경희대학교 세미나

PLAFORM + DESIGN

Youngki Lee, Inseok Hwang, Seungwoo Kang, Taiwoo Park,
Younghyun Ju, SangJeong Lee, Uichin Lee, Insik Shin,

Junehwa Song

KAIST

KAIST

Network Computing lab

Table of Contents

ACM SenSys 2015 is coming to Seoul

Seoul, Korea
November 1~4
Sensor, IoT, CPS, Mobile, Ubiquitous, Embedded,
…
Submission: April 3, April 10

Hold on a second …

- We have a lot of different emerging systems …
  - Mobile Systems
  - Internet of Things (IoT)
  - Cyber Physical Systems (CPS)
  - Ubiquitous Systems
  - Pervasive Systems
  - Embedded Systems

- How are they different?
- Are they really different?
- Why are there so many names?

PART I:

DEISGNING an EMMERGING MOBILE PLATFORM

Platform ? Design ?

- **Why Platform?**

- **Abstraction**
  - Users/developers/HW
  - Support **common functionalities**
- Optimization (runtime)

- OS .vs Platform ?

What is computer?

- What comes up to your head when you hear the word ‘computer’?
  - *What does it look like?*
  - *What do you do with it?*

Mobile/IoT Platform ?

Feature Phone

HW oriented, Communication centered

Smart-phones
iPhone, Android

Siri

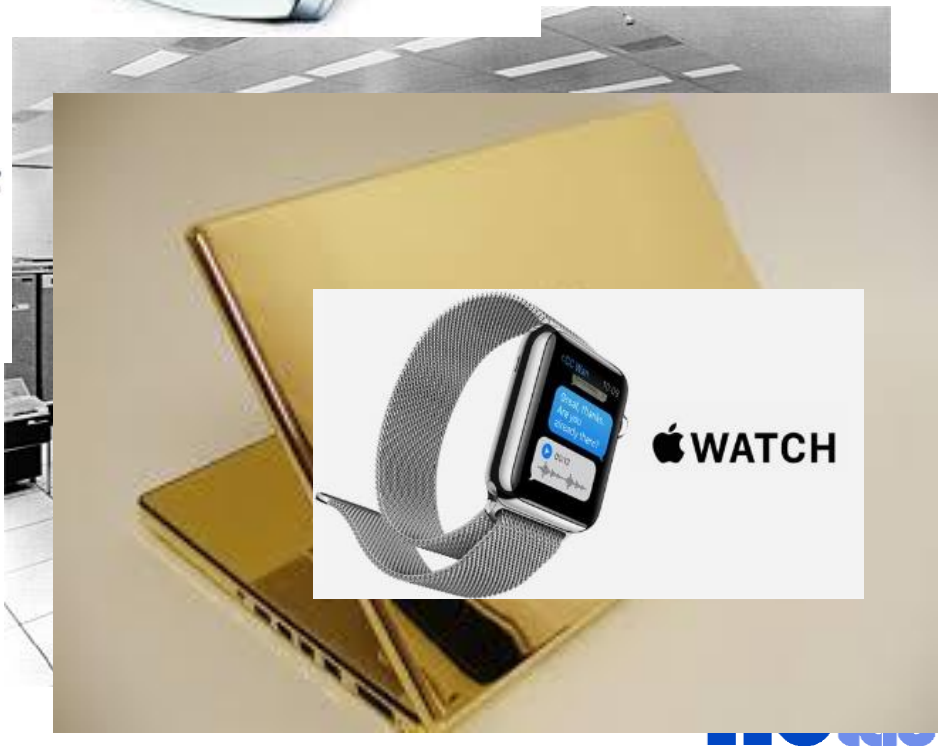
Use your voice to send messages, set remind search for inform and more.

New Mobile Interface
User satisfaction

New Mobile Platform

???

The Third Platform



the 1st generation

Laboratories

Scientists & Engineers

Complex science computation

the 2nd generation

on top of DESKS

Office workers, students

Desk tasks (pen-and-paper)

the 3rd generation

Hands or pockets

Everybody

???

the 1st generation

the 2nd generation

the 3rd generation

?

What do we do with Android?

- Early response:

  From an expert point of view, Android is not really a BIG deal. Mainly Linux with communication capability enhanced … …

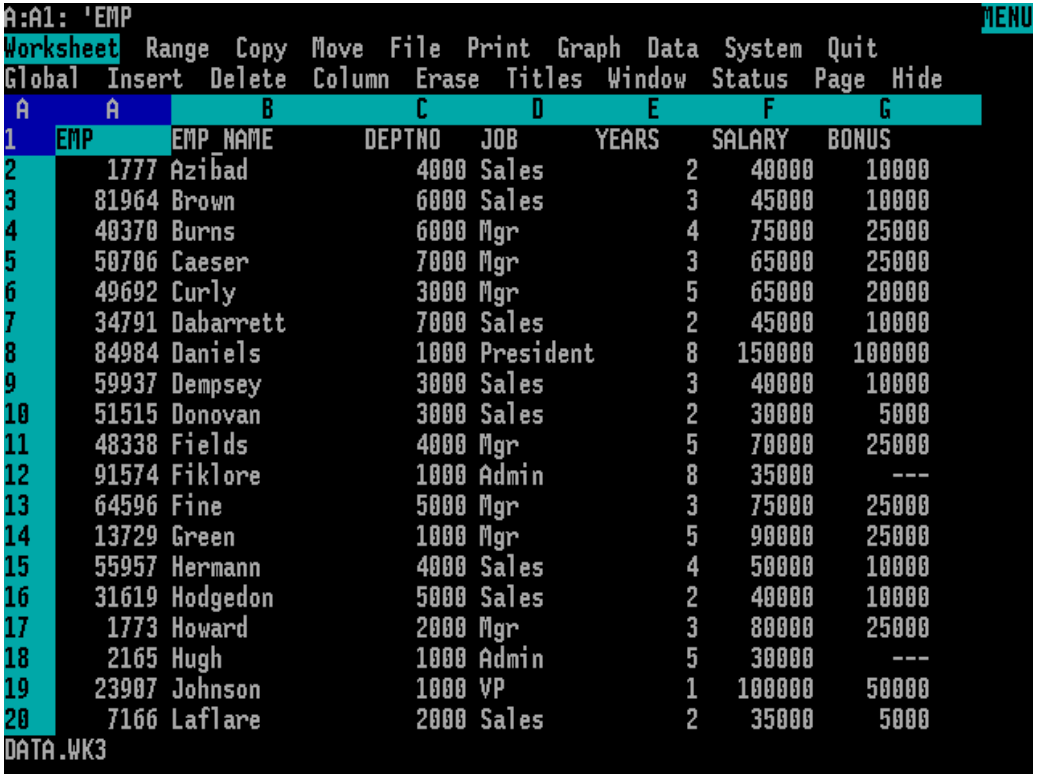
- What do we do now?

