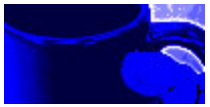
Smart-Its: Prototyping the Disappearing Computer

Hans-W. Gellersen

Lancaster University
Department of Computing
Ubiquitous Computing Research





Prototyping in Ubicomp Research

Ubicomp Research Methodology

- *"The research method for ubiquitous computing is standard experimental computer science: the construction of working prototypes of the necessary infrastructure in sufficient quantity to debug the viability of the systems in everyday use, using ourselves and a few colleagues as guinea pigs. This is an important step towards insuring that our infrastructure research is robust and scalable in the face of the details of the real world."*

Ubicomp Research Methodology

- **Roy Want**, Intel Research, formerly XeroxPARC:
  "10 Lessons Learned about Ubiquitous Computing"

  Lesson 4: Only building something actually allows you to explore its full design potential

… Lesson 9: It's a lot of work to deploy a Ubiquitous Infrastructure and it's even more work to maintain it

The Importance of Building Prototyping

Learning from building a prototype

- Feasibility, Limitations, Trade-offs, Pitfalls
- e.g. Pin&Play: Does it work? Does it scale? What is the tradeoff between size/bandwidth?

Discovering issues you'd never have thought of

- e.g. Sense Table: structural properties of table limits accuracy – no point in sampling of load sensor data at higher rate

Chance innovations

- e.g. Sense Table: designed for context acquisition – use as "mouse" emerged once it was built

The Importance of Building Prototypes

Building Prototypes

- Envisioning Design
- Tool for communication of ideas
- e.g. Sense Table inspired series of designs at RCA

The Importance of Deploying Prototypes

Deploying, not just demonstrating!

- Evolving an understanding of how a system fits into a physical and social context
  - e.g. PARCTab experience
    - in demos: "mobile email" most popular with users
    - in the deployed system:
      - Tabs had only connectivity in places that also had workstations that were better suited for email
      - exception: the meeting room, but there email was too slow because many people were using it simultaneously

The Importance of Deploying Prototypes

Evaluation in context

- Context-aware behaviour can only be evaluated in context
- Requires deployment in the real world
- Observation possibly over extended periods
  - Time to get over the wow! factor
  - Often interested in the infrequent odd cases
  - Observation difficult, not always acceptable

The Importance of Deploying Prototypes

The idea of "Living Laboratories"

- Instrumented real-world environments
- Longitudinal studies
- e.g. the Aware Home at Georgia Tech

Physical Prototyping Challenges

Physical prototyping

Ubicomp inevitably involves physical prototyping

- Pre-Ubicomp research in areas such as computing systems, human-computer interaction etc:
  - Usually dealing with software as the variable and hardware as the constant
- Ubicomp research
  - Prototyping of both software and physical system components
  - Exploring new device concepts (e.g. information appliances, wearables, …)
  - Exploring new device interactions (e.g. sensing and manipulation of the environment)

Problems with Physical Prototyping

Skills in the wrong camp!

- Ubicomp research is driven by communities that traditionally have been software-centric
- Hardware people aren't interested in Ubicomp prototyping (not challenging – they rather advance the enabling technologies)

Problems with Physical Prototyping

Construction difficult

- Hardware available typically at wrong level of abstraction, or difficult to interface
- Prototyping can involve PCB design and selection of electronic components at a very low level
- Not more than 1 device per research group per year
- Rarely iterating on design

Irreproducible

- Results can not be reproduced by others
- Demo hardware vanishes

Many issues ignored or poorly studied

- Context, interaction with environment, …

Problems with Physical Prototypes

Ubicomp Catch 22

- Ubicomp applications cannot be developed without suitable hardware
- But the design of suitable hardware depends on an understanding of how the hardware fits into the application environment

