Context-aware Mobile Computing

Benoît Garbinato
distributed object programming lab

Learning objectives

- ☐ Learn about mobile computing
- ☐ Learn about context-aware computing
- ☐ learn about related software support

Architecture evolution | Specialization

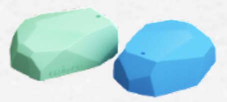
1980s: one man, one computer
- workstation, personal computers
- graphical user interfaces

1990s: the network is the computer
- the Internet accessible to all
- distributed operating systems

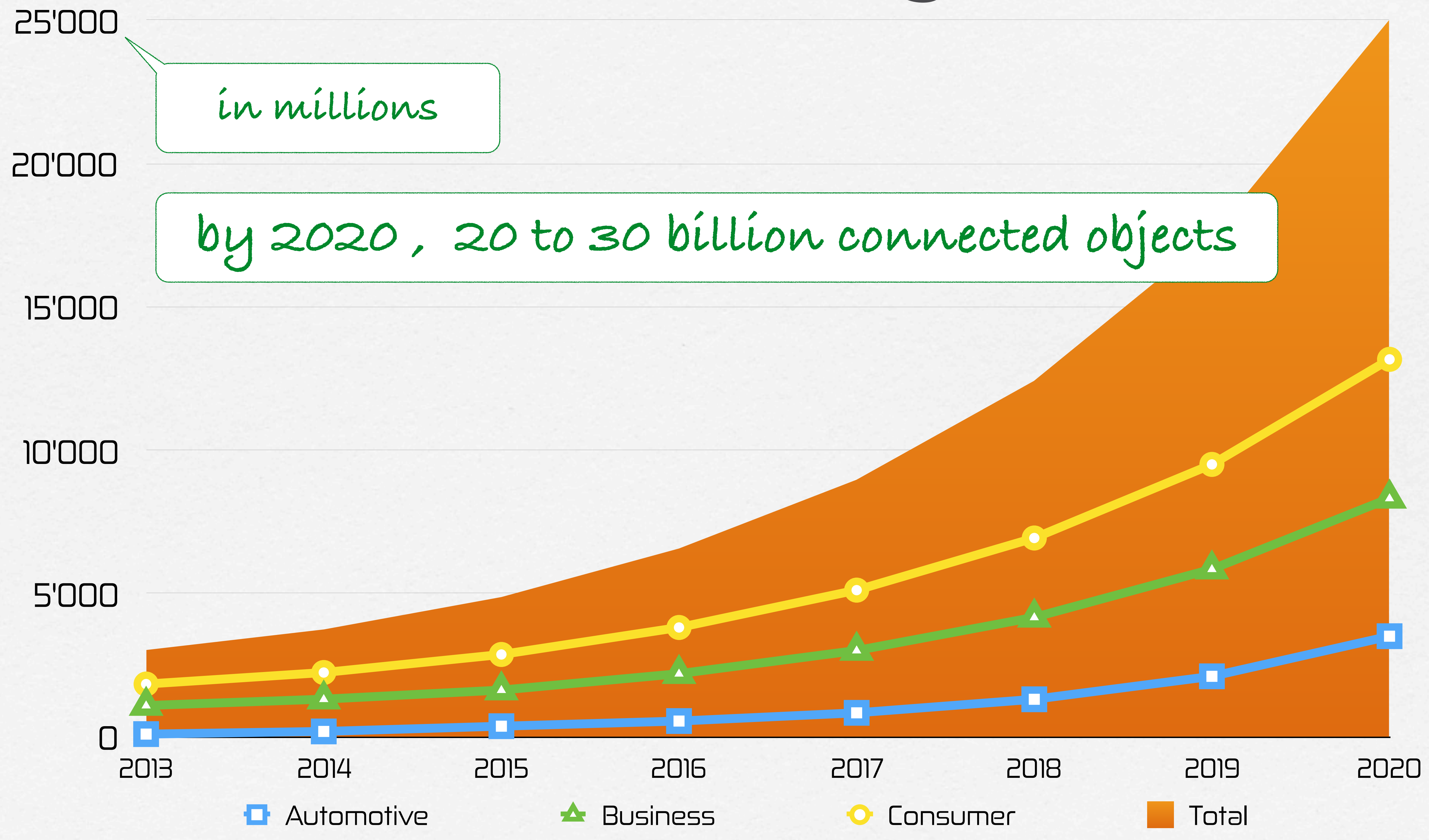
2000s: my phone is my computer
- smartphones & tablets as computers
- generalization of wireless networks

2010s: everything is a computer
- smart objects & the Internet of things
- personal networks connected to the cloud

Connected objects



Internet of Things (IoT)



Back to the future...

