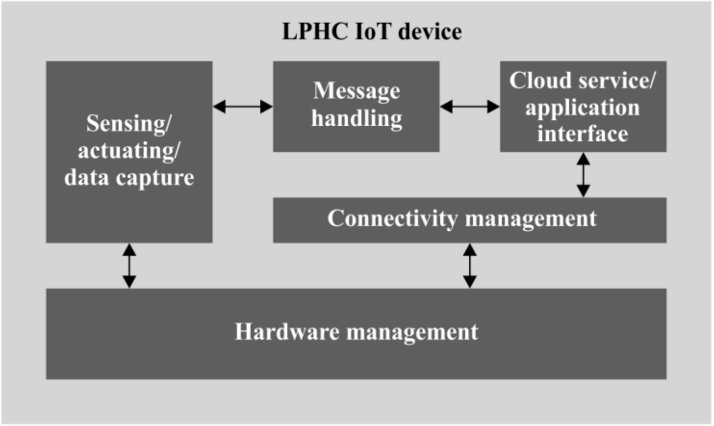
Below are descriptions for the functional entities introduced by this LPHC architectural reference model:

- **sensing/actuating/data capture functional entity**: provides functions to read data from sensors, write data to actuators and capture data from data-carrying devices or data carriers attached to physical things;
- **message handling functional entity**: provides functions to send and receive messages, by using an application layer protocol. It also can provide a state machine for handling incoming messages;
- **gateway access functional entity**: provides functions for communication management with the gateway;
- **hardware management functional entity**: provides functions for accessing the hardware (sensors and/or actuators, physical communication interfaces, hardware peripherals such as timers, analogue-to-digital converters (ADCs), etc.).

7.2 Architectural reference model for LPHC devices

These devices have enough connectivity capabilities to directly communicate with the Internet (i.e., they implement an IP stack). Thus, there is no need for gateways mediating the communication between the devices and the applications or cloud services. However, devices still do not have sufficient processing capabilities to make decisions or run complex algorithms.

Figure 2 shows the architectural reference model for an LPHC device. In this reference model, the gateway cccess functional entity is exchanged by a connectivity management functional entity. There is also a cloud service/application interface functional entity, that is responsible for understanding the application layer protocols used by the cloud service or application and its application programming interfaces (APIs) for sending/receiving messages and performing cloud services or applications operations.



Y.4460(19)_F02

Figure 2 – Architectural reference model for LPHC devices

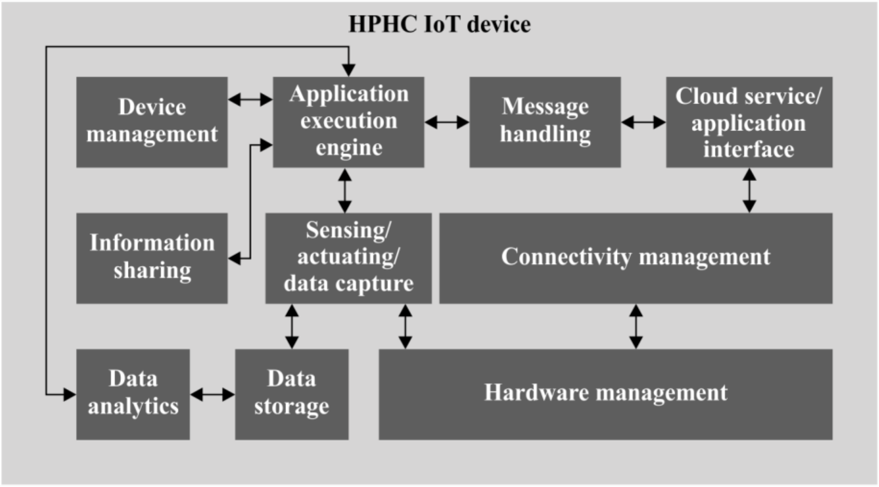
Below are descriptions for the functional entities introduced by this LPHC architectural reference model, in comparison with the reference model for an LPLC device:

- **cloud service/application interface functional entity**: provides functions to interact with the IoT cloud service or IoT application, send and receive messages to the IoT cloud service or IoT application, register/authenticate the device, etc.;
- **connectivity management functional entity**: provides functions for connectivity management between the device and the communication network.

