

# UBIQUITOUS COMPUTING

**Jong hyuk Park**

**UCS Lab**

**Tel: 970-6702**

**[jhpark1@seoultech.ac.kr](mailto:jhpark1@seoultech.ac.kr)**

# Summary

---

- Introduction to Ubiquitous Computing
- History of Ubiquitous Computing
- Challenges and Requirements
- Comparison of Technologies
- Software Engineering

# Introduction to Ubiquitous Computing

- What is
- Characteristics
- Goals

# What is (1/3)

The method of enhancing computing use by making many devices (services) available throughout the physical environment, but making them effectively invisible to the user (Mark Weiser)

# What is (2/3)

- Tries to construct a universal computing environment (UCE) that conceals:
  - ✓ Computing instruments
  - ✓ Devices
  - ✓ Resources
  - ✓ Technology
- Invisible to users

# What is (3/3)

- Computing everywhere
- Many embedded, wearable, handheld devices communicate transparently to provide different services to the users
- Devices mostly have low power and short-range wireless communication capabilities
- Devices utilize multiple on-board sensors to gather information about surrounding environments

# Characteristics of Ubicomp Applications

- Context-awareness
- Impromptu and volatile interaction
- Interactions among applications are based on specific context

# Goals

- The promise of ubiquitous computing:
  - ✓ A life in which our endeavours are powerfully, though subtly, assisted by computers
- The idealistic visions painted by the ubiquitous computing movement stand in stark contrast to what we see when we boot up our computers each day

# Summary

---

- Introduction to Ubiquitous Computing
- History of Ubiquitous Computing
- Challenges and Requirements
- Comparison of Technologies
- Software Engineering

# History of Ubiquitous Computing

- History
- Mark Weiser
- Experiments

# History

- Active Badge
  - ✓ Andy Hopper
- Xerox PARC 1991-2000
  - ✓ Mark Weiser (until, sadly, April 1999)
- Calm Technology

# Mark Weiser

- Researcher in the Computer Science Lab at Xerox's PARC (Palo Alto Research Center)
- First articulated the idea of ubiquitous computing in 1988
- Has called UC "...highest ideal is to make a computer so imbedded, so fitting, so natural, that we use it without even thinking about it."

# Experiments

- Tabs
- Pads
- Boards
- SAAMPad (Software Architecture Analysis Method Pad)
- The Conference Assistant

# Summary

- Introduction to Ubiquitous Computing
- History of Ubiquitous Computing
- Challenges and Requirements
- Comparison of Technologies
- Software Engineering

# Challenges and Requirements

- Hardware
- Applications
- User Interfaces
- Networking
- Mobility
- Scalability
- Reliability
- Interoperability
- Resource Discovery
- Privacy and Security

# Nanotechnology (1/3)

---

The trend toward miniaturization of computer components down to an atomic scale is known as nanotechnology

# Nanotechnology (2/3)

- Mobile data technology
  - ✓ GSM, GPRS, UMTS, CDMA, WAP, Imode
- Wireless data technology
  - ✓ Bluetooth, 802.11b
- Internet data technology
  - ✓ IP over optical, Broadband

# Nanotechnology (3/3)

- Content services
  - ✓ Web & WAP
- Applications
  - ✓ Multimedia, Internet messaging