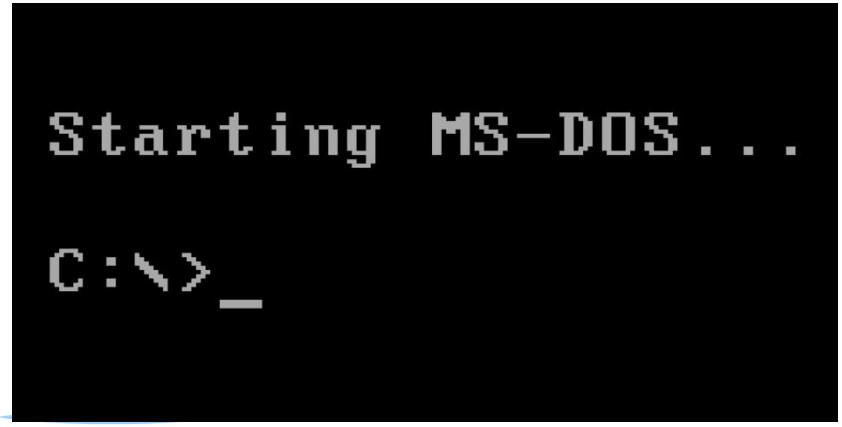
- Cf) future value
  - Experience with DOS, WWW, …

Two potentially different perspectives

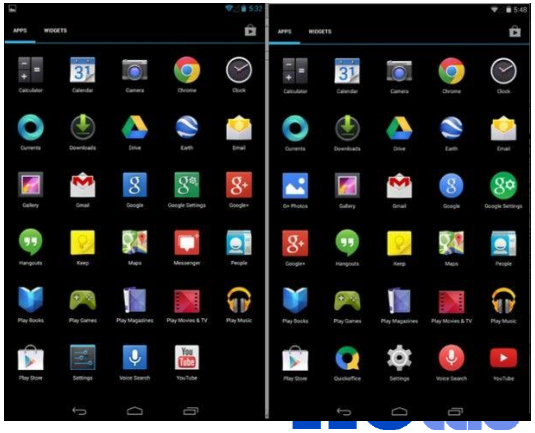
- System experts .vs **Designers** (Architects)
- Designers perspectives
  - Two platforms deployed in practice
  - A system comes with its CONCEPT
- What is a mobile platform?
  - What is the new concept of a mobile platform?

What did DOS mean?





What do iOS & Android mean?



"Mobile computing?"

People uses computers anytime and anywhere while they are moving

In reality, it means that I use computer in mobile or various other situations, i.e., in ***real life situations***

Part II:
LIFE-IMMERSIVE COMPUTING

Application Examples

Life-Immersive Computing

- What is it?

example 1) Aerobic Exercises


50%
of persons drop out of exercise
within 6 months
Wilson & Brookfield, 2009

Reasons for Exercise Drop-out

Lack of

- Social interaction
- Motivation
- Poor body image
- Expense & Time

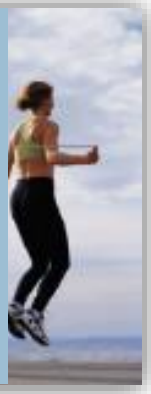
Len Kravitz
University of New Mexico

Aerobic Exercises: easily BORING

[CSCW'12, MobiSys'12, CSCW'13, CHI'14]

Repetitive-Individual-Monotonic

Multiplayer Games, i.e., *Social Exer-game*



(example 3) CHILDREN in a Kindergarten

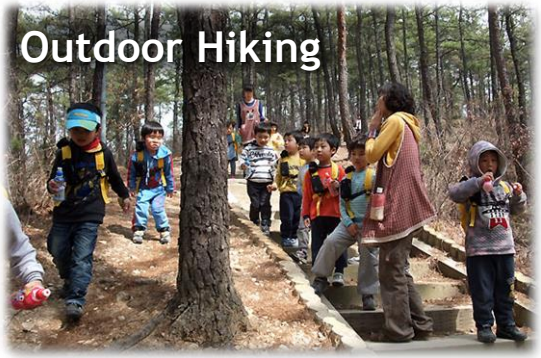
[UbiComp'10, Pervasive'12, UbiComp'12]

KINDERGARTEN

FIELD TRIPS

Setting

- ✓ 9~12 children (3~4 girls)
- ✓ 4~5-year-old
- ✓ 1~2 teachers
- ✓ Approx. 2 hours each