7.3 Architectural reference model for HPHC devices

These devices not only have high connectivity capabilities, making them able to directly connect to applications and cloud services, but also sufficiently high processing capabilities to make decisions and run complex algorithms (e.g., artificial intelligence (AI)-related algorithms). These devices are autonomous; they make decisions about their own functions and can also coordinate other devices.

Figure 3 shows the architectural reference model for an HPHC device. In this reference model, the application execution engine functional entity is the core functional entity, providing application execution capabilities and interacts directly or indirectly with all other functional entities. Besides the core functional entity, other functional entities in the architectural reference model are often commonly used.



Y.4460(19)_F03

Figure 3 – Architectural reference model for HPHC devices

Below are descriptions for the functional entities introduced by this HPHC reference model, in comparison with the reference model for an LPHC device:

- **application execution engine functional entity**: provides functions to install, delete, update and run applications on devices. Provides to applications access to other functional entities;
- **device management functional entity**: provides functions to manage other devices connected to the device and the device itself;
- **information sharing functional entity**: provides functions such as device to device interaction (data exchange between devices), service discovery, service monitoring and service discovery interoperability;
- **data analytics functional entity**: provides functions for data processing and autonomous decision by running analytics and AI algorithms;
- **data storage functional entity**: provides functions of data storing and retrieving.

8 Functional entities of devices for IoT applications

This clause describes all functional entities that are listed in the architectural reference models of devices for IoT applications in clause 7.

8.1 Sensing/actuating/data capture functional entity

This functional entity is responsible for providing capabilities for:

- reading data from sensors, for gathering information from the surrounding environment;
- writing data to actuators, in order to perform actions in the surrounding environment;
- interfacing with data capture hardware in order to gather data from physical things.

NOTE – It is not mandatory to have all functionalities present in this functional entity. The functionalities for sensing/actuating/data capture are independent and will be present depending on the intended use by the device.

8.2 Message handling functional entity

This functional entity is responsible for providing capabilities for:

- receiving data or control commands, and encapsulating then as messages according to the application layer protocols;
- encapsulating messages received according to application layer protocols, and retrieving the data and control commands from them;
- using connectivity management or gateway access functional entities for sending messages to other devices, gateways, cloud services or applications;
- receiving messages from connectivity management or gateway access functional entities.

Optionally, this functional entity can have functions for protocol handling and/or translation, including a protocol engine for handling protocol messages.

8.3 Gateway access functional entity

This functional entity is responsible for providing capabilities for:

- connection establishment and connection termination between the device and gateway, which may include authentication and authorization;
- message packetization and transferring between device and gateway.

8.4 Hardware management functional entity

This functional entity is responsible for providing capabilities for:

- providing access to the microcontroller/microprocessor peripherals such as timers, communication ports (e.g., serial, serial peripheral interface (SPI), inter-integrated circuit (I2C), etc), general-purpose input/output (GPIO), ADCs and digital-to-analogue converters (DACs), pulse-width modulations (PWMs), etc.);
- enabling access to the hardware attached to those peripherals (e.g., radio interfaces, sensors, actuators, etc.).

8.5 Connectivity management functional entity

This functional entity is responsible for providing capabilities for:

- connectivity management, such as network connection establishment and termination between this device and other devices, and IoT cloud services or IoT applications, which may include authentication and authorization;
- monitoring of the network connection status between the device and other devices, IoT cloud services or IoT applications, if needed;
- quality of service (QoS) and traffic control according to the QoS requirements (e.g., communication delay, packet loss, etc.), if needed;
- network connection performance measurement and analysis, if needed;
- message packetization and transferring between a device and other devices, IoT cloud services or IoT applications.

8.6 Cloud service/application interface functional entity

This functional entity is responsible for providing capabilities for:

- data transfer (message and commands) between a device and IoT cloud services or applications.

Device operations on IoT cloud services or IoT applications, such as: register/unregister the device, send/receive configuration for the device, etc.

8.7 Application execution engine functional entity

This functional entity is responsible for providing capabilities for:

- local processing of applications;
- local processing and execution interacting with remote IoT applications;
- providing API support for the local applications, such as: access to hardware peripherals, message handling, data storage, analytics functions, etc.

8.8 Data analytics functional entity

This functional entity is responsible for providing capabilities for:

- analysing data;
- aggregating data from devices and applications;
- data format transformation between different data formats as required by devices and applications;
- capability extension in order to support AI capabilities.