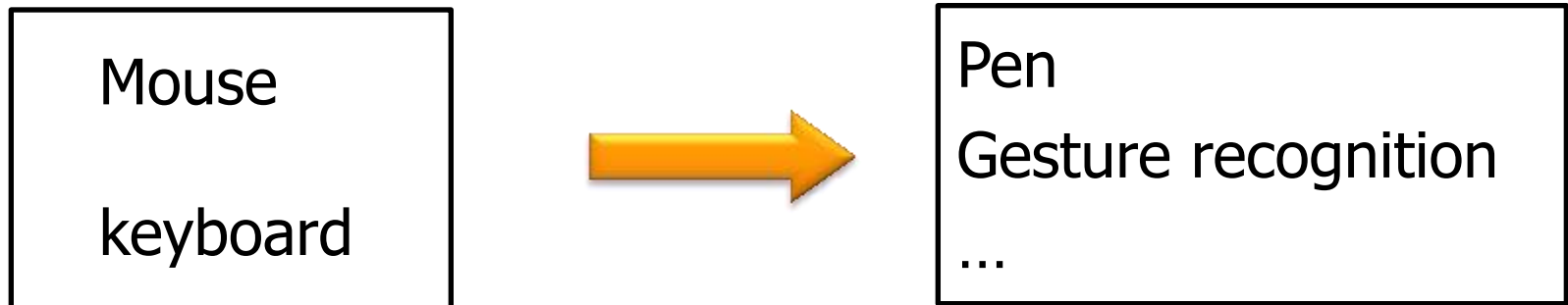
# Applications

- Main motivation of ubiquitous computing (Weiser 1993)
- Need to have an awareness of their context:

A combination of several factors, including the current location, the current user or if there are any other Ubicomp devices present in the near surroundings

# Users Interface

The multitude of different Ubicomp devices with their different sizes of displays and interaction capabilities represents another challenge



# Networking

- Another key driver for the final transition will be the use of short-range wireless as well as traditional wired technologies
- Wireless computing refers to the use of wireless technology to connect computers to a network

Mobility is made possible through wireless communication technologies

## **Problem of disconnectivity!!!**

This behaviour is an inherent property of the ubicomp concept and it should not be treated as a failure

# Scalability

- In a ubiquitous computing environment where possibly thousands and thousands of devices are part of scalability of the whole system is a key requirement
- All the devices are autonomous and must be able to operate independently a decentralized management will most likely be most suitable

# Reliability

- Thus the reliability of ubiquitous services and devices is a crucial requirement
- In order to construct reliable systems self-monitoring, self-regulating and self-healing features like they are found in biology might be a solution

# Interoperability

- This will probably be one of the major factors for the success or failure of the Ubicomp vision
- Use of technology just existed:  
JINI, CORBA, ecc...
- This diversity will make it impossible that there is only one agreed standard

# Resource Discovery

- The ability of devices to describe their behaviour to the network is a key requirement.
- On the other hand, it can not be assumed that devices in a ubiquitous environment have prior knowledge of the capabilities of other occupants.

# Privacy and Security

- In a fully networked world with ubiquitous, sensor-equipped devices several privacy and security issues arise
- The people in this environment will be worried about their privacy since there is the potential of total monitoring
- Must be understandable by the user and it must be modelled into the system architecture

# Summary

- Introduction to Ubiquitous Computing
- History of Ubiquitous Computing
- Challenges and Requirements
- Comparison of Technologies
- Software Engineering

# Comparison of Technologies

Table 1: Communication and Management architectures for Ubicomp environments

METRICS	Jini	UPnP	Salutation	Inferno	T Spaces
Scalability	✓✓✓✓	✓✓✓✓	✓✓	✓✓✓✓✓	✓✓
Reliability	✓✓✓✓✓	✓✓✓	✓✓	✓✓✓✓✓	✓✓
Configuration	✓✓✓✓	✓✓✓	<i>Unknown</i>	✓✓	✓
Administration	✓✓✓✓	✓✓✓	<i>Unknown</i>	✓✓	✓
Portability	✓✓✓✓	✓✓✓	✓✓✓✓	✓✓✓✓✓	✓✓✓
Service Discovery	✓✓✓✓✓	✓✓✓	✓✓✓	✓✓✓	✓
Interoperability	✓	–	✓✓✓✓	✓	–
Security	✓	–	–	✓✓✓	✓
TOTAL	28	19	15	26	11

Table shows a list of these APIs and technologies. The original comparison made by Olstad, Ramirez, Brady and McHollan. Without Bluetooth or IrDA.

# Summary

- Introduction to Ubiquitous Computing
- History of Ubiquitous Computing
- Challenges and Requirements
- Comparison of Technologies
- Software Engineering

# Software Engineering

- Research is inherently empirical and relies on a rapid prototyping development cycle
- Information should be pushed to user based on current task, inferences made about user's situation
- As user moves into different environments