Introduction

Lorenzo Vangelista

Dept. of Information Engineering, University of Padova, Italy

Padova, July 1, 2016

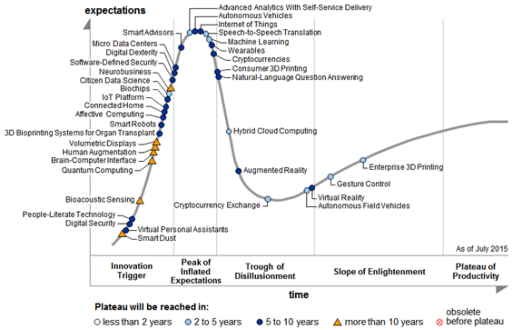
Outline

Legal notice

DISCLAIMER

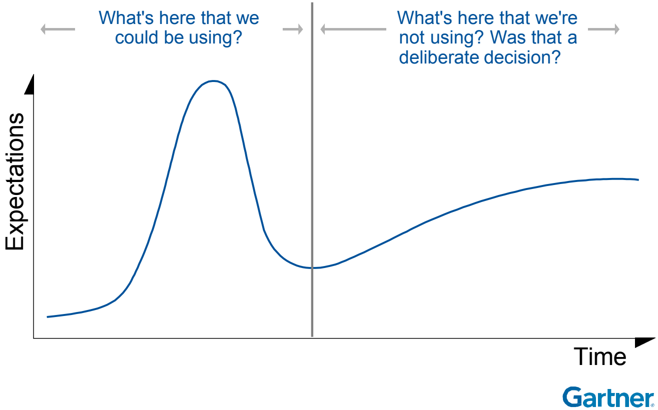
What is the Internet of Things?

- What is the **Internet of Things**?
- What is the **Machine-to-Machine-Communication**?
- Is **Internet of Things** important and worth to study?

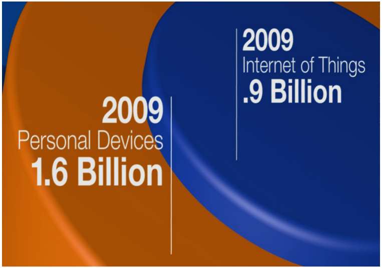


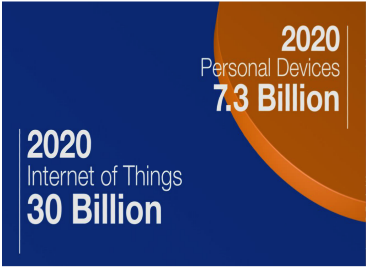
[5]

How to interpret the Gartner's hype cycle



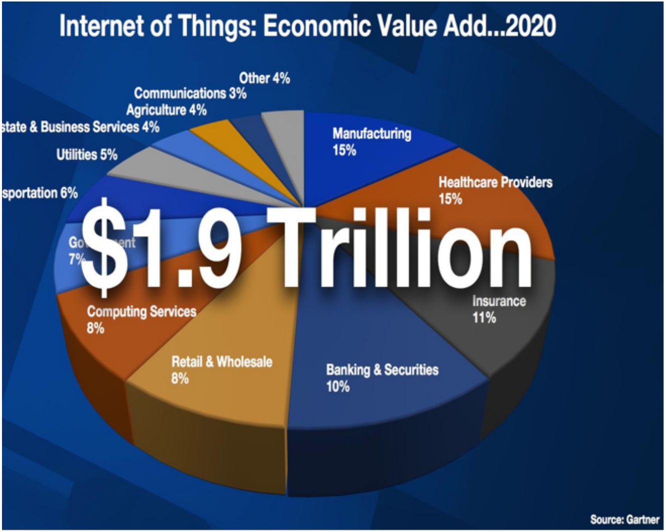
Projections on Internet of Things





[3]

How much is Internet of Things worth?



What is IoT? which "things"?

IoT is pretty young ... turning 18 ...

British entrepreneur Kevin Ashton first coined the term in 1998 while working at Procter & Gamble (then he worked at Auto-ID Labs, currently MIT Auto-ID center) referring to a global network of Radio-frequency identification (RFID) connected objects.

"Things" can be **real** or **virtual** and things are much more complex ... see below ([2])

Internet of Things and Smart Cities

Why putting them together?

- Internet of Things is the backbone of a Smart City
- In a dumb city there could be an Internet of Things
- In a smart city there MUST be an Internet of Things

What is a Smart City?

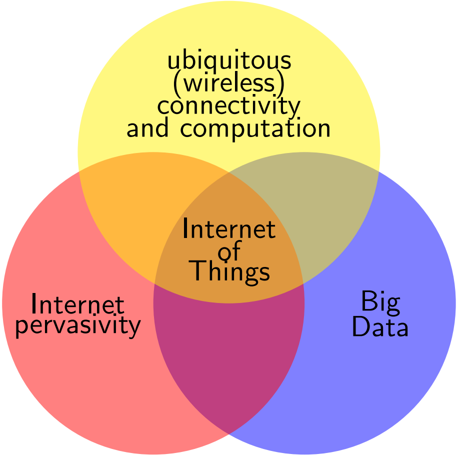
- What is the Smart City?
- How cam we measure how Smart a city is?

Smartness of a city

How smart a city is?

- A city could be smart because the municipality has an ICT system to consult the citizens about any decision to take;
  social impact
- A city could be smart because the citizens have an application to signal the potholes in the streets;
  technical enabler
- A city could be smart because it has an environmental monitoring system and change the traffic permissions according to the pollution level;
  environmental impact
- Overall we need some METRICS to measure how smart a city is.
  these metrics are "standardised" and we will see them in detail

What is IoT?


ubiquitous (wireless) connectivity and computation
Internet of Things
Internet pervasivity
Big Data

A brief history of storage

- First floppy disk 8 inches: 80 Kbyte
- mid-1970: 5 1/4 inches floppy disk: 110 kilobytes to 1.2 megabytes (1982)
- late 1980: 2 1/2 inches floppy disk: 2.4 megabyte
- By 2020 IDC predicts that 40 zettabyte of data will exist worldwide (6 terabytes per person, 1 zettabyte almost equals to 250 billion DVDs)

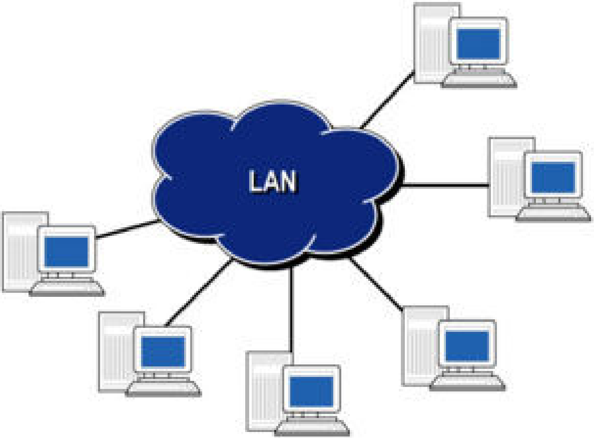
Why this is important for IoT?

- IoT generates a lot of **raw** data;
- **Machine learning** algorithms enable the "*silent machine intelligence*" (http://thesilentintelligence.com/)
- BTW: a huge shortage of people skilled in machine learning and big data (i.e., quite a lot and lucrative jobs)

A brief history of networking – the beginning

Dennis Ritchie with his colleague Ken Thompson, inventors of the C language and UNIX, working on a mainframe.

A brief history of networking – the LAN



Many "islands" not interconnected

A brief history of networking – the Inter-NET

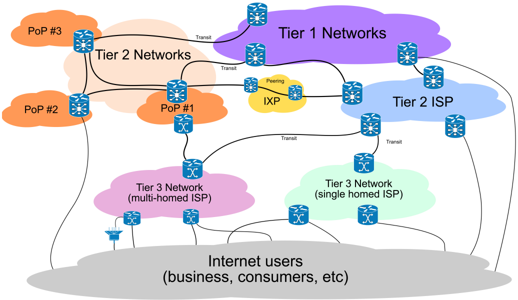


Figure: By User:Ludovic.ferre (Internet Connectivity Distribution&Core.svg) [CC BY-SA 3.0 (http://creativecommons.org/licenses/by-sa/3.0)], via Wikimedia Commons

The islands get interconnected

A brief history of networking – TCP/IP and www.

TCP/IP

Early 1970: the "lingua franca" of internet

World Wide Web

Early 1990: the HTML language is invented and the fist browser (Mosaic) appears

Today

Early 1990: 6 billion Internet users: humans?

Buzz words (but real business behind)

Platforms, REST and SOAP paradigms, Web Services, Cloud Computing, Social, Platforms

A brief history of networking – wireless and mobile

cellular network

1973 Martin Cooper (Motorola): first cellular network call

the current version of the mobile "phone"

2007: the iPhone

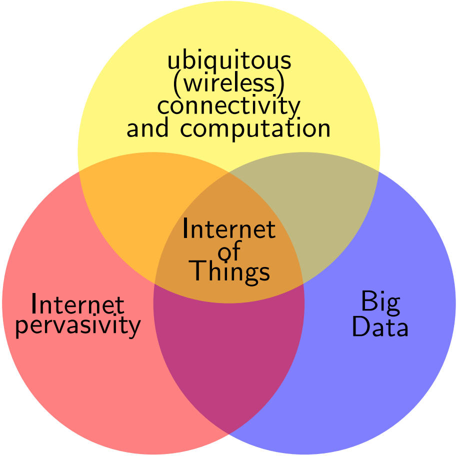
Plethora of wireless "standards"

WiFi, Bluetooth, ZigBee, NFC, RFID etc.

Most of them are found in the "mobile phone", a **real hub for the IoT**; think of wearables, e.g., FitBit

fitbit

Back to the concept of IoT



Back to the concept of IoT

Wireless IoT

It is usually taken for granted that IoT is wireless. Is that correct?

Strictly it is not: think of the domotic applications for example

But why, de facto, IoT means wireless?

Because for a "thing" we want to "**place and play**" which is different from plug and play and needs wireless.

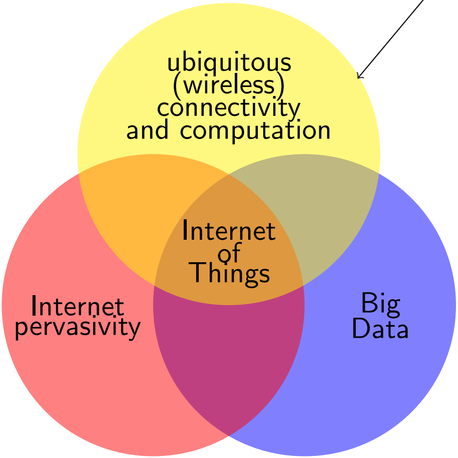
Machine to Machine Communications

IoT is often considered as a synonymous of M2M

we are going to see this is not the case

Back to the concept of IoT

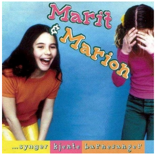
This is the **Machine – to – Machine** domain



M2M is older than IoT

In 2004, Nokia published a white paper called *Machine-to-Machine: Let Your Machine Talk* which pointed out: *It is not only people who use telecommunications and Internet technologies to communicate, but machines around us as well. This is called machine-to-machine (M2M) communication.*

According to a legend, the term *M2M* was borrowed from a then-popular Norwegian girl-pop duo of the same name that fancied by one of the Nokia's executives [6]



CAVEAT

Don't worry too much of acronyms

Unfortunately in this course you are going to encounter a lot of acronyms; don't be worried ... outside of this course you will encounter even more; we will try to keep their number at a minimum bit the IoT is pretty young and there is a lot going on

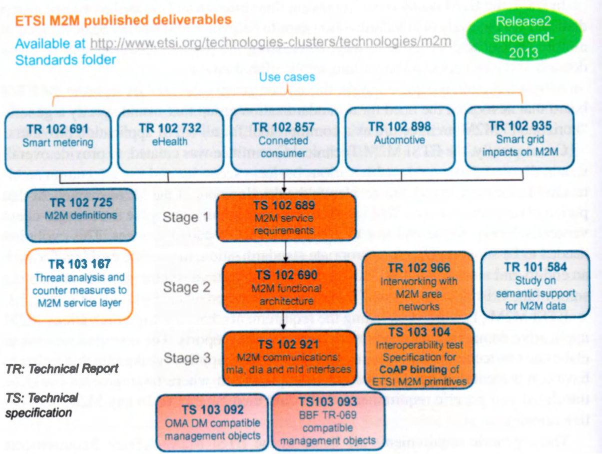
Don't underestimate the standardisation

Once upon a time standardisation $\rightarrow$ boredom; now some time the standardisation *inspire* the academic research.
Furthermore the most important issue for IoT is standardisation: *IoT needs standardisation*

ETSI

- European Telecommunications Standardisation Institute: ETSI was created by CEPT (European Conference of Postal and Telecommunications Administrations) in 1988 and is officially recognized by the European Commission. Based in Sophia Antipolis (France), ETSI is officially responsible for standardisation of Information and Communication Technologies (ICT) within Europe.
- **GSM** has been standardised by ETSI (among many other systems).
- website: `http://www.etsi.org/`

ETSI deliverables



ETSI M2M

- ETSI standardised a Machine-2-Machine framework
- It has been the first and it is quite cumbersome ...
- We will touch it briefly to get a feeling of what such a specification means and because it was the first!
- Then we will move to next era of standardisation: from "government" backed institution to "free" consortia: the successor of ETSI M2M is OneM2M `http://www.onem2m.org/`

A general architecture of M2M

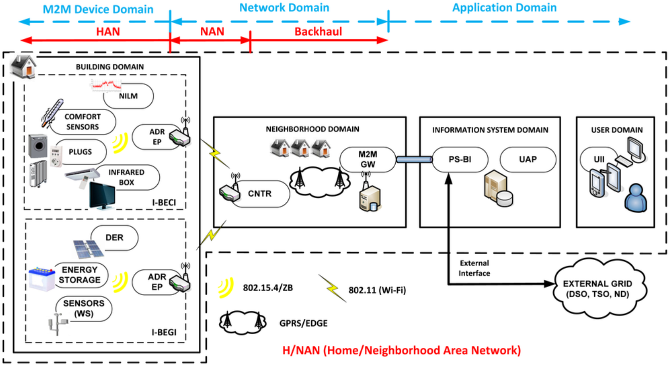


Figure: from [8]

The framework of ETSI M2M

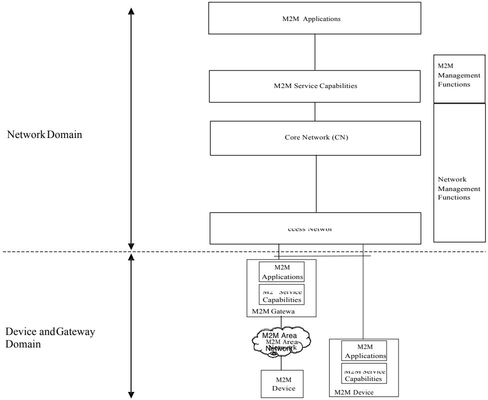


Figure: from [1]

ETSI M2M: some definitions

M2M Service Capabilities:

- Provide M2M **functions** that are to be shared by different Applications.
- Expose functions through a set of **open interfaces**.
- Use Core Network functionalities.
- Simplify and optimize application development and deployment through **hiding of network specificities**.
- Examples include: Data Storage and Aggregation, Unicast and Multicast message delivery, etc.

M2M applications: Applications that run the service logic and use M2M Service Capabilities accessible via an open interface.

ETSI M2M: some more definitions

Network Management Functions: consists of all the functions required to manage the Access and Core networks: these include Provisioning, Supervision, Fault Management, etc.
M2M Management Functions: consists of all the functions required to manage M2M Service Capabilities in the Network Domain. The management of the M2M Devices and Gateways uses a specific M2M Service Capability.

ETSI M2M: Management Functions

- The set of M2M Management Functions include a function for M2M Service Bootstrap . This function is called **MSBF (M2M Service Bootstrap Function)** and is realized within an appropriate server. The role of MSBF is to facilitate the bootstrapping of permanent M2M service layer security credentials in the M2M Device (or M2M Gateway) and the M2M Service Capabilities in the Network Domain.
- Permanent security credentials that are bootstrapped using MSBF (such as the M2M Root Key) are stored in a safe location, which is called **M2M Authentication Server (MAS)**. Such a server can be a AAA server. MSBF can be included within MAS, or may communicate the bootstrapped security credentials to MAS, through an appropriate interface (e.g. Diameter for the case where MAS is a AAA server). The corresponding permanent security credentials established in the D/G (Device / Gateway) M2M Node during the bootstrap are stored in a Secured Environment Domain of the D/G M2M Node.

ETSI M2M: Interfaces

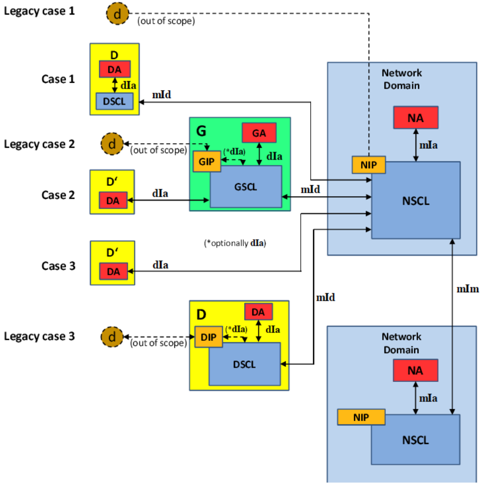


Figure: from [1]

ETSI M2M: Interfaces brief explanation

- **Gateway (G)**: shall provide M2M Service Capabilities (GSCL) that communicates to the NSCL using the **mId** reference point and to DA or GA using the **dIa** reference point.
- **Device (D)**: shall provide M2M Service Capability (DSCL) that communicates to an NSCL using the mId reference point and to DA using the **dIa** reference point.