What Researchers can do about it

- John Barton, HP Labs

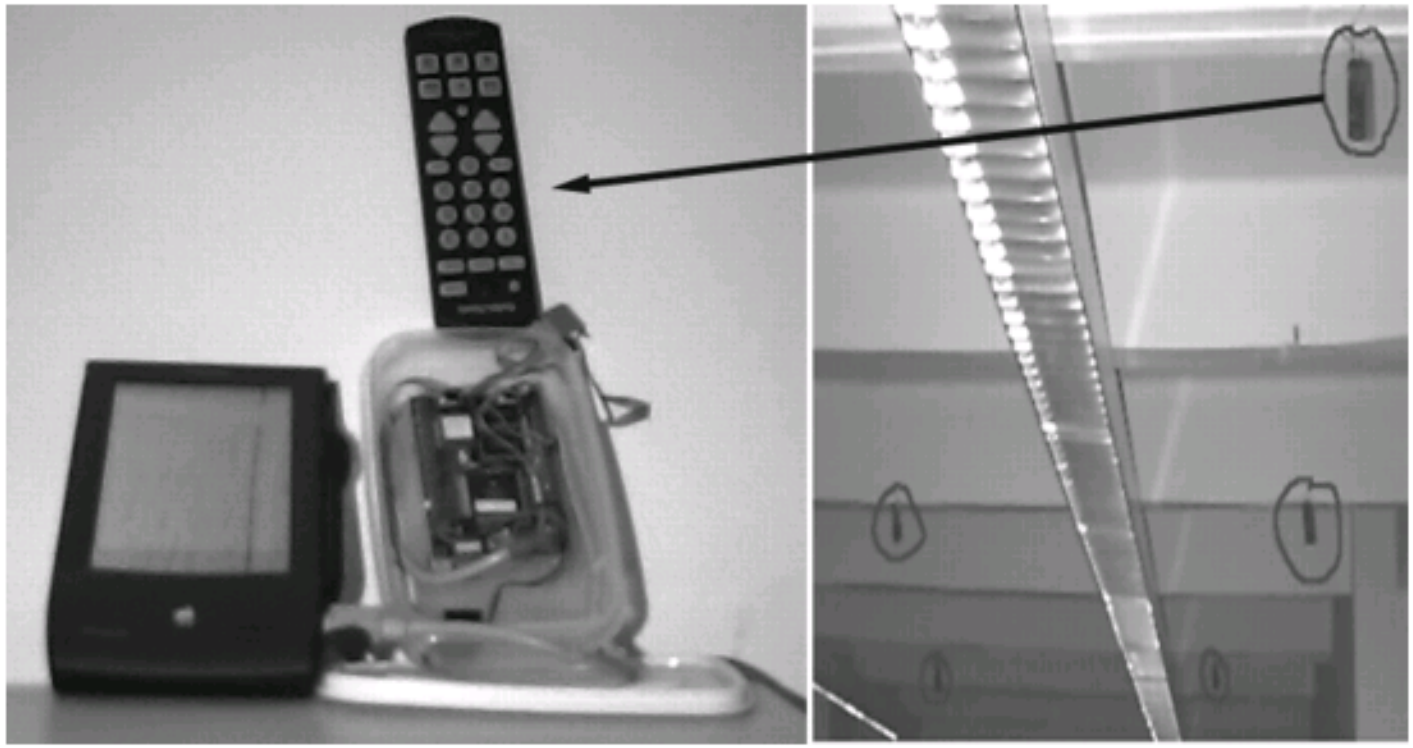
Services for wireless handheld appliances
…but you don't have any…

- Plan A: whine about it
  - Work on services; wait for appliances
- Plan B: Hack one
  - Hardware, device drivers, …
- Plan C: Build one
  - See Plan B.
- Plan D: Fake one

Plan B: Hack One

Ingeniuty diverted to Hardware Improvisation!

- e.g. TV remote controls as IR beacons (Cyberguide project, Gatech 1996)



Plan D: Fake One

Simulation of physical prototypes

- Idea: back to virtual reality to explore ubicomp environments before they are build
- Closed world, no surprises: you get what you model
- Cannot replace studies in a real context

- John Barton's Ubiwise Simulator

The Ubiwise Simulator

Two window virtual world

Close-up Device View

Physical Environment View

Back to Plan C: Build One

But lower the Hurdle

- Provide prototyping platform
  - hardware/software building blocks, architecture
- Functionality
  - embedded processing
  - wireless communication
  - physical interaction (sensors/actuators)
- Constraints
  - Price, size, energy consumption matter now (can't wait for the future to happen)

Smart-Its

Smart-Its

Architecture

Small embedded devices

- Sensor module for local observation
- Embedded perception to lift sensor data to context
- Radio module for sharing of context in ad hoc networks

Building Blocks

- Family of devices
- Software libraries
- Highly configurable sensor units
- Alternative wireless technologies

