# ARCHITECTURE

PHILIPPE LALANDA

KOBE UNIVERSITY – AUGUST 2017

# Purpose of this lecture

- Present the global architecture of pervasive systems
- Describe the different components

# Agenda

Introduction – Global architecture

Smart devices layer

Gateway layer

Cloud layer

Architectural variability

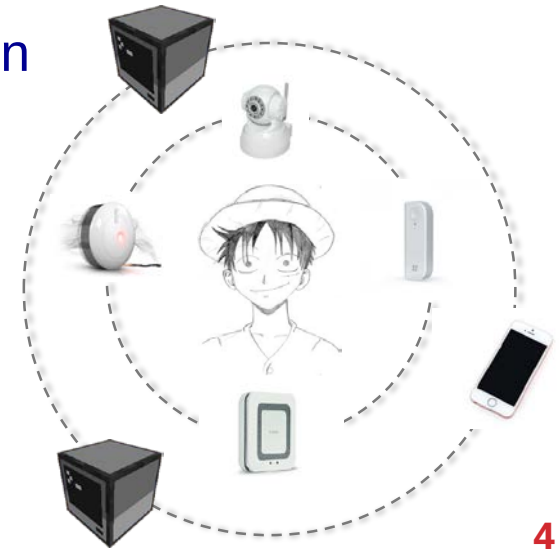
Challenges and conclusion

# Pervasive computing

Pervasive computing promotes the integration of smart, networked devices in our living environments in order to provide us services.

Those services

- are context aware
- require minimal and natural interaction
- bring real added value
- are easy to administrate by end-users



# Software complexity

**Implementing such pervasive services is very challenging. It requires to**

- get contextual information from real world
- run reactive services
- run more deliberative services
- store data for analysis and learning
- ensure non functional properties
- be secure and ensure privacy
- etc.

# Software architecture

**This requires to build and manage complex architectures, mixing software, networks, and electronics**

**Software architecture**

- describes the high level structures of a software system in terms of <u>components</u> and <u>connectors</u> between these components
- connectors are used to transmit domain data and control information

# Software architecture

**Software architectures specify**

Where are the main computing elements?

Where are the main data sources?

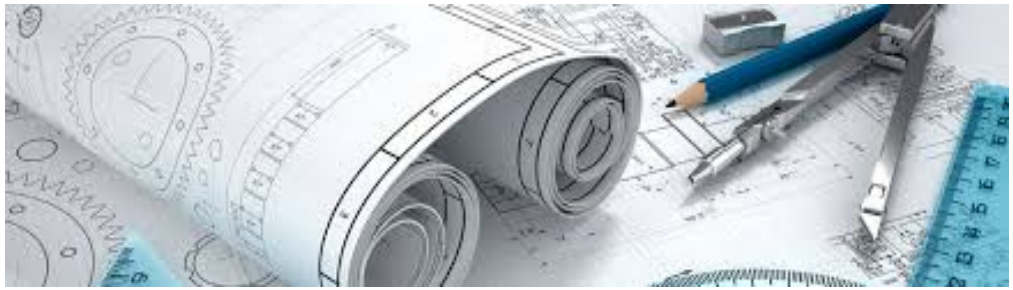
What are the main control flows?

What are the critical paths and the performance criteria?

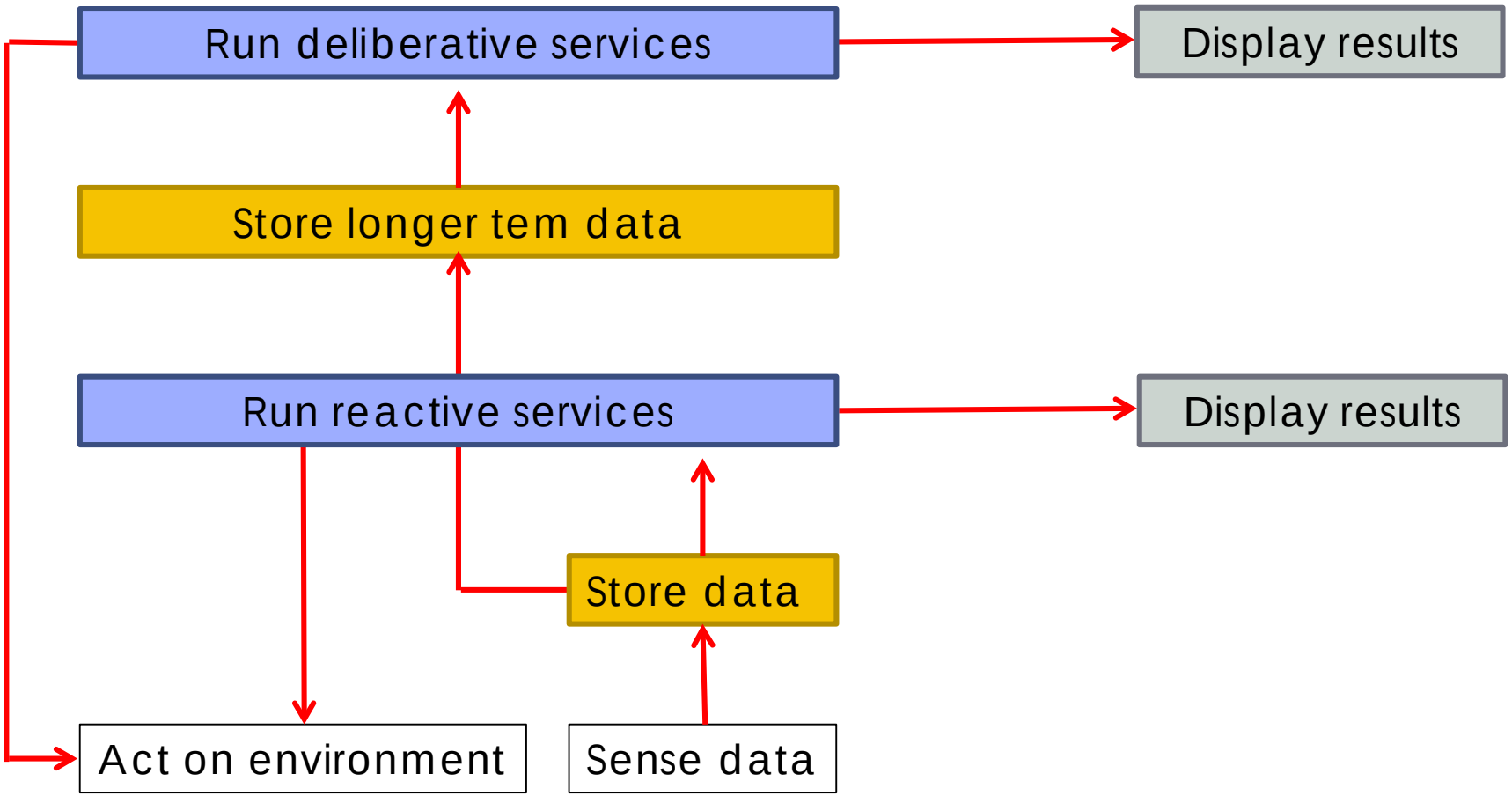
What are the needs in terms of computing platforms?

What are the levels of coupling and cohesion?