Additionally there is a non-ETSI M2M compliant device ('d') that connects to SCL using the xIP Capability (NIP, GIP, DIP). d devices do not use ETSI M2M defined reference points, however:

- GIP may either be an internal capability of GSCL or an application communicating via reference point dIa with GSCL.
- DIP may either be an internal capability of DSCL or an application communicating via reference point dIa with DSCL.

Interworking proxy capabilities (xIP) in NSCL (NIP), M2M Gateway (GIP), and or M2M Device (DIP) are optional deployed when needed required by operator policies. The NIP Capability provides the interworking between non ETSI compliant devices or gateways and the NSCL. The GIP Capability provides the following functionalities:

- Provide interworking between non ETSI compliant devices the GSCL.
- Provides interworking with one or more M2M Area Networks and the GSCL.
- GIP may either be an internal capability of GSCL or an application communicating via reference point dIa with GSCL.

The DIP Capability provides the following functionalities:

- Provide interworking between non ETSI compliant devices the DSCL.
- Provides interworking with one or more M2M Area Networks and the DSCL.

DIP may either be an internal capability of DSCL or an application communicating via reference point dIa with DSCL.

ETSI M2M: dIa example

A question: what actually do we do trough these reference points? Let's take dIa for example, the functions supported are

- Registration of D/GA to GSCL.
- Registration of DA to DSCL.
- Registration of DA to NSCL.
- *Request to Read/Write, subject to proper authorization, information in the NSCL, GSCL or DSCL.*
- *Subscription and notification to specific events.*
- Request the creation, deletion and listing of group(s).

Provisioning

What is *provisioning*?

In telecommunications, provisioning involves the process of preparing and equipping a network to allow it to provide (new) services to its users;
Let's think of connecting your ADSL modem and letting it negotiate "automatically" the parameters with the netwrok
It is of *utmost importance in real telecommunication networks,* often neglected in academia.

ETSI M2M Provisioning

Provisioning is ETSI M2M is done via two "external" standards:

- OMA-TS-DM-Protocol-V1-3: "OMA Device Management Protocol", Version 1.3.
- Broadband Forum TR-069: "CPE WAN Management Protocol" Version 1.3, Issue: 1 Amendment 4. Issue Date: July 2011.

The Open Mobile Alliance (OMA) (`http://openmobilealliance.org/`) was formed in June 2002. The Open Mobile Alliance will work towards stimulating the fast and wide adoption of a variety of new, enhanced mobile information, communication and entertainment **services**.
The Broadband Forum (`https://www.broadband-forum.org/`) is a non-profit industry consortium dedicated to developing broadband network specifications. Service provider members are primarily wire-line service providers (non-mobile) telephone companies.

A company leader in TR069 implementation is Affinegy (`http://affinegy.com/`).

But what actually are the ETSI M2M Service Capabilities and how can we develop applications?

What we have discussed so far seems to be quite generic and we don't have a grasp on how to "use" the ETSI M2M.

First of all we need to have in place the servers and the software in the devices and gateways and let's suppose to have that. For example we can use an open implementation as the Eclipse project OM2M (`http://www.eclipse.org/om2m/`) or buy from a top company like Interdigital (`http://www.interdigital.com/iot/`).

Then we may want to develop applications based on ETSI M2M. How can we do that?

We need a small detour to explain how to proceed.

The REST paradigm: from wikipedia

In computing, representational state transfer (REST) is the software architectural style of the World Wide Web. REST's coordinated set of constraints, applied to the design of components in a distributed hypermedia system, can lead to a higher-performing and more maintainable software architecture.
To the extent that systems conform to the constraints of REST they can be called RESTful. RESTful systems typically, but not always, communicate over Hypertext Transfer Protocol (HTTP) with the same HTTP **verbs** (GET, POST, PUT, DELETE, etc.) that web browsers use to retrieve web pages and to send data to remote servers. REST systems interface with external systems as web **resources** identified by **Uniform Resource Identifiers (URIs)**, for example `/people/tom`, which can be operated upon using standard verbs such as `DELETE /people/tom`.
Note 1: that the REST architecture is the counterpart of the SOAP (Simple Object Access Protocol) architecture

Note 2: in the rest architecture everything is about transactions, there is no history kept at each side about past transactions

The REST paradigm in ETSI M2M

The procedures described in ETSI M2M specifications adopt a RESTful architecture style.
This style governs how M2M Applications (DA, GA, NA) and/or M2M SCL are exchanging information with each other.
A RESTful architecture is about the transfer of representations of uniquely addressable resources (**URI**).
ETSI M2M standardized the resource structure that resides on a SCL.
In a very simplistic view, imagine that certain **resources** are buckets that can hold some application specific data. These buckets - as far as the scope of an M2M service layer is concerned - reside in the respective SCL.

CRUD verbs

When handling resources in a RESTful architecture, there are four basic methods - so called "verbs" - that could be applied to resources:

- CREATE: Create child resources.
- RETRIEVE: Read the content of the resource.
- UPDATE: Write the content of the resource.
- DELETE: Delete the resource.

Additional verbs

The additional verbs introduced are:

- NOTIFY: used to indicate the operation for reporting a notification about a change of a resource as a consequence of a subscription. This verb would either map to a response of a RETRIEVE method in case that the long polling mechanism is used, or to an UPDATE method in case that the asynchronous mechanism is used.
- EXECUTE: for executing a management command/task which is represented by a resource. This verb corresponds to an UPDATE method without any payload data.

ETSI M2M resource structure

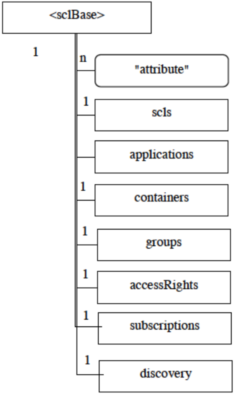


Figure: from [1]

Please note the **discovery**

ETSI M2M scls resource structure

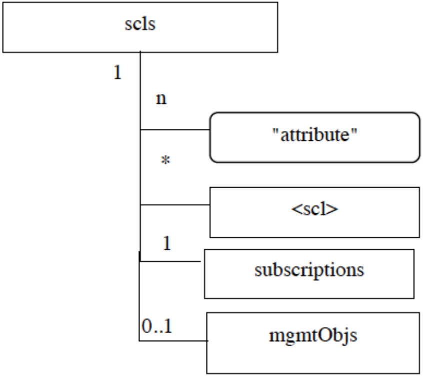


Figure: from [1]

Please note the **mgmtObjs**

ETSI M2M mgmtObjs resource structure

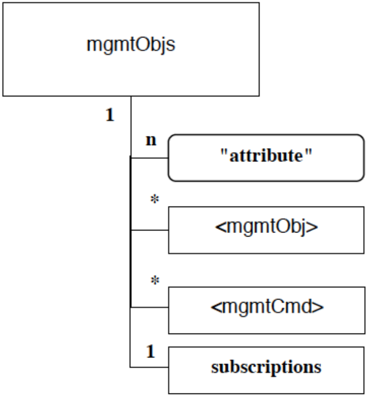


Figure: from [1]

Please note the **mgmtCmd**

ETSI M2M mgmt commands (TR-069)

BBF TR-069 RPC Methods	Meaning
FactoryReset	Used by the ACS [10] to reset the CPE [10] to its factory default state.
Reboot	Used by the ACS to cause the CPE to reboot.
Upload	Used by the ACS to cause the CPE to upload a specified file to the designated location. It can be cancelled by BBF TR-069 CancelTransfer RPC.
Download	Used by the ACS to cause the CPE to download a specified file from the designated location. It can be cancelled by BBF TR-069 CancelTransfer RPC.
ScheduleDownload	Used by the ACS to cause the CPE to download a specified file from the designated location and apply it within either one or two specified time Windows. It can be cancelled by BBF TR-069 CancelTransfer RPC.
ScheduleInform	Used by the ACS to request the CPE to schedule a one-time Inform method call sometime in the future.
ChangeDUState	Used by an ACS to trigger the explicit state transitions of Install, Update, and Uninstall for a Deployment Unit (DU), i.e. installing a new DU, updating an existing DU, or uninstalling an existing DU.

Figure: from [1]

ETSI M2M enable applications with clear interfaces and abstracting from the underlying networks

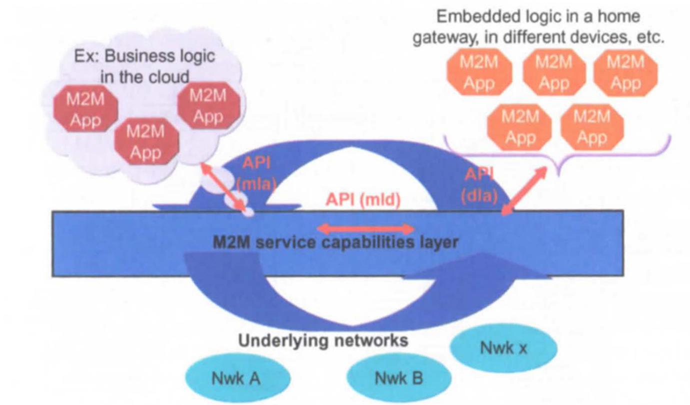


Figure: from [4]

Note that we have **not said a single word on the underlying networks**

OneM2M

OneM2M

The successor of ETSI M2M for worldwide compatibility.

From Europe to Worldwide

OneM2M specifications phase 1

Reference	Version	Title	Date	ARIB	ATIS	CCSA	ETSI	TIA	TTA	TTC
TS 0001	1.6.1	Functional Architecture	01/2015		ATIS.oneM2M.TS0001V161-2015		TS 118 101 V1.0.0		TTAT.MM-TS.0001	TS-M2M-0001v1.6.1
TS 0002	1.0.1	Requirements	01/2015		ATIS.oneM2M.TS0002V101-2015		TS 118 102 V1.0.0		TTAT.MM-TS.0002	TS-M2M-0002v1.0.1
TS 0003	1.0.1	Security Solutions	01/2015		ATIS.oneM2M.TS0003V101-2015		TS 118 103 V1.0.0		TTAT.MM-TS.0003	TS-M2M-0003v1.0.1
TS 0004	1.0.1	Service Layer Core Protocol Specification	01/2015		ATIS.oneM2M.TS0004V101-2015		TS 118 104 V1.0.0		TTAT.MM-TS.0004	TS-M2M-0004v1.0.1
TS 0005	1.0.1	Management Enablement (OMA)	01/2015		ATIS.oneM2M.TS0005V101-2015		TS 118 105 V1.0.0		TTAT.MM-TS.0005	TS-M2M-0005v1.0.1
TS 0006	1.0.1	Management Enablement (BBF)	01/2015		ATIS.oneM2M.TS0006V101-2015		TS 118 106 V1.0.0		TTAT.MM-TS.0006	TS-M2M-0006v1.0.1
TS 0008	1.0.1	CoAP Protocol Binding	01/2015		ATIS.oneM2M.TS0008V101-2015		TS 118 108 V1.0.0		TTAT.MM-TS.0008	TS-M2M-0008v1.0.1
TS 0009	1.0.1	HTTP Protocol Binding	01/2015		ATIS.oneM2M.TS0009V101-2015		TS 118 109 V1.0.0		TTAT.MM-TS.0009	TS-M2M-0009v1.0.1
TS 0010	1.0.1	MQTT Protocol Binding	01/2015		ATIS.oneM2M.TS0010V101-2015		TS 118 110 V1.0.0		TTAT.MM-TS.0010	TS-M2M-0010v1.0.1
TS 0011	1.2.1	Common Terminology	01/2015		ATIS.oneM2M.TS0011V121-2015		TS 118 111 V1.0.0		TTAT.MM-TS.0011	TS-M2M-0011v1.2.1

Looks like too abstract? Are you in love with Open Source?

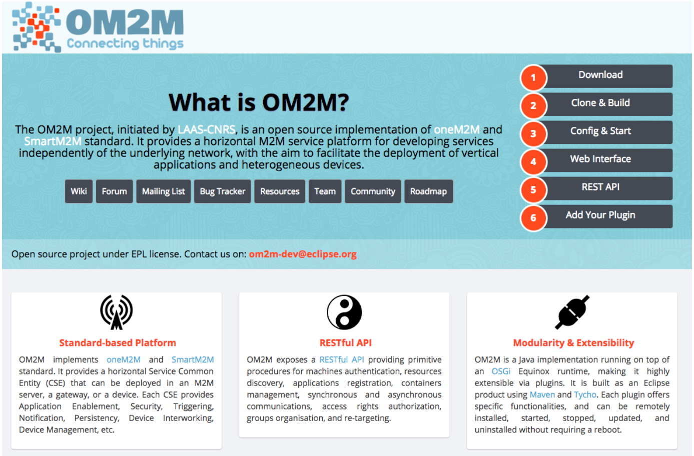


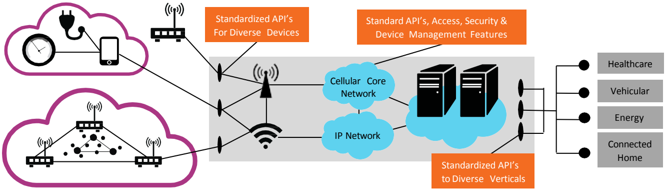
Figure: from `http://www.eclipse.org/om2m/`

OM2M roadmap

Version 0.8 is supporting ETSI M2M
Version 1.0 will support OneM2M and non longer ETSI M2M

A commercial implementation fo OneM2M

ONEMPOWER an InterDigital Solution is an End-to-End IoT Platform Solution



Devices	Gateway	Cloud Server	Apps
Cellular, WLAN, WPAN (Zigbee, 6LoWPAN, Bluetooth), Wireline	Enables cellular & non-cellular M2M devices to communicate through operator networks. Provides localized Service Capabilities to offload network	Service Provider's M2M Service Platform, offering Service Capabilities to Diverse M2M Verticals (Device/ Data Access, Device Management, Security, Billing Service Discovery, etc.)	Build apps that support a rich mashup of devices and data across multiple verticals

Standardisation organizations

Standardisation organizations

One of the major problems of IoT is the fragmentation of standards. We try to provide an overview

International Telecommunication Union (ITU)

ITU (`http://www.itu.int/`) is the United Nations specialized agency for information and communication technologies - ICTs. ITU allocate global radio spectrum and satellite orbits, develop the technical standards that ensure networks and technologies seamlessly interconnect, and strive to improve access to ICTs to underserved communities worldwide.

ITU is committed to connecting all the world's people – wherever they live and whatever their means. ITU protects and supports everyone's fundamental right to communicate.

ITU has a long history, it is probably the first SDO, see `http://www.itu.int/en/history/Pages/Home.aspx`

ITU sectors

ITU is made up of 3 sectors

- **ITU-T** Radiocommunication
- **ITU-T** Telecommunication Standards
- **ITU-D** Telecommunications Development

ITU on IoT and Smart Cities

ITU is actively working on IoT and Smart Cities. Please see the portal `http://www.itu.int/en/ITU-T/ssc/resources/Pages/default.aspx`.

Internet Engineering Task Force (IETF)

The mission of the IETF is to make the Internet work better by producing high quality, relevant technical documents that influence the way people design, use, and manage the Internet. The IETF is a loosely self-organized group of people who contribute to the engineering and evolution of Internet technologies. It is the principal body engaged in the development of new Internet standard specifications. The IETF is unusual in that it exists as a collection of happenings, but is not a corporation and has no board of directors, no members, and no dues.

To learn more about IETF please visit the following page
Newcomers page

I E T F®

IETF *standardization* process

The basic steps for getting something published as an IETF standard are as follows:

1. Publish the document as an Internet-Draft.
2. Receive comments on the draft.
3. Edit your draft based on the comments.
4. Repeat steps 1 through 3 a few times.
5. Ask an Area Director (a person appointed by the Internet Engineering Steering Group) to take the draft to the IESG (if it's an individual submission). If the draft is an official Working Group product, the WG chair asks the AD to take it to the IESG.
6. If the Area Director accepts the submission, they will do their own initial review, and maybe ask for updates before they move it forwards.
7. Get reviews from the wider IETF membership. In particular, some of the Areas in the IETF have formed review teams to look over drafts that are ready to go to the IESG. Two of the more active review teams are from the Security Directorate ("SecDir") and the General Area Review Team (Gen-Art).
8. Discuss concerns with the IESG members. Their concerns might be resolved with a simple answer, or they might require additions or changes to the document.
9. Wait for the document to be published by the RFC Editor.

