The Internet/Web of Things

Niels Olof Bouvin

Overview

- **What is the Internet of Things?**
- The vision
- Domains of the Internet of Things
- The challenges of the Internet of Things
- The Proto Web of Things
- The challenge of the Internet of Things
- The Web as IoT architecture
- Milestone 1: meet the Raspberry Pi!

What is the Internet of Things?

- (CERP-IoT 2009): *"Internet of Things (IoT) is an integrated part of Future Internet and could be defined as a dynamic global network infrastructure with self configuring capabilities based on standard and interoperable communication protocols where physical and virtual* **'things' have identities, physical attributes, and virtual personalities and use intelligent interfaces, and are seamlessly integrated into the information network.** *In the IoT, 'things' are expected to become active participants in business, information and social processes where they are enabled to interact and communicate among themselves and with the environment by exchanging data and information 'sensed' about the environment, while reacting autonomously to the 'real/physical world' events and influencing it by running processes that trigger actions and create services with or without direct human intervention. Interfaces in the form of services facilitate interactions with these 'smart things' over the Internet, query and change their state and any information associated with them, taking into account security and privacy issues."*

Many definitions for IoT

- **The link between the real and the digital world**
- **Machines talking to machines (M2M)**
- **Everyone and everything connected via the Internet**
- ***The Internet of Things is a system of physical objects that can be discovered, monitored, controlled, or interacted with by electronic devices that communicate over various networking interfaces and eventually can be connected to the wider internet.*** **[Guinard & Trifa, eds.]**

Constituent parts of the Internet of Things

- Identity
- Connectivity
- Capability

Identity

- **Primary requirement**
- **Scannable ID, e.g., RFID, barcode, QR-code, etc**
  - cheap, often limited, "dumb" objects
- **Inherent ID, e.g., MAC address (WiFi, Bluetooth LE, etc), assigned identity**
  - more expensive, more capable, "smart" objects

Connectivity

- How can we address the object?
- IR, Bluetooth (LE), Zigbee, WiFi, Thread, etc.
- Internet Protocol, or more specialised protocols (for resource constrained devices)

Capability

- What can the object *do*?
- Simple: Identification
- Intermediate: Sensing
- Advanced: Reacting

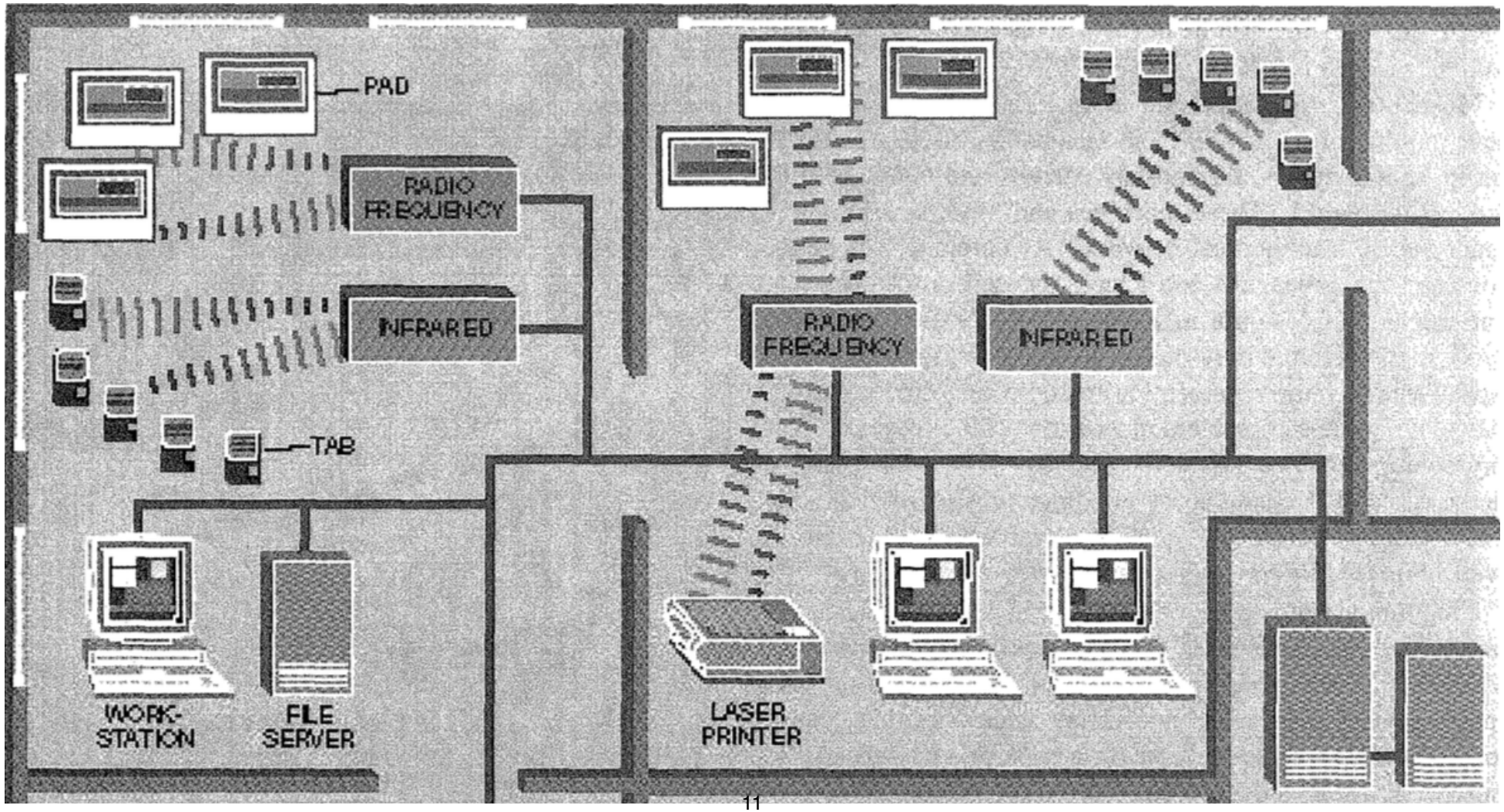
Overview

- What is the Internet of Things?
- **The vision**
- **Domains of the Internet of Things**
- **The challenges of the Internet of Things**
- **The Proto Web of Things**
- **The challenge of the Internet of Things**
- **The Web as IoT architecture**
- **Milestone 1: meet the Raspberry Pi!**

Towards the Internet of Things

Visions for the Internet of Things

- **Mark Weiser: The computer for the 21st century (1991)**
  - a paper that would herald what came to be known as "pervasive computing"