8.9 Device management functional entity

This functional entity is responsible for providing capabilities for:

- configuration management, performance management, fault management, etc.;
- device self-management and remote maintenance;

- management of other devices connected to the device;
- device self-configuration;
- configuration of other devices connected to the device.

8.10 Information sharing functional entity

This functional entity is responsible for providing capabilities for:

- interfaces for device to device (D2D) interaction (data exchange between devices);
- automatic discovery of services enabled in devices;
- monitoring services enabled in devices;
- providing information related to services enabled in devices to other devices connected to the device.

8.11 Data storage functional entity

This functional entity is responsible for providing capabilities for:

- storing device data in a structured and persistent way;
- retrieving previously stored device data in an addressable and structured way;
- updating or deleting previously stored device data.

9 Logical flows of functional entities

This clause describes the logical flow of functional entities considering the types of devices defined on clause 7.

9.1 Logical flow of functional entities for LPLC devices

9.1.1 Message handling

The message handling functional entity interacts with sensing/actuating/data capture functional entity in order to:

- get data from sensors/data capture devices and pack as messages;
- receive and unpack messages for set data into actuators.

The message handling functional entity interacts with gateway access functional entity in order to:

- send messages to the gateway;
- receive messages from the gateway.

9.1.2 Gateway access

The gateway access functional entity interacts with hardware management functional entity in order to access the communication network hardware.

9.1.3 Sensing/actuating/data capture

The sensing/actuating/data capture functional entity interacts with hardware management functional entity in order to access the sensors/actuators/data capture hardware.

9.2 Logical flow of functional entities for LPHC devices

9.2.1 Message handling

The message handling functional entity interacts with sensing/actuating/data capture functional entity in order to:

- get data from sensors/data capture devices and pack as messages;

- receive and unpack messages for set data into actuators.

The message handling functional entity interacts with cloud service/application interface functional entity in order to:

- send messages to the cloud service or application;
- receive messages from the cloud service or application.

9.2.2 Cloud service/application interface

The cloud service/application interface functional entity interacts with connectivity management functional entity in order to access the communication network.

9.2.3 Connectivity management

The connectivity management functional entity interacts with hardware management functional entity in order to access the communication network hardware.

9.2.4 Sensing/actuating/data capture

The sensing/actuating/data capture functional entity interacts with hardware management functional entity in order to access the sensors/actuators/data capture hardware.

9.3 Logical flow of functional entities for HPHC devices

9.3.1 Application execution engine

The application execution engine functional entity interacts with message handling functional entity in order to:

- pack messages from the device for sending;
- unpack received messages.

The application execution engine functional entity interacts with sensing/actuating/data capture functional entity in order to:

- get data from sensors/data capture;
- set data into actuators.

The application execution engine functional entity interacts with device management functional entity in order to:

- manage the device itself (including configuration management, performance management, fault management, etc.);
- manage other devices connected to this device including configuration management of those devices;
- manage gateways connected to this device.

The application execution engine functional entity interacts with information sharing functional entity in order to:

- exchange data with other devices;
- search for services enabled in devices;
- monitoring services enabled in devices.

The application execution engine functional entity interacts with data analytics functional entity in order to perform a request for executing data analysis algorithms or AI algorithms.

9.3.2 Message handling

The message handling functional entity interacts with cloud service/application interface functional entity in order to:

- send messages to the cloud service or remote application;
- receive messages from the cloud service or remote application.

9.3.3 Cloud service/application interface

The cloud service/application interface functional entity interacts with the connectivity management functional entity to access the communication network.

9.3.4 Connectivity management

The connectivity management functional entity interacts with hardware management functional entity in order to access the communication network hardware.

9.3.5 Sensing/actuating/data capture

The sensing/actuating/data capture functional entity interacts with hardware management functional entity in order to access the sensors/actuators/data capture hardware.

The sensing/actuating/data capture functional entity interacts with data storage functional entity in order to store/retrieve data from sensors/actuators/data capture hardware.

9.3.6 Device management

The device management functional entity interacts with information sharing functional entity in order to:

- perform interactions (data exchange) with devices and/or applications running on other devices;
- perform operations for automatic discovery of services enabled in devices;
- perform operations for collecting information of services and monitoring service status in devices.

9.3.7 Data analytics

The data analytics functional entity interacts with data storage functional entity to access the data from sensors/actuators/data capture that are stored in the data storage functional entity.