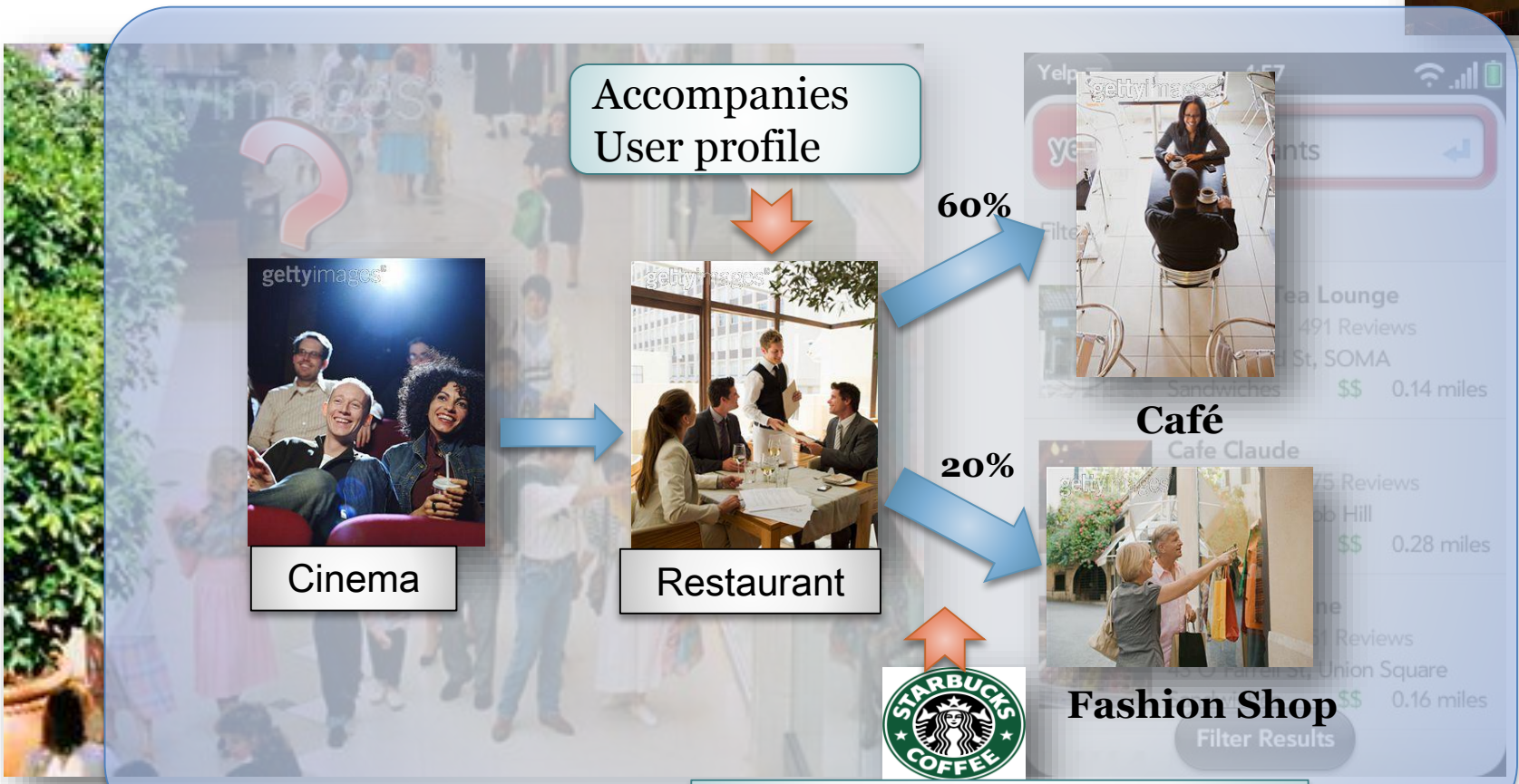



Head-counting


Potential dangers

(example 4) People in the COEX Mall

- AdNext+ : Next Visit-Predictive Ads [HotMobile'11]



* Extension of "AdNext: Visit-Pattern-aware Mo... Complexes, HotMobile 2011

PART III:

A Platform for **Life-immersive** Mobile Computing

Life-Immersive Computing

- I use a computer in mobile or various other situations, *i.e.*, in *real life* situations
  - Help users enhance their real life experiences …

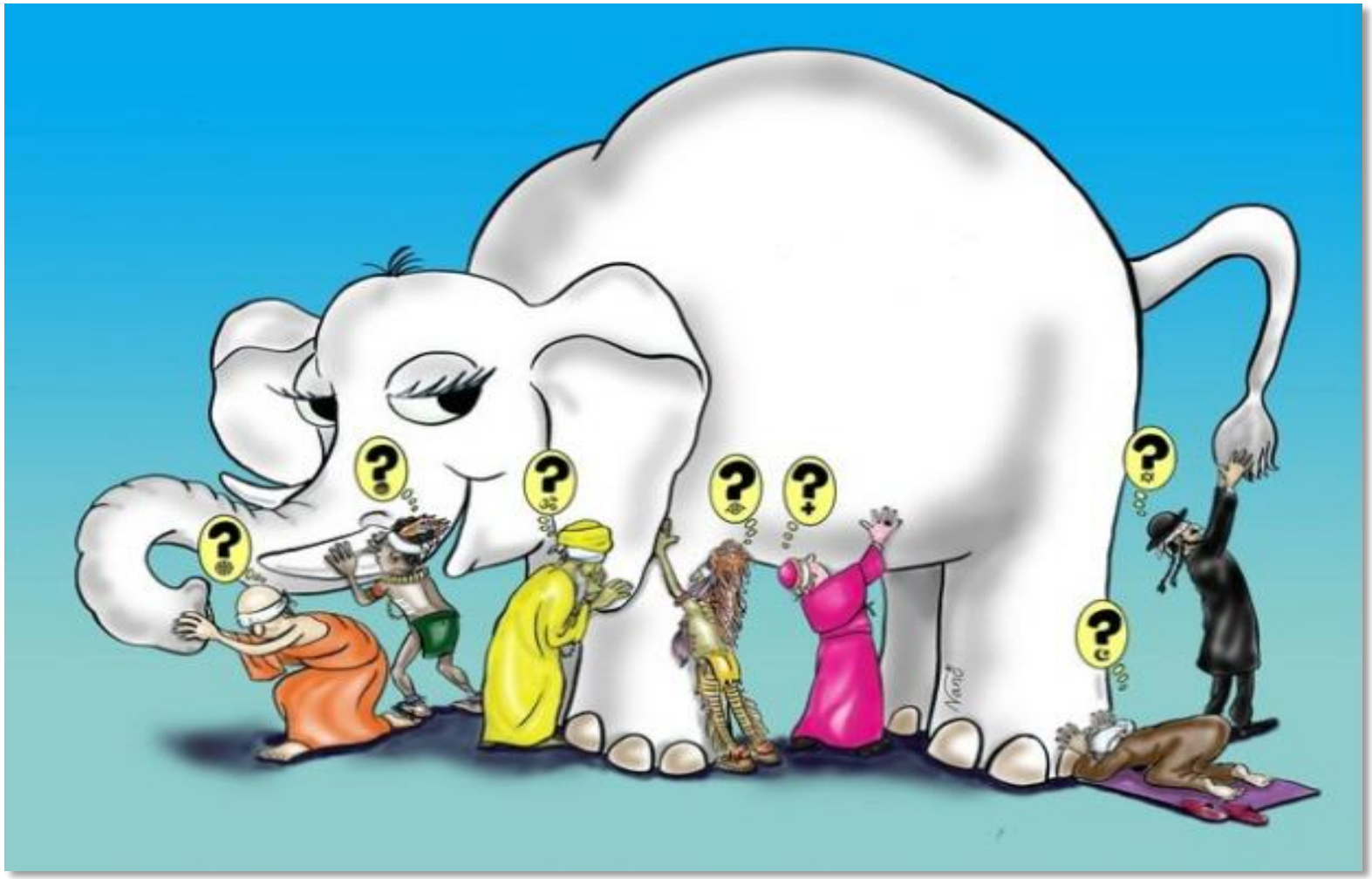
- **iOS** and **Android**
  - preliminary stage; re-produce PC apps; 'just-a-moment' (잠깐만) mobile computing !
    - MP3, AnyPang, AngryBird, Twitter, KaKaoTalk, …
  - Re-active