In the late nineties, Brian Halla from National Semiconductor wrote:

- ☐ processors will continue to become cheaper and faster,
- ☐ general-purpose PCs will eventually disappear,
- ☐ ubiquitous processors will be given for free by service providers.

Reference: Brian Halla, How the PC Will Disappear, IEEE Computer, vol. 31, no. 12, December 98.

What definition(s)?

mobile computing

ambient intelligence

sensor networks

ubiquitous computing

context-aware computing

location-aware computing

pervasive computing

nomadic computing

mobile ad hoc networks

Mobile vs. nomadic computing

- ❑ In common: anytime and anywhere
- ❑ Difference:

  nomadic: multiple fixed locations

  mobile: continuous on-the-move operation

Ubiquitous computing

Ubiquitous computing is the method of enhancing computer use by making many computers available throughout the physical environment, but making them effectively invisible to the user.

Mark Weiser, the "father" of ubiquitous computing

http://www.ubiq.com/hypertext/weiser/acmfuture2endnote.htm

Weiser’s vision

- Notion of “calm” technology, i.e., disappearing, invisible technology
- The computing devices is no longer at the center of our attraction, i.e., the best tools are those invisible to their users

Ubiquitous computing

- Computing devices are immersed in an real human-based environment*
- Devices have limited resources, e.g., power supply, memory, bandwidth, cpu, etc.
- Devices are mobile and wireless, and may reside on a person (wearable computing)

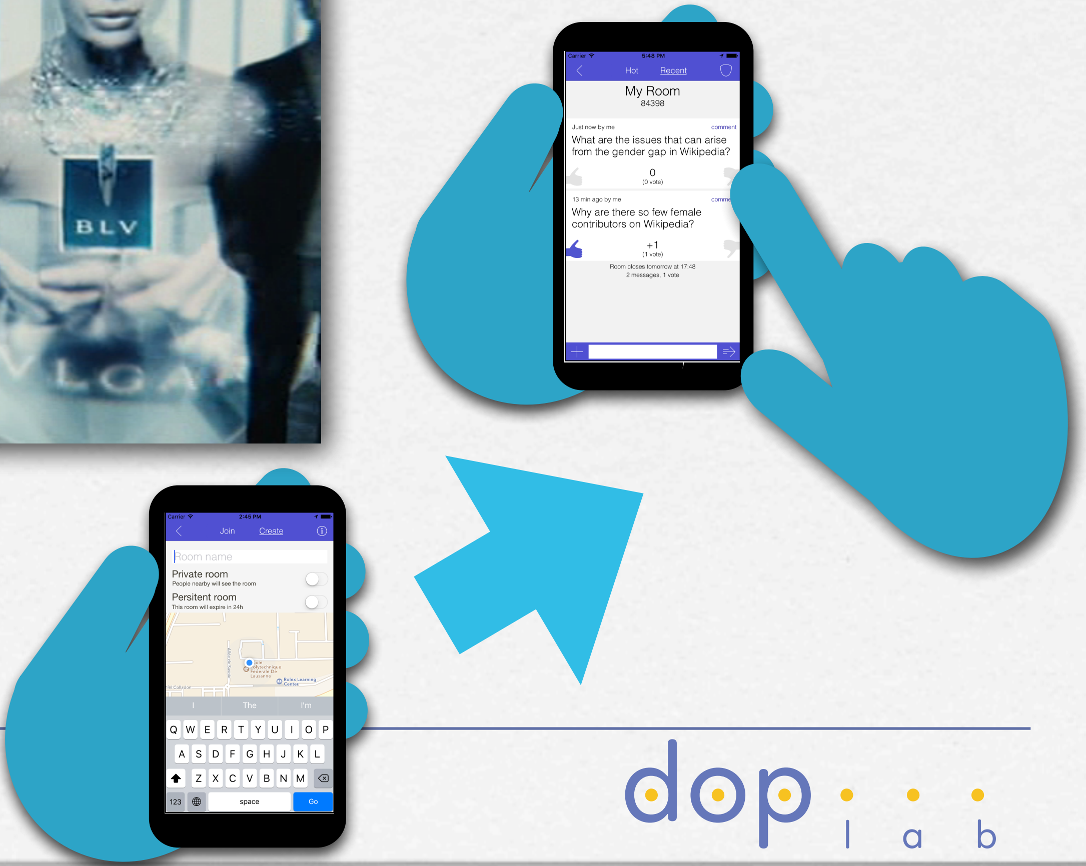
*somehow the dual of virtual reality, where humans are immersed in a virtual computer-based environment