Smart-Its

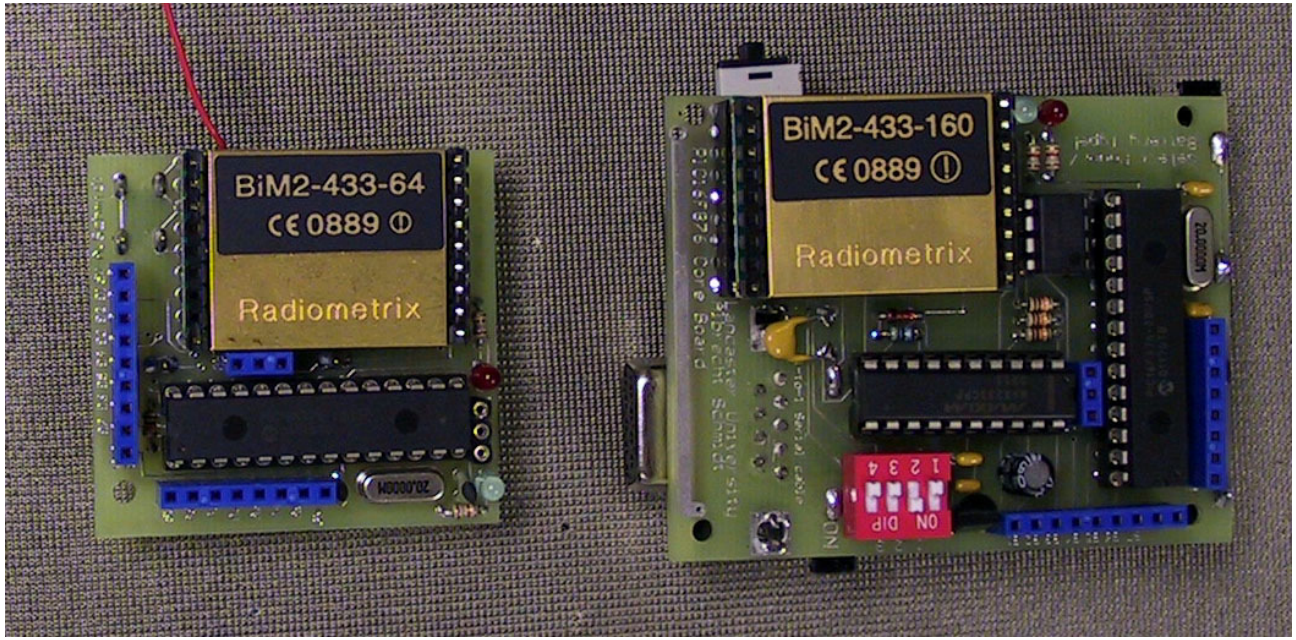
Smart-Its Background

- Mediacup Project, 1999-2001
- Demonstrator for Digital Presence

Mediacup Implementation

- Cup with embedded sensors, processor, wireless comm., and wireless charging
- Autonomous computation of user-level context („filled up", „gone cold", ...)

Smart-Its Platform



All base boards

- Microcontroller
- RAM
- Analog Inputs
- Digital I/O
- Wireless communication
- All boards are software and hardware compatible

Small portable unit

- 45mm x 50mm x 19mm
- 29g with battery

Base station/debug unit

- 55mm x 70mm x 29mm
- 110g with 4x AAA
- RS232 connector
- DC Power Connector

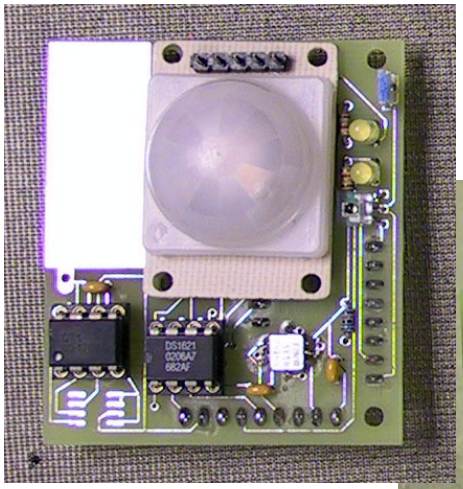
Sensor boards
Add-ons to the Core Smart-Its

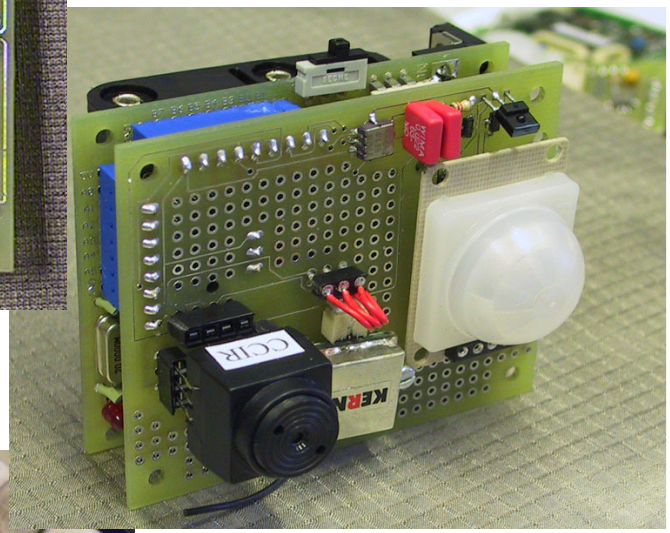
Hardware
- Much simpler

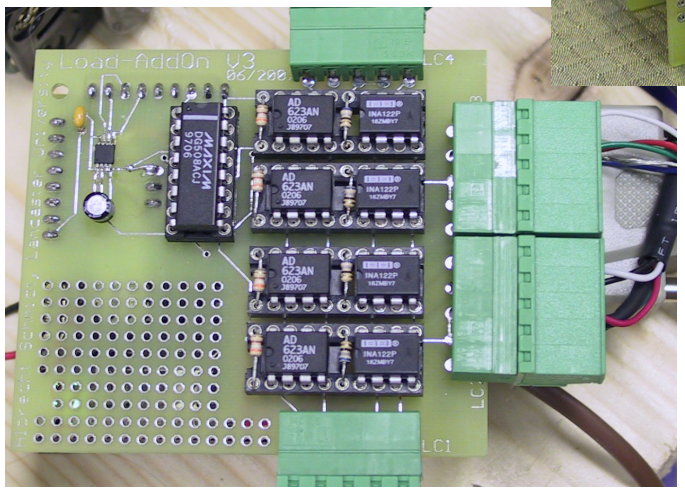
Software
- Build upon frameworks

Communication
- Basic functions available

Examples
- General sensors
- Vision / Camera
- Load sensing
- Weather board
- Motion sensing
- Actuator boards
- ...