- Early examples of life-immersive apps
  - Navigator
    - satisfactory?
  - Quantified-self apps

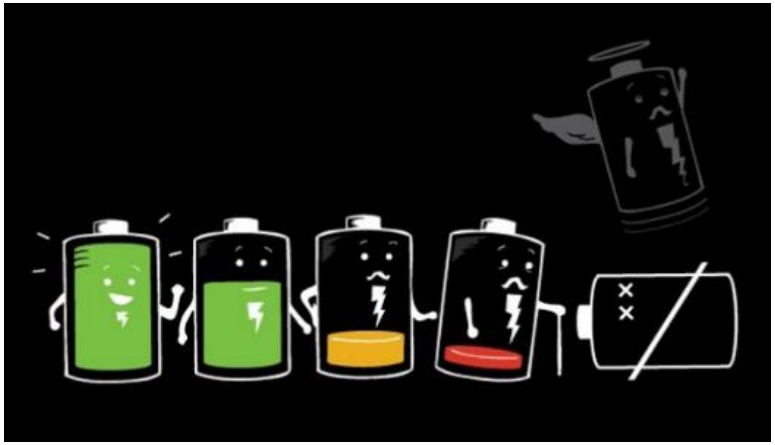
Requirements

- System in general
  - Performance (throughput, responsiveness, …)
  - Reliability
  - Scalability
- Mobile systems
  - Mobility
  - Energy
- Life immersive-ness, i.e., immersion to real life situations
  - **Situation conformity**
  - **Attention management**
  - **Privacy**
  - Situation awareness
  - Proactive-ness & automation
  - Natural interaction

Challenge: Limited REAL-WORLD Awareness

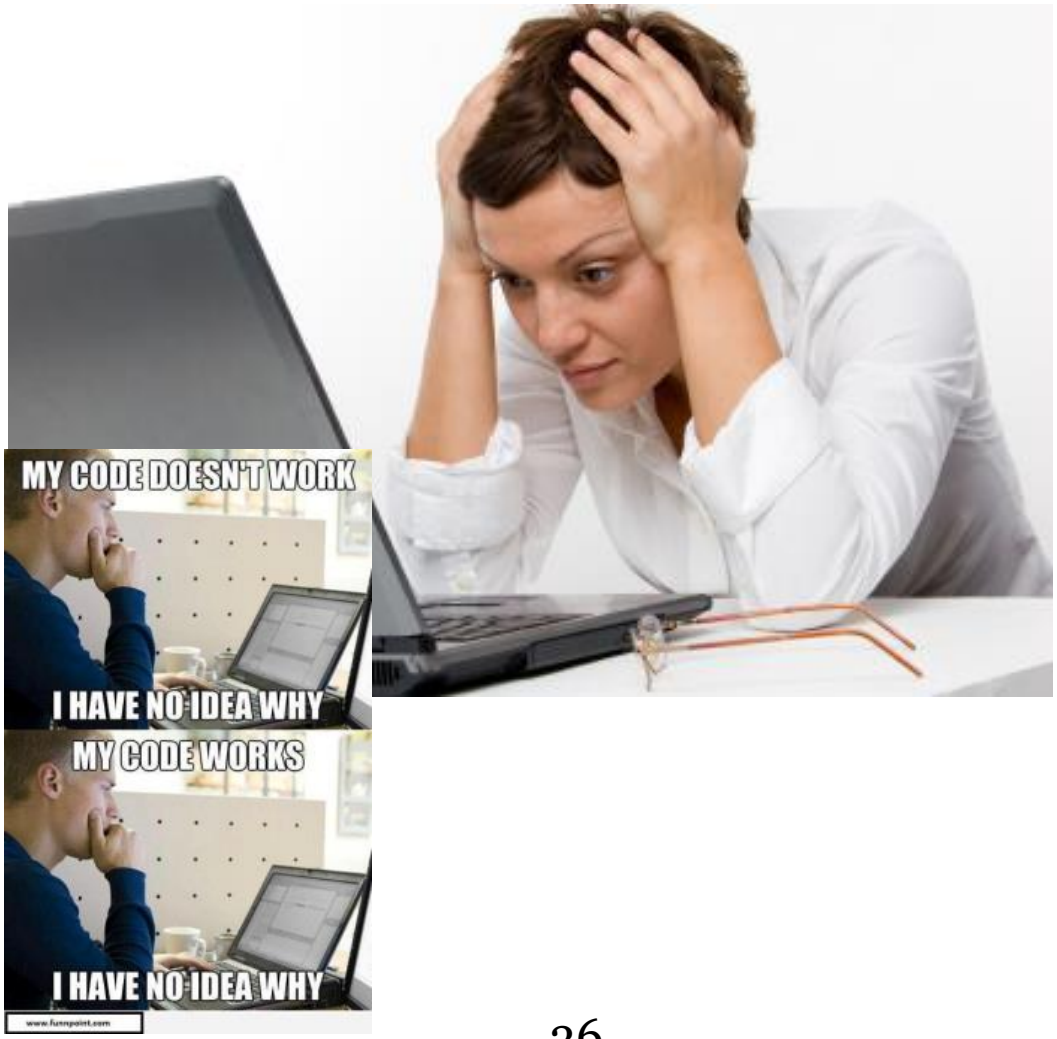


Challenge: High Resource Demand under High Limitedness



Challenge: Limited Interface

Can Applications solve these problems? Impossible!