Some IETF IoT *standards*

- CoAP (Constrained Application Protocol) and DTLS (CoRE group)
- RPL (ROLL group)
- 6TiSCH group (IP over TSCH)

International Organization for Standardization - ISO

ISO is an independent, non-governmental international organization with a membership of 162 national standards bodies. Through its members, it brings together experts to share knowledge and develop voluntary, consensus-based, market relevant International Standards that support innovation and provide solutions to global challenges.
UNI (Ente Nazionale Italiano di Unificazione) is the Italian representative.
The ISO/IEC Joint Technical Committee 1 (ISO/IEC JTC1) is developing standards on ICT. The Italian representative in IEC is the Comitato elettrotecnico italiano (CEI).
Is developed significant standards like MPEG. We are interested in Wireless MBus (CEI UNI EN 13757) for smart metering

ETSI


Security
Home & Office
Better Living with ICT
Public Safety
Content Delivery
Interoperability
ETSI
Networks
Connecting Things
Transportation
Wireless Systems

ETSI regulations

Final draft **ETSI EN 300 220-1** V2.4.1 (2012-01)

Electromagnetic compatibility
and Radio spectrum Matters (ERM);
Short Range Devices (SRD);
Radio equipment to be used in the 25 MHz to 1 000 MHz
frequency range with power levels ranging up to 500 mW;
Part 1: Technical characteristics and test methods

ETSI *sister* societies worldwide

- **TTC** The Telecommunication Technology Committee, is a standardization organization established in 1985 and authorized by Japan's Ministry of Internal Affairs and Communications to conduct research and to develop and promote standards for telecommunications in Japan.
- **TSDSI** Telecommunications Standards Development Society, India
- **TIA** The Telecommunications Industry Association (USA)
- **CCSA** China Communications Standards Association
- **ARIB** Association of Radio Industries and Business (Japan)
- **ATIS** Alliance for Telecommunications Industry Solutions (USA)

3GPP

The 3^{rd} Generation Partnership Project (3GPP) unites 7 telecommunications standard development organizations (ARIB, ATIS, CCSA, ETSI, TSDSI, TTA, Telecommunications Technology Association (Korea), TTC), known as "Organizational Partners" and provides their members with a stable environment to produce the Reports and Specifications that define 3GPP technologies.

The original scope of 3GPP (1998) was to produce Technical Specifications and Technical Reports for a 3G Mobile System based on evolved GSM core networks and the radio access technologies that they support (i.e., Universal Terrestrial Radio Access (UTRA) both Frequency Division Duplex (FDD) and Time Division Duplex (TDD) modes).

The scope was subsequently amended to include the maintenance and development of the Global System for Mobile communication (GSM) Technical Specifications and Technical Reports including evolved radio access technologies (e.g. General Packet Radio Service (GPRS) and Enhanced Data rates for GSM Evolution (EDGE)).

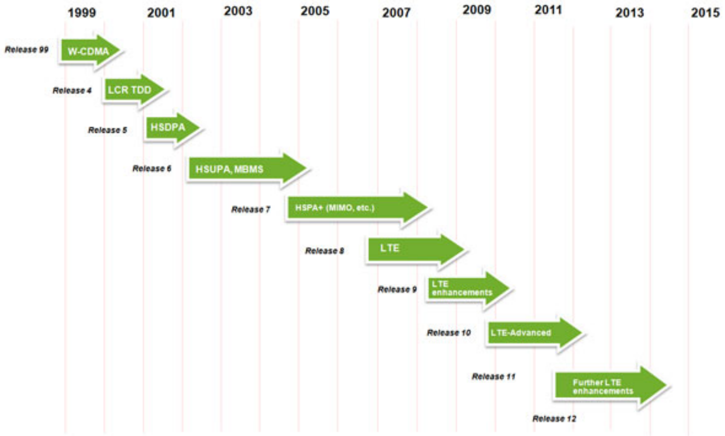
3GPP Technical Specification Groups

3GPP specifications and studies are contribution-driven, by member companies, in Working Groups and at the Technical Specification Group level.

The Four Technical Specification Groups (TSG) in 3GPP are;

- Radio Access Networks (RAN),
- Service & Systems Aspects (SA),
- Core Network & Terminals (CT) and
- GSM EDGE Radio Access Networks (GERAN).

3GPP releases



3GPP and cellular IoT

Release 13 is supposed to be frozen last March 11^{th}. Release 14 is expected to be frozen on June 2017. 3GPP is working on enabling IoT in LTE specifications: see, for example, 3GPP TR 45.820 *Cellular system support for ultra-low complexity and low throughput Internet of Things (CIoT)* at
`http://www.3gpp.org/DynaReport/45820.htm`
Candidates technologies are:

1. EC-GSM
2. N-GSM
3. NB-M2M
4. NB-OFDMA
5. NB-IoT
6. Cooperative UNB
7. NB-LTE

NB-IoT

NB-IoT has been selected and "standardized" on June, 2016. Specification not yet available. Huawei is the market leader.

3GPP2

The Third Generation Partnership Project 2 (3GPP2) is:

- a collaborative third generation (3G) telecommunications specifications-setting project
- comprising North American and Asian interests developing global specifications for ANSI/TIA/EIA-41 Cellular Radio telecommunication Intersystem Operations network evolution to 3G
- and global specifications for the radio transmission technologies (RTTs) supported by ANSI/TIA/EIA-41.

3GPP2 partners

- **ARIB** Association of Radio Industries and Business (Japan)
- **CCSA** China Communications Standards Association (China)
- **TIA** The Telecommunications Industry Association (USA)
- **TTA** Telecommunications Technology Association (Korea)
- **TTC** The Telecommunication Technology Committee, (Japan)

IEEE 802 Working Groups and Study Groups

- 802.1 Higher Layer LAN Protocols Working Group
- *802.3 Ethernet Working Group*
- **802.11 Wireless LAN Working Group**: general interest and special interest in 802.11ah
- **802.15 Wireless Personal Area Network (WPAN) Working Group**: general interest and special interest in 802.15.4
- 802.16 Broadband Wireless Access Working Group
- 802.18 Radio Regulatory TAG
- 802.19 Wireless Coexistence Working Group
- 802.21 Media Independent Handover Services Working Group
- 802.22 Wireless Regional Area Networks
- 802.24 Vertical Applications TAG

IEEE ® IEEE 802

Thread

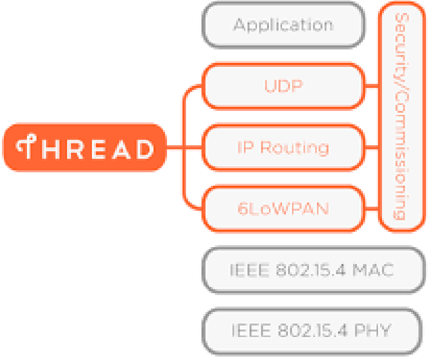


Figure: http://www.threadgroup.org/

ZigBee Alliance

Figure: http://www.zigbee.org

	ZigBee RF4CE		ZigBee PRO						ZigBee IP
Application Standard	ZigBee Remote Control	ZigBee Input Device	ZigBee Building Automation	ZigBee Health Care	ZigBee Home Automation	ZigBee Retail Services	ZigBee Smart Energy 1.x	ZigBee Telecom Services	ZigBee Smart Energy 2.0
Network	ZigBee RF4CE		ZigBee PRO						ZigBee IP
MAC	IEEE 802.15.4 – MAC								IEEE 802.15.4 - MAC
PHY	IEEE 802.15.4 Sub-GHz (specified per region)		IEEE 802.15.4 – 2.4 GHz (worldwide)						IEEE 802.15.4 2006 - 2.4GHz or other

Bluetooth

Bluetooth is a Special Interest Group maintaining the Bluetooth specifications `https://www.bluetooth.com`

Please note that the Physical layer and MAC are part of the IEEE 802.15.1 specifications

AllSeen Alliance

The AllSeen Alliance is a non-profit organization dedicated to making it easy for devices, appliances and apps to connect to the Internet of Things.

Figure: `https://allseenalliance.org`

DIPARTIMENTO
DI INGEGNERIA
DELL'INFORMAZIONE

Open Connectivity Foundation (former Open Internet Consortium)

The Open Connectivity Foundation (formerly Open Interconnect Consortium) is an industry group whose stated mission is to develop standards and certification for devices involved in the Internet of Things (IoT) based around CoAP.

OPEN
CONNECTIVITY
FOUNDATION™

Figure: `http://openconnectivity.org/`

Open Mobile Alliance

The Open Mobile Alliance is an industry consortium which proposed a standard - LWM2M – Light Weight Machine-to-Machine - based around CoAP.

Figure: http://openmobilealliance.org/

GSM Alliance

The GSMA represents the interests of mobile operators worldwide, uniting nearly 800 operators with more than 250 companies in the broader mobile ecosystem, including handset and device makers, software companies, equipment providers and internet companies, as well as organisations in adjacent industry sectors.
Recently GSMA proposed the 'Mobile IoT Initiative', designed to accelerate the commercial availability of Low Power Wide Area (LPWA) solutions in *licensed spectrum*.

Figure: http://www.gsma.com/

DIPARTIMENTO
DI INGEGNERIA
DELL'INFORMAZIONE

Industrial Internet Consortium

IIC consortium aims to bring IoT in industrial environments.

Figure: http://www.iiconsortium.org/

Low Power Wide Area Networks in unlicensed spectrum

Lo-Ra	`https://www.lora-alliance.org/`
SigFox	`http://www.sigfox.com/`
Weigthless	`http://www.weightless.org/`
Ingenu	`http://www.ingenu.com/`

Mesh networking for IoT

The first wave of IoT networks are mesh networks ...

A lot of study and implementation work has been done at the beginning of the IoT based on a MESH architecture. Nowadays ...

After 802.11, probably the most popular standard

IEEE 802.15.4 defines **low-rate wireless personal area network (LR-WPAN)**

Current IEEE 802.15.4 standard

You can get it at `http://standards.ieee.org/about/get/802/802.15.html`
The *baseline* is the 2011 version. Then there are quite few *amendments*

1. IEEE 802.15.4e-2012 IEEE Standard for Local and metropolitan area networks–Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs) Amendment 1: MAC sublayer
2. IEEE 802.15.4f-2012 IEEE Standard for Local and metropolitan area networks– Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs) Amendment 2: Active Radio Frequency Identification (RFID) System Physical Layer (PHY)
3. IEEE 802.15.4g-2012 IEEE Standard for Local and metropolitan area networks–Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs) Amendment 3: Physical Layer (PHY) Specifications for Low-Data-Rate, Wireless, Smart Metering Utility Networks
4. IEEE 802.15.4j-2013 IEEE Standard for Local and metropolitan area networks - Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs) Amendment 4: Alternative Physical Layer Extension to Support Medical Body Area Network (MBAN) Services Operating in the 2360 MHz – 2400 MHz Band
5. IEEE 802.15.4k-2013 IEEE Standard for Local and metropolitan area networks— Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs)—Amendment 5: Physical Layer Specifications for Low Energy, Critical Infrastructure Monitoring Networks.
6. IEEE 802.15.4m-2014 IEEE Standard for Local and metropolitan area networks - Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs) - Amendment 6: TV White Space Between 54 MHz and 862 MHz Physical Layer
7. IEEE 802.15.4p-2014 IEEE Standard for Local and metropolitan area networks - Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs) - Amendment 7: Physical Layer for Rail Communications and Control (RCC)

Latest release IEEE 802.15.4 standard

It is IEEE 802.15.4 - 2015, frozen last October, going trough the approval process.

It incorporates and harmonize all of the amendments.

The following presentation is based on this release. No books yet available.

Summary of the latest release of IEEE 802.15.4

1. General description
   1. Introduction
   2. Specific applications
   3. Network topologies
   4. Architecture
   5. Functional overview
   6. Concept of primitives
2. MAC functional description
3. MAC frame formats
4. MAC services
5. Security
6. Physical Layers

Introduction

An LR-WPAN is a simple, **low-cost** communication network that allows wireless connectivity in applications with limited power and relaxed throughput requirements. The main objectives of an LR-WPAN are ease of installation, reliable data transfer, extremely low cost, and a reasonable battery life, while maintaining a simple and flexible protocol.

This standard defines multiple PHYs operating in a variety of frequency bands.

Two different device types can participate in an IEEE 802.15.4 network: a full-function device (FFD) and a reduced-function device (RFD). An FFD is a device that is capable of serving as a personal area network (PAN) coordinator . An RFD is a device that is not capable of serving as a PAN coordinator . An RFD is intended for applications that are extremely simple, such as a light switch or a passive infrared sensor; it does not have the need to send large amounts of data and only associates with a single FFD at a time. Consequently, the RFD can be implemented using minimal resources and memory capacity.

Specific applications

Although this standard is intended address many diverse application spaces, some application spaces have unique requirements which required specific elements added to this standard

- Smart metering utility network (SUN)
- Rail Communications and Control (RCC)
- Television White Space (TVWS)
- (active) Radio Frequency Identification (RFID)
- Low Energy Critical Infrastructure Monitoring (LECIM)
- Medical body area network (MBAN) services

Components of the IEEE 802.15.4 WPAN

A system conforming to this standard consists of several components.
The most basic is the device. A device has a single radio interface that implements an IEEE Std 802.15.4 MAC and PHY.
Two or more devices communicating on the same physical channel constitute a WPAN. A WPAN includes at least one FFD, operating as the PAN coordinator.

IEEE 802.15.4 WPAN topology

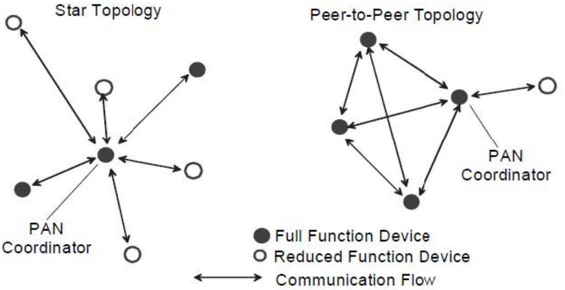


Figure: from [7]

IEEE 802.15.4 WPAN topology: important remarks

All devices operating on a network of either topology have unique addresses, referred to as extended addresses. In addition, a device can be assigned a short address during the association process. A device will use either the extended address or the short address for communication within the PAN. The transmission of the extended and short address fields is optional for RFD-TX devices.
A peer-to-peer network allows multiple hops to route messages from any device to any other device on the network.
Such functions can be added at the higher layer, but they are not part of this standard.
Each independent PAN selects a unique identifier. This PAN identifier (ID) allows communication between devices within a network using short addresses and enables transmissions between devices across independent networks.
The mechanism by which identifiers are chosen is out of scope.
The network formation is performed by the higher layer, which is not part of this standard.

IEEE 802.15.4 WPAN topology: extending the coverage

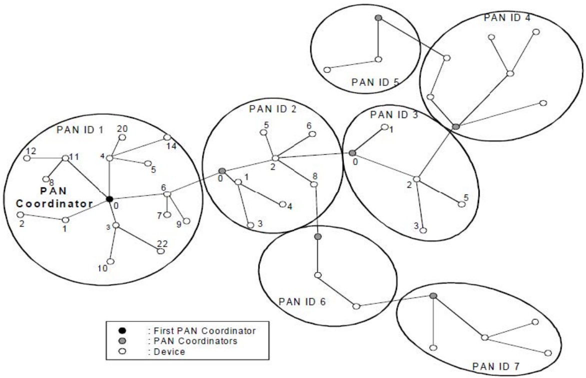


Figure: from [7]

IEEE 802.15.4 device architecture

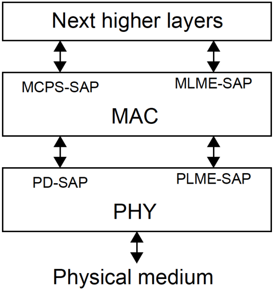


Figure: from [7]

MCPS-SAP: MAC common part sublayer service access point
MLME-SAP: MAC sublayer management entity service access point
PD-SAP: physical layer data service access point
PLME-SAP physical layer management entity service access point

IEEE 802.15.4 MAC sublayer services

The MAC sublayer provides two services:

- the MAC data service; the MAC data service enables the transmission and reception of MAC protocol data units (MPDUs) across the PHY data service.
- the MAC management service interfacing to the MAC sublayer management entity (MLME) service access point (SAP) (known as MLME-SAP).

IEEE 802.15.4 MAC superframe

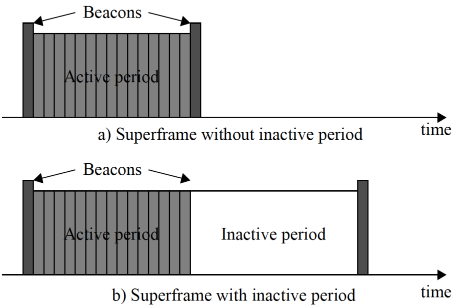


Figure: from [7]

The format of the superframe is defined by the coordinator. The superframe is bounded by beacons (either Beacon frames or Enhanced Beacon Frames) sent by the coordinator.
Optionally, the superframe can have an active and an inactive portion.
If a coordinator does not wish to use a superframe structure, it will turn off the beacon transmissions.
The beacons are used to synchronize the attached devices, to identify the PAN, and to describe the structure of the superframes.

IEEE 802.15.4 MAC superframe with guaranteed time slots (GTSs).

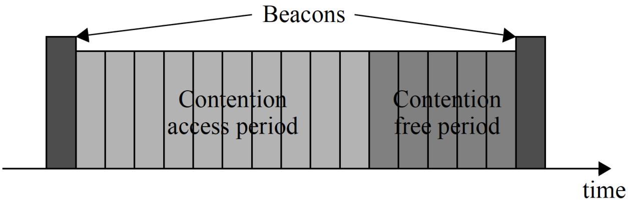


Figure: from [7]

Any device wishing to communicate during the contention access period (CAP) between two beacons competes with other devices using a slotted CSMA-CA or ALOHA mechanism, as appropriate. For low-latency applications or applications requiring specific data bandwidth, the PAN coordinator dedicates portions of the active superframe to that application. These portions are called guaranteed time slots (GTSs). The GTSs form the contention-free period (CFP), which always appears at the end of the active superframe starting at a slot boundary immediately following the CAP

IEEE 802.15.4 MAC deterministic and synchronous multi-channel extension (DSME) multi-superframe.