10 Security considerations

All architectural reference models presented on this Recommendation should take into account information security best practices including a risk assessment in order to implement appropriate safeguards. As presented in this Recommendation, IoT devices have varying requirements which impacts their capabilities, including those that support information security management best practices. Therefore, it is important that information security management be considered at all three architectural reference models presented in this Recommendation.

Appendix I

Use cases for architectural reference models of devices for IoT applications

(This appendix does not form an integral part of this Recommendation.)

I.1 Use case for architectural reference model for LPLC devices

LPLC devices are suitable to the IoT use cases in environments with no direct Internet connection. As one of the most significant IoT use cases that fit in these characteristics, the smart agriculture sector has high demand for connected objects in different intelligent applications.

An use case for the architectural reference model of LPLC devices can be presented from an irrigation monitoring system, an intelligent irrigation system built from connected devices in the central irrigation pivot of plantations. The monitoring device is constructed using a low-cost resource constrained microcontroller integrated with a long-range RF module. The primary functions of the device allow sending data captured from sensors, such as on/off status, geolocation, water pressure, irrigation speed and direction of the pivot.

The device's software deploys the LPLC architecture presented on this Recommendation with the following functional entities:

- **sensing/actuating/data capture functional entity**: reads data from global positioning system (GPS), speed sensor, direction sensor, and water pressure sensor;
- **message handling functional entity**: packs data read from sensors into messages to be sent to the gateway;
- **gateway access functional entity**: establishes gateway connection and send data messages to the gateway;
- **hardware management functional entity**: implements the functions for interfacing with all sensors and the long-range RF radio;
- **security management functional entity**: performs device authentication and authorization with the gateway.

All information collected from the different pivot devices is sent to the gateway module, located at the control room of the farm, with direct Internet connectivity. It is configured with an antenna and long-range RF concentrator module. The gateway provides data to the IoT cloud service that connects to an IoT application. The gateway authenticates all the message traffic generated by the LPLC devices in an interaction with the cloud service. The irrigation monitor system is depicted in Figure I.1.

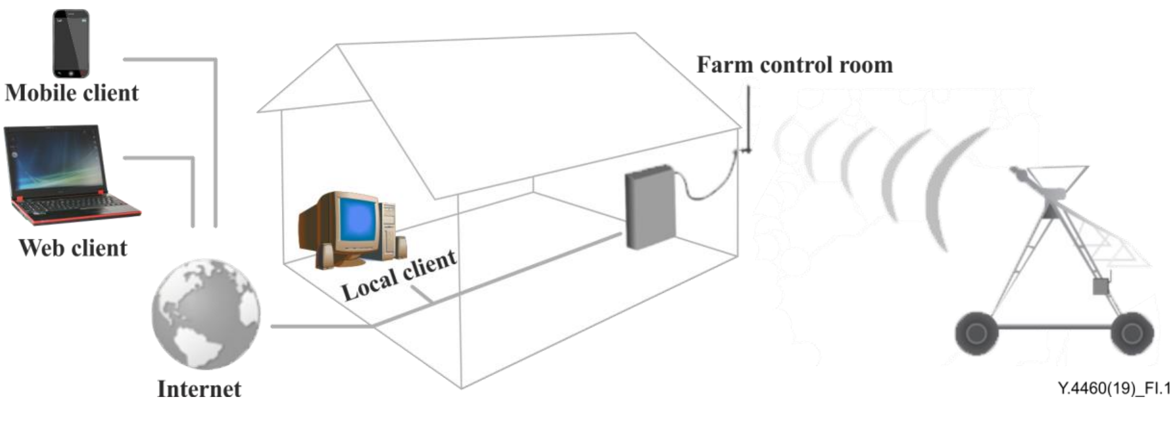
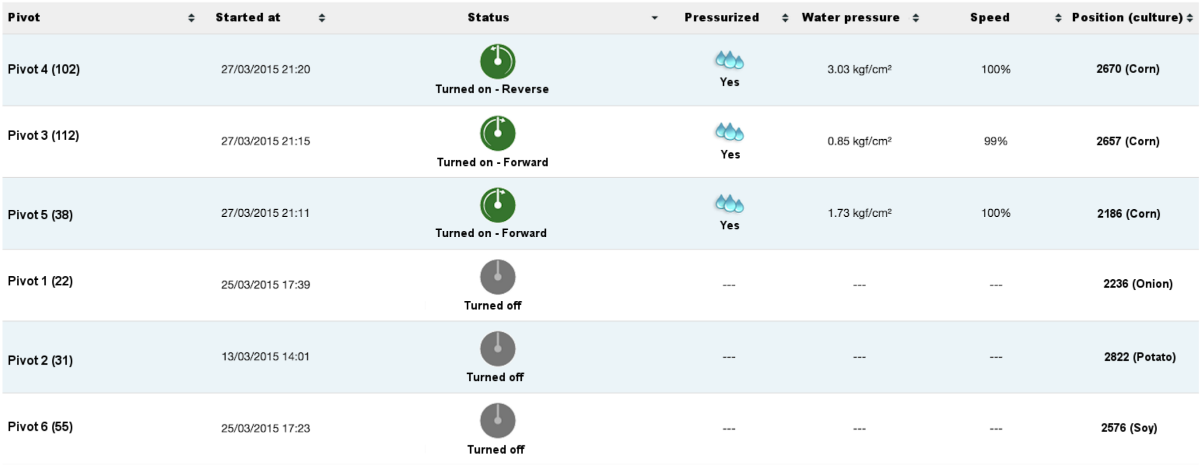