Context/location awareness

- <u>Context awareness</u>: the computing system is aware of its environment and acts accordingly, e.g., time, temperature, device capability, location, user interests, activity, etc.
- <u>Location-awareness</u>: a special case of context awareness (see location-based pub/sub as an example)

Some scenarios…

R.C. Central Park
50 Central Park South, New York
The R.C. New York Central Park is a luxurious haven in the heart of the city.



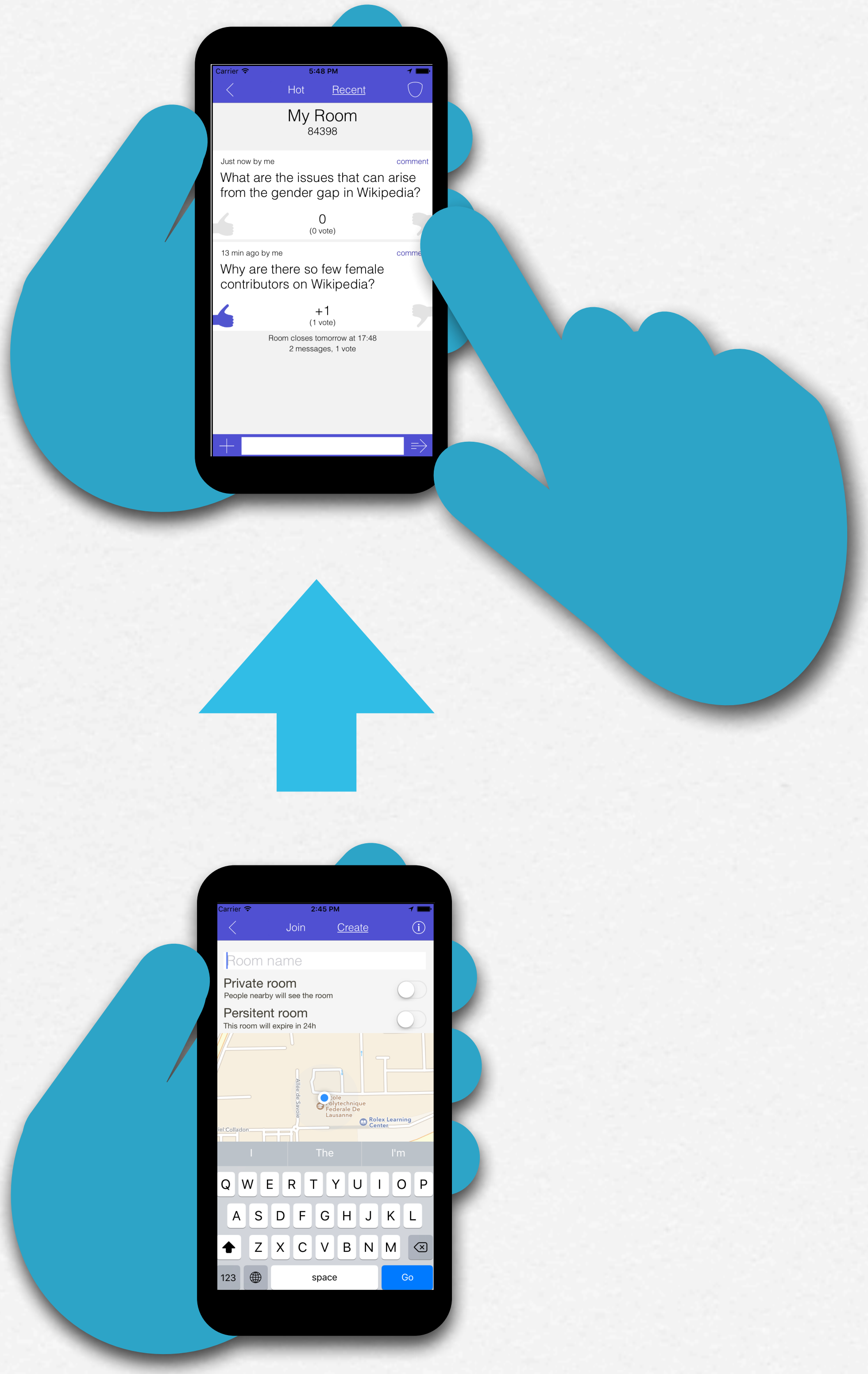
My Room
84398

Just now by me
What are the issues that can arise from the gender gap in Wikipedia?
0
(0 vote)