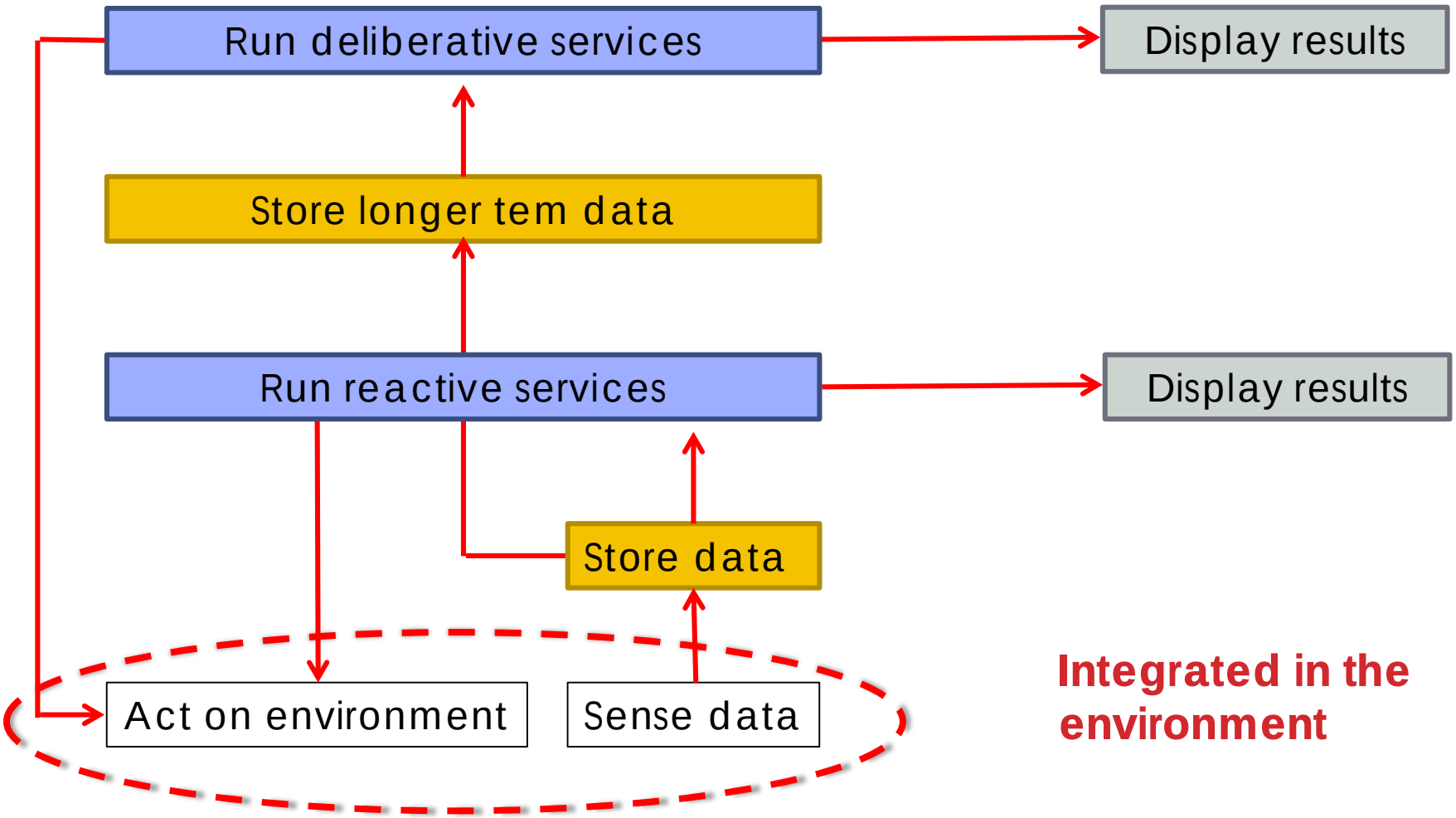
**A good practice to start designing is to identify the main functions**