Figure I.1 – Irrigation monitoring system using LPLC architectural reference model

In addition to the data collection process, the irrigation monitoring system provides advantages, such as reducing the cost of water and electricity consumption used in the plantations. These features are achieved by the IoT web application which is responsible for analyzing pivot data and generating useful information through dashboards. Figure I.2 shows a view of the irrigation monitoring system dashboard with information from different LPLC devices.



Pivot	Started at	Status	Pressurized	Water pressure	Speed	Position (culture)
Pivot 4 (102)	27/03/2015 21:20	Turned on - Reverse	Yes	3.03 kgf/cm^2	100%	2670 (Corn)
Pivot 3 (112)	27/03/2015 21:15	Turned on - Forward	Yes	0.85 kgf/cm^2	99%	2657 (Corn)
Pivot 5 (38)	27/03/2015 21:11	Turned on - Forward	Yes	1.73 kgf/cm^2	100%	2186 (Corn)
Pivot 1 (22)	25/03/2015 17:39	Turned off	---	---	---	2236 (Onion)
Pivot 2 (31)	13/03/2015 14:01	Turned off	---	---	---	2822 (Potato)
Pivot 6 (55)	25/03/2015 17:23	Turned off	---	---	---	2576 (Soy)

Figure I.2 – Irrigation monitoring system dashboard example

In addition to the general information collected from the devices, the irrigation monitoring system also shows the region covered by the monitoring devices and its current status. Figure I.3 shows the geolocation status plot example.

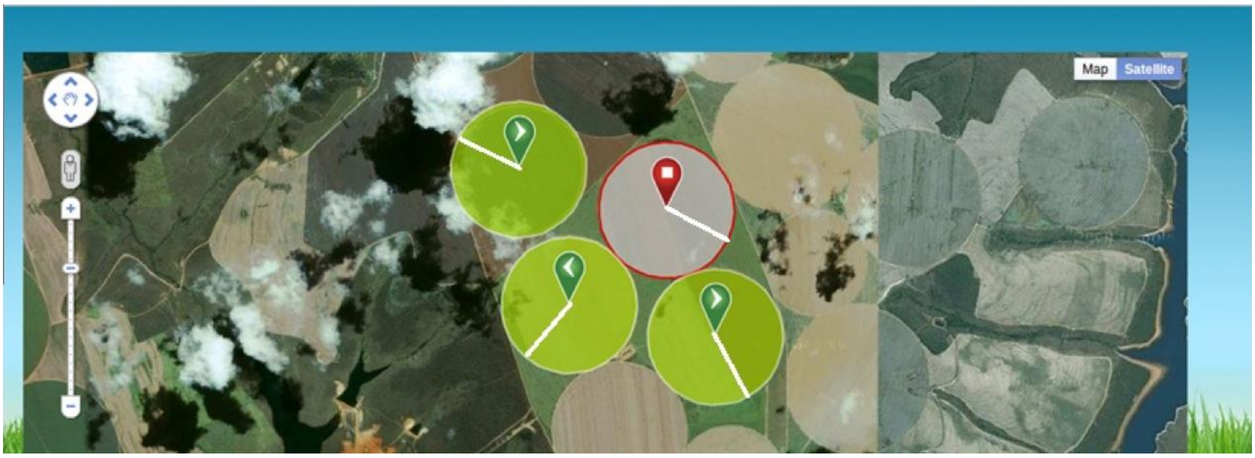


Figure I.3 – Irrigation monitoring system geolocation status example

I.2 Use case for architectural reference model of LPHC devices

LPHC devices are suitable for IoT uses cases in environments with a direct Internet connection and predictable tasks are executed. As one of the most significant IoT use cases that fits in these characteristics, the smart home sector presents high demand for connected objects in different intelligent applications.