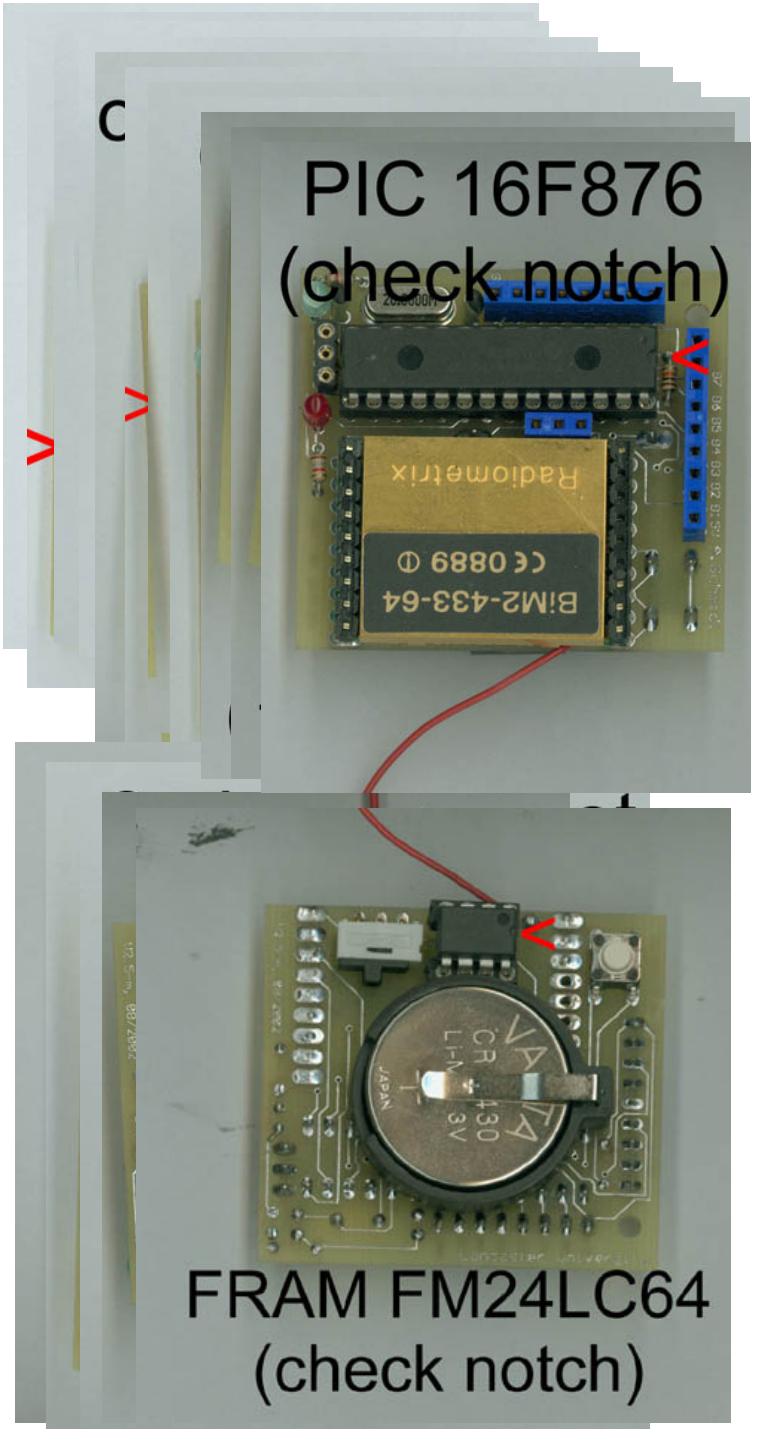
Hardware
DIY Approach

Selected requirements

- Understandable with a CS background
- Minimal electronics skills
- Similar to electronic kits
- Easing embedding of sensors and actuators
- Reusable
- Basic hardware and software should run within a week for most scenarios