# Pervasive computing – main functions

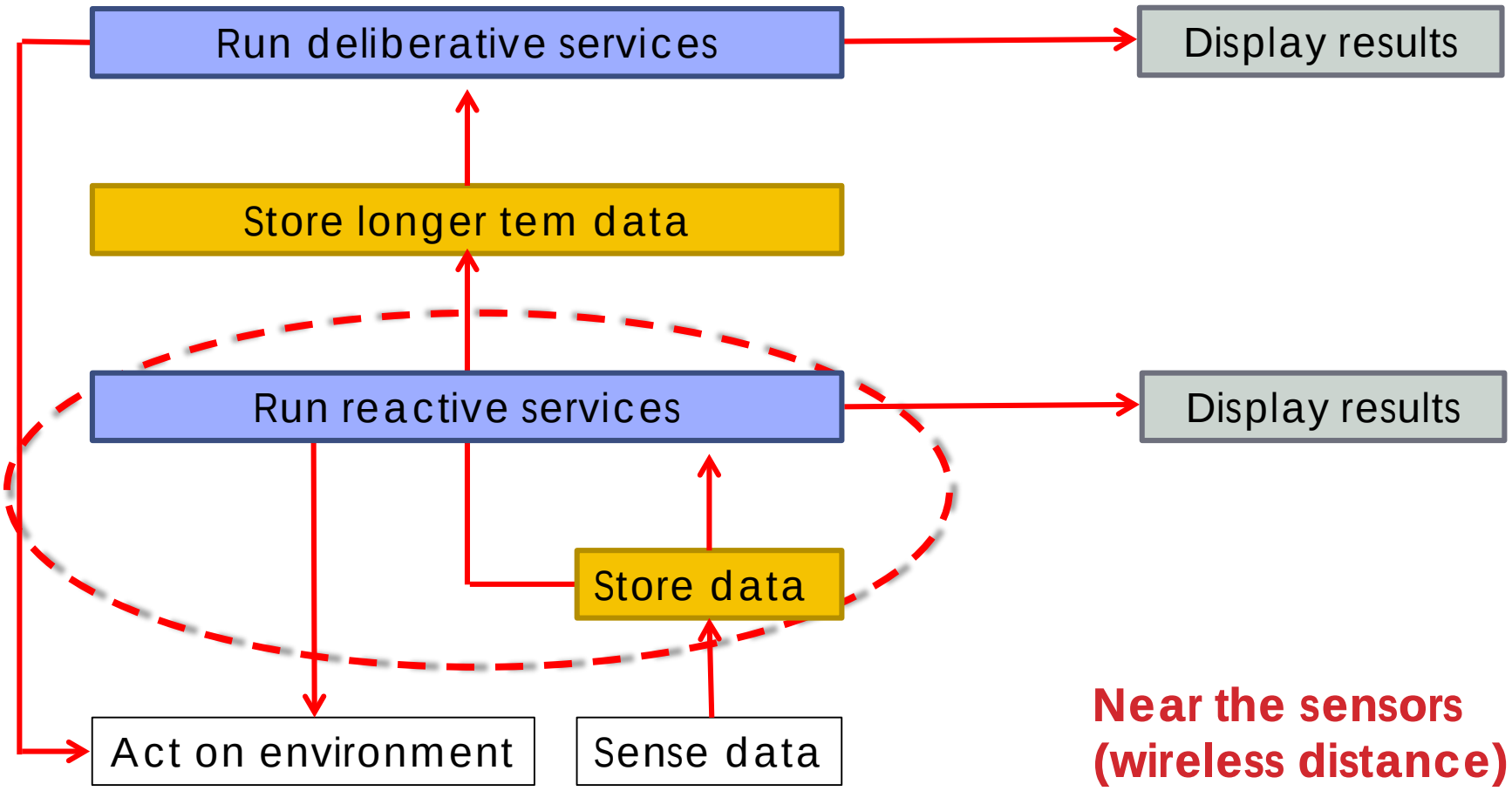


# Contraints



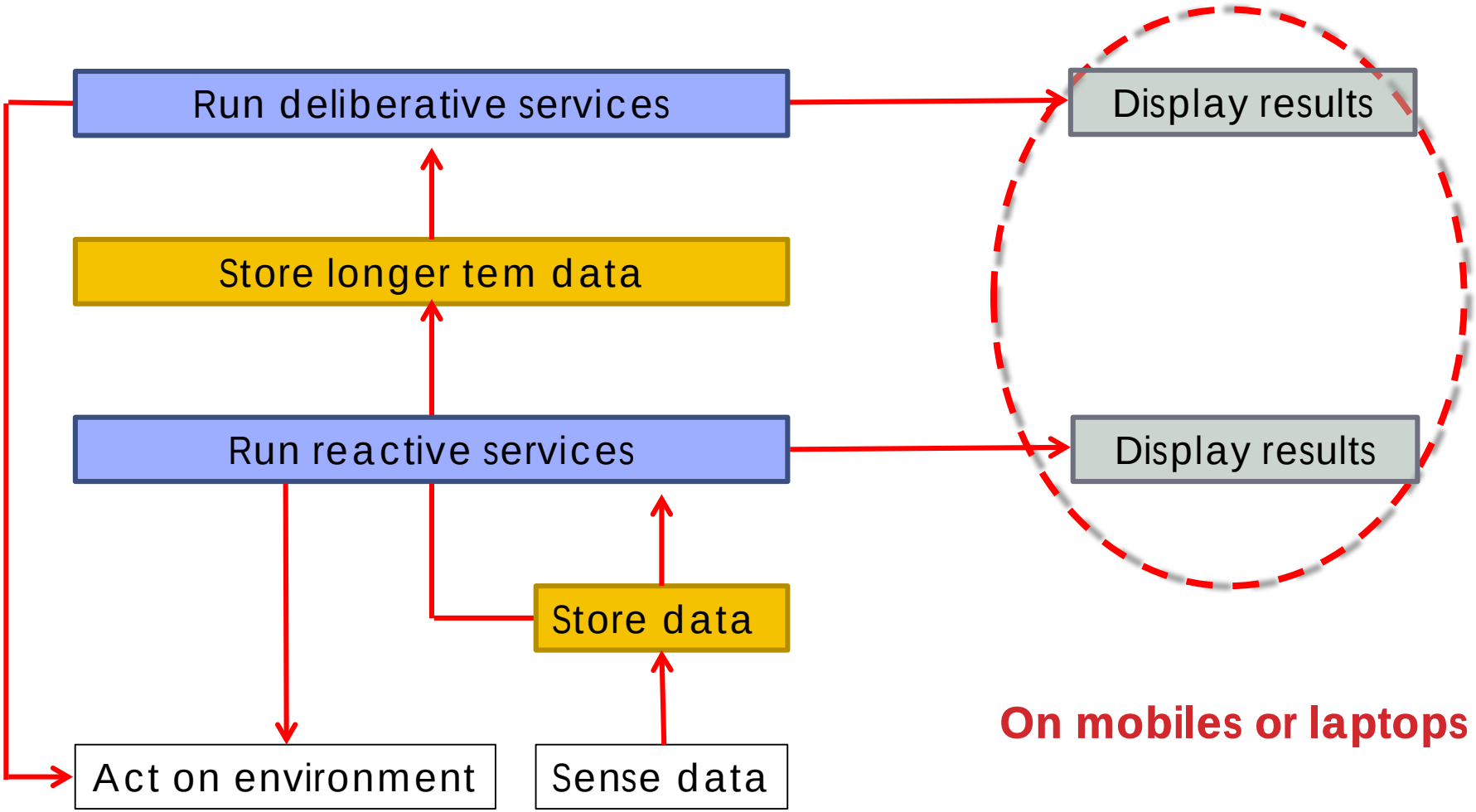

**Integrated in the environment**

# Contraints



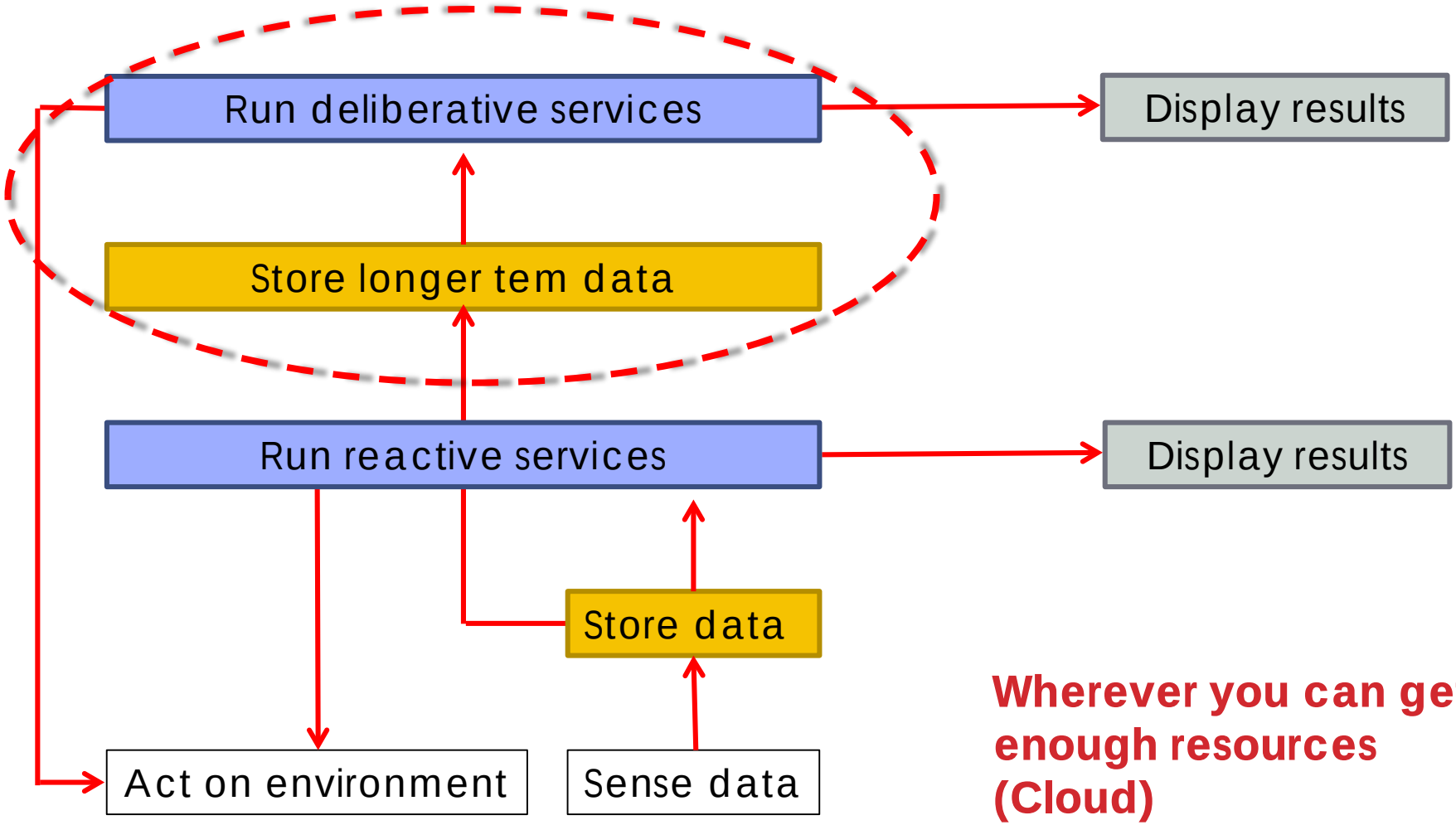

**Near the sensors (wireless distance)**

# Contraints



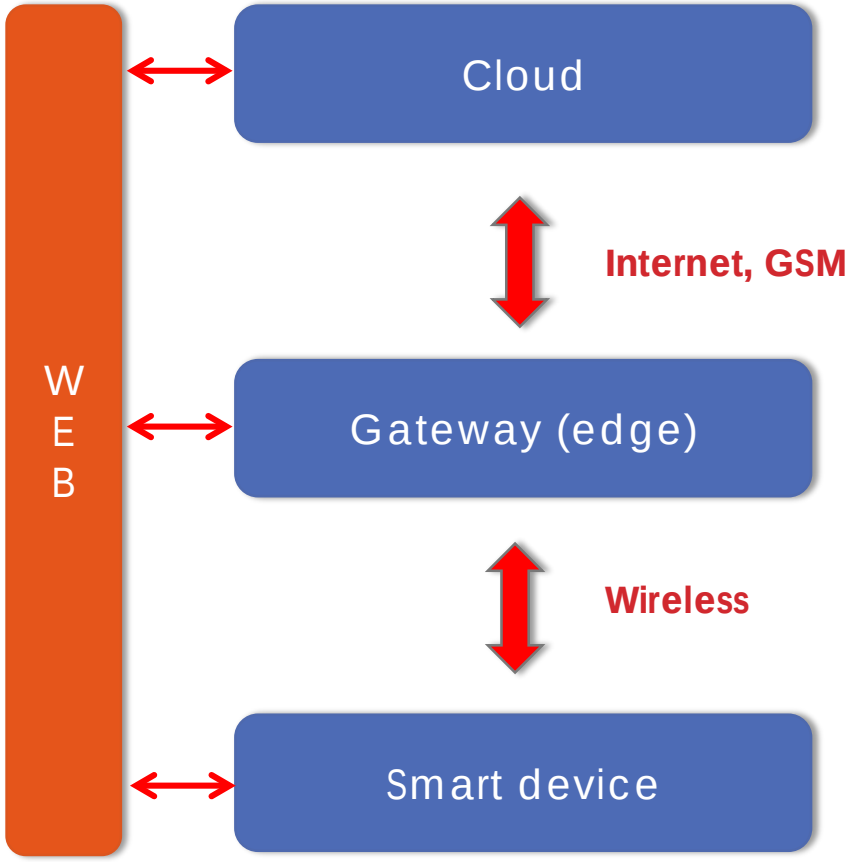

**On mobiles or laptops**

# Contraints




**Wherever you can get enough resources (Cloud)**

# High level architecture




This layer stores and uses to the data collected from all layers. It provides more deliberative services based on big data, learning, advanced processing.

This layer runs local, real-time services and enables the stream of data to move from one level to the next for more processing.

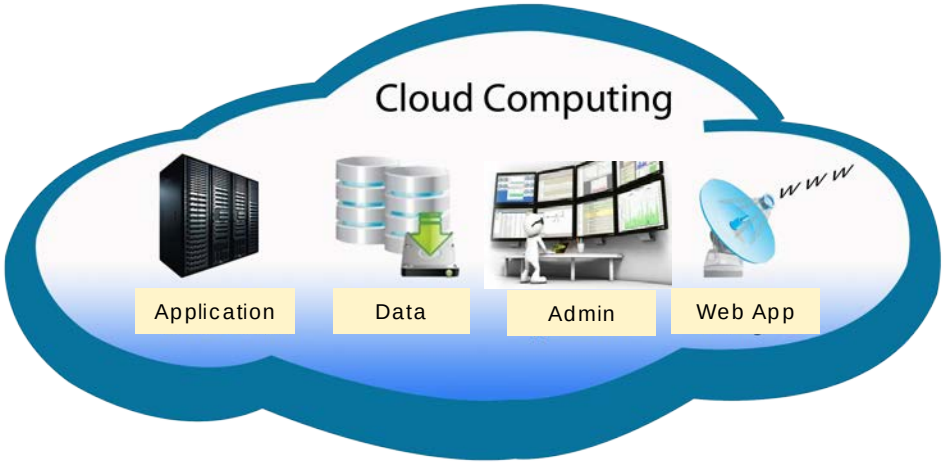
This layer in made of objects integrated in our physical environment, generating a stream of relevant data.

Web presentation, multi-modalities, additional services (WS).