Vision: Full-Fledged Platform for Life-Immersive APPs

Life-Immersive apps

AdNext+ ••• 

SympaThing ••• 

Fall monitoring

Simple and Intuitive API

Platform Support (Common Functionalities)

Abstraction of Runtime Sensing Resources

Rich sensor devices

Broader View and Limitations

Life-Immersive Apps

Specification/ Application Interface/ Usability

Conflict Coordination

Sensor Fusion/ Stream Processing

Cloud Support

Energy Efficiency

Inference Precision

Privacy/ Security

Resource Adaptation

Effective Learning

Communication

Sensor Device Abstraction

Rich Set of Sensor Hardware

Step I

FIRST ROUND ATTEMTS

First round attempts

- Various early attempts
  - Localization
  - Energy savings
  - Mobile cloud computing
  - Still going on …

  - API and languages
  - Technologies
  - Knowledge

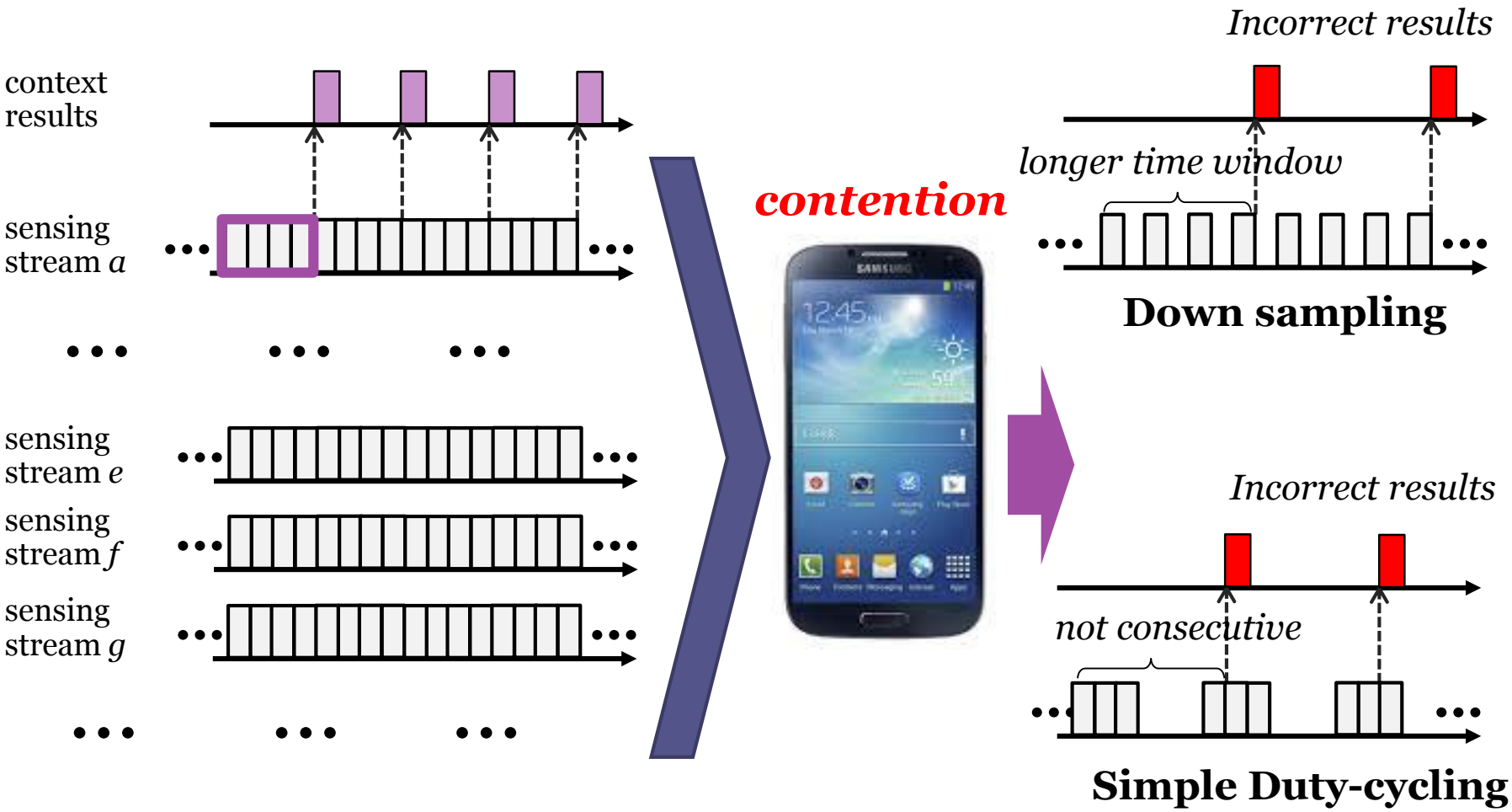
Findings from the platform design (step 1)

- New knowledge learned and technologies developed for a new platform

A. Language: CMQ interface and new APIs
B. Runtime design and implementation techniques
   1. Context monitoring (.vs context recognition) [MobiSys'08, TMC'10]
   2. Context continuity; eventually boils down to high degree of *locality*
   3. Essential sensor set
   4. Translation-based context processing
      1) Plurality of processing logics and
      2) Plurality of sensing modalities
   5. Frame-based flow scheduling with flame externalization [SenSys'12]
   6. Synchronous flow processing [PerCom'10, TMC'14]
   7. Dynamic sensor abstraction
   8. Collaborative gesture architecture with closed-loop fusion [SenSys'11]
   9. Real-world (.vs in-lab) mobile experiments

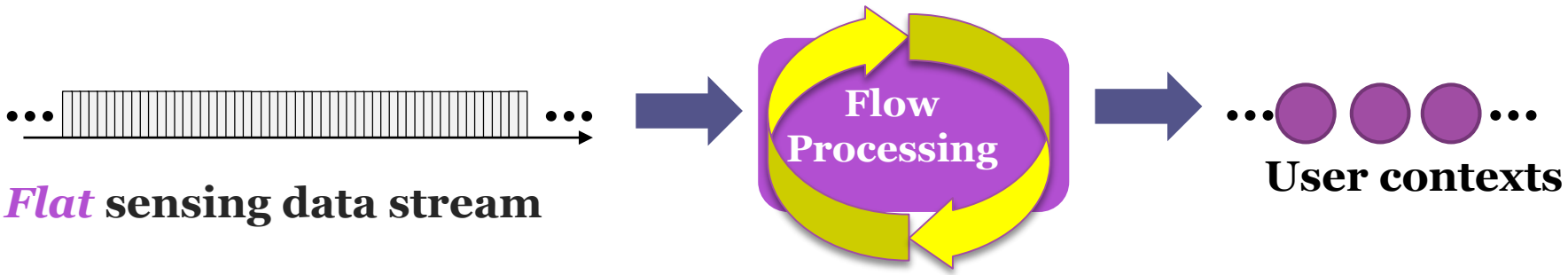
Sensing Stream Scheduling and Processing [SenSys'12]