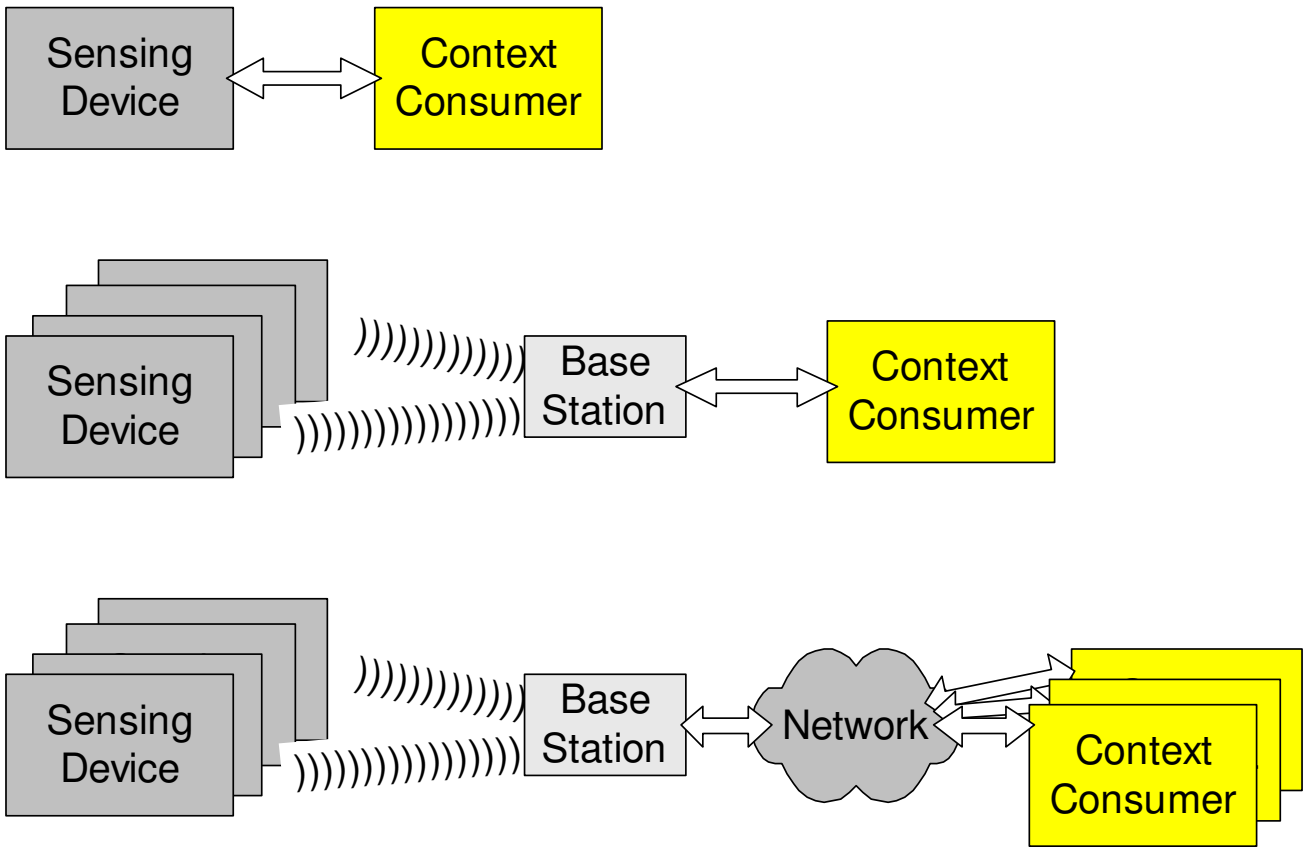
Beyond rapid prototypes

- All components as CAD unit
- Different physical shapes (of the same HW) are easy to do
- Software compatible to modules