13 min ago by me
Why are there so few female contributors on Wikipedia?
+1
(1 vote)

Room closes tomorrow at 17:48
2 messages, 1 vote

Private room
People nearby will see the room
Persitent room
This room will expire in 24h



Just now by me — comment

What are the issues that can arise from the gender gap in Wikipedia?

0 (0 vote)

13 min ago by me — comme

Why are there so few female contributors on Wikipedia?

+1 (1 vote)

Room closes tomorrow at 17:48
2 messages, 1 vote

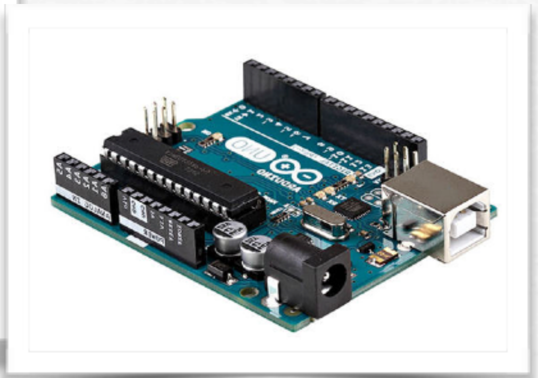
Private room
People nearby will see the room

Persitent room
This room will expire in 24h

SpeakUp

http://speakup.info

What devices?



- PDA and smart phones
- Smart devices/cards, e.g., Java card, iButtons, etc.
- Radio Frequency ID tags (RFIDs)
- Sensors networks

- Embedded systems, e.g., in the automotive industry
- ...

Distributed computing issues

- Remote communication (RMI, MOM, etc.)
- Naming of distributed entities/services
- Distributed data management, e.g., distributed file systems, distributed transactions, etc.
- Reliability, availability, security
- Caching (for performance)

Mobile computing issues

- Networking: mobile IP address, TCP de-/re-connection, performance, etc.
- Information access (bandwidth)
- Power consumption (variable cpu/disk speed, network de-/re-connection,etc.)
- Location awareness and resource discovery
- Mobile ad hoc networks & topology control

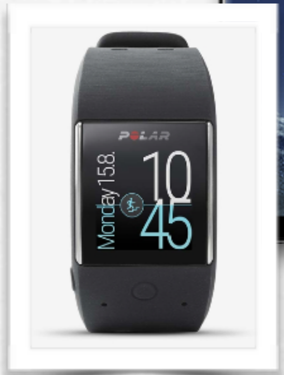
Context-aware computing issues

- ☐ Constraints on sensor design (size, cost, power consumption, etc.)
- ☐ Mobile ad hoc networks & topology control
- ☐ localized scalability (greater distance ⇒ less communication)
- ☐ Invisibility (millions of sensors should not distract the user)

What software support?

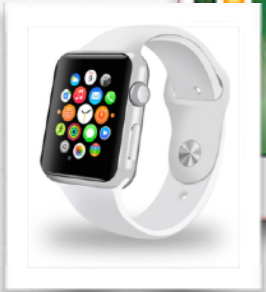
- The main challenge for a software supporting mobile & ubiquitous computing lies in the heterogeneity of devices
- Today, two software platforms have survived :
  - Android platform
  - Apple platform

Android platform



- Based on the acquisition of a small startup by Google in 2005
- In 2007, the Open Handset Alliance was funded to drive the development of open standards for mobile devices
- In 2008, Android became an open source project
- The development framework is based on the Java programming languages but not on standard Java APIs (neither Java SE nor Java ME)
- This is not (yet?) a curated platform

Apple Platform



- Based on Mac OS
- Based on Objective-C frameworks, now Swift
- Integrated in XCode, together with an emulator and a set of deployment options
- Comes with a business & application provisioning model, "à la" iTunes Store
- This is a curated platform, with an innovative revenue sharing models for developers