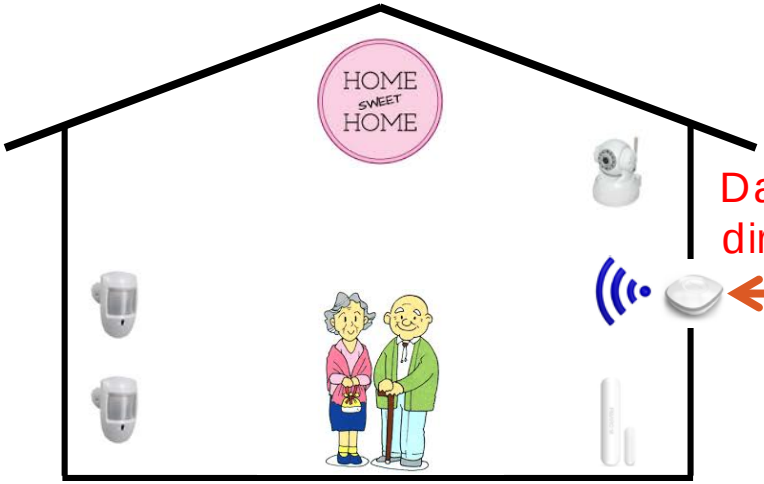
# Example: Activity tracking at home

Activity tracker

Web presentation




Deliberative services
Data storage
Administration




Smart devices providing contextual information

Data re-direction

Gateway running reactive services (fall detection)

# Software based

**This architecture is very challenging regarding software**

most non functional aspects are treated at the software level

**Focus of next lecture**

**For now, let us focus on**

business value is in software

description of each layer

main components of each layer

# Agenda

Introduction – Global architecture

Smart devices layer

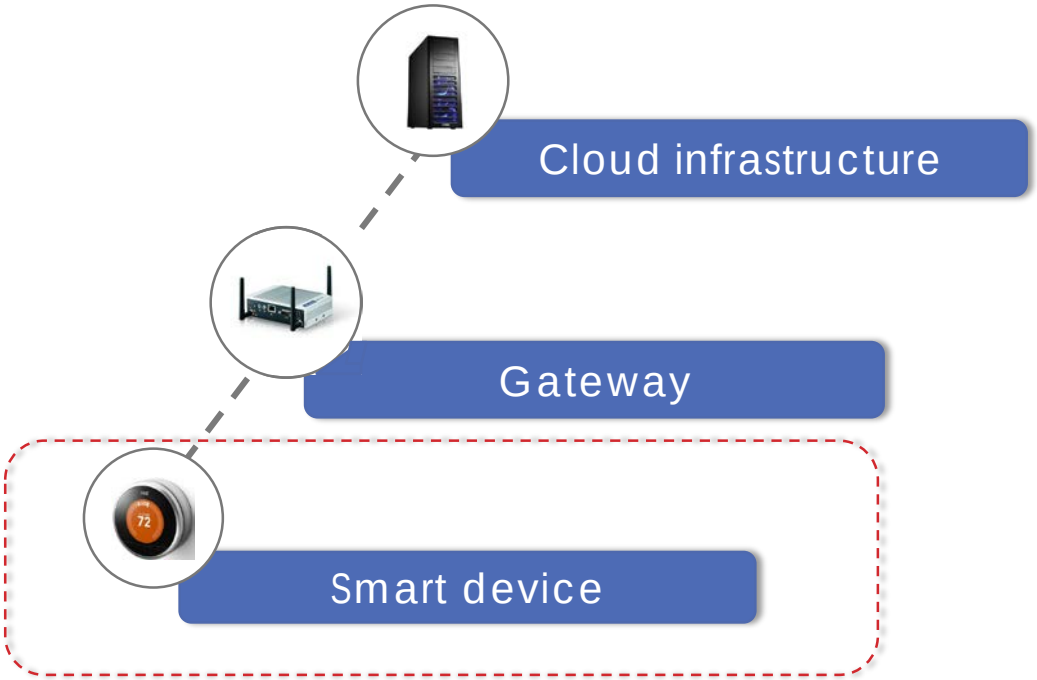
Gateway layer

Cloud layer

Architectural variability

Challenges and conclusion

# Smart devices layer




This layer in made of objects integrated in our physical environment, generating a stream of relevant data.

# Smart devices - definition

**The first layer is made of a number of smart, communication-enabled devices connected via wireless protocols to**

the gateway layer

possibly to each other (device-to-device)

Smart device = sensing + communicating capabilities

Smoke and CO2 detector

Cam indoor


Learning thermostat

Smart fridge

# Smart devices

**Smart devices can be installed everywhere in the physical environment**

in a door, in a field, in a tree, ...

**They can be mobile**

smart clothes, smart phones

**They depend on**

software (to prune/aggregate/smooth captured data)

electricity (battery or main)

# Smart devices - functions

**Main functions:**

Get data from the environment

Process and store data

Send data to IoT gateways or other devices

Receive information from IoT gateways or other devices

Panasonic smart devices for home

# Sensing function

Acoustic, sound, vibration

Thermal, heat, temperature

Humidity, moisture, barometric pressure

Light, images

Fluid velocity and flow

Position, displacement

Speed, acceleration, angle

Proximity, presence

Electric current, potential

Radiation, chemical

...

Bundle of popular sensors (for Arduino)

# Communication function

**Main functions:**

Allow data transmission between devices/computers

Domain data and control data

quality of service

# Networking

## LAN
**Short range**

Good for:

- Buildings, plants
- Mobile
- Cost

Not good for:

- Battery life
- Wide area

## Cellular/Satellite
**Long range**

Good for:

- Coverage
- High data range

Not good for:

- Battery life
- cost

## Low Power WAN
**Long range**

Good for:

- Coverage
- Battery life
- Cost

Not good for:

- High data rate

# Selecting a smart device

**Smart devices must meet applications requirements**

- many smart devices around (some are multi-sensors)
- varying technical and functional properties
- getting the right ones is a crucial step

**Aspects to consider**

- accuracy
- reliability
- durability
- consumption
- platform (Arduino: 5V sensor is more convenient than 3V3)
- Aesthetics

# Accuracy – health application

**Heart beat rate**


**+/- 1**


**+/- 5**

**+/- 20**

# Reliability – temperature application

**Magnetic sensibility**

Smart factory

Smart building

# Durability – Fire management application

**Moisture sensibility**

Tropical forest

Smart house

# Consumption – parking help application

**Accessibility**

In the concrete

In the ceiling

# Aesthetic and comfort

**Heart beat rate**

Cardio strap

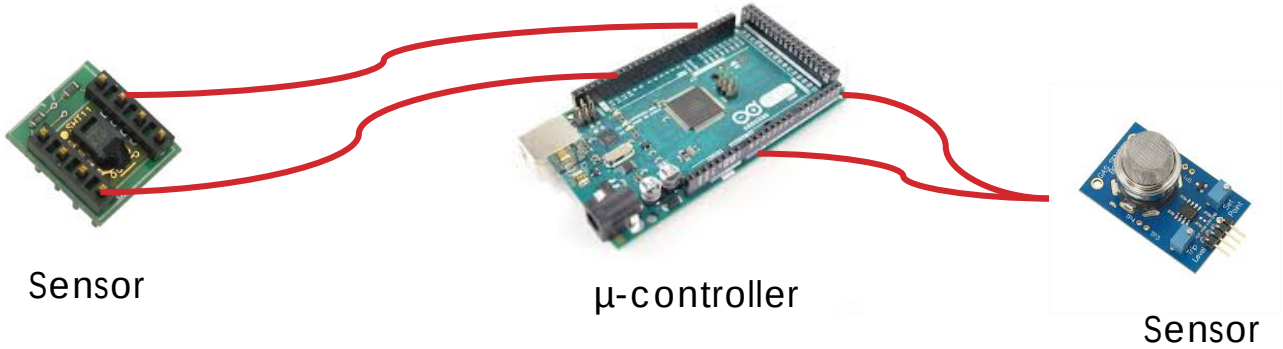

Cardio watch

# Building smart devices

**Smart device = (sensors + micro-controller)**

sensors deal with the capture of environmental information

micro-controller deals with computing, storage and networking




**Note**: A µ-controller is a small computer on a single integrated circuit.
It includes one or more CPU, memory, peripherals and network chips.
Low consumption. Cheaper, slower than multipurpose micro-processors.