Software
Context Acquisition Systems

Architectures & Software Frameworks



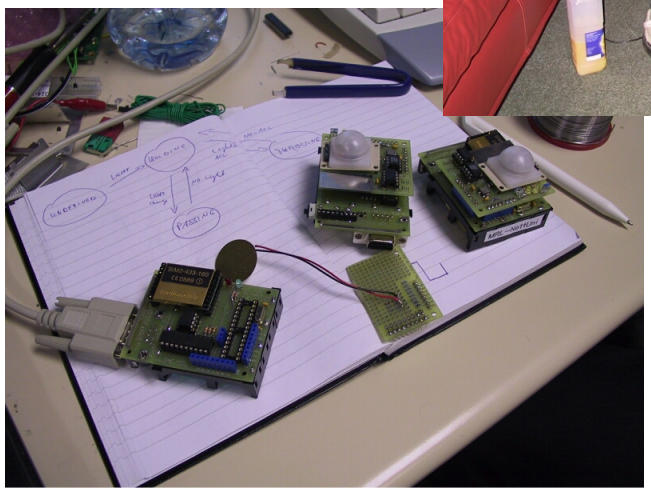
Platform Evaluation

Evaluation Method

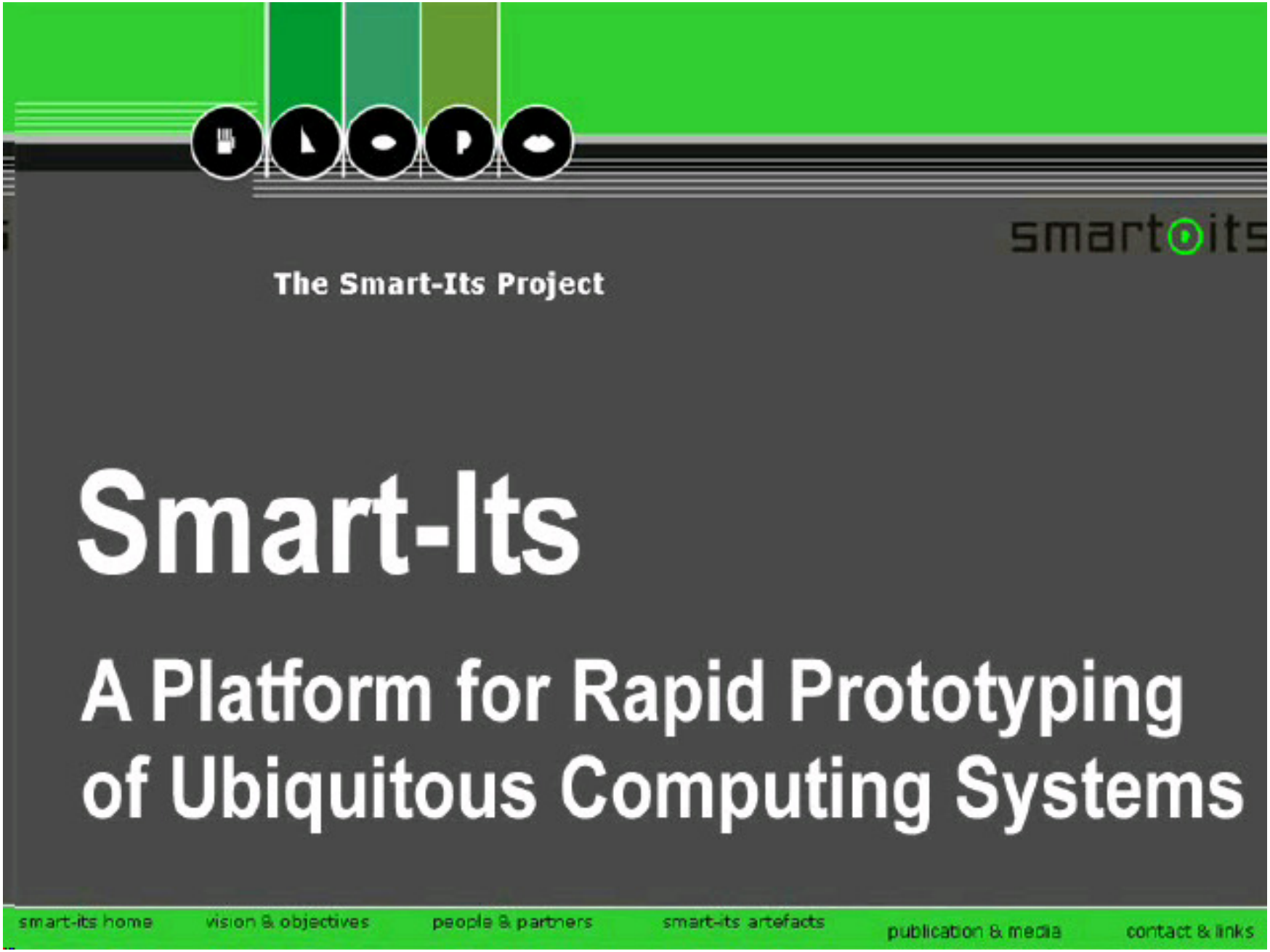
- Workshop with users from other projects
- 2,5 days for development of mini-demos from scratch

Results

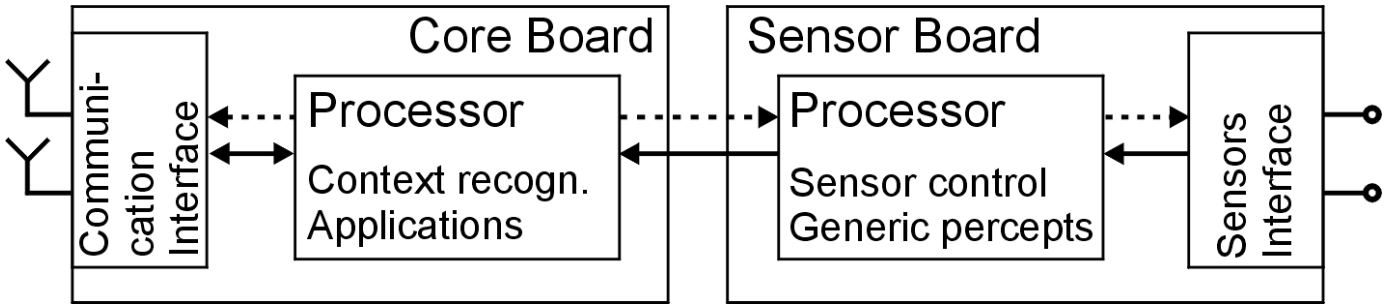
- Hardware: new add-on devices
  - e.g. RFID reader
- Application demos
  - Smart Ball
  - Wireless Gesture Input
  - Perception of walking in groups
  - ...