- *Contention* over limited resources



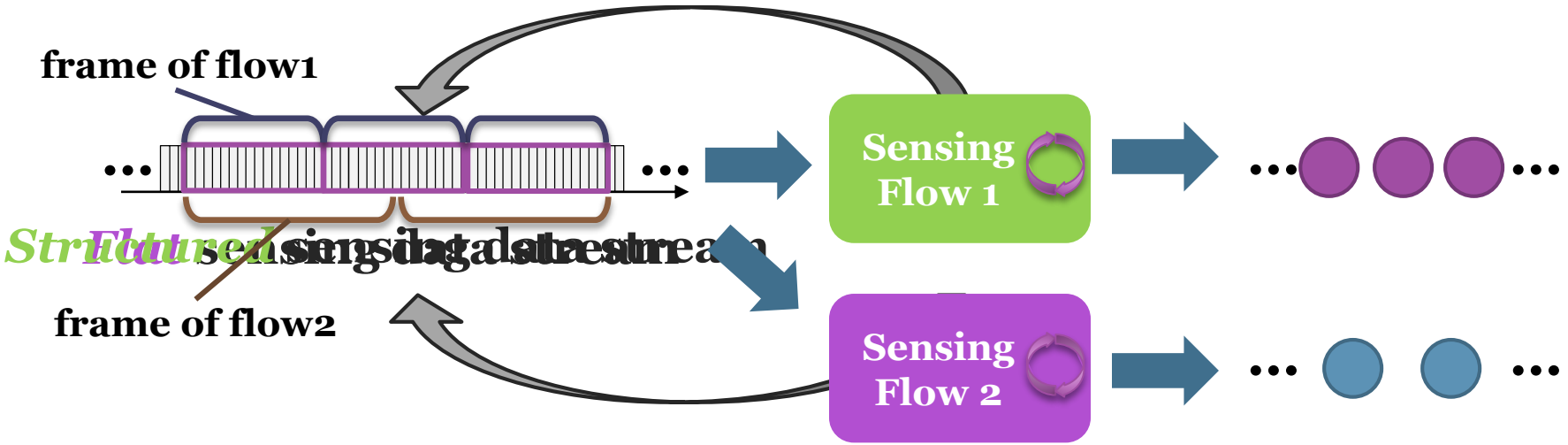
Unstructured stream

- Scheduling units?
- Processing units?



Frame Externalization

- Externalize semantic structures embedded in sensing data stream
  - provides useful hints for sensing flow coordination