# The ARDUINO example

**Single-board microcontroller**

- Open source
- Equipped with sets of digital/analog I/O
- Several versions
- Development kit (C)

Plug sensors

Program a function based on collected data

USB connection (to dev. PC)

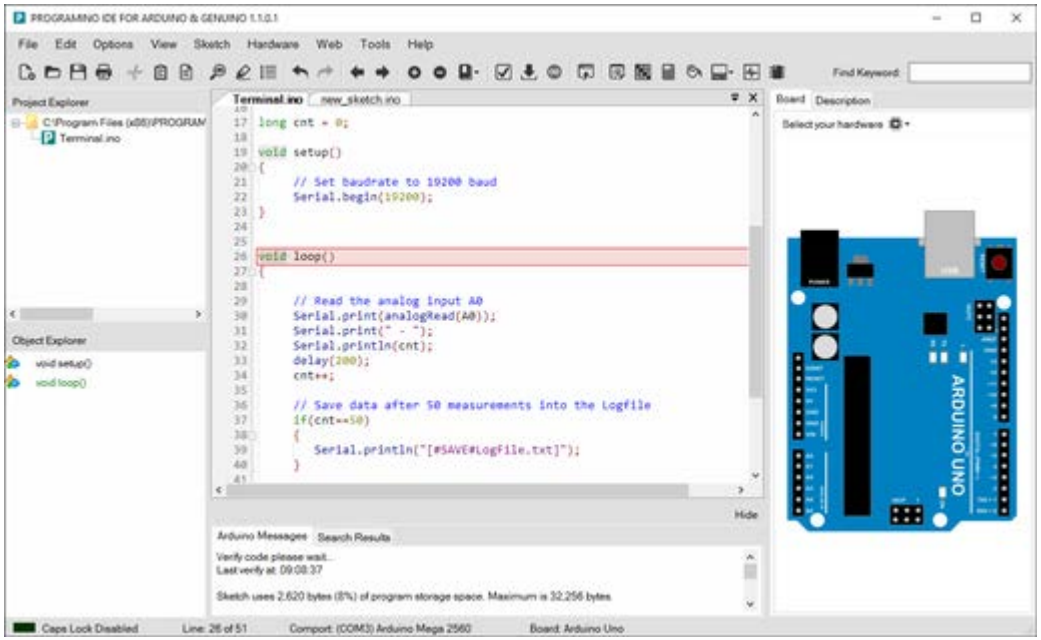
# The ARDUINO programming environment

**A usual development environment**

C programming language (with restrictions)

loop-based: read/process/store

easy but requires specific skills



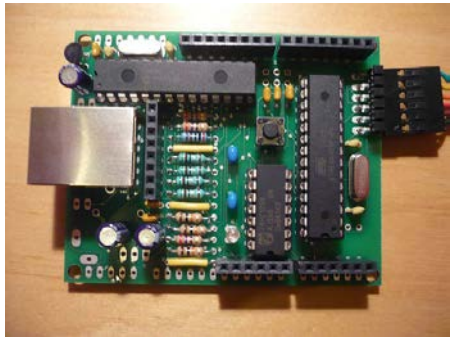
# ARDUINO products

| | |
|---|---|
| ENTRY LEVEL<br>$25<br>($10) | UNO, LEONARDO, 101, ROBOT, ESPLORA, MICRO, NANO, MINI<br>MKR2UNO ADAPTER, STARTER KIT, BASIC KIT, LCD SCREEN |
| ENHANCED FEATURES<br>30$<br>($12) | MEGA, ZERO, DUE, MEGA ADK, PRO, M0, M0 PRO, MKRZERO, PRO MINI<br>MOTOR SHIELD, USB HOST SHIELD, PROTO SHIELD, MKR PROTO SHIELD, 4 RELAYS SHIELD<br>MEGA PROTO SHIELD, MKR RELAY PROTO SHIELD, ISP, USB2SERIAL MICRO<br>USB2SERIAL CONVERTER |
| INTERNET OF THINGS<br>60$ | YÚN, ETHERNET, TIAN, INDUSTRIAL 101, LEONARDO ETH, MKRFOX 1200, MKR1000<br>YUN MINI, WIFI SHIELD, WIFI 101 SHIELD, YÚN SHIELD, WIRELESS SD SHIELD<br>WIRELESS PROTO SHIELD, ETHERNET SHIELD V2, GSM SHIELD V2, MKR1000 BUNDLE |

# Alternatives


LaunchPad MSP430
Texas Instrument
$4

(for simple work)



Nanode
Ethernet connection
Compatible with Arduino
$55

(for web-based sensors)

STM 32 Discovery
ST Microelectronics
$10

(more powerful)

# Conclusion about smart devices

**There are hundreds of smart devices on the market**

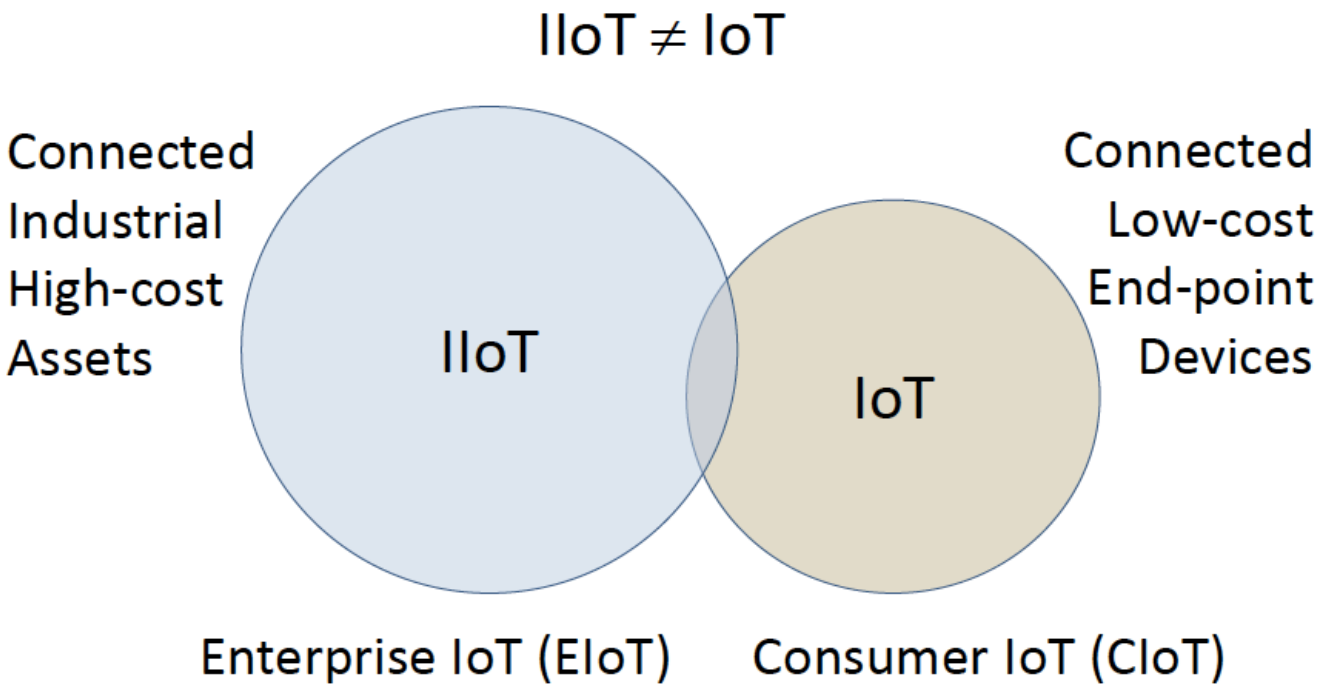
- different sensors and µ-controllers
- different prices and qualities
- different networking choices