Prototyping Exercise - Impressions

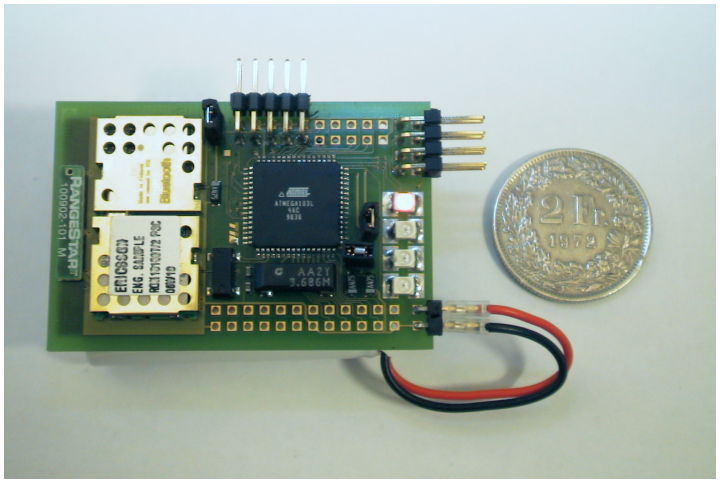


More Smart-Its Devices



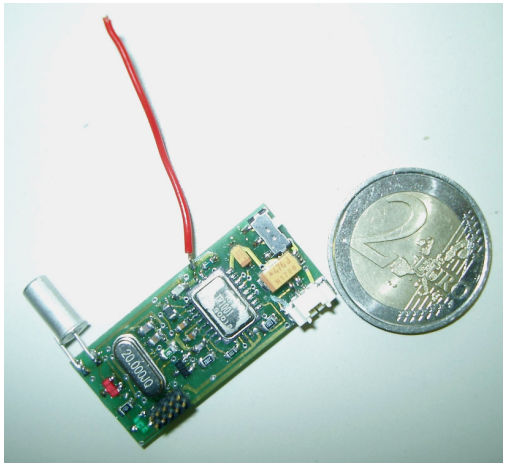
ETH Zurich

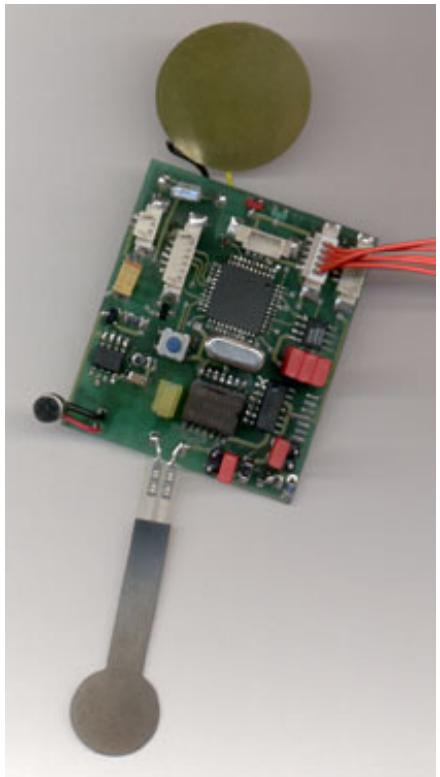
- Bluetooth node
- Interoperability



TecO, Uni KA

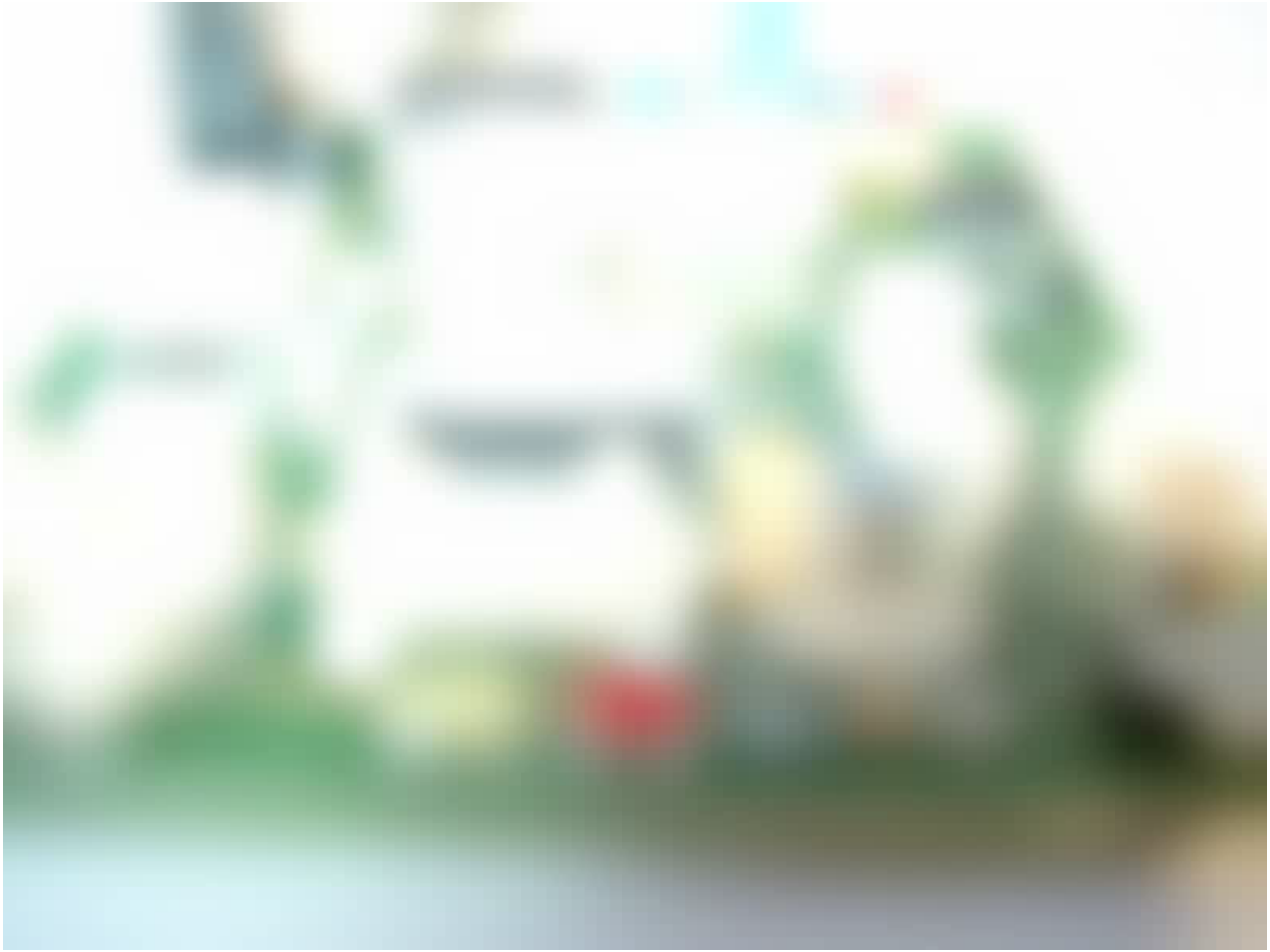
- Minimal size
- Minimal energy





Beyond Prototyping

- Development of Prototyping Platform is vehicle for further research in the Smart-Its project
- Exploring "Smart Artefacts" Vision
  - Applications composed of common objects that have processing and context-awareness embedded
- Studying Collective Awareness
  - Context-awareness based on shared observation
  - Integration of diverse types of sensor
  - Perception processes that scales with number of spontaneously connected sensors



Smart-Its

Acknowledgements

- University of Karlsruhe, Germany
- ETH Zurich, Switzerland
- Interactive Institute, Sweden
- VTT Electronics, Finland

More information

- http://www.smart-its.org/
  and partner sites

Summary

- The importance of building and deploying prototypes in ubicomp research
- Challenges associated with physical prototyping
- The Smart-Its platform designed to lower the hurdle for building, trying out and evaluating prototypes of the Disappearing Computer