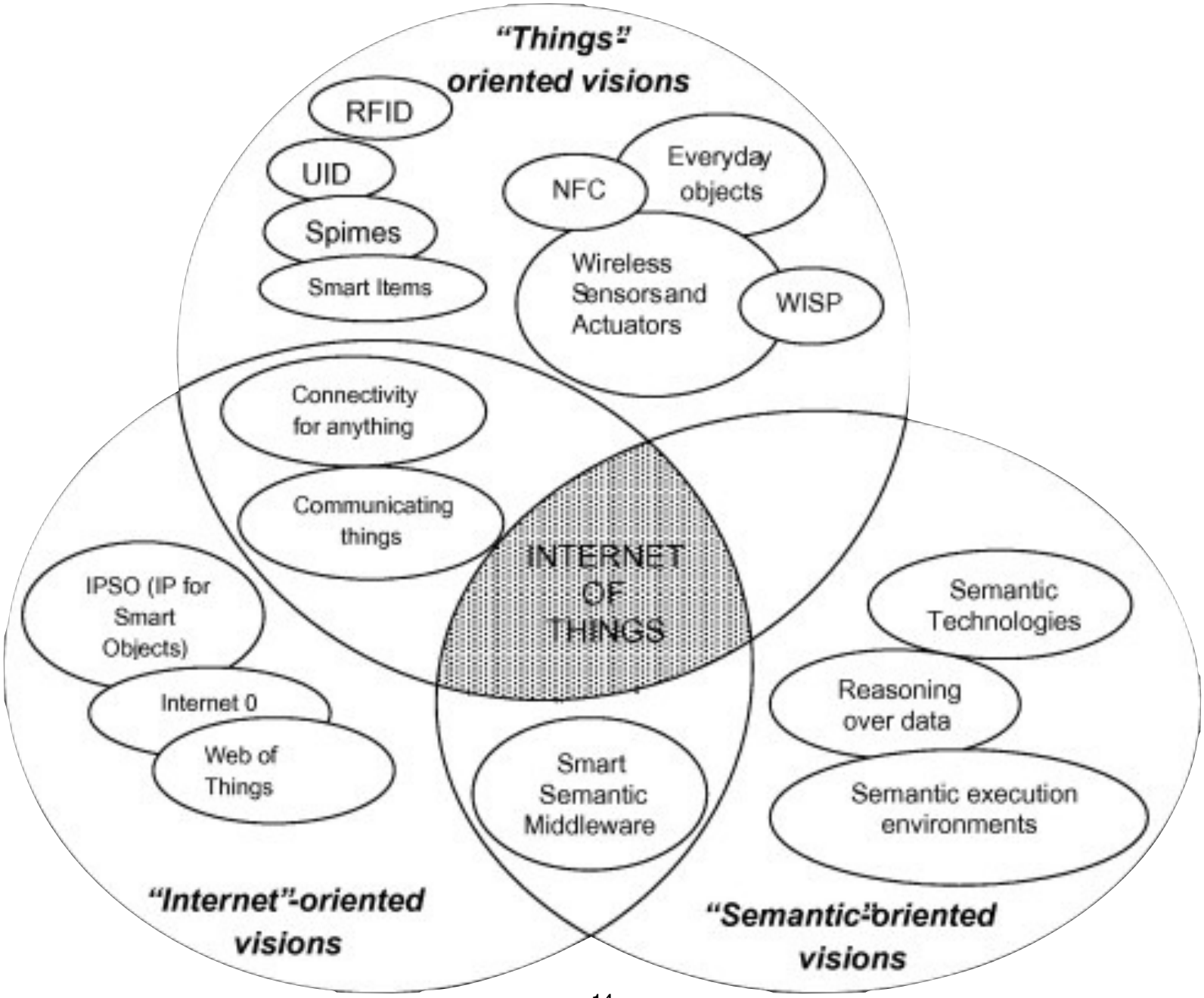
The early days of the Internet of Things

- **Motivated by caffeine and sloth…**
- **CMU Coke Machine**
  - CMU CS Department, U.S.A.,1982- (several iterations)
- **The Trojan Room Coffee Pot Camera**
  - Computer Science Lab, University of Cambridge, U.K. 1991-2001

Microsoft SPOT

- **Microsoft Smart Watch SPOT 2004-8**
  - Smart Personal Objects Technology
  - general platform—watches **and** coffeemakers
  - data broadcast over FM band in USA (DirectBand—12Kb/s)
  - watches from Swatch, Suunto, Tissot, and Fossil
  - data feed subscription based ($60/year)

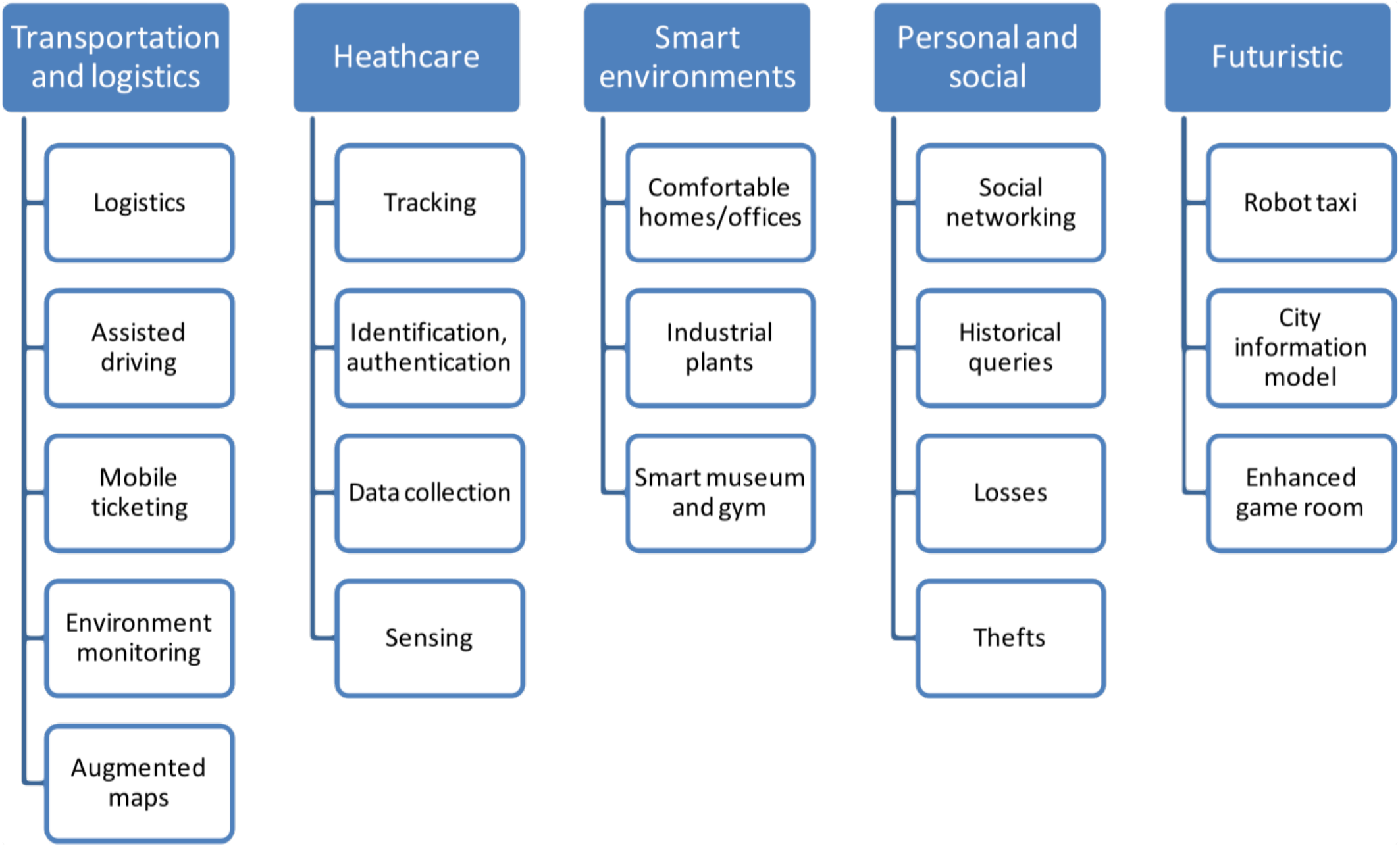
The Internet of Things: at the crossroads



Overview

- What is the Internet of Things?
- The vision
- **Domains of the Internet of Things**
- **The challenges of the Internet of Things**
- **The Proto Web of Things**
- **The challenge of the Internet of Things**
- **The Web as IoT architecture**
- **Milestone 1: meet the Raspberry Pi!**