A use case for the architectural reference model of LPHC devices can be presented from a smart window system, built from connected devices in an ordinary home. The monitoring device is constructed using a 16-bit microcontroller with wireless fidelity (Wi-Fi) module. The functionalities of the device allow it to monitor indoor air quality and weather conditions (e.g., sensing for rain), and open/close the windows accordingly. Additionally, the windows check the weather prediction and are remotely controlled over the Internet. Furthermore, the windows can send data collected from sensors, such as on/off status, rain sensors, liquefied petroleum gas (LPG) levels present in the air, and internal temperature.

The device's software implements the LPHC architecture presented on this Recommendation, with the following functional entities:

- **sensing/actuating/data capture functional entity**: reads data from the temperature, rain sensor and LPG levels;
- **message handling functional entity**: packs data read from sensors into messages to be sent to the IoT cloud;
- **hardware management functional entity**: implements the functions for interfacing with all sensors and the Wi-Fi radio;
- **connectivity management:** establishes communication between the device and the communication network (Internet);
- **cloud service/application interface**: interacts with the IoT cloud that stores the information sent by the device.

The data collected from the devices are sent to the cloud, which authenticates all the message traffic generated by the LPLC devices in an interaction with the cloud service. The smart window system is depicted in Figure I.4.

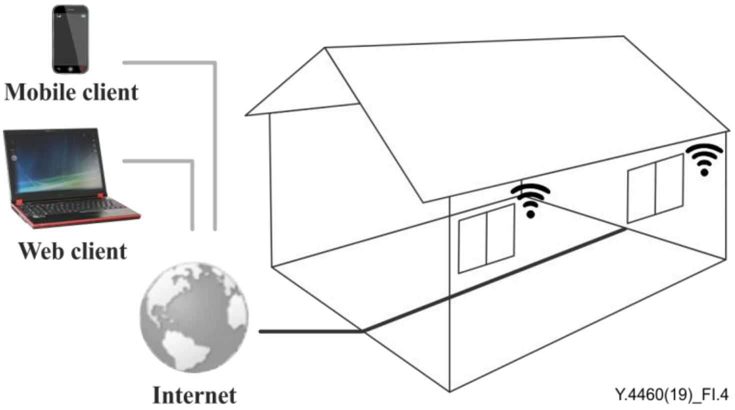
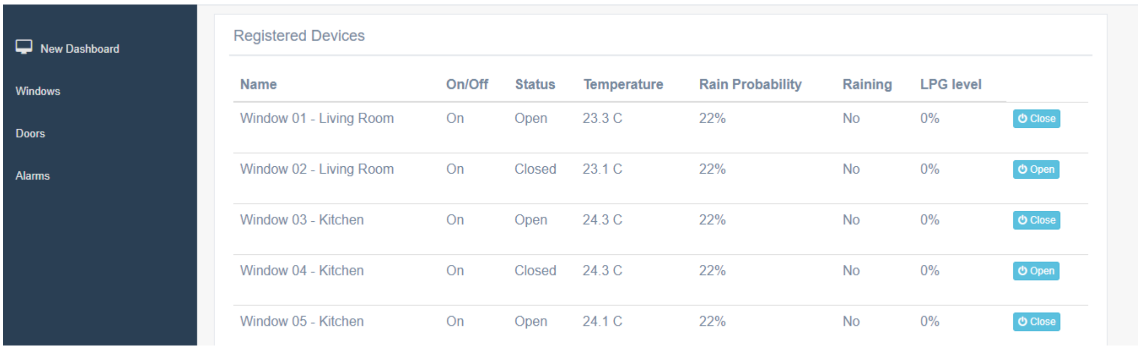


Figure I.4 – Smart window system using LPHC architectural reference model where autonomous windows can be remotely managed and monitored through the Internet

In addition to the data collection process, the smart window system provides advantages, such as being remotely monitored and controlled, which offer comfort and safety for the end-user who no longer has to worry about forgetting if the windows are open. These features are achieved by the IoT web application that is responsible for analyzing data and generating useful information through dashboards. Figure I.5 shows a view of the smart window system dashboard with information from different LPHC devices.