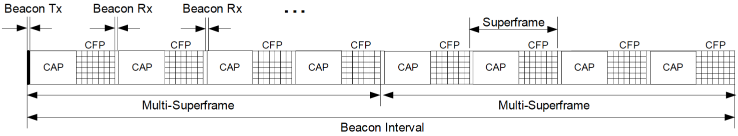


Figure: from [7]

Idea from previous IEEE 802.15.4e-2012: each coordinator periodically transmits a specific beacon frame to inform the nodes in coverage about the exact time instant that the multi-superframe structure starts. When a new device with the role of coordinator wants to join the network, it listens to the physical medium waiting for the reception of this beacon frame. Then, once the starting time of the multi-superframe is known, the new coordinator searches for an empty slot within the multisuperframe to transmit its own beacons and superframes. In addition, to further reduce collisions from devices transmitting in the same coverage area, DSME provides two possible solutions, on the one hand, the channel hopping technique on the other hand, the channel adaptation method based on the concept that nodes always remain in the same channel as long as the interference level experienced is kept within an acceptable range

IEEE 802.15.4 MAC: TSCH and slotframes.

Timeslotted channel hopping (TSCH) PANs are topology independent and can be used in star topologies as well as partial or full mesh topologies.
In a TSCH PAN, the concept of the superframe is replaced with a slotframe. The slotframe also contains defined periods of communications between peers that are either CSMA-CA or guaranteed, but the slotframe automatically repeats based on the participating devices' shared notion of time. Unlike the superframe, slotframes and a device's assigned timeslot(s) within the slotframe can be initially communicated by beacon, but are typically configured by a higher layer as the device joins the network. Because all devices share common time and channel information, devices hop over the entire channel space to minimize the negative effects of multipath fading and interference, and do so in a slotted way to avoid collisions, minimizing the need for retransmissions. Both of these features are desirable for operation in a **harsh industrial environment**.

IEEE 802.15.4 Data transfer types

The data transfer models used in this standard are:

- transfer to a coordinator in which a device transmits the data.
- transfer from a coordinator in which the device receives the data.
- transfer between two peer devices.

If a frame that is correctly received requests an acknowledgment, the device that received the frame will respond with one of the acknowledgment (Ack) frames: immediate acknowledgment (Imm-Ack), enhanced acknowledgment (Enh-Ack) or fragment acknowledgement.

IEEE 802.15.4 Data transfer from/to a Coordinator

When a **device wishes to transfer data to a coordinator** in a beacon-enabled PAN, it first listens for the beacon. When the beacon is found, the device synchronizes to the superframe structure. At the appropriate time, the device transmits its Data frame to the coordinator.
When a **device wishes to transfer data** in a nonbeacon-enabled PAN, it transmits its Data frame to the coordinator.

When the **coordinator wishes to transfer data to a device** in a beacon-enabled PAN, it indicates in the beacon that the data message is pending. The device periodically listens to the beacon and, if a message is pending, transmits a Data Request command. The pending Data frame is then sent by the coordinator. Upon successful completion of the data transaction, the message is removed from the list of pending messages in the beacon.
When a **coordinator wishes to transfer data to a device** in a nonbeacon-enabled PAN, it stores the data for the appropriate device to make contact and request the data. A device requests data by transmitting a Data Request command to its coordinator. If a Data frame is pending, the coordinator transmits the Data frame. If a Data frame is not pending, the coordinator indicates this fact either in the Ack frame following the Data Request command, if an acknowledgment was requested, or in a Data frame with a zero-length payload.

IEEE 802.15.4 Frame structure.

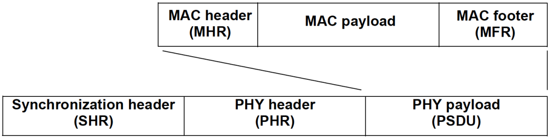


Figure: from [7]

IEEE 802.15.4 makes use of Information Elements (IEs) to transfer formatted data between layers and between devices. IEs consist of an identification, a length, and the IE content. Devices can accept or discard a particular element if the ID is known, and skip over unknown ID elements

IEEE 802.15.4 access methods

The access methods defined in this standard are:

- unslotted CSMA-CA used in nonbeacon-enabled PANs
- slotted CSMA-CA used in beacon-enabled PANs
- TSCH CCA used in non-shared slots in a TSCH PAN
- TSCH CSMA-CA used for shared slots in a TSCH PAN
- CSMA-CA with PCA (priority channel access) in for critical events
- LECIM ALOHA with PCA,

IEEE 802.15.4 acknoledgments

A successful reception and validation of a frame is **optionally** confirmed with an acknowledgment.
The receiving device can insert additional content in an Enh-Ack frame encapsulated as IEs. If the originator does not understand the IE content of the Enh-Ack frame, it is ignored, but the transmission is considered successful.
If the originator does not receive an acknowledgment after some period, it assumes that the transmission was unsuccessful and retries the frame transmission. If an acknowledgment is still not received after several retries, the originator can choose either to terminate the transaction or to try again. When the acknowledgment is not required, the originator assumes the transmission was successful.

IEEE 802.15.4 Frak and Data Verification

The **Frak** is used during the fragment sequence transfer to determine which fragments have been received successfully and which fragments need to be retransmitted. A Frak includes the status of one or more fragments.
A cyclic redundancy check (CRC) is used to detect errors in every PSDU. To accommodate individual fragment acknowledgments, a fragment integrity check sequence (FICS) is included with each fragment. The recipient uses the FICS and fragment number to determine which fragments of the sequence have been received correctly and which are missing.

IEEE 802.15.4 Low Power considerations

Low-energy mechanisms, coordinated sampled listening (CSL-from 802.15.4e) and receiver initiated transmission (RIT) are provided to further reduce energy consumption by allowing devices to communicate while maintaining low duty cycles.
CSL allows receiving devices to periodically sample the channel(s) for incoming transmissions at low duty cycles. The receiving device and the transmitting device are coordinated to reduce transmitting overhead.
RIT (from LCIM) allows receiving devices to periodically broadcast data request frames, and transmitting devices only transmit to a receiving device upon receiving a data request frame. RIT is suitable for the following application scenarios:

- Low data traffic rate and loose latency requirement, where a few seconds of latency is allowable by application.
- Local regulations restricting the duration of continuous radio transmissions.

IEEE 802.15.4 Security considerations - I

The cryptographic mechanism in IEEE 802.15.4 is based on symmetric-key cryptography and uses keys that are provided by higher layer processes. **The establishment and maintenance of these keys are outside the scope of IEEE 802.15.4 standard**.

The mechanism assumes a secure implementation of cryptographic operations and secure and authentic storage of keying material.

The cryptographic mechanism provides particular combinations of the following security services:

- Data confidentiality: Assurance that transmitted information is only disclosed to parties for which it is intended.
- Data authenticity: Assurance of the source of transmitted information (and, hereby, that information was not modified in transit).
- Replay protection: Assurance that duplicate information is detected.

The actual frame protection provided can be adapted on a frame-by-frame basis and allows for varying levels of data authenticity (to minimize security overhead in transmitted frames where required) and for optional data confidentiality. When nontrivial protection is required, replay protection is always provided.

IEEE 802.15.4 Security considerations - II

The actual frame protection provided can be adapted on a frame-by-frame basis and allows for varying levels of data authenticity (to minimize security overhead in transmitted frames where required) and for optional data confidentiality. When nontrivial protection is required, replay protection is always provided.

Cryptographic frame protection uses either a key shared between two peer devices (link key) or a key shared among a group of devices (group key), thus allowing some flexibility and application-specific tradeoffs between key storage and key maintenance costs versus the cryptographic protection provided. If a group key is used for peer-to-peer communication, protection is provided only against outsider devices and not against potential malicious devices in the key-sharing group.

Implementations should include mechanisms to prevent unauthorized access to locally stored keys.

IEEE 802.15.4 Primitives' types

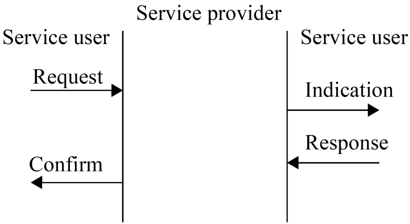


Figure: from [7]

A service is specified by describing the service primitives and parameters that characterize it. A primitive can be one of four generic types:

- **Request** : The request primitive is used to request that a service is initiated.
- **Indication** : The indication primitive is used to indicate to the user an internal event.
- **Response** : The response primitive is used to complete a procedure previously invoked by an indication primitive.
- **Confirm** : The confirm primitive is used to convey the results of one or more associated previous service requests.

IEEE 802.15.4 superfame structure.

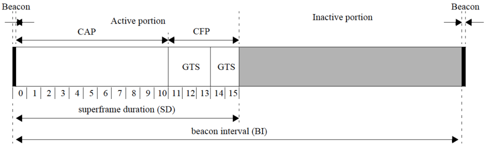


Figure: from [7]

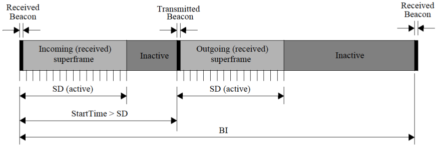


Figure: (from [7]) On a beacon-enabled PAN, a coordinator that is not the PAN coordinator shall maintain the timing of both the superframe in which its coordinator transmits a beacon (the incoming superframe) and the superframe in which it transmits its own beacon (the outgoing superframe).

IEEE 802.15.4 conventions

Constants and PAN information base (PIB) attributes that are specified and maintained by the MAC sublayer or PHY layer are written in the text in italics. Constants have a general prefix of "*a*", e.g., *aBaseSlotDuration*.

MAC PIB attributes have a general prefix of "*mac*", e.g., *macExtendedAddress*.

PHY PIB attributes have a general prefix of "*phy*", e.g., *phyCurrentChannel*.

IEEE 802.15.4 separation between two successive frames

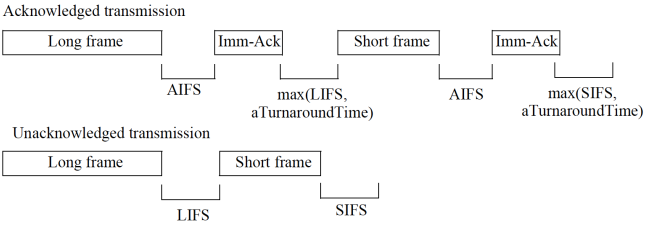


Figure: from [7]

The MAC sublayer needs a finite amount of time to process data received by the PHY. To allow for this, two successive frames transmitted from a device shall be separated by at least an IFS period; if the first transmission requires an acknowledgment, the separation between the Ack frame and the second transmission shall be at least the acknowledgment interframe spacing (AIFS). The length of the IFS period is dependent on the size of the frame that has just been transmitted. Frames (i.e., MPDUs) of up to *aMaxSIFSFrameSize* shall be followed by a short interfame space (SIFS) period of a duration of at least *max(macSifsPeriod, aTurnaroundTime)*. Frames (i.e., MPDUs) with lengths greater than *aMaxSIFSFrameSize* shall be followed by the maximum of long interframe spacing (LIFS) of a duration of at least *max(macLifsPeriod, aTurnaroundTime)*.

IEEE 802.15.4 channel access: CSMA/CA

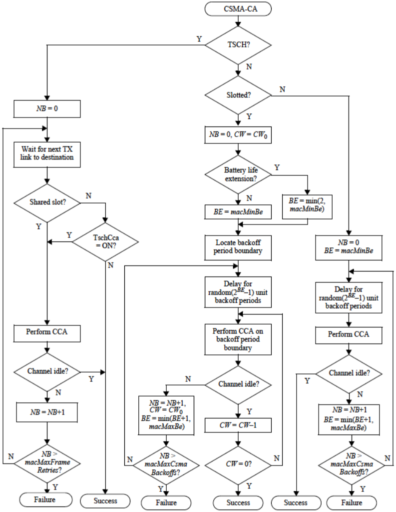


Figure: from [7] valid with and without beacons

IEEE 802.15.4 conventions

Each device shall maintain three variables for each transmission attempt: NB, CW, and BE. NB is the number of times the CSMA-CA algorithm was required to back off while attempting the current transmission; this value shall be initialized to zero before each new transmission attempt. CW is the contention window length, defining the number of backoff periods that need to be clear of channel activity before the transmission can commence. The value of shall CW_0 be initialized to two before each transmission attempt and reset to CW_0 each time the channel is assessed to be busy. The CW variable is only used for slotted CSMA-CA. BE is the backoff exponent, which is related to how many backoff periods a device shall wait before attempting to assess a channel

IEEE 802.15.4 other channel access modes

We will not go into details for channel access on networks with TSCH, as well as we will not touch the CSMA-CA with PCA and LECIM Aloha PCA.

For these please refer to [7]

IEEE 802.15.4 TSCH time slot

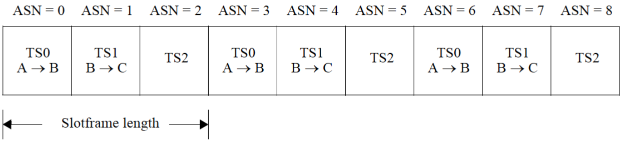


Figure: from [7]

IEEE 802.15.4 TSCH multiple time slot

	ASN = 0	ASN = 1	ASN = 2	ASN = 3	ASN = 4	ASN = 5	ASN = 6	ASN = 7	
Slotframe 1 5 slots	TS0	TS1	TS2	TS3	TS4	TS0	TS1	TS2	...
Slotframe 2 3 slots	TS0	TS1	TS2	TS0	TS1	TS2	TS0	TS1	...

Figure: from [7]

In TSCH mode, a given network using timeslot-based access may contain several concurrent slotframes of different sizes. Multiple slotframes may be used to define a different communication schedule for various groups of nodes or to run the entire network at **different duty cycles** by giving some devices many active timeslots in a slotframe, and others few or none. A network device may participate in one or more slotframes simultaneously, and not all devices need to participate in all slotframes. By configuring a network device to participate in multiple overlapping slotframes of different sizes, it is possible to establish different communication schedules and connectivity matrices that all work at the same time.

IEEE 802.15.4 starting a PAN

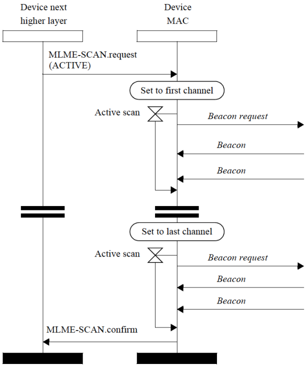


Figure: from [7]: active scan; with no beacon request we have passive scan; an Energy Detetction scan allows a device to obtain a measure of the peak energy in each requested channel; this could be used by a prospective PAN coordinator to select a channel on which to operate prior to starting a new PAN.

IEEE 802.15.4 Device discovery

The PAN coordinator or a coordinator indicates its presence on a PAN to other devices by transmitting Beacon frames. This allows the other devices to perform device discovery. A coordinator that is not the PAN coordinator shall begin transmitting Beacon frames only when it has successfully associated with a PAN. The transmission of Beacon frames by the device is initiated through the use of the MLME-START.request primitive with the PANCoordinator parameter set to FALSE

Specific procedures should be used for TSCH PANs

IEEE 802.15.4 device association

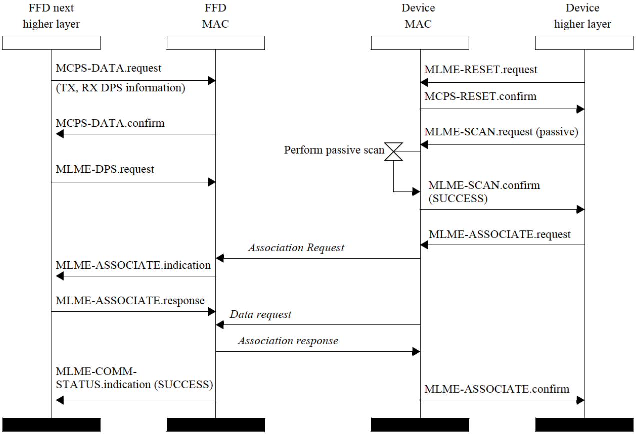


Figure: from [7] if the device does not listen to beacons the association is started from a FFD

IEEE 802.15.4 Transaction handling

Because the IEEE 802.15.4 standard favors very low cost devices that, in general, will be battery powered, **transactions can be instigated from the devices themselves rather than from the coordinator**. In other words, either the coordinator needs to indicate in its beacon when messages are pending for devices or the devices themselves need to poll the coordinator to determine whether they have any messages pending. Such transfers are called indirect transmissions

IEEE 802.15.4 Transmission, reception, and acknowledgment- I

If the frame is to be transmitted on a beacon-enabled PAN, the transmitting device shall attempt to find the beacon before transmitting. If the beacon is not found after a time specified in the standard, the device shall transmit the frame following the successful application of the unslotted version of the CSMA-CA algorithm. Once the beacon has been found, either after a search or due to its being tracked, the frame shall be transmitted in the appropriate portion of the superframe using the slotted CSMA-CA algorithm.

If the frame is to be transmitted on a nonbeacon-enabled PAN, the frame shall be transmitted following the successful application of the unslotted version of the CSMA-CA algorithm.

For either a beacon-enabled PAN or a nonbeacon-enabled PAN, if the transmission is direct and originates due to a primitive issued by the next higher layer and the CSMA-CA algorithm fails, the next higher layer shall be notified. If the transmission is indirect and the CSMA-CA algorithm fails, the frame shall remain in the transaction queue until it is requested again and successfully transmitted or until the transaction expires

IEEE 802.15.4 Transmission, reception, and acknowledgment- II

A transceiver task shall be defined as a transmission request with acknowledgment reception, if required, or a reception request. On completion of each transceiver task, the MAC sublayer shall request that the PHY enables or disables its receiver.
Due to the nature of radio communications, a device with its receiver enabled will be able to receive and decode transmissions from all devices complying with this standard that are currently operating on the same channel and are in its radio communications range, along with interference from other sources. The MAC sublayer shall, therefore, be able to filter incoming frames and present only the frames that are of interest to the next higher layer.