The Internet of Things: domains



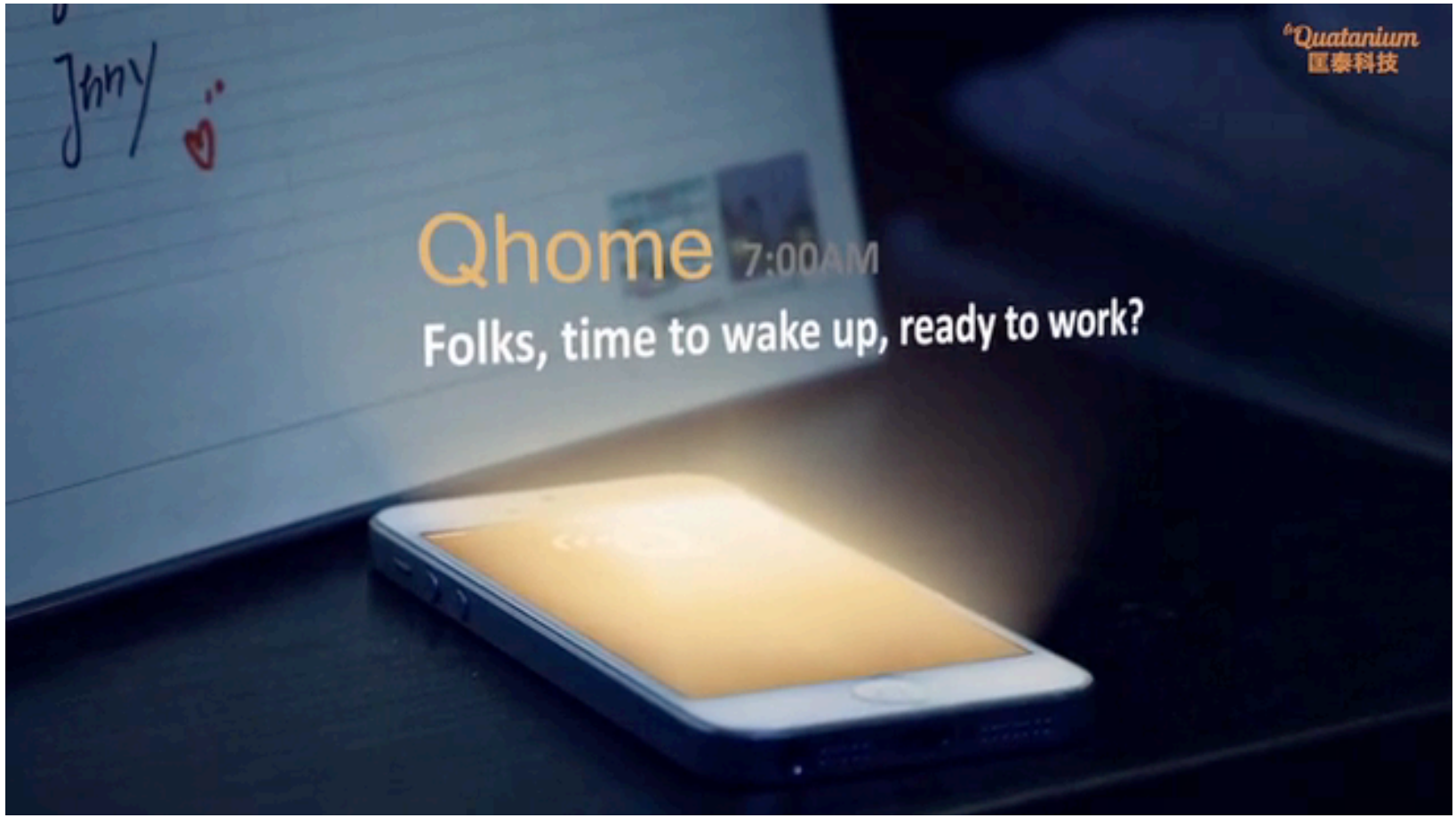
Logistics and supply chain

- **Precise tracking within and across organisations**
- **Shipping and transportation**
- **Manufacturing**
  - asset tracking: precise knowledge of components, both being ordered and delivered from suppliers, and on the shop floor
- **Post manufacturing**
  - delivery to customer
  - precise knowledge of constituent parts, their origins, and histories
  - monitoring of product during its lifetime to ensure quality and proper disposal

Home automation/Smart home

- **Smart metering**
  - electricity, water, heat
  - benefit in a fluctuating energy market
- **Home control**
  - builtin, or through after-market add-ons
- **Home surveillance**
  - fire, water leakage, intruders

QHome


Jenny
Qhome 7:00AM
Folks, time to wake up, ready to work?
Quatanium
匡泰科技

Healthcare

- **Sensors, wearable or otherwise, enabling high fidelity surveillance of the sick, the injured, and the elderly**
  - detecting things, before they become an issue
  - enabling patients to live normal lives away from hospitals, yet still monitored
  - automated systems alerting patients without the need for a doctor
- **Tracking doctors, nurses, orderlies, patients, medicine, and equipment to ensure efficient and correct procedures at hospitals**
  - decrease dangerous or costly mistakes

The Quantifiable Life

- Keeping track of caloric intake, weight, exercise, sleep, etc, etc
- Wearable sensors (exercise monitors, smart watches)
- Mobile phones (GPS, accelerometers, …)

Wearables

- **Smart watches and other devices**
- **Always available, continually sensing**
- **Typically, small interface connected to smartphone gateway**
  - conserves battery, provides richer interface on larger device
- **Modern examples**
  - Pebble Watch; Apple Watch ; Google Wear; Samsung Gear
  - The Dash headphones
  - Fitbit, and other fitness trackers

Wireless Sensor Networks

- **Underlying many scenarios, many sensors distributed in an area monitoring and measuring the environment, digital or physical**
  - RFID tags, Bluetooth IDs, . . .
  - temperature, humidity, vibration,. . .
- **Zero (or very low) configuration, self-organising network**

Smart Infrastructure

- **Smart Grid**
  - aligning production and consumption of electricity
  - across borders
  - especially crucial with renewable energy sources
- **Support for planning and living**
  - Smart Cities
  - traffic analysis based on crowd-sourced sensing
  - improved real-time data for commuters
- **Transportation**
  - fleet management
  - self-driving cars, etc.

Overview

- What is the Internet of Things?
- The vision
- Domains of the Internet of Things
- **The challenges of the Internet of Things**
- **The Proto Web of Things**
- **The challenge of the Internet of Things**
- **The Web as IoT architecture**
- **Milestone 1: meet the Raspberry Pi!**

Challenges for the Internet of Things

- **The Internet of Things has many forms, many domains, and many associated challenges**
  - technological as well as legal and social

Energy usage and scavenging

- **A sensor with no power is no good**
- **Energy conservation**
  - long-lived batteries; highly frugal devices; low-energy networking and routing
- **Energy capture**
  - through radio signals (e.g., passive RFID)
  - through induction, photovoltaic, motion, . . .

Too many devices, too few IP addresses

- **We have nearly run out of (IPv4) IP-addresses**
  - 32-bit addresses seemed/was big enough 40 years ago. Today? Not so much
  - $2^{32} = 4.294.967.296$
- **Interim solution: use gateways to "hide" devices**
- **IPv6 to the rescue!**
  - 128-bit address space: We are not going to run out of IP-addresses anytime soon
  - $2^{128} = 340.282.366.920.938.463.463.374.607.431.768.211.456$
  - Transition painful, tedious, expensive, necessary, and taking place
- **6LoWPAN**
  - IPv6 over Low power Wireless Personal Area Networks

Naming and discovery

- IPv6 may address unique identities (10^{38} is **big**), but that is not enough:
- How are devices found?
- How are they named?
- How are their abilities discovered?
- How is interoperability ensured?

Standards, rather than standardisation

- **The Internet of Things is seen (perhaps rightly so) as a huge future growth opportunity across many fields**
  - to hold the keys to that growth is highly desirable
- **Result: Major players (Microsoft, Google, Intel, Apple, Samsung, Qualcomm, ARM, TI, etc.) present their own vision, tools, and systems**
  - promising startups are being bought by big companies (e.g., Nest acquired by Google)
  - small companies are interested in interoperability, big companies often less so
- **So far, no one solution dominant enough to force *de facto* adherence**
  - will this be a case for market forces or international agreements?