**Challenges**

- getting more and more accurate and reliable
- lower energy

→ **Good devices are essential**

# IIoT versus IoT



# Agenda

Introduction – Global architecture

Smart devices layer

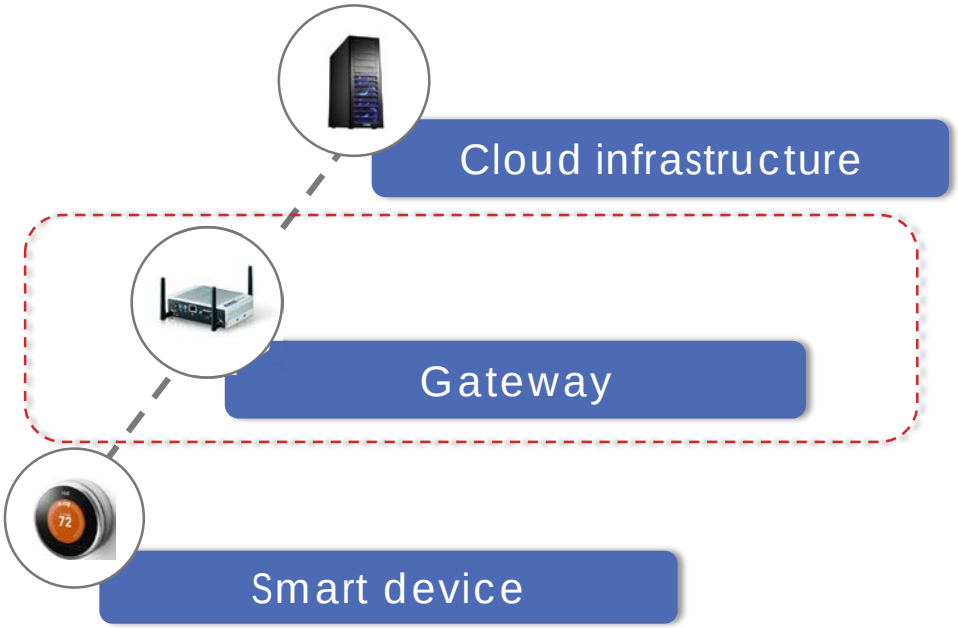
Gateway layer

Cloud layer

Architectural variability

Challenges and conclusion

# Smart devices layer



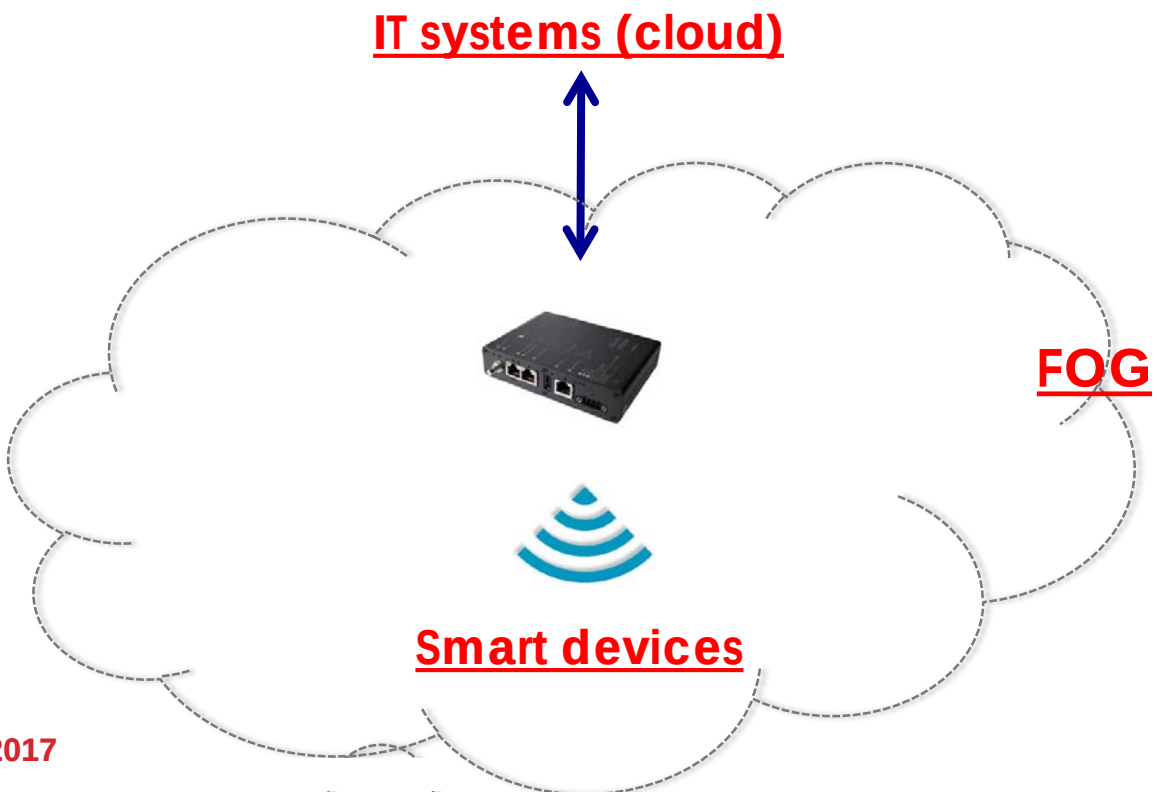

This layer runs local, real-time services and enables the stream of data to move from one level to the next for more processing.

# IoT gateways - definition

**Electronic device serving as a connection point between pervasive devices (floor) and IT systems (cloud)**

a crucial element to handle the explosion of devices

and to implement real-time services



# IoT gateways - functions

**Main functions:**

Implement multiple protocols

device connectivity (ZigBee, Bluetooth, Ethernet, LoRa,...)

Internet connectivity(Web, Rest, MOM)

Gather, aggregate, process data from devices

Receive information/requests from the Cloud


DELL edge gateways

# Fog computing

## Traditional model

Data centers, clouds

Assumes infinite bandwidth and no delay

Endpoints

## Pervasive model

Data centers, clouds

Assumes limited bandwidth, possible delays, intermittent connectivity

Fog (gateways)

Assumes limited bandwidth, possible delays, intermittent connectivity

Devices

# Benefits of fog computing (gateways)

**Security**

- protect data moving to the cloud from leaks
- prevent control of devices from outside parties

**Responsiveness**

- less networking delays

**Scalability**

- Data grows faster than bandwidth

**Resiliency**

- can resist to network problems (though caching for instance)

# Smart IoT gateways

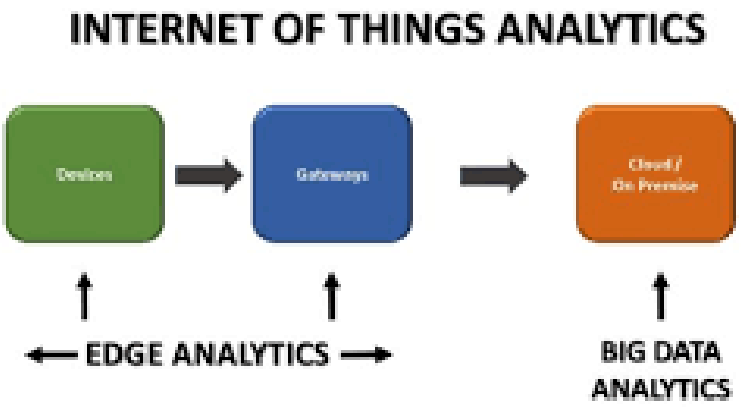
**Traditional IoT gateways are not very smart**