IEEE 802.15.4 Transmission, reception, and acknowledgment- II

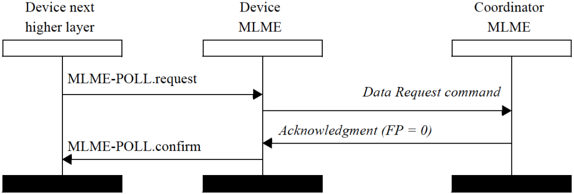


Figure: from [7]: requesting data from the coordinator when the coordinator does not have data pending

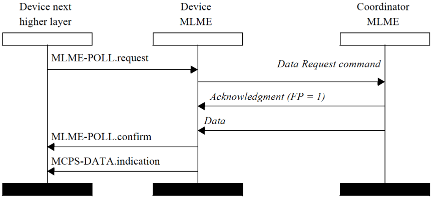


Figure: from [7]: requesting data from the coordinator when the coordinator has data pending

IEEE 802.15.4 Transmission, reception, and acknowledgment- III

If a single transmission attempt has failed and the transmission was indirect, the coordinator shall not retransmit the frame. Instead, the frame shall remain in the transaction queue of the coordinator and can only be extracted following the reception of a new Data Request command. If a new Data Request command is received, the originating device shall transmit the frame using the same DSN as was used in the original transmission.

If a single transmission attempt has failed and the transmission was direct, the device shall repeat the process of transmitting the frame and waiting for the acknowledgment, up to a maximum of *macMaxFrameRetries* times

IEEE 802.15.4 general MAC frame format

Octets: 1/2	**0/1**	**0/2**	**0/2/8**	**0/2**	**0/2/8**	**variable**	**variable**		**variable**	**2/4**
Frame Control	Sequence Number	Destination PAN ID	Destination Address	Source PAN ID	Source Address	Auxiliary Security Header	IE		Frame Payload	FCS
		Addressing fields					Header IEs	Payload IEs		
MHR								MAC Payload		MFR

Figure: from [7]

IEEE 802.15.4 MAC frame control field

Bits: 0–2	3	4	5	6	7	8	9	10–11	12–13	14–15
Frame Type	Security Enabled	Frame Pending	AR	PAN ID Compression	Reserved	Sequence Number Suppression	IE Present	Destination Addressing Mode	Frame Version	Source Addressing Mode

Figure: from [7]

IEEE 802.15.4 MAC services -I

The MAC sublayer provides an interface between the next higher layer and the PHY. The MAC sublayer conceptually includes a management entity called the MLME. This entity provides the service interfaces through which layer management may be invoked. The MLME is also responsible for maintaining a database of managed objects pertaining to the MAC sublayer. This database is referred to as the MAC sublayer PIB.
The MAC sublayer provides two services, accessed through two SAPs:

- The MAC data service, accessed through the MAC common part sublayer (MCPS) data SAP (MCPS-SAP)
- The MAC management service, accessed through the MLME-SAP

These two services provide the interface between the next higher layer and the PHY. In addition to these external interfaces, an implicit interface also exists between the MLME and the MCPS that allows the MLME to use the MAC data service.

IEEE 802.15.4 MAC services -II

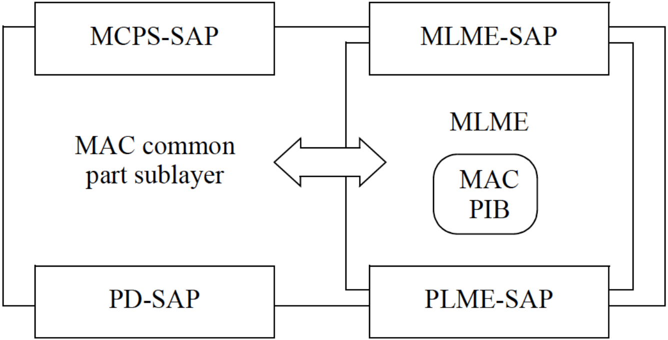


Figure: from [7] - MAC sublayer reference model

IEEE 802.15.4 Security

We skip the description, please check [7].

IEEE 802.15.4 Physical Layer

The PHY is responsible for the following tasks:

- Activation and deactivation of the radio transceiver
- Energy detection (ED) within the current channel
- Link quality indicator (LQI) for received packets
- Clear channel assessment (CCA) for carrier sense multiple access with collision avoidance (CSMA-CA)
- Channel frequency selection
- Data transmission and reception
- Precision ranging for ultra-wide band (UWB) PHYs

IEEE 802.15.4 Physical Layer Modulations and Frequency Bands-I

- O-QPSK PHY: direct sequence spread spectrum (DSSS) PHY employing offset quadrature phaseshift keying (O-QPSK) modulation, operating in the 780 MHz band, 868 MHz band, 915 MHz band, 2380 MHz band, and 2450 MHz band. **This is the most used, raw bit rate 250kbit/s**
- BPSK PHY: DSSS PHY employing binary phase-shift keying (BPSK) modulation, operating in the 868 MHz band, and 915 MHz band
- ASK PHY: parallel sequence spread spectrum (PSSS) PHY employing amplitude shift keying (ASK) and BPSK modulation, operating in the 868 MHz band and 915 MHz band
- CSS PHY: chirp spread spectrum (CSS) employing differential quadrature phase-shift keying (DQPSK) modulation, operating in the 2450 MHz band

IEEE 802.15.4 Physical Layer Modulations and Frequency Bands-II

- HRP UWB PHY: combined burst position modulation (BPM) and BPSK modulation, operating in the sub-gigahertz and 3–10 GHz bands
- MPSK PHY: M-ary phase-shift keying (MPSK) modulation, operating in the 780 MHz band
- GFSK PHY: Gaussian frequency-shift keying (GFSK), operating in the 920 MHz band
- MSK PHY: minimum shift keying (MSK) PHY
- LRP UWB PHY: low rate pulse UWB PHY

IEEE 802.15.4 Physical Layer Modulations and Frequency Bands-III

- SUN FSK PHY: SUN FSK PHY operating in multiple over-the-air data rates in support of SUN applications
- SUN OFDM PHY: SUN OFDM PHY operating in multiple over-the-air data rates in support of SUN applications
- SUN O-QPSK PHY: SUN O-QPSK PHY operating in multiple over-the-air data rates in support of SUN applications.

IEEE 802.15.4 Physical Layer Modulations and Frequency Bands-IV

- LECIM DSSS PHY: DSSS PHY operating with characteristics that enable support of LECIM applications.
- LECIM FSK PHY: FSK PHY operating with characteristics that enable support of LECIM applications.
- TVWS-FSK PHY: FSK PHY operating in multiple over-the-air data rates in support of various applications in TVWS.
- TVWS-OFDM PHY: OFDM PHY operating in multiple over-the-air data rates in support of various applications in TVWS.
- TVWS-NB-OFDM PHY: narrow band OFDM (NB-OFDM) PHY operating in multiple over-the air data rates in support of various applications in TVWS.

IEEE 802.15.4 Physical Layer Modulations and Frequency Bands-V

- RCC LMR PHY: land mobile radio (LMR) for use in rail communications and control (RCC) applications using one of Gaussian minimum shift keying (GMSK), 4FSK, QPSK, $\pi/4$ differential quadrature phase-shift keying (DQPSK), or DSSS employing DPSK.
- RCC DSSS BPSK PHY: a DSSS BPSK PHY for use in RCC applications.

IEEE 802.15.4 Physical Layer Frequency Bands - I

Band designation	Frequency band (MHz)
169 MHz	169.400–169.475
433 MHz	433.05–434.79
450 MHz	450–470
470 MHz	470–510
780 MHz	779–787
863 MHz	863–870
868 MHz	868–868.6
896 MHz	896–901
901 MHz	901–902
915 MHz	902–928
917 MHz	917–923.5
920 MHz	920–928

IEEE 802.15.4 Physical Layer Frequency Bands - II

Band designation	Frequency band (MHz)
928 MHz	928–960[1]
1427 MHz	1427–1518a
2380 MHz	2360–2400
2450 MHz	2400–2483.5
HRP UWB	sub-gigahertz 250–750
HRP UWB	low band 3244–4742
HRP UWB	high band 5944–10 234
LRP UWB	6289.6–9185.6

[1] non contiguous.

6LoWPAN - RPL

We are just at the MAC level, we need the upper layers

- 6LoWPAN basically introduce the IP connectivity in mesh (IEEE 802.15.4) IoT networks; it does not cover routing
- RPL is the routing protocol which normally goes with 6LoWPAN

What is 6LoWPAN

6LoWPAN is an acronym of IPv6 over Low power Wireless Personal Area Networks.
6LoWPAN is the name of a concluded working group in IETF
`https://datatracker.ietf.org/wg/6lowpan/documents/`

> *6LoWPAN is a protocol definition to enable IPv6 packets to be carried on top of low power wireless networks, specifically IEEE 802.15.4. The concept was born from the idea that the Internet Protocol could and should be applied to even the smallest of devices. [9]*

But what is really needed to get IPv6 on Low power Wireless Personal Area Networks? Which problem does 6LoWPAN solve?

Problem statement

The main problem

The IPv6 header is **40 bytes long**. We cannot carry the IPv6 on and IEEE 802.1.5.4 (127 bytes maximum MAC payload size) because e basically consume 30% of the bandwidth just because of the header.

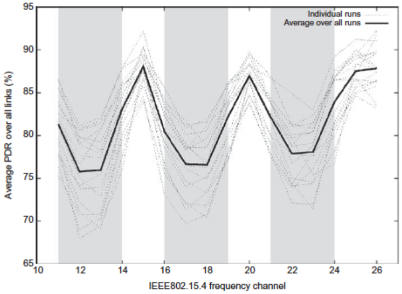


Figure: from [11]

The basic of the solution

The main problem

We need a **header compression** mechanism

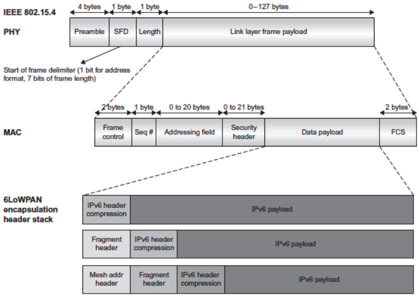
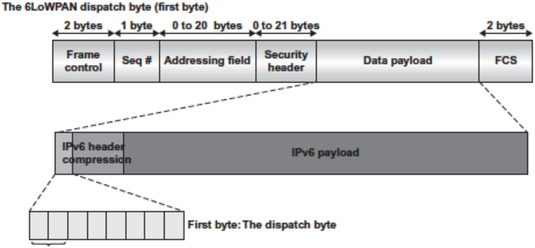


Figure: from [11]

The summary of the solution



00	Not a 6LoWPAN frame
01	IPv6 addressing header
10	Mesh header
11	Fragmentation header (6 lower bits are 100xxx)

Figure: from [11]

However many other problem arise when making and adaptation layer-I

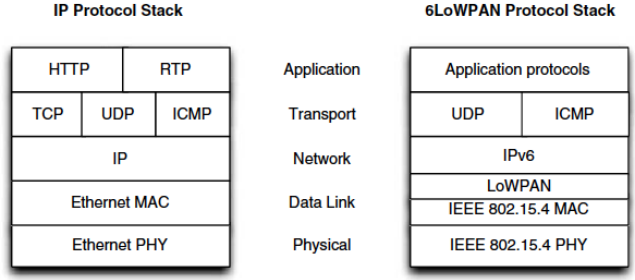


Figure: from [11]

However many other problem arise when making and adaptation layer-II

addressing Once the IP layer has decided on the IP address of the next hop for a packet, one of the tasks of the adaptation layer is to find out to which link-layer address the packet needs to be addressed to so that it advances on its way to the intended IP-layer destination

forwarding While LoWPANs are not connection-oriented, in a Mesh-Under situation the adaptation layer may have to figure out the next L2 hop and may need to provide that hop with information about the further direction to forward the packet on.

fragmentation In order to be able to transport larger IPv6 packets, there needs to be a way to carve up the L3 packets and put their contents into multiple L2 packets.

header compression IP was designed so that each packet stands completely on its own. This leads to a header that may contain a lot of information that could be inferred from its context. In a LoWPAN, the typical IP/UDP header size of 48 bytes already consumes a significant part of the payload space available in a single IEEE 802.15.4 packet, leaving little for applications before fragmentation has to set in. A better approach is to eliminate large parts of the redundancy at the L3–L2 interface. Existing IETF standards for header compression (such as ROHC) are too heavyweight for LoWPAN Nodes; therefore 6LoWPAN comes with its own header compression

Pre-requisites for 6LoWPAN

Prerequisites

You need to have a decent familiarity with the IPv6 and related protocols.

Internet and 6LoWPAN

The 6LoWPAN architecture is made of 6LoWPANs (low power wireless area networks) which are IPv6 *stub* network.
A 6LoWPAN network is a collection of 6LoWPAN nodes which share a common IPv6 address *prefix* (i.e., the first 64 bits of a IPv6 address).

Three types of 6LoWPAN networks

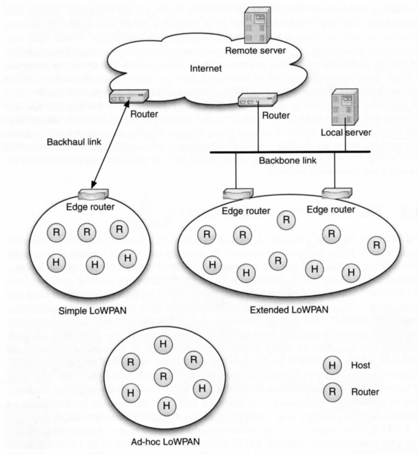


Figure: from [10]

6LowPAN addressing: distinguishing 6LoWPAN packets: the *dispatch* byte

The basic data packet format defined by IEEE 802.15.4. This format does not contain any fields that further identify the payload that a data packet is carrying: there is no multiplexing information that would allow a receiver to distinguish 6LoWPAN packets from any other data packets that might be sent, or to distinguish between different kinds of 6LoWPAN packets.

The first byte of the payload is used as a **dispatch** byte providing both a type identifier and possibly further information within the subtype.

The *dispatch* byte

Table 2.2 Current and proposed dispatch byte allocations.

From	To	Allocation
00 000000	00 111111	NALP – Not a LoWPAN frame (NALP)
01 000000		reserved for future use
01 000001		IPv6 – uncompressed IPv6 packets
01 000010		LOWPAN_HC1 – compressed IPv6, see Section 2.6.1
01 000011	01 001111	reserved for future use
01 010000		LOWPAN_BC0 – broadcast, see Section 2.8
01 010001	01 011111	reserved for future use
01 100000	01 111111	proposed for LOWPAN_IPHC, see Section 2.6.2
01 111111		ESC – Additional Dispatch byte follows (preempted by IPHC)
10 000000	10 111111	MESH – Mesh header, see Section 2.5
11 000000	11 000111	FRAG1 – Fragmentation Header (first), see Section 2.7
11 001000	11 011111	reserved for future use
11 100000	11 100111	FRAGN – Fragmentation Header (subsequent), see Section 2.7
11 101000	11 101011	proposed for fragment recovery [ID-thubert-sfr]
11 101100	11 111111	reserved for future use

Figure: from [10]

6LowPAN addressing: derivation of IPv6 addresses from link-layer addresses -II

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                            Prefix                             +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    OUI     |U|M|         OUI (cont.)          |               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+               +
|                     extension identifier                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+