Data silos

- **Pre-Web Internet**
  - data flowed evenly across the hosts
- **Present Internet**
  - data flows from content providers to end-users; data about habits collected
- **Future (IoT) Internet**
  - countless sensors and devices streaming data towards central repositories (ie., clouds)
- **If major players succeed in creating dominant standards and systems, the collected data will end up on their servers**
  - which, presumably, is *the whole point* of playing for some of them…

Security and privacy

- **If I generate data, surely that data is mine?**
  - or is it? Better read up on the fine print in that EULA…
  - even if it *is* mine, where is it stored? How securely is it stored?
- **If my home, or the infrastructure I depend on, is "smart", it is also hackable and vulnerable**
  - security becomes a paramount concern
  - smart devices become potential vectors of attack
  - smaller devices cannot implement sophisticated security
  - smart grids must be protected at a high national and international level
- **Industrial espionage and sabotage**
  - stuxnet and its heirs

Summary

- **The Internet of Things is characterised by**
  - identity, connectivity, and capability
- **It posits a huge set of different devices collaborating and coexisting, collecting and correlating data**
- **The technical challenges are significant, as are the social and legal ones**
- **Who will hold the keys to the IoT? Will we see a balkanisation of systems, a centralisation, or is there a third way?**

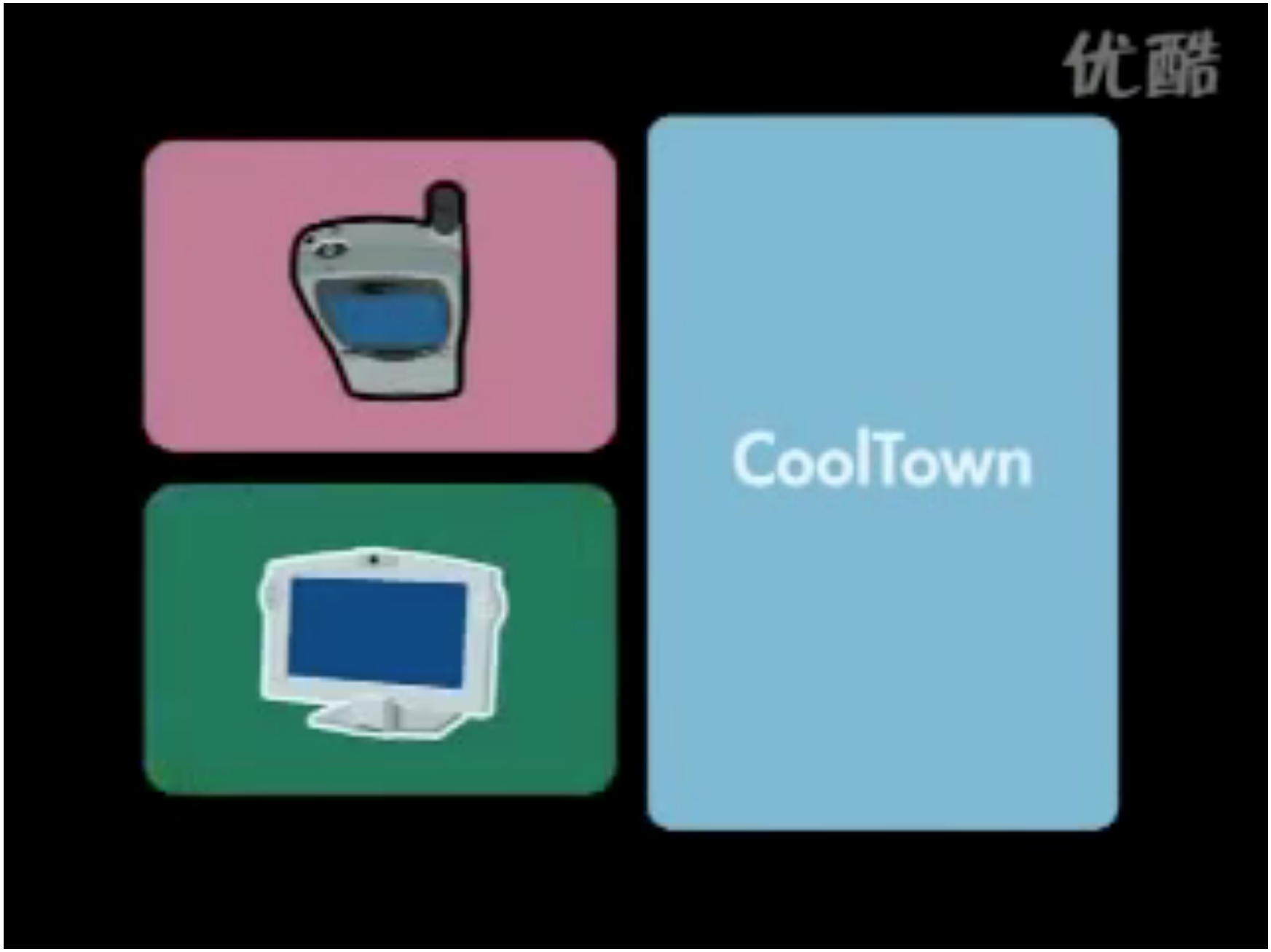
Overview

- What is the Internet of Things?
- The vision
- Domains of the Internet of Things
- The challenges of the Internet of Things
- **The Proto Web of Things**
- **The challenge of the Internet of Things**
- **The Web as IoT architecture**
- **Milestone 1: meet the Raspberry Pi!**

Hewlett Packard CoolTown (2001)

- **Everything has a Web-presence**
  - embedded server, or
  - scannable URL (optical or IR based)
- **The state of all things can be inspected**
  - well-defined semantics ⇒ interoperability between devices
- **Devices communicate with each other within the context of their use (e.g., owner's identity)**
  - software agents can work on behalf of their users

HP CoolTown



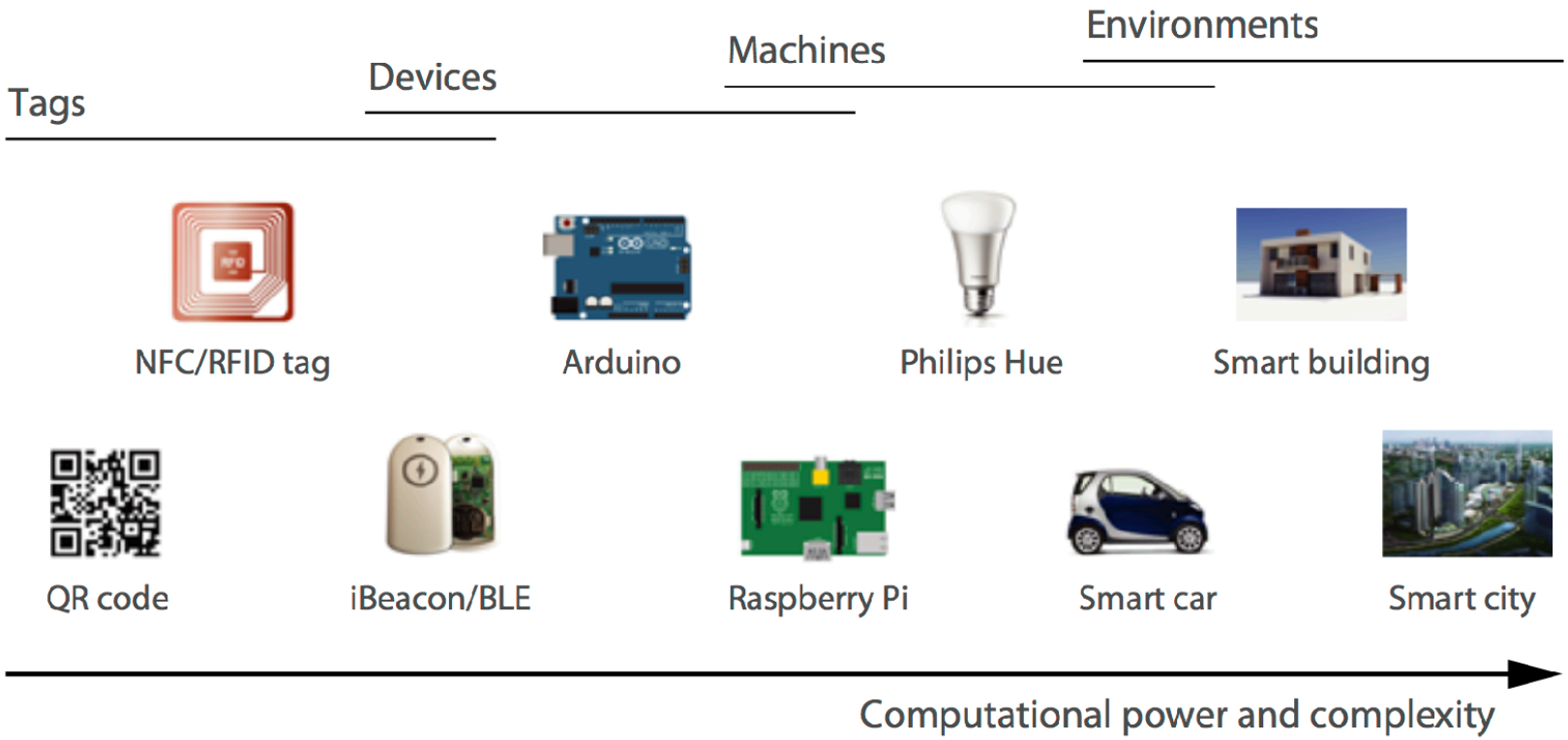
Overview

- What is the Internet of Things?
- The vision
- Domains of the Internet of Things
- The challenges of the Internet of Things
- The Proto Web of Things
- **The challenge of the Internet of Things**
- **The Web as IoT architecture**
- **Milestone 1: meet the Raspberry Pi!**