Registered Devices

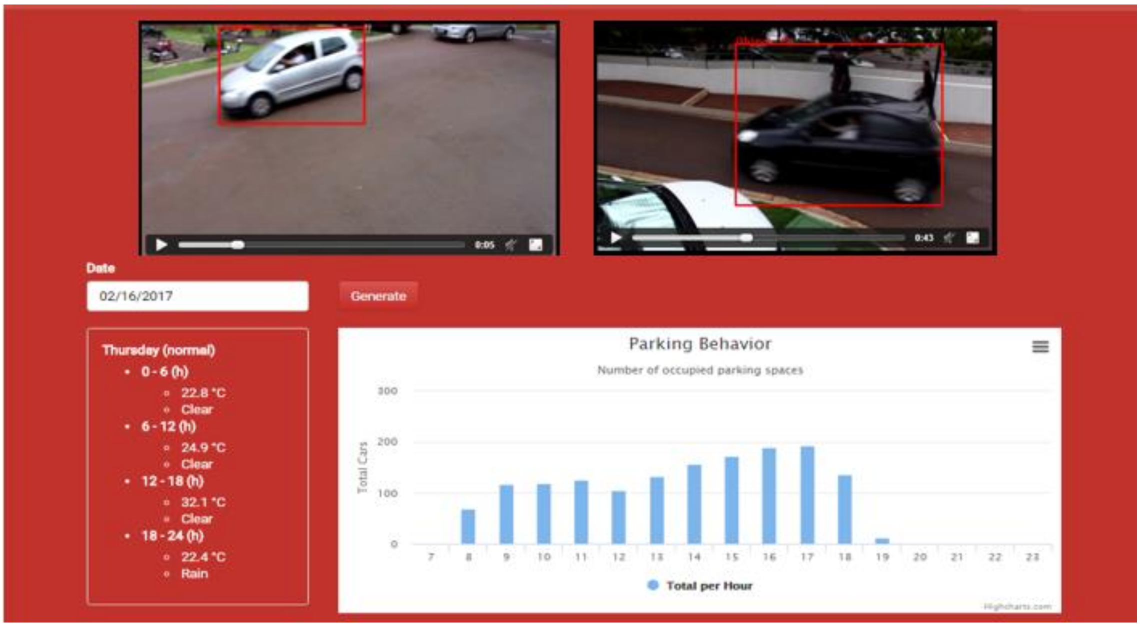
Name	On/Off	Status	Temperature	Rain Probability	Raining	LPG level	
Window 01 - Living Room	On	Open	23.3 C	22%	No	0%	⏻ Close
Window 02 - Living Room	On	Closed	23.1 C	22%	No	0%	⏻ Open
Window 03 - Kitchen	On	Open	24.3 C	22%	No	0%	⏻ Close
Window 04 - Kitchen	On	Closed	24.3 C	22%	No	0%	⏻ Open
Window 05 - Kitchen	On	Open	24.1 C	22%	No	0%	⏻ Close

Figure I.5 – Smart windows system dashboard example

I.3 Use case for architectural reference model of HPHC devices

HPHC devices are suitable for IoT uses cases in environments with available direct internet connection, and where data analysis algorithms are required to be executed as close to the data source as possible, to allow time sensitive actions to be performed quickly. As one of the most significant IoT use cases that fits in these characteristics, the smart building sector presents high demand for connected objects that need the ability to perform analysis and act upon partial system outage scenarios, for example, when the network is not available.

In the context of architectural reference model of HPHC devices, a smart building where the parking lot has limited number of parking spots is presented as a use case. The smart parking lot system consists of two monitoring devices with computer vision capabilities, where cars are detected based on neural networks that can sense cars in multiple form factors and in multiple weather conditions. This allows for the deployment of only two devices, one at the entrance and the other at the exit, to monitor 200 parking lot spaces and provides an example of the HPHC architecture. Figure I.6 shows the smart parking lot monitoring system example.