2001:0DB8:0BAD:FADE::                   -- Prefix (/64)
                    ACDE:4812:3456:7890 -- EUI-64
2001:0DB8:0BAD:FADE:AEDE:4812:3456:7890 -- IPv6 Address
```

Figure: derivation of the IPv6 address from the EUI-64

In 6LoWPAN, this derivation of IPv6 addresses from link-layer addresses is mandatory, as the 6LoWPAN optimizations to the Neighbor Discovery protocol rely on this mapping

6LowPAN routing - I

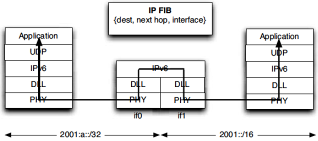


Figure: two interfaces routing - from [10]

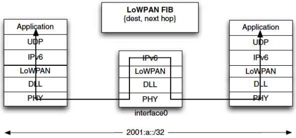


Figure: 6LoWPAN one interface routing - from [10]

6LowPAN forwading - L2 - a

Strictly speaking, 6LoWPAN does not address routing, performed by a specific protocol called RPL (next lesson). However the 6LoWPAN adaptation layer need to take care of routing in case of *mesh-under* routing. Since each forwarding step overwrites the link-layer destination address by the address of the next hop and the link-layer source address by the address of the node doing the forwarding, this information needs to be stored somewhere else. 6LoWPAN defines the mesh header for this.

6LowPAN forwarding - L2 - b

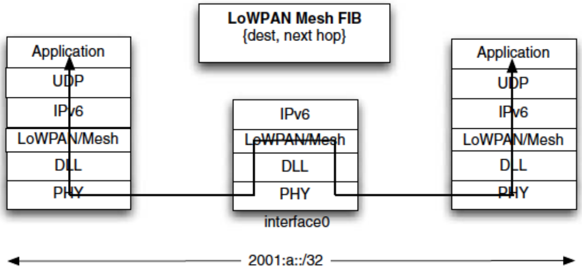


Figure: 6LoWPAN mesh-under - from [10]

6LowPAN header compression hop by hop

Header compression can be performed end-to-end, but is then limited to compressing the headers that are within the payload of the IP header, as the routers on the way between compressor and decompressor still have to see full IP headers. Since the largest header in many IPv6 header stacks is the IP header itself, this is not very efficient. Instead, most header compression schemes operate hop-by-hop, i.e. as part of the adaptation layer. This allows compressing the full header stack including the IP header immediately before sending the packet on a link, and decompressing and thus reconstructing the header stack in full before the packet is possibly routed and sent on via a different link, quite likely with a different header compression scheme or at least different header compression parameters. The beauty of this approach is that deploying header compression becomes a local decision between two neighbors: only the two nodes on the ends of a link need to agree on its use (and the specific parameters to be used).

6LowPAN header compression summary

The 6LoWPAN format specification [RFC4944] defines two header compression schemes that are designed to work together: HC1 to compress IPv6 headers, and HC2 to compress UDP headers. HC2 can optionally be used in packets that also use HC1. HC1 is selected by using a dispatch byte of LOWPAN_HC1 (01000010). The next byte then selects various options; the final bit, if set, indicates that HC2 is used as well, using another byte of options. The objective of HC1/HC2 was to enable header compression in an entirely stateless fashion. In other words, there is no requirement for previous agreement between nodes exchanging the compressed packets. So HC1/HC2 can essentially just exploit internal redundancies in the packet, or possibly encode more likely variations of the packet in fewer bits. The most useful redundancy in a 6LoWPAN packet is caused by the way IP addresses are formed from layer-2 addresses, both of which are present in the overall layer-2 packet. When a 6LoWPAN host sends a packet, the layer-2 source address of the packet will reflect the MAC address of the host.

6LowPAN fragmentation and re-assembly summary

Instead of providing a more-fragments flag, like IPv4, 6LoWPAN copies the size of the packet to be reassembled (IPv6 header + IPv6 payload) into every fragment. This enables the receiving end to allocate a buffer for the whole reassembly unit upon reception of the first fragment, independent of which of the fragments actually arrives first. RFC 4944 somewhat misleadingly calls the reassembled packets *datagrams* and therefore calls the size field *datagram_size*. It is 11 bits in length, which would allow reassembled units of up to 2047 bytes (fitting nicely the actual IP-layer MTU of 6LoWPAN interfaces, which is defined at 1280 bytes). Similar to the IPv4 Identification field, a 16-bit *datagram_tag*, combined with the sender's link layer address, the destination's link layer address and the *datagram_size*, is used to distinguish the different packets to be reassembled.
We do not provide the full details in this presentation. Please refer to [10].

The need for a 6LowPAN neighbor discovery

The IPv6 Neighbor Discovery protocol [RFC4861] is a focal point in bootstrapping an IPv6 network. A node uses Neighbor Discovery (ND) to discover other nodes on the same link, to determine their link-layer addresses, to find routers, and to maintain reachability information about the paths to neighbors that the node is actively communicating with. ND can be combined with other protocols such as DHCPv6 to obtain additional node configuration information; for resource-limited nodes in a LoWPAN, such a combination of mechanisms is often more overhead than desired. The ND specification states that it applies to all types of links, unless specified otherwise in the relevant IP-over-X specification [RFC4861, section 1].

6LowPAN neighbor discovery

The base ND protocol divides nodes into the traditional roles of host and router, where only the router forwards IP packets that are not addressed to itself. Routers have to perform certain additional functions in ND as compared to hosts. As many nodes in LoWPANs will be limited in their capabilities, 6LoWPAN-ND introduces a third role, that of the edge router, which is specialized to perform some of the more complex functions of 6LoWPAN-ND, reducing the complexity of the tasks to be performed by the other routers and in particular by the hosts. The main new concept is that of a *whiteboard* maintained by the edge routers, centralizing some of the protocol state. Also, some simplifying assumptions are made that relieve the ND protocol of some of its tasks entirely. Finally, 6LoWPAN-ND can be used to disseminate the context information that enables the higher compression efficiency of context-based header compression

6LowPAN neighbor discovery summary

In IPv4, a node that did not have an address configured for itself had to use the dynamic host configuration protocol (DHCP) to obtain one from a DHCP server. DHCP employs a four-way message exchange to select one of possibly multiple DHCP servers and to obtain a time-bounded lease on an address assigned by the selected server. DHCP has been ported to IPv6 as DHCPv6 [RFC3315]. However, the larger address size of IPv6 enables the use of a simpler mechanism for address configuration: Stateless Address Autoconfiguration (SAA) [RFC4862]. 6LoWPAN simplifies the IPv6 addressing model by requiring that the node addresses are formed from an interface ID built from a MAC layer address and a prefix. Two such addresses are needed for each 6LoWPAN interface: a link-local address, constructed from the prefix FE80::/10, and (usually) a globally routable address, constructed from the globally routable prefix of the LoWPAN. How does a node find out that prefix? In standard ND, routers periodically send router announcements (RAs), and, if they don't want to wait for the periodical RA, nodes can solicit one using a Router Solicitation (RS) message. Both messages are usually multicast. In this specific case, this does not pose a problem even in LoWPANs: the communication occurs between host and first-hop router, so no expensive multihop forwarding of the messages is needed.

Not covered in the presentation

- More details on 6LoWPAN-ND
- Security
- Mobility and routing

CoAP

We are just at the network level, we need the upper layers

The CoAP standard enable the application level. We directly refer to a presentation by one of its main inventors.

Low Power Wide Area Networks

The new kid in town

From a Mesh topology to a Star topology

LPWA target characteristics

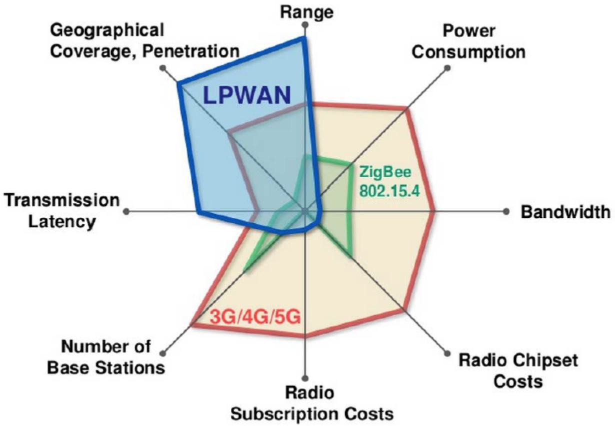


Figure: from [?]

LPWAN in licensed and unlicensed bands

LPWAN can be implemented in licensed or unlicensed bands

1. licensed bands: C-IoT a.k.a. NB-IoT standardised in release 13 by 3GPP on June 2016
2. unlicensed bands: SigFox, Lo-Ra, Weightless, Ingenu and many others

The source of the regulation

CEPT ECC

Electronic Communications Committee

ERC Recommendation

70-03

Unlincensed bands in Europe

Frequency Band	ERP	Duty Cycle	Channel Bandwidth	Remarks
433.05 – 434.79 MHz	+10 dBm	<10%	No limits	No audio and voice
433.05 – 434.79 MHz	0 dBm	No limits	No limits	≤– 13 dBm/10 kHz, no audio and voice
433.05 – 434.79 MHz	+10 dBm	No limits	<25 kHz	No audio and voice
868 – 868.6 MHz	+14 dBm	< 1%	No limits	
868.7 – 869.2 MHz	+14 dBm	< 0.1%	No limits	
869.3 – 869.4 MHz	+10 dBm	No limits	< 25 kHz	Appropriate access protocol required
869.4 – 869.65 MHz	+27 dBm	< 10%	< 25 kHz	Channels may be combined to one high speed channel
869.7 -870 MHz	+7 dBm	No limits	No limits	
2400 – 2483.5 MHz	+7.85 dBm	No limits	No limits	Transmit power limit is 10-dBm EIRP

Duty Cycle

Duty Cycle Limit	Total *On* Time Within One Hour	Maximum *On* Time of One Transmission	Minimum *Off* Time of Two Transmission
< 0.1%	3.6 seconds	0.72 seconds	0.72 seconds
< 1%	36 seconds	3.6 seconds	1.8 seconds
< 10%	360 seconds	36 seconds	3.6 seconds

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

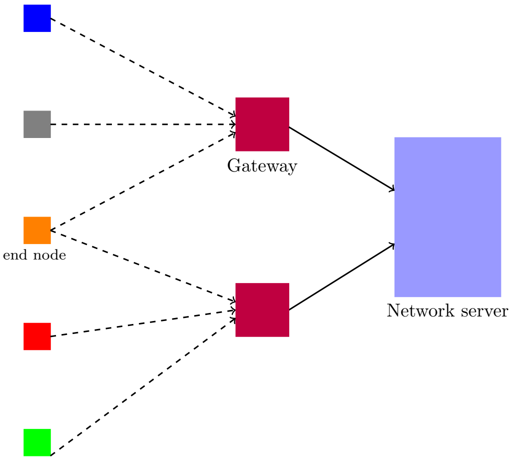
Specific Italian regulation

TABELLA DI ATTRIBUZIONE DEL

PIANO NAZIONALE DI RIPARTIZIONE DELLE FREQUENZE

BANDA DI FREQUENZE (MHz)	SERVIZIO	GESTORE	UTILIZZAZIONI	NORMATIVA INTERNAZIONALE
	MOBILE escluso mobile aeronautico 101B 103		-Reti mobili ad uso privato	
	Radiolocalizzazione	Ministero difesa		
446,0000 - 448,0000	FISSO 78 100B 101C	MiSE	-PMR 446 Analogico -PMR 446 Digitale --Ponti radio monocanali e sistemi rurali multiaccesso per collegamento di abbonato -Reti fisse ad uso privato	ERC/DEC/(98)25 ECC/DEC/(05)12 ECC/DEC/(06)06
	Radiolocalizzazione	Ministero difesa		
448,000 - 450,000	FISSO 78 100B 102	MiSE	- Reti fisse ad uso privato	ECC/DEC/(06)06
	Radiolocalizzazione 102	Ministero difesa		
450,0000 - 470,0000	MOBILE 46 78 85 100B 102 103 104 105 106 107	MiSE	-Cerca persone -Comunicazioni a bordo di imbarcazioni -Reti mobili ad uso privato -Ricerca spaziale	ECC/DEC/(04)06 ECC/DEC/(06)06
470,0000 - 608,0000	RADIODIFFUSIONE 58 87A 87B	MiSE	-Collegamenti audio a larga banda temporanei -Radiodiffusione televisiva -SRD - Radiomicrofoni	Piano di Ginevra 2006 ERC/REC 70-03
608,0000 - 614,0000	RADIODIFFUSIONE 59 87A 87B	MiSE	-Collegamenti audio a larga banda temporanei -Radiodiffusione televisiva -SRD- Radiomicrofoni	Piano di Ginevra 2006 ERC/REC 70-03
	Radioastronomia 108			
614,0000 - 790,0000	RADIODIFFUSIONE 58 87A 87B 109	MiSE	-Collegamenti audio a larga banda temporanei -Radiodiffusione televisiva -SRD - Radiomicrofoni	Piano di Ginevra 2006 ERC/REC 70-03
790,0000 - 862,0000	MOBILE escluso mobile aeronautico 112 112B 112C	MiSE	-Servizi di comunicazioni elettroniche terrestri -IMT -PMSE- Radiomicrofoni	RR 5.314 RES 224 RR(Rev. WRC-12) Piano di Ginevra 2006 2010/267/CE 2014/641/UE ECC/DEC/(09)03
862,0000 - 876,0000	FISSO 110 110A 110B 110C	Ministero difesa	-SRD - identificazione a radio frequenza (RFID)	RES 224 RR(Rev. WRC-12) 2006/771/CE 2006/804/CE ERC/REC 70-03
	MOBILE escluso mobile aeronautico 110 110A 110B 110C			

Architecture for LPWAN



SigFox

SIGFOX
One network A billion dreams

SigFox in Italy

NETROTTER
CONNECTING THINGS

EI TOWERS

Società appartenente al gruppo BERLUSCONI SILVIO

Struttura del gruppo

Società presenti nel gruppo: 86
Società controllate direttamente: 50
Società controllate indirettamente: 3
Società partecipate dal gruppo: 29
Profondità del gruppo: 6

L'impresa è parte di un gruppo societario

Acquista Gruppo Societario

Capogruppo

BERLUSCONI SILVIO
Codice Fiscale: BRLSLV36P29F205W
Nato il: 29/09/1936 **a:** MILANO (MI)

SigFox Physical Layer

The UNB PHY layer used in SigFox's network is very simple. Binary data are broadcast with a BPSK modulation at a very low rate $R_b = 100$ bit/s. The transmitted signal thus occupies a band of about $B = 100$ Hz.
As narrow band signals may suffer from flat fading, Frequency Hopping (FH) is supported to introduce diversity and improve the reliability.
To improve reliability, each message can be sent up to 3 times on different frequencies.

Sigfox MAC: Random Frequency and Time Division Multiple Access

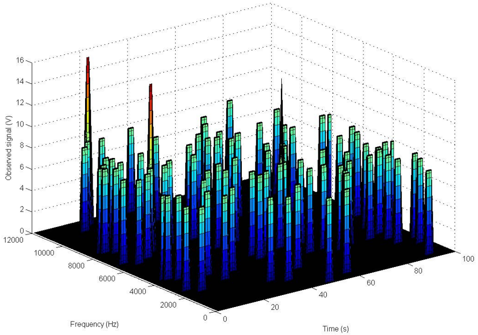
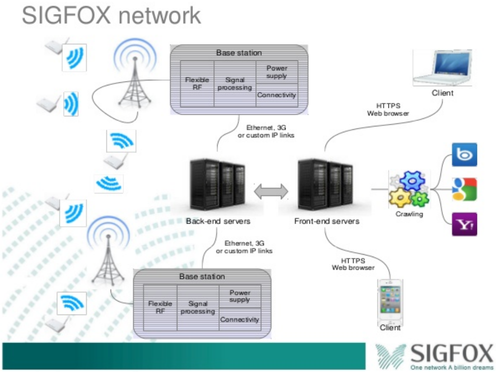


Figure: from [?]

Practical SigFox

SigFox is an **operator**

- messages of 12 bytes
- maximum 140 messages in uplink per day, no acknowledgement
- maximum 4 messages per day downlink



Weigthless

WEIGHTLESS™

Weightless SIG is an organization which aims at providing Weightless standards for IoT networks. These standards are based on the technology initially developed by the British company Neul, recently acquired by Huawei in fall 2014.
3 different standards are proposed : Weightless-N, Weightless-P and Weightless-W.

Weightless N

This standard is based on narrow band technology, with a differential binary phase shift keying (DBPSK) digital modulation scheme, and is mainly based on Nwave technology. The data rates are 30 - 100 kbps. Transmissions are performed in the sub-GHz ISM bands 868 MHz. This leads to a range of 5 kms even in challenging urban environments. A FH algorithm is used to counteract interference and fading. This standard is intended for devices that need one-way communications at a very low cost.

Weightless P

This second standard improves the first one by allowing two-way communications. This permits to enhance the reliability by using acknowledgment protocols. A multiple access is performed with FDMA+TDMA in 12:5kHz narrow band channels. Thus the base stations are time-synchronized to schedule the transmissions in the slots. However, the range is slightly reduced to 2 km in urban environment.

Weightless W

The last standard permits to take advantage of the available white-space in the spectrum use. Use of TV white spaces is the primary idea of the founding members of Weightless. The modulation modes vary from 16-QAM to DBPSK, in the 470 – 790 MHz TV band. Besides, spreading is also enabled (with a spreading factor up to 1024) to dynamically adapt the rate and the range to the actual needs. One may note that the Base Station (BS) transmission power is 20 dB higher than the end-device's one. To balance the link budget, the end-device uses a channel with a bandwidth 64 times smaller than the BS one. Time Division Duplexing is used to provide uplink and downlink pairing, as spectrum is not guaranteed in the TV white space. The indoor (resp. outdoor) range is up to 5 km (resp. 10 km).

Ingenu

Fron On-Ramp to Ingenu

RPMA (Random Phase Multiple Access) was developed by On-Ramp Wireless. This American company was founded in 2008, to provide connectivity to oil and gas actors. In September 2015, it was renamed Ingenu, and targets to extend its technology to the IoT and M2M market.
RPMA is based on DSSS. Transmissions are made in the **2.4 GHz ISM band**. Data are first encoded (1=2 rate) and interleaved. The resulting stream is then D-BPSK modulated, before being spread by a Gold Code. The signal is then **randomly delayed before transmission**.

RPMA

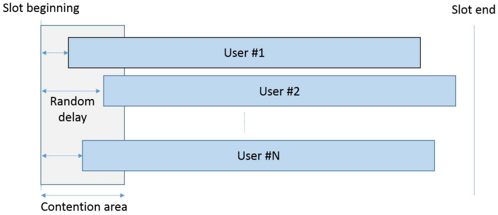


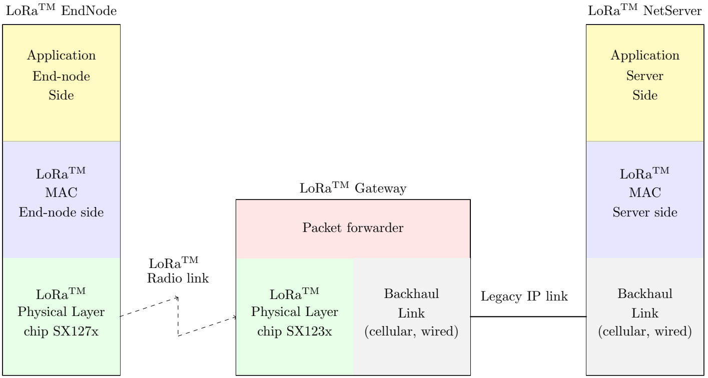
Figure: from [?]

RPMA in Italy: Meterlinq (Padova) has an exclusive deal with Ingenu

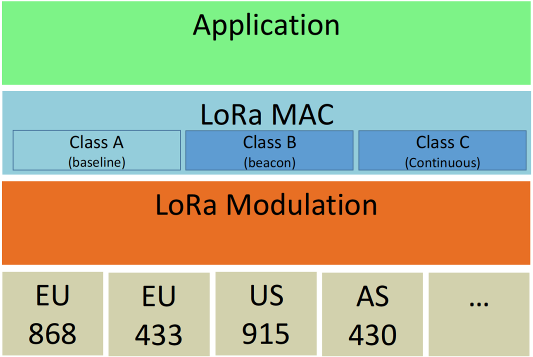
Lo-Ra WAN general description

LoRaWAN networks typically are laid out in a **star-of-stars topology** in which **gateways relay messages between end-devices and a central network server at the backend**. Gateways are connected to the network server via standard IP connections while end-devices use single-hop LoRa or FSK communication to one or many gateways.
All communication is generally bi-directional, although uplink communication from an end9 device to the network server is expected to be the predominant traffic. Communication between end-devices and gateways is spread out on different frequency channels and data rates. The selection of the data rate is a trade-off between communication range and message duration, communications with different data rates do not interfere with each other. **LoRa data rates range from 0.3 kbps to 50 kbps**.
To maximize both battery life of the end-devices and overall network capacity, **the LoRa network infrastructure can manage the data rate and RF output for each end-device individually by means of an adaptive data rate (ADR) scheme.**

Lo-Ra WAN protocol stack