The definition of IoT by Guinard & Trifa

The Internet of Things is a system of physical objects that can be discovered*,* monitored*,* controlled*, or* interacted *with by electronic devices that communicate over various networking interfaces and eventually can be connected to the wider internet.*

The scope of the Internet of Things



- **A wide range of uses & devices—from tags to cities**
  - all connected to the Internet in one form or another
- **A large set of associated technologies, data, and communication standards, companies, and stakeholders**

The Intranet of Things

- **The set of technologies and uses is too diverse to have one single standard of communication across the entire stack**
- **Companies are inclined to prefer their own solutions**
- **Thus, we face the Intranet of Things**
  - islands of devices
- **If a company goes out of business, is bought, or abandons a 'smart' product…**
  - the device may well stop functioning

An arbitrary example

Philip Hue	4 apps, hardware controllers	Accessible through HomeKit and Siri
AirTunes	Own app, iTunes	
Elgato sensors and switches	Own app	
Squeezebox Radio	Own app (by third party)	
Anova Sous Vide	Own app	
SmartHalo Bike Nav	Own app	HealthKit
AV Equipment	Own app, Harmony remote	
Pebble watch	Own app	HealthKit

Overview

- What is the Internet of Things?
- The vision
- Domains of the Internet of Things
- The challenges of the Internet of Things
- The Proto Web of Things
- The challenge of the Internet of Things
- **The Web as IoT architecture**
- **Milestone 1: meet the Raspberry Pi!**

The Web of Things

- **So…**
  - a myriad of devices from different vendors generating huge amounts of data
  - multitudes of users accessing and manipulating these devices and their data from a similarly broad palette of systems
- **If only we had some system that managed just that…**
- **…also known as the World Wide Web**
- **Why invent new technologies and protocols, when we have adequate, *extremely* widespread systems already?**

A return to the Cool part of Town…

- Use the web for **application layer communication**
- Use web browsers to inspect devices and their capabilities (M2H)
- Use standard web protocols, naming conventions, and data formats to access, explore, and control devices (M2M)
- Not necessary to reinvent security once again
- Better than an app for every device…

One size fits all?

- **No. There is still a need for specialised communication standards for, e.g., energy constrained devices**
- **But, as long as we maintain an application layer based on web standards, it does not matter what is underneath**
- **The web has continued to evolve and grow since its inception, but it is still highly interoperable**
  - especially since web standards became something to follow 5-10 years ago
- **Loose coupling helps ensuring continuity**

A WoT Thing

My WoT Camera | Home | Book | Code

Device: My WoT Camera!

This is a WoT Device (URL: http://devices.webofthings.io/camera/).

Description: A simple WoT-connected camera..

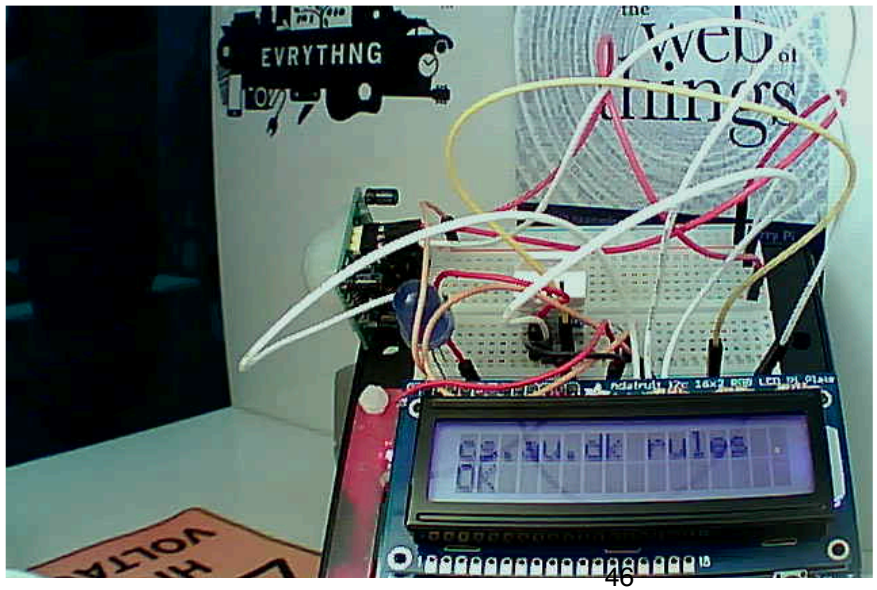
Tags: ["camera","WoT"].

Sensor: Camera Sensor

Description: Takes a still picture with the camera.

1. Type: image
2. Recorded at: 2017-09-06T21:33:00.751Z
3. Value: http://devices.webofthings.io:9090/snapshot.cgi?user=snapshots&pwd=4MXfTSr0gH

Sensor Value



URIs for smart devices

- **Devices have resources**
  - such as sensors that can take measurements,
  - states, that can be read or set, or
  - rules, that can be modified
- **These can be named systematically using URIs**
  - `http://…/sunspots/spot1/sensors/` (all sensors provided by spot1)
  - `http://…/sunspots/spot1/sensors/temperature` (spot1's temperature sensor)
  - `http://…/sunspots/spot1/actuators/leds/2` (spot1's second led)
- **Hierarchical structured, easily readable for humans, and easily transversed by machine**
  - pages contain links to parents and children

Representing resources

- HTTP provides the accept header to signify what data formats the recipient prefers/can handle
- This allows for flexible representations of the same resources, increasing levels of interoperability
- When a device is being queried by a web browser (i.e., a human), it should return HTML describing the state of the resource
- When a device is being queried by another device, JSON or XML is much better

Different representations for different purposes

Operating on a device

- **HTTP supports four main methods:**
- GET
  - retrieve the state of a resource — don't change the resource
- PUT
  - update existing resource, or create new resource with an identifier
- POST
  - create new resource, do not specify identifier
- DELETE
  - remove a resource

(A few) HTTP Status Codes

- **`200 OK`**
  - Standard response for successful HTTP requests. The actual response will depend on the request method used. In a GET request, the response will contain an entity corresponding to the requested resource. In a POST request the response will contain an entity describing or containing the result of the action
- **`201 Created`**
  - The request has been fulfilled and resulted in a new resource being created
- **`301 Moved Permanently`**
  - This and all future requests should be directed to the given URI
- **`404 Not Found`**
  - The requested resource could not be found but may be available again in the future.

Many clients, weak device?