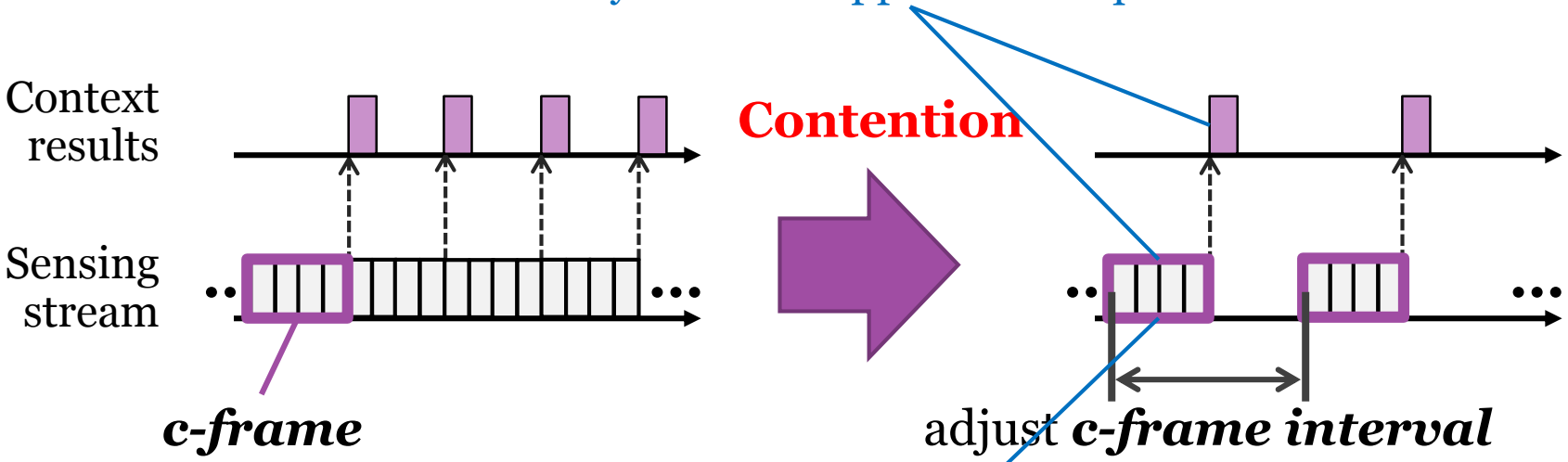


Different structures for different applications

Frame-based Flow Scheduling

- *c-frame* as the basic unit of resource allocation

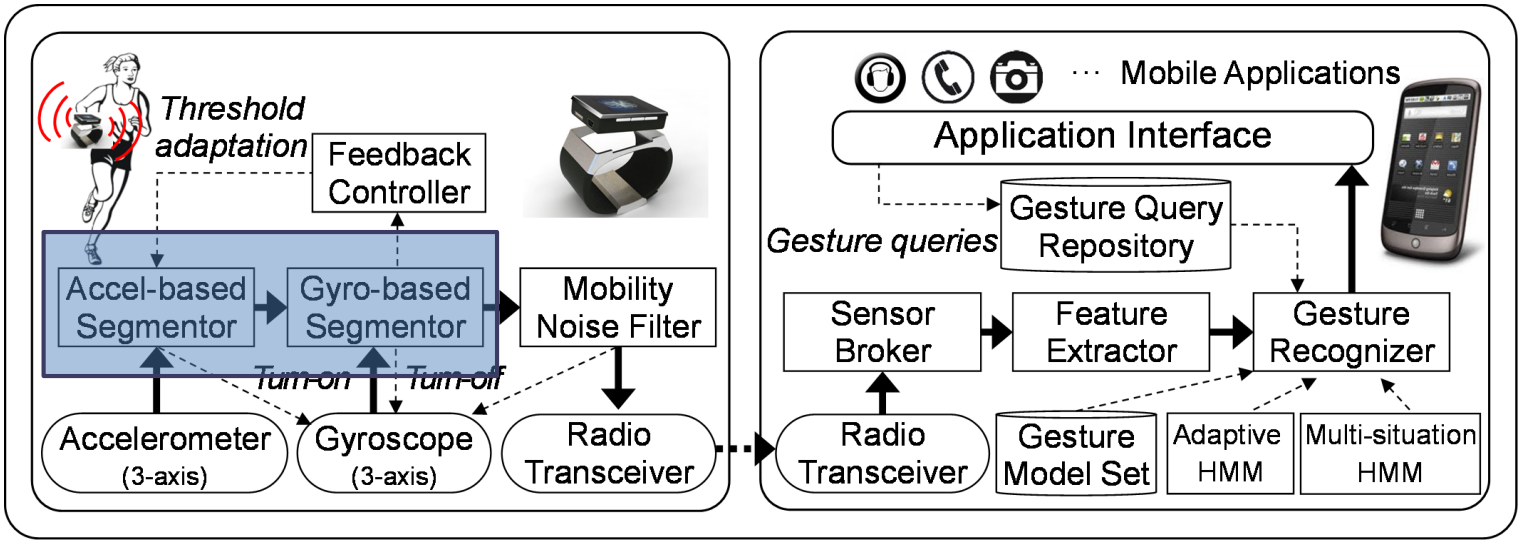
A c-frame = A result
→ Easy to reflect application requirements



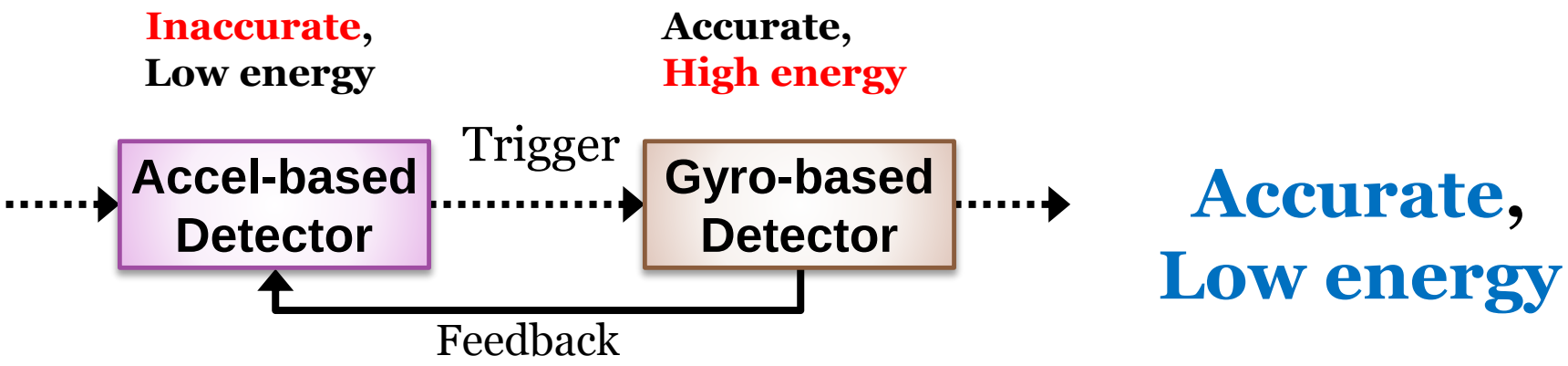
Ensure the deserved amount of resources
for an allocated c-frame → Correct results

Gestural Interaction [SenSys'11]

- excessive energy overhead: continuous monitoring
- Collaborative fusion architecture
  - Inter-device Collaboration & Inter-modality Collaboration
  - Closed-loop collaboration architecture of hybrid sensing modalities



Closed-loop Collaborative Detector



Performance-preserving, Energy-saving Collaborative Sensor Fusion

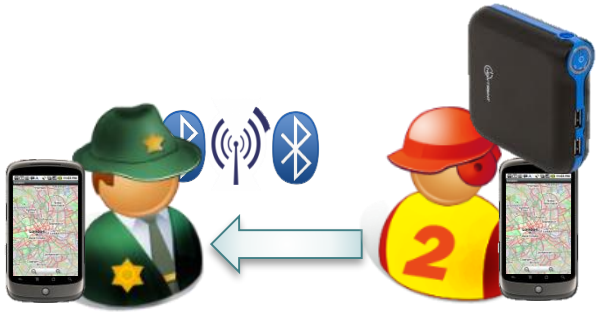
Step II

MULTI-DEVICE COLLABORATION ARCHITECTURE for Mobile Platforms

CoMon: In-Situ Collaboration Architecture [MobiSys'12]

Expected Power Savings

I Travel Alone

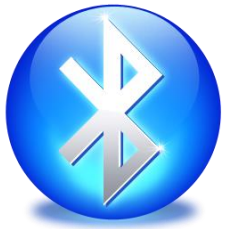


I Meet Young

I Meet Brian

≈ 440 mW

≈ 315 mW



≈ 80 mW

≈ 110 mW

≈ 315 mW

CoMon: Cooperative Ambience Monitoring Platform


GPS
Pulse meter


Air Quality
(dust)

...


Heartrate
Sound

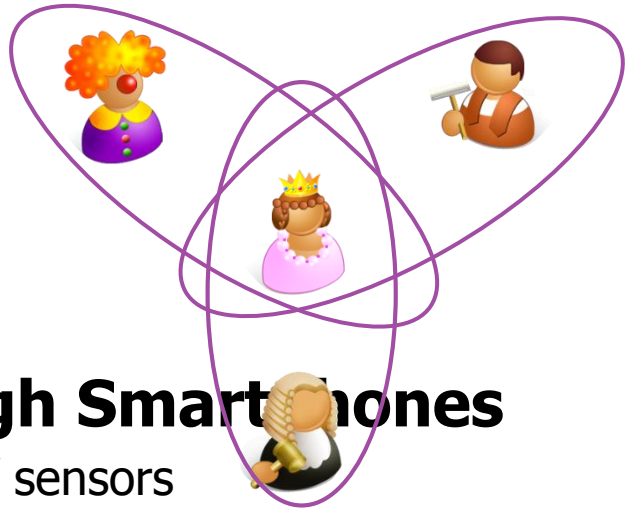
- **Collaborative Sensing as a Platform**
  - *reqContextUpdates("Loc", 10 sec, hdler);*
- **Support Diverse Ambience Contexts**