Lo-Ra WAN devices classes



Lo-Ra physical layer

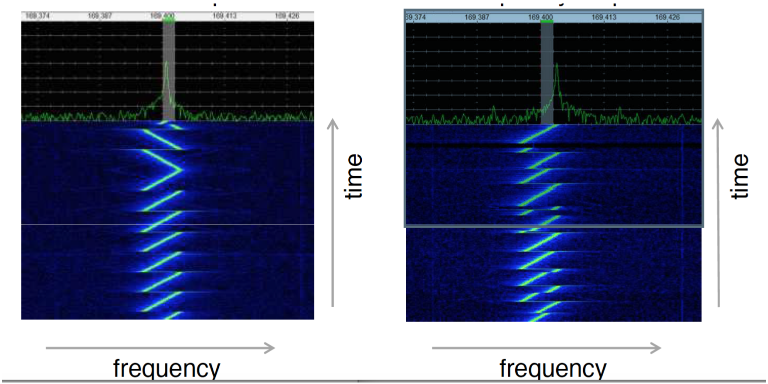


Figure: preamble and data packets: chirp modulation

Data rates as a function of the bandwidth, the channel coding and spreading factor

Spreading factor	Chips /symbol	125kHz (coding 4/5)	250kHz (coding 4/5)	500kHz (coding 4/5)
6	64	9380bps (-127.5dBm)	18750bps (-124.5dBm)	37500bps (-111.5dBm)
7	128	5469bps(-130.0dBm)	10938bbs (-127.0dBm)	21875bps (-124.0dBm)
8	256	3125bps(-132.5dBm)	6250bps (-129.5dBm)	12500bps (-126.5dBm)
9	512	1758bps(-135.0dBm)	3516bps (-132dBm)	7031bps (-129.0dBm)
10	1024	977bps(-137.5dBm)	1953bps (-134.5dBm)	3906bps (-131.5dBm)
11	2048	537bps(-140.0dBm)	1074bps (-137.0dBm)	2148bps (-134.0dBm)
12	4096	293bps (-142.5dBm)	586bps(-139.5dBm)	1172bps (-136.5dBm)

SNR for 32 octets @ 10% PER

Physical Messages format

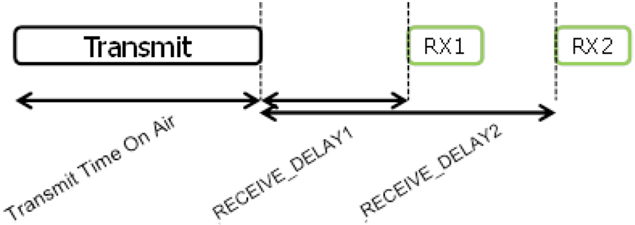
Uplink PHY:

Preamble	PHDR	PHDR_CRC	PHYPayload	CRC

Downlink PHY:

Preamble	PHDR	PHDR_CRC	PHYPayload

Receive windows



Lo-Ra message format

Radio PHY layer:

Preamble	PHDR	PHDR_CRC	PHYPayload	CRC*

Figure 5: Radio PHY structure (CRC* is only available on uplink messages)

PHYPayload:

MHDR	MACPayload	MIC

Figure 6: PHY payload structure

MACPayload:

FHDR	FPort	FRMPayload

Figure 7: MAC payload structure

FHDR:

DevAddr	FCtrl	FCnt	FOpts

Figure 8: Frame header structure

MAC header

Bit#	7..5	4..2	1..0
MHDR bits	MType	RFU	Major

MType	Description
000	Join Request
001	Join Accept
010	Unconfirmed Data Up
011	Unconfirmed Data Down
100	Confirmed Data Up
101	Confirmed Data Down
110	RFU
111	Proprietary

FCtrl content: the foundation of ADR

For downlink frames the FCtrl content of the frame header is:

Bit#	7	6	5	4	[3..0]
FCtrl bits	ADR	ADRACKReq	ACK	FPending	FOptsLen

For uplink frames the FCtrl content of the frame header is:

Bit#	7	6	5	4	[3..0]
FCtrl bits	ADR	ADRACKReq	ACK	RFU	FOptsLen

Adaptive data rate

If an end-device whose data rate is optimized by the network to use a data rate higher than its default data rate, it periodically needs to validate that the network still receives the uplink frames. Each time the uplink frame counter is incremented (for each new uplink, repeated transmissions do not increase the counter), the device increments an ADR_ACK_CNT counter. After ADR_ACK_LIMIT uplinks (ADR_ACK_CNT $\geq$ ADR_ACK_LIMIT) without any downlink response, it sets the ADR acknowledgment request bit (ADRACKReq). The network is required to respond with a downlink frame within the time set by the ADR_ACK_DELAY, any received downlink frame following an uplink frame resets the ADR_ACK_CNT counter. The downlink ACK bit does not need to be set as any response during the receive slot of the end-device indicates that the gateway has still received the uplinks from this device. If no reply is received within the next ADR_ACK_DELAY uplinks (i.e., after a total of ADR_ACK_LIMIT + ADR_ACK_DELAY), the end-device may try to regain connectivity by switching to the next lower data rate that provides a longer radio range. The end-device will further lower its data rate step by step every time ADR_ACK_LIMIT is reached. The ADRACKReq shall not be set if the device uses its default data rate because in that case no action can be taken to improve the link range.

ADR

Link Adaptative Data Rate (ADR)

The LinkADRReq command is sent by the network towards a node to change the node data rate, RF transmission power or channel used.

Octet	4	4	16
LinkADRReq payload	DataRate	TxPower	ChMask

The requested date rate (DataRate) and TX output power (TXPower) are encoded as follows:

Data rate (4 MSBs)	Requested mode
0	LoRa™: SF12/125kHz
1	LoRa™: SF11/125kHz
2	LoRa™: SF10/125kHz
3	LoRa™: SF9/125kHz
4	LoRa™: SF8/125kHz
5	LoRa™: SF7/125kHz
6	LoRa™: SF7/250kHz
7	FSK: 100kbps
8 ..15	Reserved

Table 6 – LinkADRReq data rate correspondence

TX power	
0	20dBm (if supported)
1	14dBm
2	11dBm
3	8dBm
4	5dBm
5	2dBm
6..15	Reserved

Table 7 – LinkADRReq Tx power correspondence

MAC commands

CID	Command	Transmitted by End-device	Transmitted by Gateway	Short Description
0x02	***LinkCheckReq***	x		Used by an end-device to validate its connectivity to a network.
0x02	***LinkCheckAns***		x	Answer to LinkCheckReq command. Contains the received signal power estimation indicating to the end-device the quality of reception (link margin).
0x03	***LinkADRReq***		x	Requests the end-device to change data rate, transmit power, repetition rate or channel.
0x03	***LinkADRAns***	x		Acknowledges the LinkRateReq.
0x04	***DutyCycleReq***		x	Sets the maximum aggregated transmit duty-cycle of a device
0x04	***DutyCycleAns***	x		Acknowledges a DutyCycleReq command
0x05	***RXParamSetupReq***		x	Sets the reception slots parameters
0x05	***RXParamSetupAns***	x		Acknowledges a RXSetupReq command
0x06	***DevStatusReq***		x	Requests the status of the end-device
0x06	***DevStatusAns***	x		Returns the status of the end-device, namely its battery level and its demodulation margin
0x07	***NewChannelReq***		x	Creates or modifies the definition of a radio channel
0x07	***NewChannelAns***	x		Acknowledges a NewChannelReq command
0x08	***RXTimingSetupReq***		x	Sets the timing of the of the reception slots
0x08	***RXTimingSetupAns***	x		Acknowledge RXTimingSetupReq command
0x80 to 0xFF	Proprietary	x	x	Reserved for proprietary network command extensions

Joining the network

Octets	1	8	8	2	4
Contents	MHDR Frame type = 000	AppEui	DevEui	DevNonce	MIC

Octets	1	6	4	2	4
Contents	MHDR Frame type = 001	AppNonce	DevAddr	Reserved	MIC

Figure 17: Join Response MAC frame format

The join-accept message contains an application nonce (AppNonce) of 6 octets and a device address (DevAddr) as described above. The AppNonce is a random value or some form of unique ID provided by the network server and used by the end device to derive the two session keys NwkSKey and AppSKey as follows:

NwkSKey = aes128_encrypt(AppKey, AppNonce | DevNonce | pad_{16})

AppSKey = aes128_encrypt(AppKey, DevNonce | AppNonce | pad_{16})

The MIC value for a join-accept message is calculated as follows [RFC4493]:

cmac = aes128_cmac(AppKey, MHDR | AppNonce | DevAddr | RFU)

MIC = cmac[0..3]

The join-accept message itself is encrypted with the AppKey as follows[2]:

aes128_decrypt(AppKey, AppNonce | DevAddr | RFU | MIC)

Platforms for IoT

we have dealt with moving data, but how to make them really usable?

We need to make the data usable, readable and actionable

Why Platforms?

We have seen different application frameworks like ETSI M2M, OneM2M and CoAP.

What is missing? **Platforms**

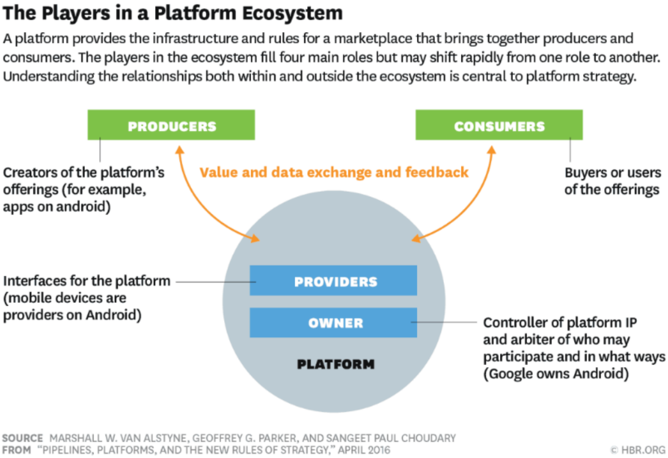
Platform definition

In a platform approach **a company's product operates as a facilitator** and large part of the value is created by the participants instead of the company itself, which can be Facebook users, Uber drivers or makers of Apple-compatible accessories.

Android and iOS have won the mobile OS war with their ability to **attract and keep developers** on their platforms rather than BlackBerry or Windows Mobile

What is new is that the platform approach have become the **standard practice in the digital B2C space** to compete for companies aiming to disrupt the market, or create a new one.

Android as a platform



Why we need a platform for Internet of Things?

The IoT community needs a tool for fast, secure, easy deployment of **IoT applications**.

ETSI M2M, OneM2M and CoAP are too complicated and **not visual**.

The first IoT platform: Patchube now Xively

Founded in 2007 as a data infrastructure and community for the Internet of Things.
In July 2011, Pachube announced that they had been acquired by LogMeIn and renamed to Cosm.
Cosm came out of beta development and was rebranded as Xively in May 2013.

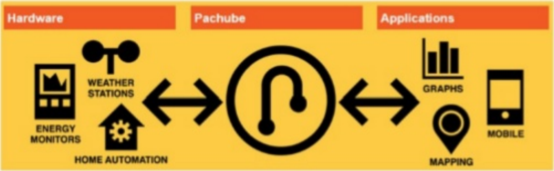
Patchube - I


pachube :: connecting environments, patching the planet
http://www.pachube.com/
pachube.apps | pachube.blog | pachube.community
username
password
login
forgot password?
pachube
about | tutorials | API | software/hardware | contact
Store, share & discover realtime sensor, energy and environment data from objects, devices & buildings around the world.
Pachube is a convenient, secure & scalable platform that helps you connect to & build the 'internet of things'. Sign up here!
Map view | List view
search
all | agriculture | building | device | energy | environment | transport | other
Satellite
Terrain
input - register a feed
If you have a device, building, environment or sensor (either physical or virtual) that is connected to the internet (wired, wireless or via SMS gateway) and you want to store, share, graph and distribute its datastreams in realtime...
... then register a feed!
output - use a feed
If you have a device, building, environment or actuator (either physical or virtual) that is connected to the internet, or if you have a website, and you want to embed, monitor or connect to an existing environment...
... then use a feed!
latest news
Triggers bring 'push' capabilities to Pachube
As promised a few weeks ago, we're going to be rolling out a whole host of new features th... » more
Google
Search
Terms of Use
send | link
RSS | Atom
Sign up and use Pachube to...

Patchube - II

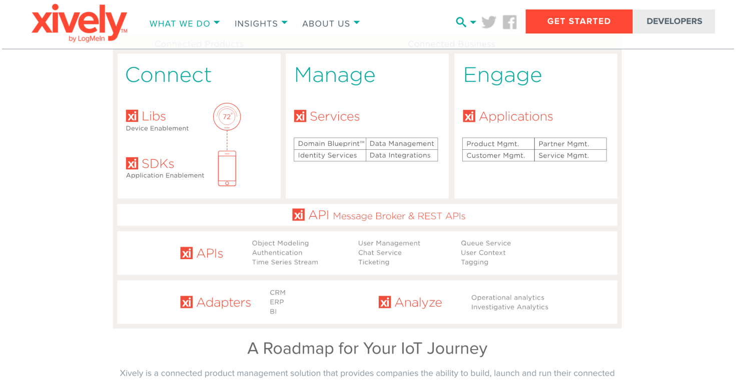
Example 1: Pachube

- **"The Internet of Things Real-Time Web Service and Applications"**
  - Platform to connect sensors and other hardware
  - Platform to build IoT services and applications
  - RESTful APIs

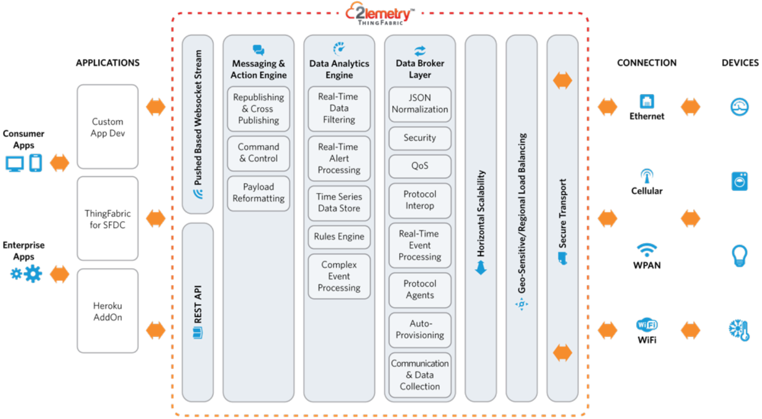


Source: https://pachube.com/

Xively


xively by LogMeIn
WHAT WE DO
INSIGHTS
ABOUT US
GET STARTED
DEVELOPERS
Connect
xi Libs
Device Enablement
xi SDKs
Application Enablement
Manage
xi Services
Domain Blueprint™
Data Management
Identity Services
Data Integrations
Engage
xi Applications
Product Mgmt.
Partner Mgmt.
Customer Mgmt.
Service Mgmt.
xi API Message Broker & REST APIs
xi APIs
Object Modeling
Authentication
Time Series Stream
User Management
Chat Service
Ticketing
Queue Service
User Context
Tagging
xi Adapters
CRM
ERP
BI
xi Analyze
Operational analytics
Investigative Analytics
A Roadmap for Your IoT Journey
Xively is a connected product management solution that provides companies the ability to build, launch and run their connected

2lemetry


2lemetry
THINGFABRIC
APPLICATIONS
Consumer Apps
Enterprise Apps
Custom App Dev
ThingFabric for SFDC
Heroku AddOn
Pushed Based Websocket Stream
REST API
Messaging & Action Engine
Republishing & Cross Publishing
Command & Control
Payload Reformatting
Data Analytics Engine
Real-Time Data Filtering
Real-Time Alert Processing
Time Series Data Store
Rules Engine
Complex Event Processing
Data Broker Layer
JSON Normalization
Security
QoS
Protocol Interop
Real-Time Event Processing
Protocol Agents
Auto-Provisioning
Communication & Data Collection
Horizontal Scalability
Geo-Sensitive/Regional Load Balancing
Secure Transport
CONNECTION
Ethernet
Cellular
WPAN
WiFi
DEVICES

2lemetry ...

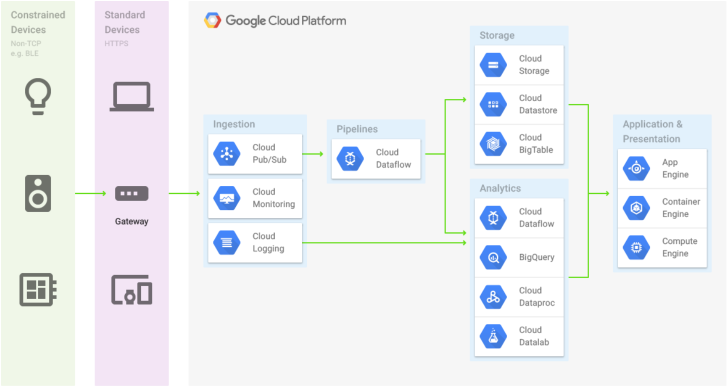
Amazon IoT components -I

- **Message broker** Provides a secure mechanism for things and IoT applications to publish and receive messages from each other. You can use the MQTT protocol to publish and subscribe. You can use the HTTP REST interface to publish.
- **Rules engine** Provides message processing and integration with other AWS services. You can use a SQL-based language to select data from message payloads, process the data, and send the data to other services, such as Amazon S3, Amazon DynamoDB, and AWS Lambda. You can also use the message broker to republish messages to other subscribers.
- **Thing Registry** Sometimes referred to as the Device Registry. Organizes the resources associated with each thing. You register your things and associate up to three custom attributes with each thing. You can also associate certificates and MQTT client IDs with each thing to improve your ability to manage and troubleshoot your things

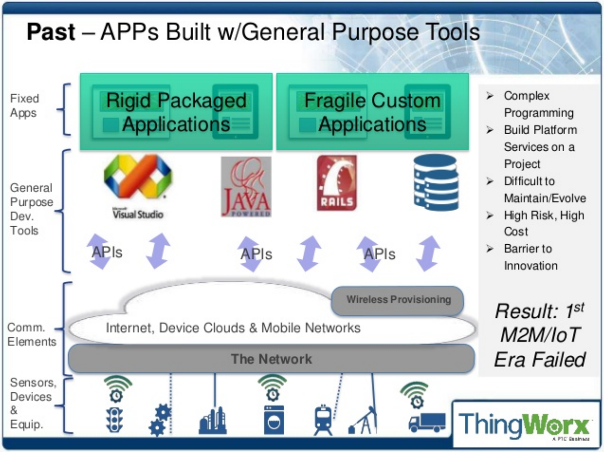
Amazon IoT components -II

- **Thing Shadows service** Provides persistent representations of your things in the AWS cloud. You can publish updated state information to a thing shadow, and your thing can synchronize its state when it connects. Your things can also publish their current state to a thing shadow for use by applications or devices.
- **Thing shadow** sometimes referred to as a *device shadow*. A JSON document used to store and retrieve current state information for a thing (device, app, and so on).
- **Device gateway** Enables devices to securely and efficiently communicate with AWS IoT
- **Security and identity service** Provides shared responsibility for security in the AWS cloud. Your things must keep their credentials safe in order to send data securely to the message broker. The message broker and rules engine use AWS security features to send data securely to devices or other AWS services.

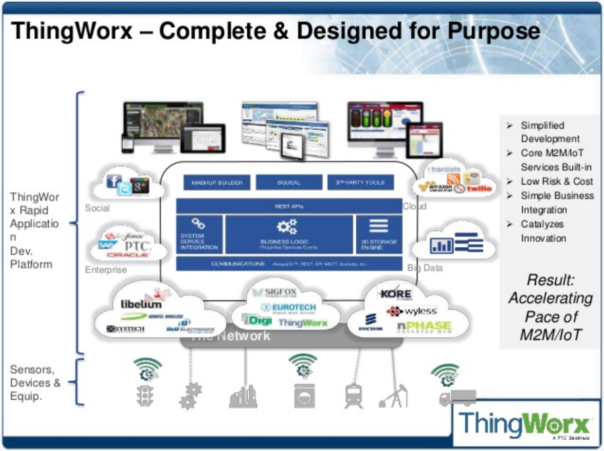
The other cloud giant: Google



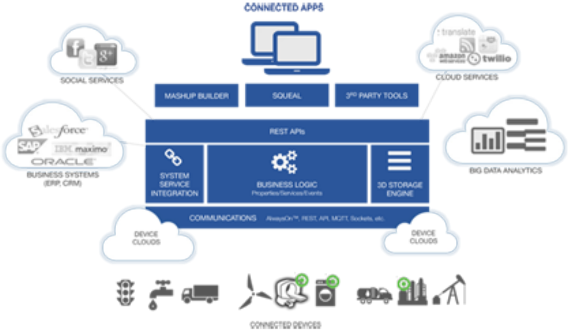
ThingWorx : the problem



ThingWorx : the solution



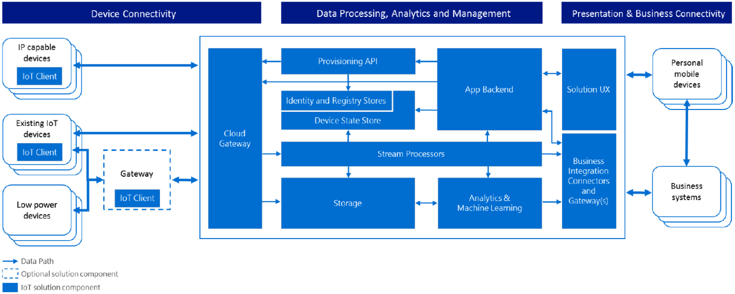
ThingWorx - Mashup builder



Jasper is now ... Cisco

SAN JOSE, Calif. — March 22, 2016, Cisco completed its acquisition of Jasper.

The elephant ... Microsoft Azure IoT



Many other bigs are providing their own IoT platform

- Ericsson
- IBM
- Oracle
- Bosch
- Telit (now partnering with Google)

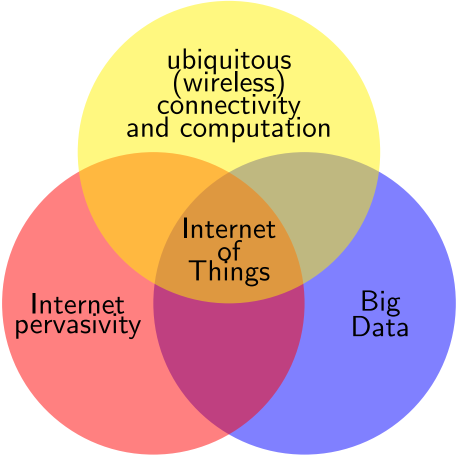
Kaa - an open source project

Big Data and IoT

IoT is generating really a lot of data, but how to make them really usable?

We need to make the data usable, readable and actionable

What is IoT?


ubiquitous (wireless) connectivity and computation
Internet pervasivity
Big Data
Internet of Things

Big Data - Definition

"Big Data represents the Information assets characterized by such a High **Volume, Velocity and Variety** to require specific Technology and Analytical Methods for its transformation into Value". (definition from META Group (now Gartner) analyst Doug Laney) Additionally, a new V "Veracity" is added by some organizations to describe it, revisionism challenged by some industry authorities. The 3Vs have been expanded to other complementary characteristics of big data:

- Volume: big data doesn't sample; it just observes and tracks what happens
- Velocity: big data is often available in real-time
- Variety: big data draws from text, images, audio, video; plus it completes missing pieces through data fusion
- **Machine Learning**: big data often doesn't ask why and simply detects patterns
- Digital footprint: big data is often a cost-free **byproduct of digital interaction**

Big Data and Open Data

Very relevant for Big Data coming from Smart Cities (who owns the data?)

Open data is any dataset that becomes freely available online. Open data are typically expressed in RDF an expressive and flexible data model
`https://it.wikipedia.org/wiki/Resource_Description_Framework`

A couple of interesting readings

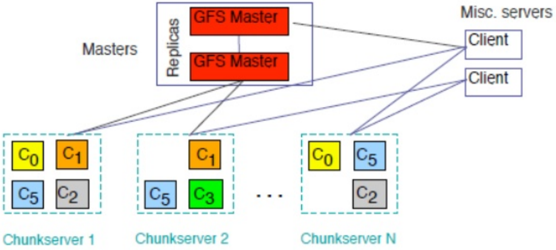
An interesting whitepaper
`http://cra.org/ccc/wp-content/uploads/sites/2/2015/05/bigdatawhitepaper.pdf`
An interesting book:
Donald Miner and Adam Shook. *MapReduce Design Patterns:Building Effective Algorithms and Analytics for Hadoop and Other Systems*, Publisher: O'Reilly Media.

It is all about storage and processing

1. How do we store Big Data? with a distributed file system
2. How do we process Big Data? with a parallel/distributed computing system/framework

Storage example

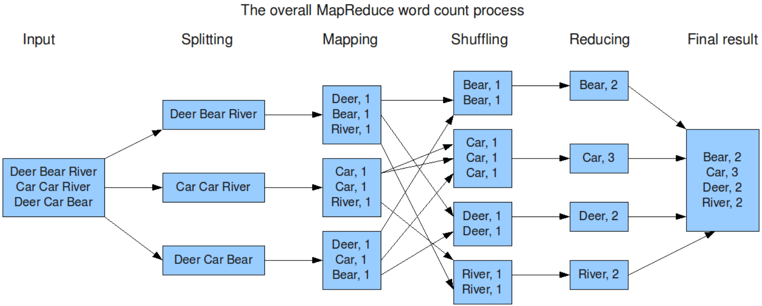
GFS (Google File System) Design



- Master manages metadata
- Data transfers happen directly between clients/chunk servers
- Files broken into chunks (typically 64 MB)

10

Map-Reduce by example



Map Reduce Steps

1. **Prepare the Map() input** – the "MapReduce system" designates Map processors, assigns the input key value K1 that each processor would work on, and provides that processor with all the input data associated with that key value.
2. **Run the user-provided Map() code** – Map() is run exactly once for each K1 key value, generating output organized by key values K2.
3. **"Shuffle" the Map output to the Reduce processors** – the MapReduce system designates Reduce processors, assigns the K2 key value each processor should work on, and provides that processor with all the Map-generated data associated with that key value.
4. **Run the user-provided Reduce() code** – Reduce() is run exactly once for each K2 key value produced by the Map step.

Another example - SQL case

Imagine that for a database of 1.1 billion people, one would like to compute the average number of social contacts a person has according to age.