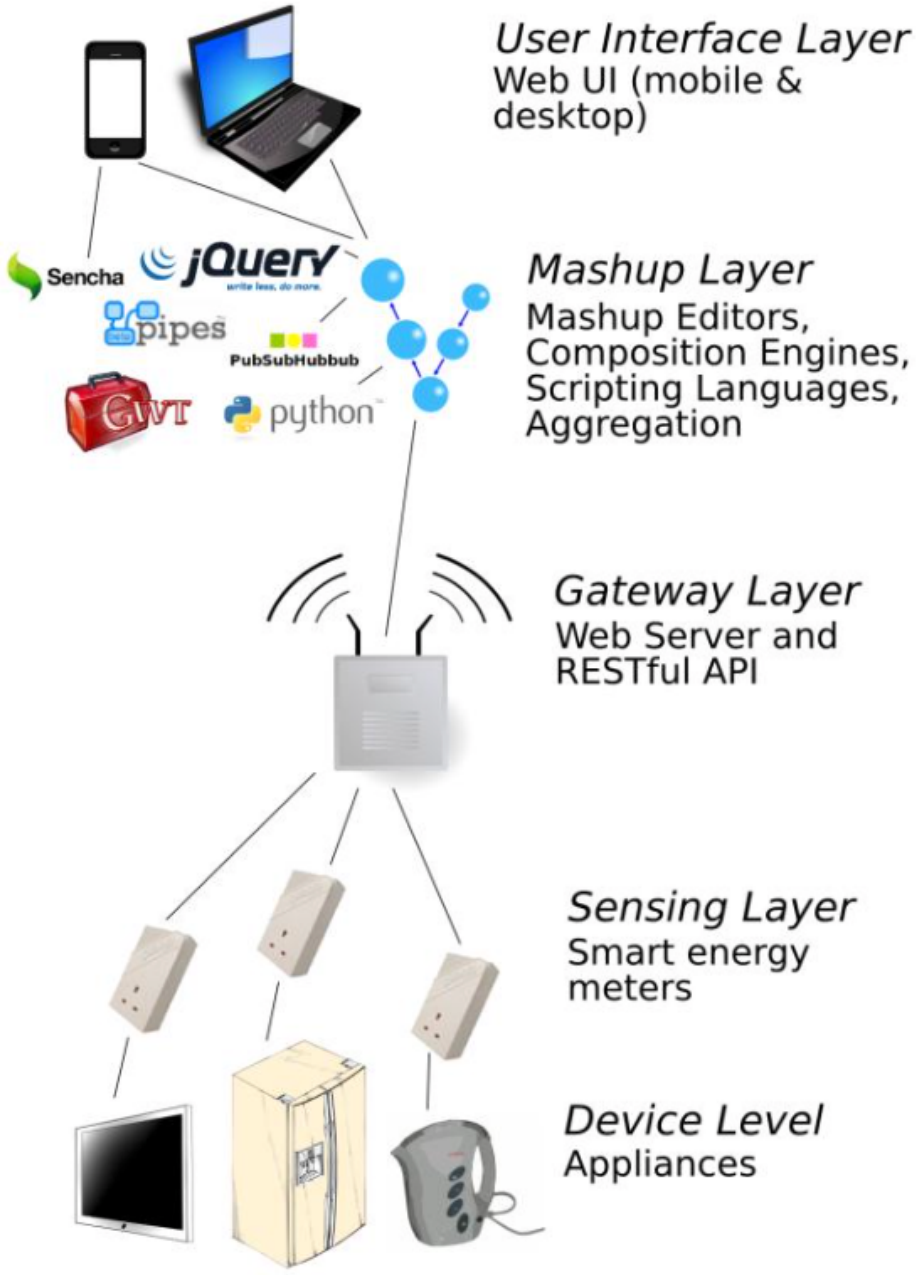
- Register device with Atom feed on server
- Whenever a specified state changes on the device, it is pushed to the server
- The resource is published by the server, conserving the device's resources

Streaming data?

- Connect through a WebSocket
- Pass data back and forth as needed, possibly through an intermediary

Wrapping existing systems

- Devices not adhering to the RESTful approach can be handled through the use of a gateway
- Thus, proprietary systems can be wrapped and used in a wider context



Advantages of the Web of Things

- **Existing Web based tools, frameworks, and methodologies just work**
  - including proxies and caching
  - and, crucially, security
- **Simple to do "mash-ups", connecting data sources with other tools**
- **Easy to do development by exploration**
- **If a device has a Web browser, it can be used to explore the Internet of Things**

Disadvantages of the Web of Things

- **HTTP is a fairly heavy protocol for small devices**
  - though small devices are getting more powerful
- **Streaming (sensor) data is not what HTTP was originally designed to do**
  - Web Sockets helps, as will HTTP/2
- **Discovery**
  - devices may be described using micro formats, which can be systematically indexed
- **Security**
  - use existing frameworks, including authentication through social network sites
  - hide devices behind a proxy that requires proper authentication from, e.g., Facebook

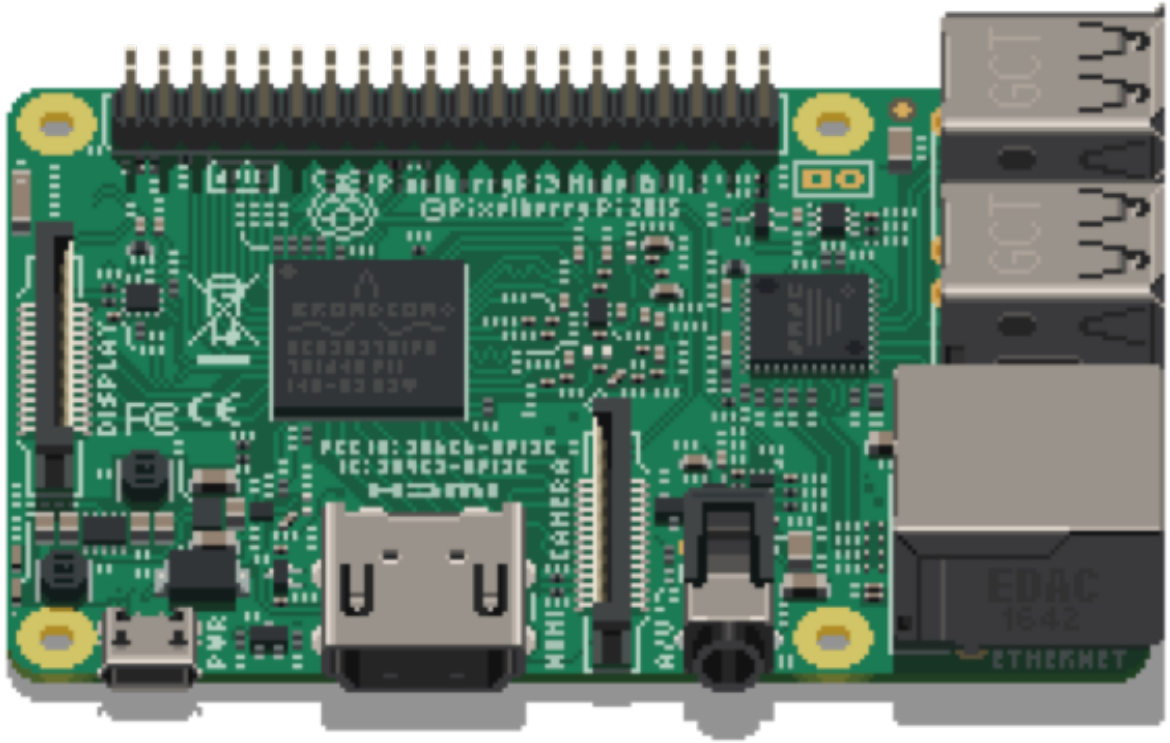
The *Internet* of Things

- The premise and success of the Internet is founded on open standards
- Open, shared standards are required for the Internet of Things to succeed outside of specialist or vendor-specific domains
- Hopefully, if no shared standards can be agreed upon in advance, market realities will force interoperability