aggregate and send data

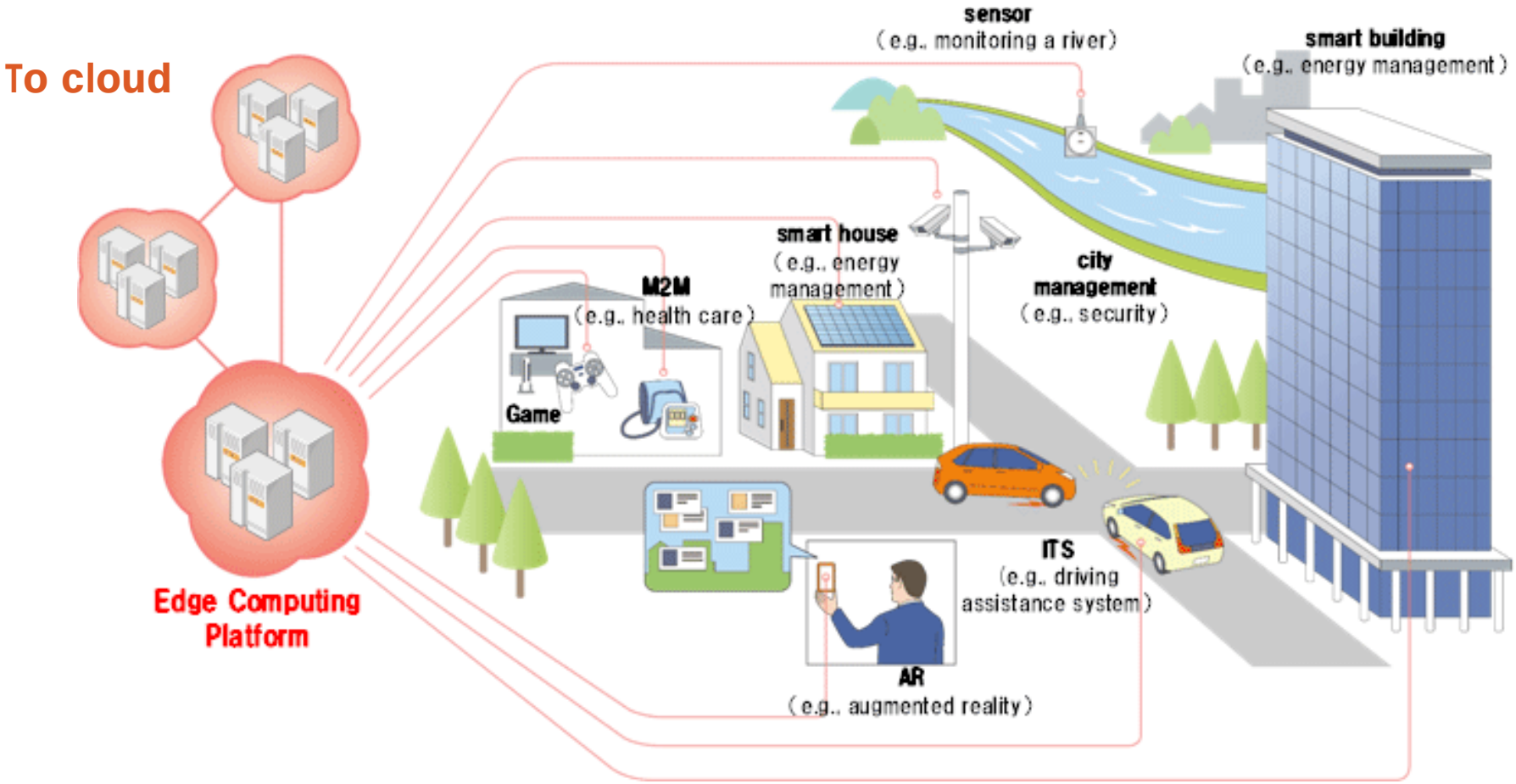
perform simple services

**They are getting smarter**

performing edge analytics before pushing data to the cloud



# Edge analytics is the future

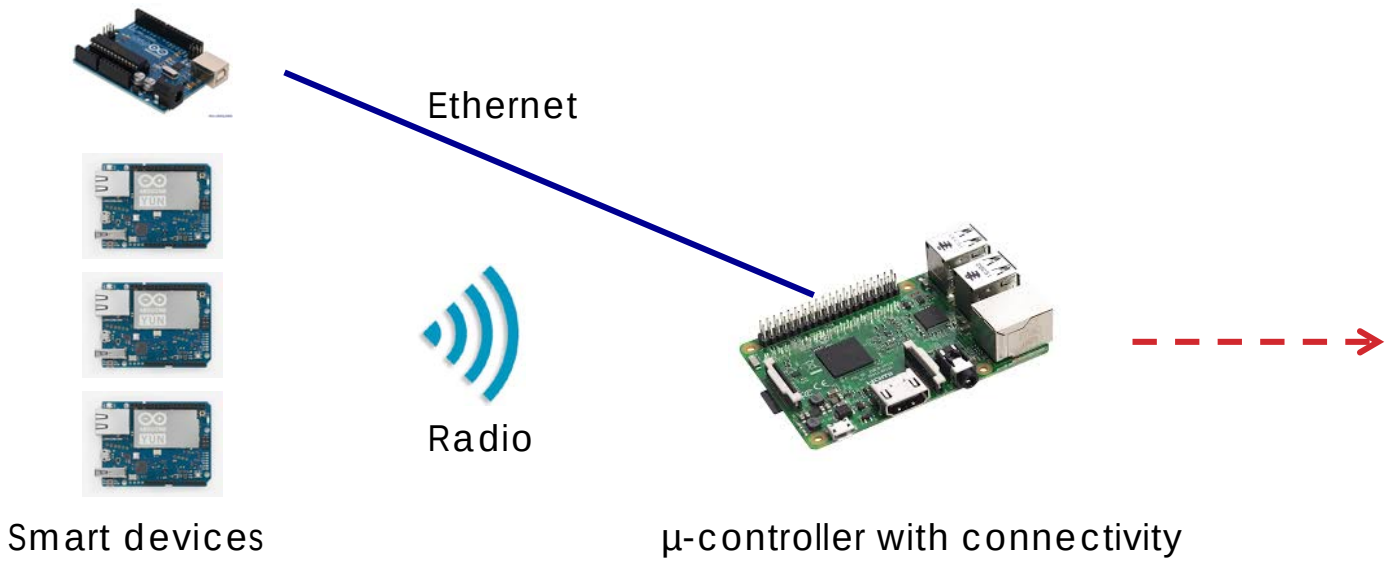


# Building a pervasive gateway

**Gateway= (micro-controller + network chips)**

network chips with multiple connectivity

micro-controller deals with computing and storage



# The Raspberry Pi example

**Single-board microcontroller**

runs Linux

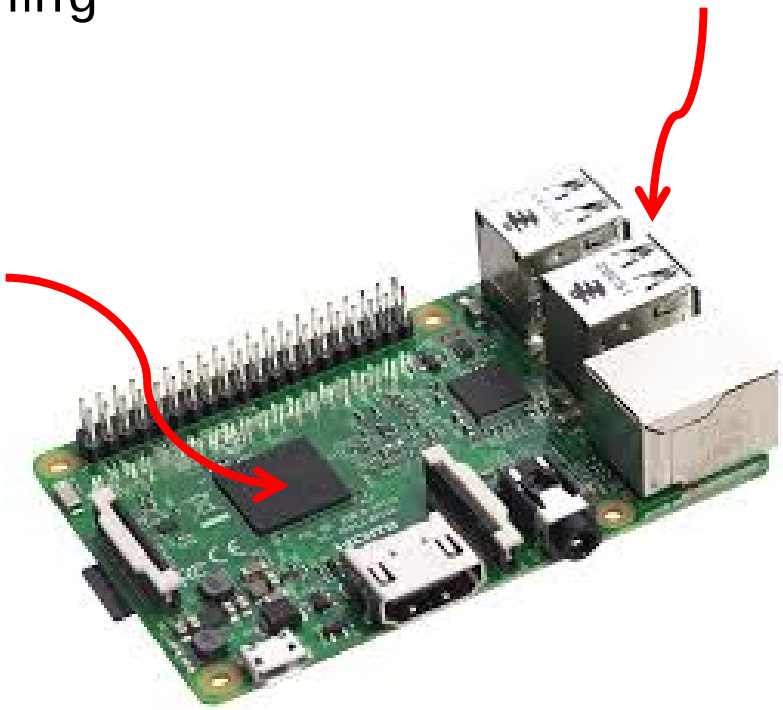
Full networking system

software oriented programming

Development kit

Networks

Program advanced function

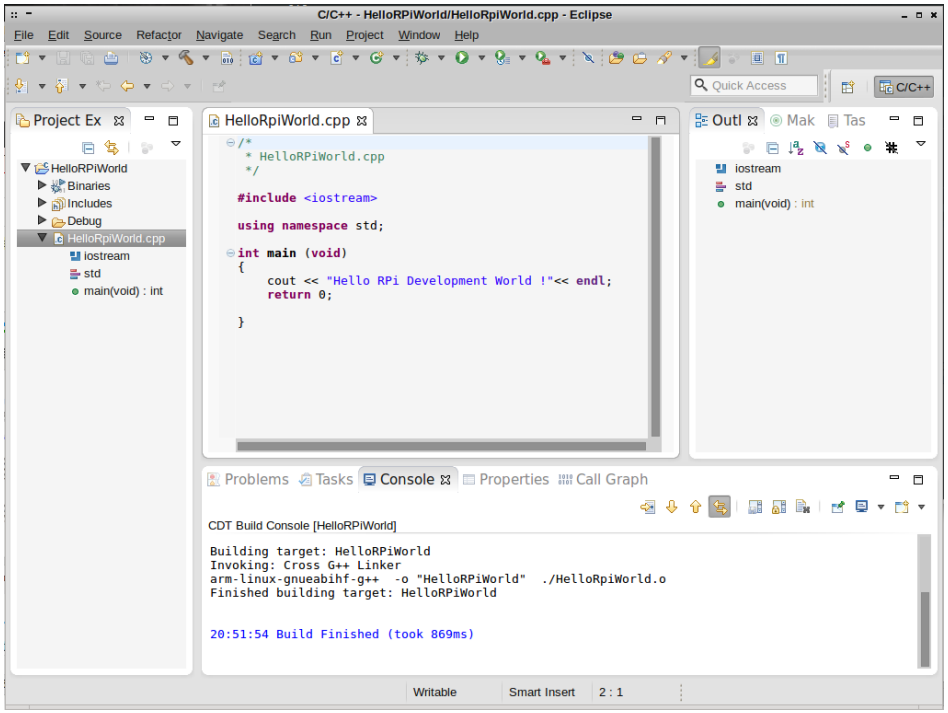


# The Raspberry Pi programming environment

**Software-oriented development environment**

C/C++, Java

modern tools (Eclipse, etc.)



# Alternatives

Raspberry Pi
Intel
$35

900 MHz ARM
Ethernet
1 GB RAM


Edison
Intel
$70

1 GHz Dual core
WiFi
1 GB RAM
4 GB Flash

XU4
Odroid
$75

1 GHz ARM
Ethernet
2 GB RAM
4 GB Flash

# Smart devices from a software perspective

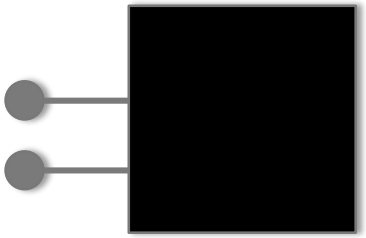
Program

```
end Calculator()
    end addition(self, other)
        return self + other
    end

    /** Create a list of 2 numbers. */
    end makeFraction(numerator, denominator)
        return [numerator, denominator]
    end

    /** Warning: Destroys original fraction! */
    end multiplyFrac(frac, otherFrac)
        frac[0] *= otherFrac[0]
        frac[1] *= otherFrac[1]
        return frac
    end
end

end InfinityCalculator()
    inherit Calculator()
    /** Create a list of 2 numbers. */
    end makeFraction(numerator, denominator)
        if denominator == 0
            /* The user is trying to divide by 0.
             * Use Java's way of handling this: */
            import math into mathematics
            return mathematics.INFINITY
        end
        return [numerator, denominator]
    end
end
```

Method

Class

Smart device



Availability?
Sleeping/Sleeping/waking cyle?
API?
Events?
Semantics?
...