02/16/2017

Thursday (normal)

- 0 - 6 (h)
  - 22.8 °C
  - Clear
- 6 - 12 (h)
  - 24.9 °C
  - Clear
- 12 - 18 (h)
  - 32.1 °C
  - Clear
- 18 - 24 (h)
  - 22.4 °C
  - Rain

Figure I.6 – Example of a smart parking lot monitoring system

The monitoring devices are constructed using a multi-core processor and 1 Gb RAM. This processor and memory are part of a single-board computer; it also has a small 8-megapixel camera. The application software is hosted on an operating system, and both the neural networks and the picture capturing software modules are written in a high-level programming language.

The device's software implements the HPHC architecture presented on this Recommendation, with the following functional entities:

- **application execution engine functional entity**: services to update the neural networks car recognition models for improved detection are available for external servers that may improve the model based on historical data;
- **device management functional entity**: services and functions that allow camera configuration, providing adaptation to external light changes due to changing light characteristics linked to weather conditions;
- **information sharing functional entity**: provides interaction between the entrance and the exit cameras allowing better vehicle detection. The detection of cars entering and leaving in a short period of time signals a full parking lot, allowing the prediction models to self-correct;
- **data analytics functional entity**: implements the functions for running the vehicle detection neural networks model;
- **data storage functional entity**: stores data including parking lot predictions and images of positives (and in some cases negatives) to allow access from servers that can process the data for improving the detection and prediction models.

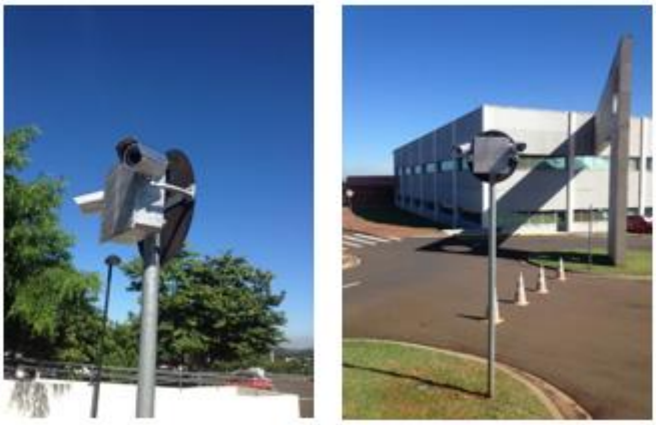


Figure I.7 – Monitoring devices field installation example

The primary function of this device is to detect cars, not motorcycles or service trucks, and keep a counter for the number of cars that entered the building; this allows drivers to avoid time wasted by entering the parking lot when it is full. On the described smart sensing devices, it is also possible to apply recurrent neural network models to predict times of parking lot fullness, providing users the probability of encountering a parking spot based on their estimated time of arrival. Figure I.7 shows the HPHC monitoring devices (cameras) deployed in a real scenario.

Bibliography

[b-ITU-T Y.2012]	Recommendation ITU-T Y.2012 (2010), *Functional requirements and architecture of next generation networks*.
[b-ITU-T Y.2091]	Recommendation ITU-T Y.2091 (2011), *Terms and definitions for next generation networks*.
[b-ITU-T Y.3500]	Recommendation ITU-T Y.3500 (2014), *Information technology – Cloud computing – Overview and vocabulary*.
[b-ITU-T Y.4050]	Recommendation ITU-T Y.4050 (2012), *Terms and definitions for the Internet of things*.
[b-ITU-T Y.4100]	Recommendation ITU-T Y.4100 (2014), *Common requirements of the Internet of things*.
[b-ITU-T Y.4101]	Recommendation ITU-T Y.4101 (2017), *Common requirements and capabilities of a gateway for Internet of things applications*.
[b-ITU-T Y.4111]	Recommendation ITU-T Y.4111 (2016), *Semantics based requirements and framework of the Internet of things*.

SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
Series D	Tariff and accounting principles and international telecommunication/ICT economic and policy issues
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	Transmission systems and media, digital systems and networks
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
Series J	Cable networks and transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Environment and ICTs, climate change, e-waste, energy efficiency; construction, installation and protection of cables and other elements of outside plant
Series M	Telecommunication management, including TMN and network maintenance
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
Series Q	Switching and signalling, and associated measurements and tests
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	Data networks, open system communications and security
Series Y	**Global information infrastructure, Internet protocol aspects, next-generation networks, Internet of Things and smart cities**
Series Z	Languages and general software aspects for telecommunication systems

Printed in Switzerland
Geneva, 2019