Overview

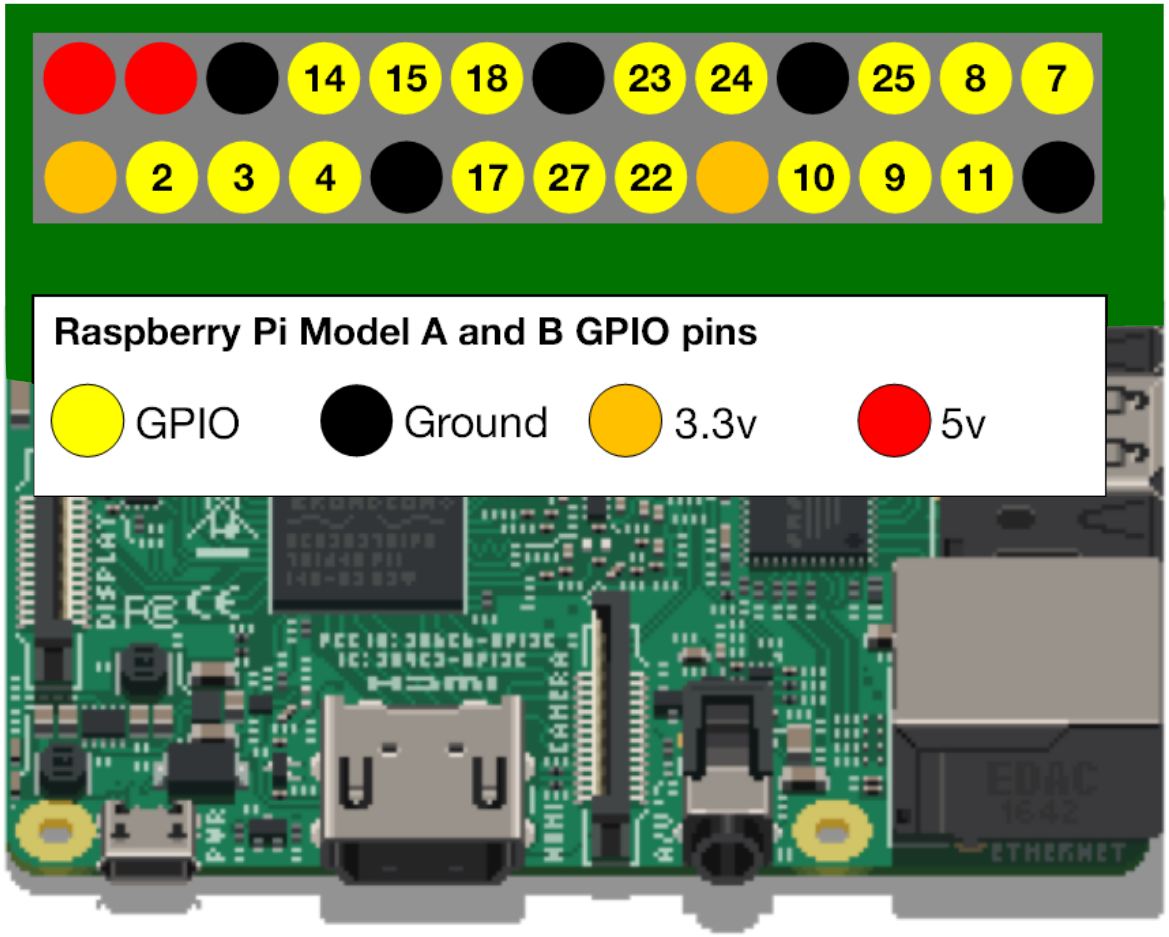
- What is the Internet of Things?
- The vision
- Domains of the Internet of Things
- The challenges of the Internet of Things
- The Proto Web of Things
- The challenge of the Internet of Things
- The Web as IoT architecture
- **Milestone 1: meet the Raspberry Pi!**

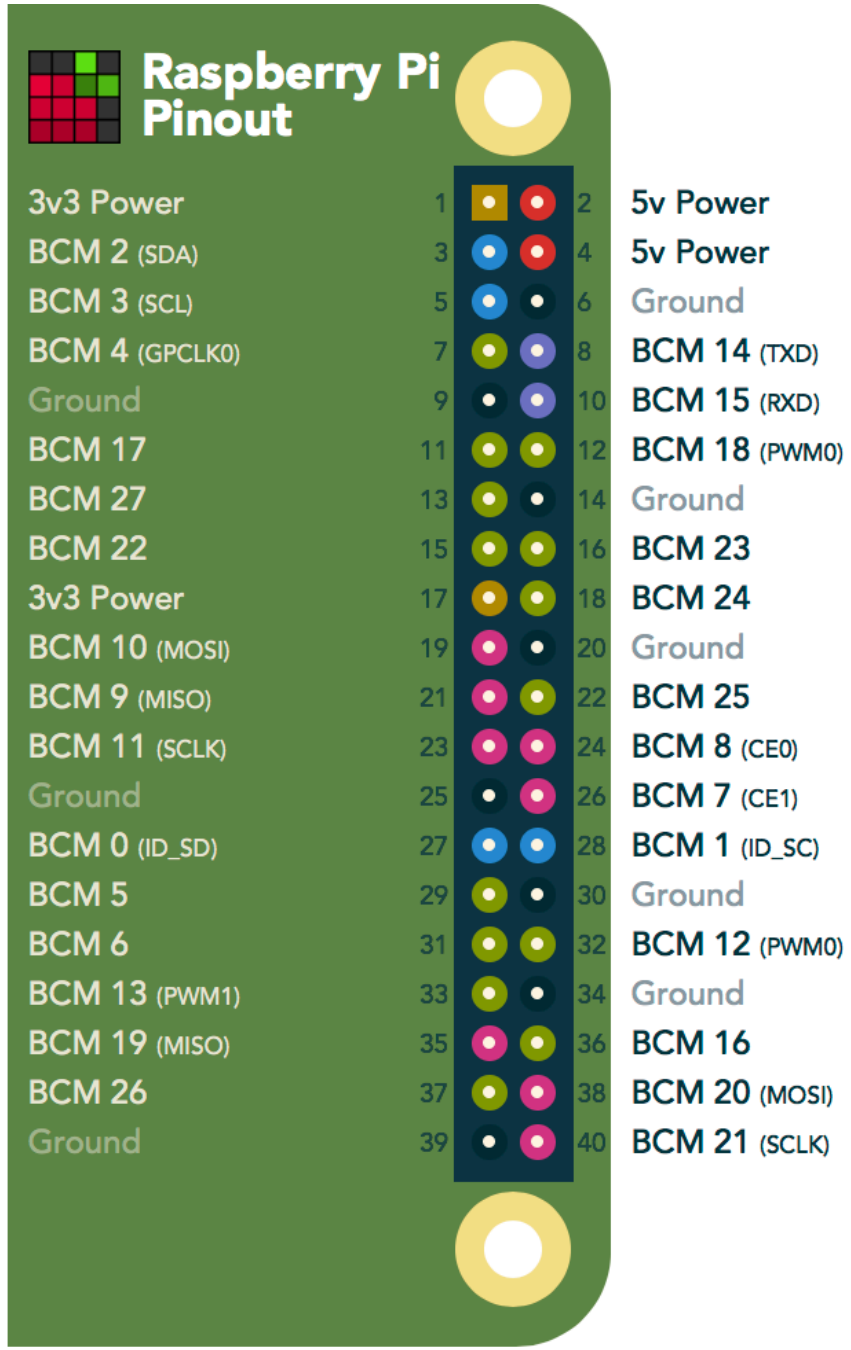
Meet the Raspberry Pi 3 B+



- **Single Board Computer**
  - Quad-core 1,4 GHz ARMv8, 1 GB RAM, 802.11 b/g/n/ac, Bluetooth 4.2 LE, Ethernet, microSD, 4×USB 2.0, 3,5 mm audio jack, Camera Input, HDMI 1080p, GPIO
  - Runs all sorts of (largely Linux-based) operating systems—we'll be using the standard Debian based Raspian (I have put a system image on the Website)
- **>15 million Raspberry Pis have been sold worldwide**

Input and Output on a RPi





- **GPIO**
  - General Purpose Input/Output
  - 5 and 3.3 volt
  - only digital, no AD on a Raspberry Pi

Talking to the sensors/actuators

- **Node.js does, of course, not know the GPIO, so it is necessary to install drivers and modules to access it**
- **Such requirements are quite common for JavaScript projects (regardless of whether they use GPIO or not), and the** package.json **file is the answer together with** npm