# Conclusion about IoT gateways

**High fragmentation in the gateway market**

- specialized in domains
- specialized in networks (IoRa vs. Sigfox for instance)

**Challenges**

- get smarter
- get bigger
- get secure

**Constant progress but high instability**

# Agenda

Introduction – Global architecture

Smart devices layer

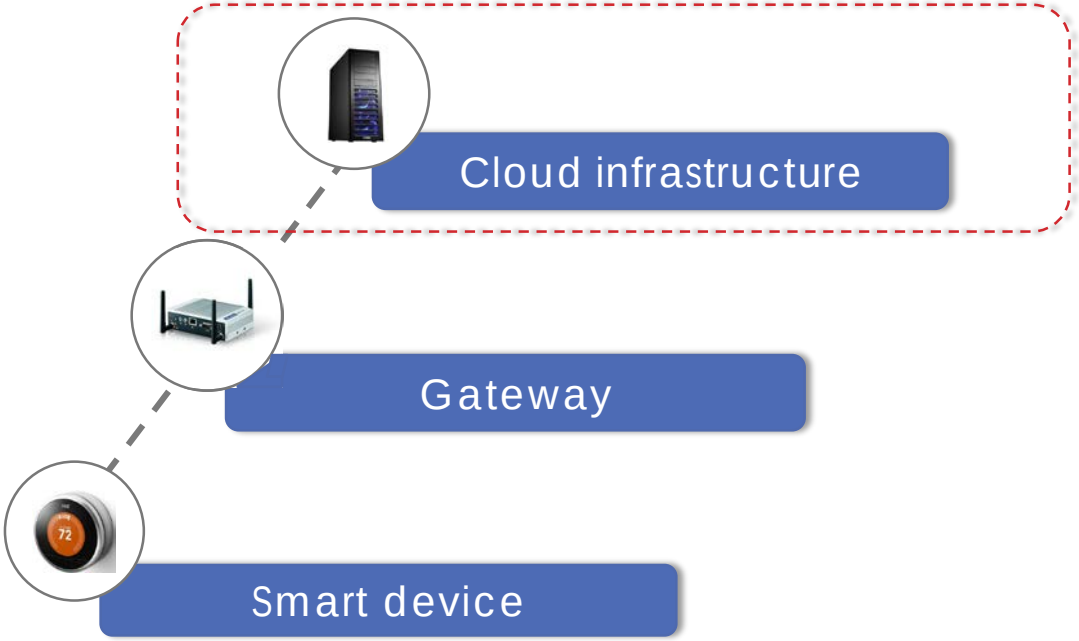
Gateway layer

Cloud layer

Architectural variability

Challenges and conclusion

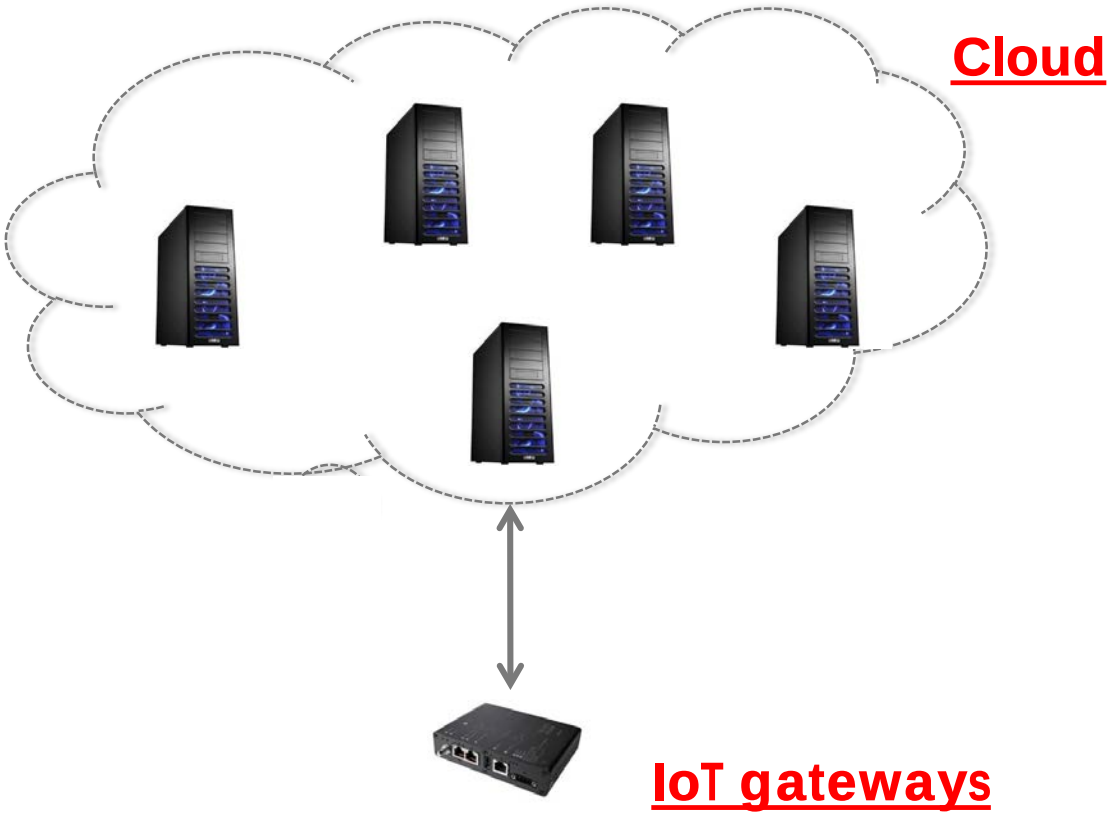
# Smart devices layer




This layer stores and uses to the data collected from all layers. It provides more deliberative services based on big data, learning, advanced processing.

# Cloud computing - definition

**The practice of using a network of remote servers hosted on the Internet to store, manage, and process data.**



# Cloud computing - functions

**Main functions:**

Internet connectivity

Receive data from gateways

Send information/requests to gateways

Store data

Analyze data

Provide added-value services

The cloud infrastructure also provides good scalability.

# Benefits of cloud computing

**Core advantages:**

Power

Speed

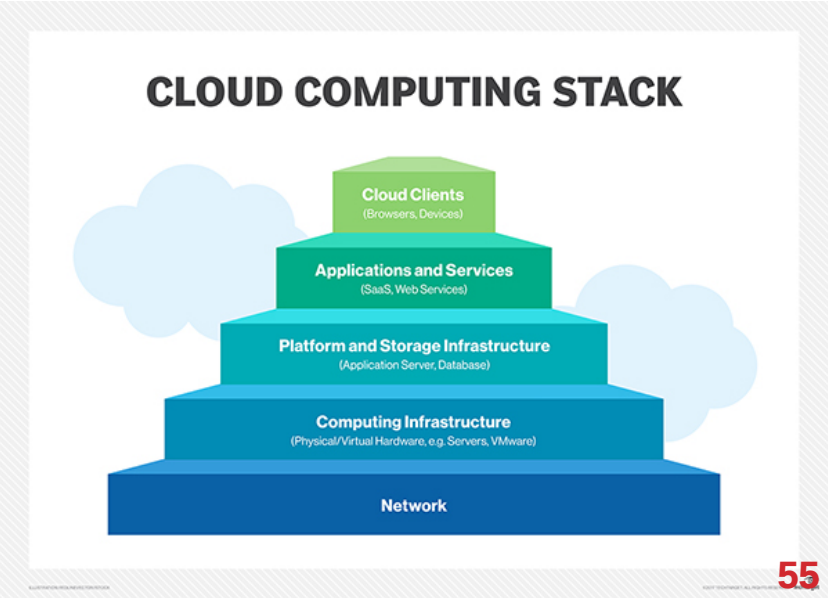
Cost

Infrastructure management

Deployment

Continuous enhancement

Easier administration



# Big data

**A special kind of pervasive service**

- collection of data
- use of analytics algorithms
- parallel processing

**Allow to get expected and unexpected findings.**

**Very greedy algorithms**

# Example - IBM Watson

# Conclusion about cloud computers

**Cloud infrastructure provide**

- computing and storage facilities
- advanced services (complex algorithms)
- advanced infrastructure
- advanced management tools

**Questions**

- what to do with the data?
- which data should be stored?

**Anarchic today?**

# Agenda

Introduction – Global architecture

Smart devices layer

Gateway layer

Cloud layer

Architectural variability

Challenges and conclusion

# Architectures are big

| Size | Duration | Programmers | LOC | Examples |
|---|---|---|---|---|
| Very small | 4 months | 1 | 2000 | Course project |
| Small | 2 years | 3 | 50K | Pace maker |
| Medium | 3 years | 10 | 100K | Optimizing compiler |
| Large | 5 years | 100 | 1M | MS Word, Excel |
| Very large | 10 years | 1000 | 10M | Air traffic control<br>Space shuttle |
| Very very large | 15+ years | 1000 | 35M | W2K |
| Ultra large | ? | ? | ? | Pervasive computing<br>Connected health |

# Big diversity

**There are multiple variations for this architecture**

networks, devices, environment, ...