```sql
  SELECT age, AVG(contacts)
    FROM social.person
GROUP BY age
ORDER BY age
```

Another example - Map Reduce case

```
function Map is
    input: integer K1 between 1 and 1100, representing a batch of 1 million social.pers
    for each social.person record in the K1 batch do
        let Y be the person's age
        let N be the number of contacts the person has
        produce one output record (Y,(N,1))
    repeat
end function

function Reduce is
    input: age (in years) Y
    for each input record (Y,(N,C)) do
        Accumulate in S the sum of N*C
        Accumulate in C_new the sum of C
    repeat
    let A be S/C_new
    produce one output record (Y,(A,C_new))
end function
```

The MapReduce System would line up the 1100 Map processors, and would provide each with its corresponding 1 million input records. The Map step would produce 1.1 billion (Y,(N,1)) records, with Y values ranging between, say, 8 and 103. The MapReduce System would then line up the 96 Reduce processors by performing shuffling operation of the key/value pairs due to the fact that we need average per age, and provide each with its millions of corresponding input records. The Reduce step would result in the much reduced set of only 96 output records (Y,A), which would be put in the final result file, sorted by Y.

Don't reduce to much and keep track of the right data

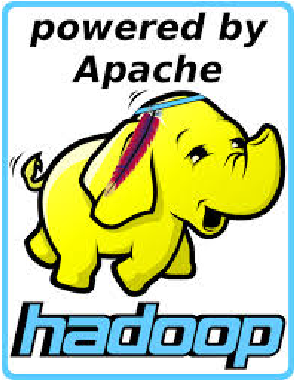
Suppose we don't keep track of the number of records (C)

```
-- map output #1: age, quantity of contacts
10, 9
10, 9
10, 9
```

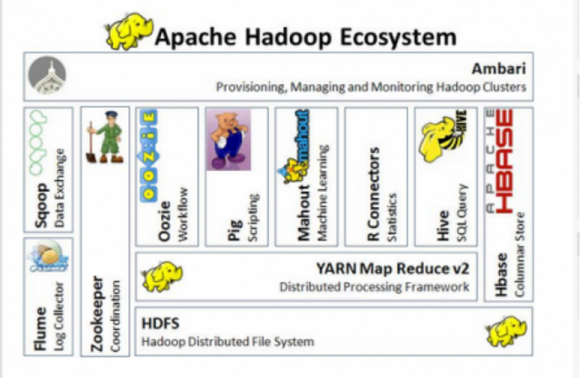
```
-- map output #2: age, quantity of contacts
10, 9
10, 9
```

```
-- map output #3: age, quantity of contacts
10, 10
```

Map Reduce, invented by Google now Open Source



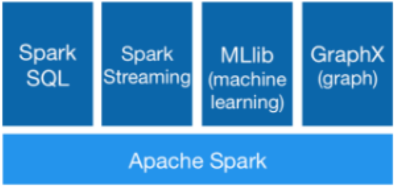
Hadoop Eco-System



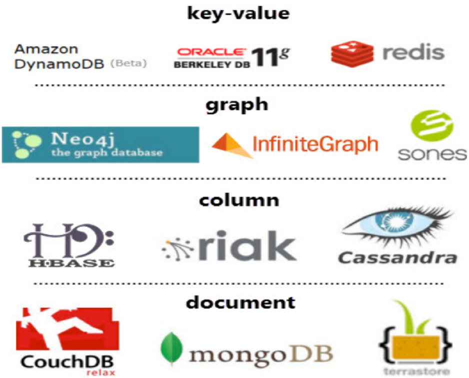
Spark

Spark highlights

Apache Spark is a fast and general engine for large-scale data processing. Run programs up to 100x faster than Hadoop MapReduce in memory, or 10x faster on disk.



No-SQL databases



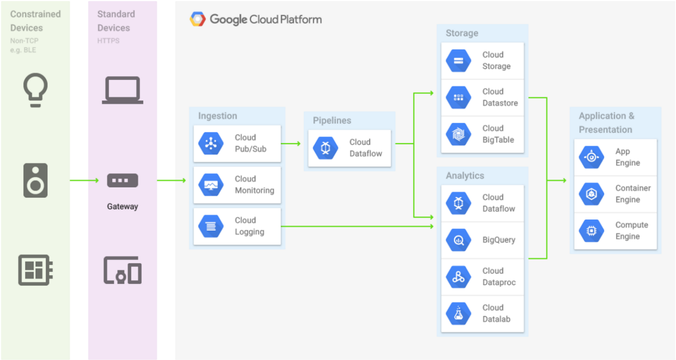
Column database

RowId	EmpId	Lastname	Firstname	Salary
001	10	Smith	Joe	40000
002	12	Jones	Mary	50000
003	11	Johnson	Cathy	44000
004	22	Jones	Bob	55000

```
001:10,Smith,Joe,40000;
002:12,Jones,Mary,50000;
003:11,Johnson,Cathy,44000;
004:22,Jones,Bob,55000;
```

```
10:001,12:002,11:003,22:004;
Smith:001,Jones:002,Johnson:003,Jones:004;
Joe:001,Mary:002,Cathy:003,Bob:004;
40000:001,50000:002,44000:003,55000:004;
```

Google


Constrained Devices
Non-TCP
e.g. BLE
Standard Devices
HTTPS
Gateway
Google Cloud Platform
Ingestion
Cloud Pub/Sub
Cloud Monitoring
Cloud Logging
Pipelines
Cloud Dataflow
Storage
Cloud Storage
Cloud Datastore
Cloud BigTable
Analytics
Cloud Dataflow
BigQuery
Cloud Dataproc
Cloud Datalab
Application & Presentation
App Engine
Container Engine
Compute Engine

References

ETSI TR 102 690: Machine-to-Machine communications (M2M); Functional architecture .
2013.

Mehdi Mohammadi Mohammed Aledhari Ala Al-Fuqaha, Mohsen Guizani and Moussa Ayyash.
Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications.
EEE COMMUNICATION SURVEYS & TUTORIALS, VOL. 17, NO. 4, FOURTH QUARTER 2015, 2015.

Gartner Alvaro Mello, VP Research.
Negocios Digitais na Era da Inovacao, 2014.

Carles Anton-Haro and Mischa Dohler, editors.
Machine-to-Machine (M2M Communications - Architectures, Performance and Applications.
Woodhead Publishing, 2015.

References (cont.)

- Hype Cycle for Emerging Technologies, 2015, Gartner.
- Daniel Kellmereit and Daniel Odovoski.
  The Silent Intelligence - The Internet of Things.
  DnD Ventures, 2013.
- LAN/MAN Standards Committee of the IEEE Computer Society.
  IEEE P802.15.4-REVc/D02^{TM} Draft Standard for Low-Rate Wireless Personal Area Networks (WPANs).
  2015.
- Gregorio López, Pedro Moura, José Ignacio Moreno, and José Manuel Camacho.
  Multi-faceted assessment of a wireless communications infrastructure for the green neighborhoods of the smart grid.
  Energies, 7(5):3453–3483, 2014.

References (cont.)

Geoff Mulligan.
The 6LoWPAN Architecture.
In *Proceedings of the 4th Workshop on Embedded Networked Sensors*, EmNets '07, pages 78–82, New York, NY, USA, 2007. ACM.

Z. Shelby and C. Bormann.
6LoWPAN: The Wireless Embedded Internet.
Wiley Series on Communications Networking & Distributed Systems. Wiley, 2011.

J.P. Vasseur and A. Dunkels.
Interconnecting Smart Objects with IP: The Next Internet.
Elsevier Science, 2010.