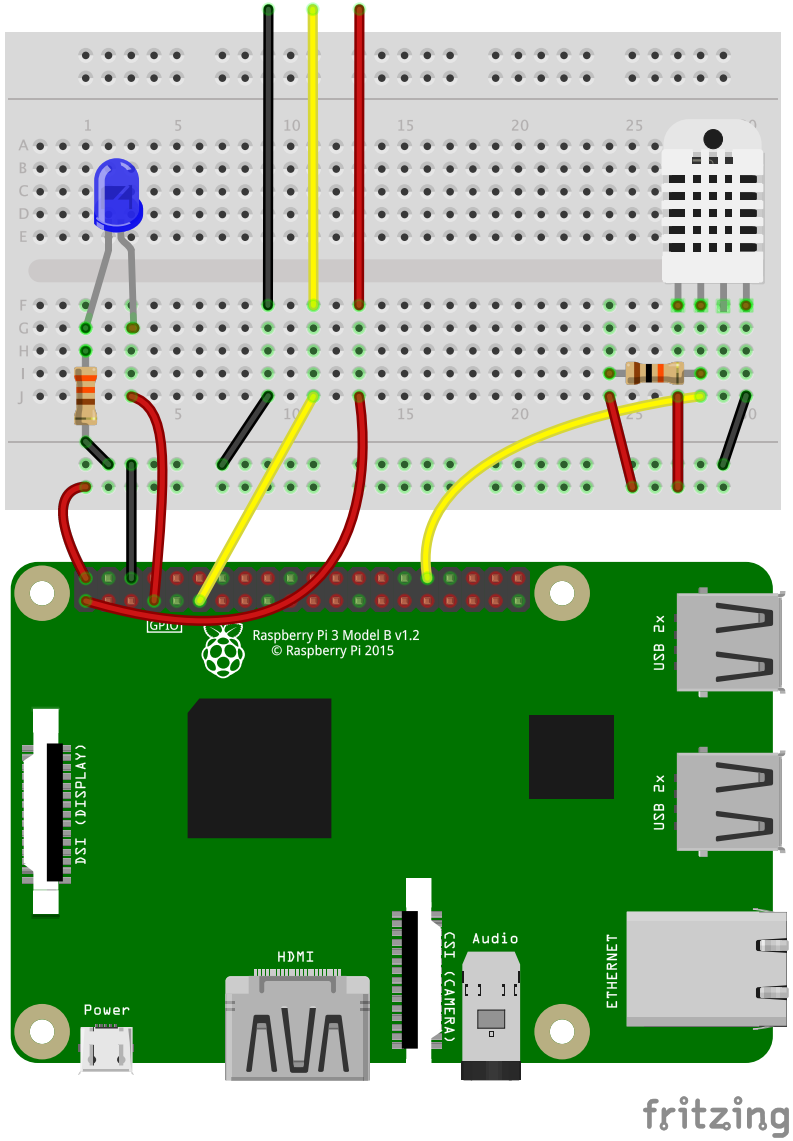
Installing required drivers & modules

```
{
  "name": "wot-book-gpio",
  "description": "Examples for the Web of Things Book",
  "author": "Dominique Guinard <dom@guinard.org>",
  "author": "Vlad Trifa <vladounet@gmail.com>",
  "repository" : "https://github.com/webofthings/wot-book.git",
  "dependencies": {
    "onoff": “^1.2.0",
    "node-dht-sensor": "^0.0.33"
  },
  "engine": "node >= 0.12.0"
}
```

- **Specifies information about a project, especially dependencies**
  - modules that should be retrieved, built, and installed for the project to work
- `npm install`
  - will install the required modules in the current folder

Connecting sensors to the RPi

- **A blue LED**
  - short leg conn. to 330Ω resistor conn. to GND
  - long leg conn. to pin 7 (GPIO 4)
- **A temperature humidity sensor**
  - first leg conn. to 3.3 V
  - second leg conn. to 10kΩ resistor conn. to 3.3V
  - second leg conn. to pin 32 (GPIO 12)
  - fourth leg conn. to GND
- **PIR sensor**
  - first leg conn. to GND
  - second leg conn. to pin 11 (GPIO 17)
  - third leg conn. to 5V



Blinkenlights

```
const Gpio = require('onoff').Gpio // #A

const led = new Gpio(4, 'out') // #B
let interval

interval = setInterval(() => { // #C
  let value = (led.readSync() + 1) % 2 // #D
  led.write(value, () => { // #E
    console.log('Changed LED state to: ' + value)
  })
}, 2000)

process.on('SIGINT', () => { // #F
  clearInterval(interval)
  led.writeSync(0) // #G
  led.unexport()
  console.log('Bye, bye!')
  process.exit()
})

// #A Import the onoff Gpio library
// #B Initialise pin 4 to be an output pin
// #C This interval will be called every 2 seconds
// #D Synchronously read the value of pin 4 and transform 1 to 0 or 0 to 1
// #E Asynchronously write the new value to pin 4
// #F Listen to the event triggered on CTRL+C
// #G Cleanly close the GPIO pin before exiting
```

Temperatur & humidity

```
const sensorLib = require('node-dht-sensor')

sensorLib.initialize(22, 12) // #A
const interval = setInterval(() => { // #B
  read()
}, 2000)

function read () {
  let readout = sensorLib.read() // #C
  console.log('Temperature: ' + readout.temperature.toFixed(2) + 'C, ' + // #D
    'humidity: ' + readout.humidity.toFixed(2) + '%')
};

process.on('SIGINT', () => {
  clearInterval(interval)
  console.log('Bye, bye!')
  process.exit()
})

// #A 22 is for DHT22/AM2302, 12 is the GPIO we connect to on the Pi
// #B create an interval to read the values every 2 seconds
// #C read the sensor values
// #D readout contains two values: temperature and humidity
```

Is there anybody out there?

```
const Gpio = require('onoff').Gpio
const sensor = new Gpio(17, 'in', 'both')    // #A

sensor.watch((err, value) => { // #B
  if (err) exit(err)
  console.log(value ? 'there is someone!' : 'not anymore!')
})

function exit (err) {
  if (err) console.log('An error occurred: ' + err)
  sensor.unexport()
  console.log('Bye, bye!')
  process.exit()
}
process.on('SIGINT', exit)

// #A Initialize pin 17 in input mode, 'both' means we want to handle
//    both rising and falling interrupt edges
// #B Listen for state changes on pin 17, if a change is detected
//    the anonymous callback function will be called with the new value
```

Sensors in action

```
pi@raspberrypi: ~/Development/itWoT/my-first-WoT-code (ssh)
make: Leaving directory '/home/pi/Development/itWoT/my-first-WoT-code/node_modules/node-dht-sensor/b
uild'
npm notice created a lockfile as package-lock.json. You should commit this file.
npm WARN wot-book-gpio@ No license field.

added 5 packages in 35.945s
pi@raspberrypi:~/Development/itWoT/my-first-WoT-code $ node blink.js
Changed LED state to: 1
Changed LED state to: 0
Changed LED state to: 1
Changed LED state to: 0
Changed LED state to: 1
```

```
pi@raspberrypi: ~/Development/itWoT/my-first-WoT-code (ssh)
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sun Feb 11 23:17:07 2018 from 192.168.1.217

SSH is enabled and the default password for the 'pi' user has not been changed.
This is a security risk - please login as the 'pi' user and type 'passwd' to set a new password.

pi@raspberrypi:~ $ cd Development/itWoT/my-first-WoT-code/
pi@raspberrypi:~/Development/itWoT/my-first-WoT-code $ node dht.js
Temperature: 19.10C, humidity: 42.20%
```

```
pi@raspberrypi: ~/Development/itWoT/my-first-WoT-code (ssh)
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sun Feb 11 22:37:28 2018

SSH is enabled and the default password for the 'pi' user has not been changed.
This is a security risk - please login as the 'pi' user and type 'passwd' to set a new password.

pi@raspberrypi:~ $ cd Development/itWoT/my-first-WoT-code/
pi@raspberrypi:~/Development/itWoT/my-first-WoT-code $
```

Connecting sensors for Milestone 1

- Getting started with Raspberry Pi development
- I would suggest you use Docker throughout the project, as that makes it easier to handle dependencies etc
- There is now an official AU gitlab repository:
  - https://gitlab.au.dk/
  - it supports private repositories for groups, so perfect for our purposes