Location	Indoor Place	...	Air Quality
Events (Music, ...)	Group Activity		Crowdedness

- **User-Unobtrusive Collaboration**
  - Automated cooperation when people meet
  - No manual initiation by users
- **Energy Benefits and More**
  - Cooperate when beneficial (e.g., saving battery)

Design Choices

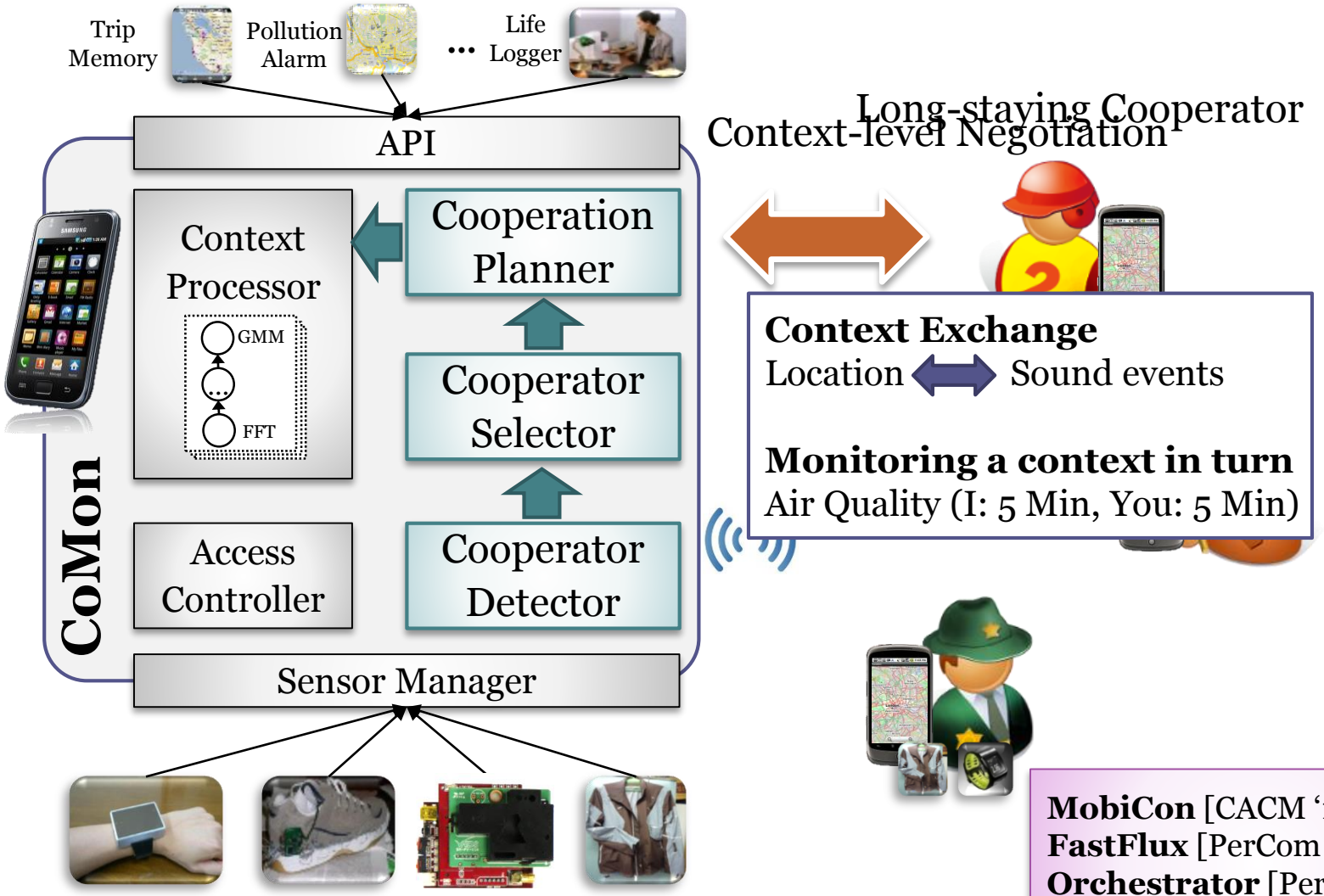
- **Cooperation with Long Stayers**
  - Low overheads for cooperation
    - Less frequent discovery, negotiation, and connection mgm.
  - Little service vacancy for continuous sensing



- **Pair-wise Cooperation**
  - Less venerable to user mobility
  - Low negotiation complexity

- **Sharing High-level Context Through Smartphones**
  - Hide heterogeneity and dynamics of cooperators' sensors
  - Save energy by not exchanging raw sensor data (e.g., sound)

Architecture



MobiCon [CACM '12]
FastFlux [PerCom '12]
Orchestrator [PerCom '10]
SeeMon [MobiSys '08]

Mobile Multi-speaker Audio Apps: a camping scenario

[UbiComp'14]

Mobile Maestro:

Enabling Mobile Multi-Speaker Audio Applications on Commodity Mobile Devices

Hyosu Kim[1], SangJeong Lee[2], Jung-Woo Choi[3], Hwidong Bae[1], Jiyeon Lee[1], Junehwa Song[1], and Insik Shin[1]

[1] Department of Computer Science, KAIST
[2] Software Center, Samsung Electronics
[3] Department of Mechanical Engineering, KAIST

Part IV

Research OUTLOOK: LESSONS, DISCUSSION

Reflection

- A novel mobile device that Korean industry designed …
  - Was it successful?
  - Why?
  - Why not?

Google Glasses

- 100 reasons they are good
- 100 reasons they are bad


58°
6
Subway service suspended
jessica wants to talk to you.


Paul is 402 ft away
jessica wants to talk to you
Accept

Design of a Future Platform (System)

- Workload .vs HW (environments)
- Application .vs Platform
  - (How) Do we know future applications?
  - Can we come up with a new platform without knowing new applications
- Who defines a new platform?
  - Expert .vs Designer (Architect)
  - Value creation?
- What is the notion of PERFORMANCE?
  - Up to Human Satisfaction

THANK YOU VERY MUCH

Questions?

Junehwa Song

junesong@cs.kaist.ac.kr
School of Computing
KAIST
http://nclab.kaist.ac.kr
TEL) 042 350 3546