- More information, links to project sites
  http://www.comp.lancs.ac.uk/~hwg/
- Contact
  hwg@comp.lancs.ac.uk

Thank you!

Readings

Ubicomp Prototyping

[1] R. Want, "Ten Lessons Learned about Ubiquitous Computing", Invited Talk at Dagstuhl Seminar on Ubiquitous Computing, Sept 2001, http://www.inf.ethz.ch/vs/events/dag2001/slides/roy-lessons.pdf

[2] N. Davies and H. Gellersen, "Beyond Prototypes: Challenges in Deploying Ubiquitous Systems"

[3] C. Kidd, G. Abowd et al "The aware home: A living laboratory for ubiquitous computing research", Proc CoBuild 1999, Springer-Verlag

[4] J. Barton, "The Ubiwise Simulator", Lecture at Dagstuhl Summer School on Ubiquitous Computing, Aug. 2002, http://www.inf.ethz.ch/vs/events/dag2002/

[5] S. Long et al "Rapid Prototyping of Mobile Context-Aware Applications: The Cyberguide Case Study", Proc. Intl. Conf. on Mobile Computing (MOBICOM'96), ACM Press, pp. 97-107

Smart-Its

[5] http://www.smart-its.org/

[6] L.E. Holmquist et al, "Smart-Its Friends: A Technique for Users to Easily Establish Connections between Smart Artefacts", Proc. Ubicomp 2001, Springer-Verlag

[7] "Smart-Its: Technology for Smart Artefacts with Collective Awareness", Technical Report, http://www.comp.lancs.ac.uk/~hwg/Smart-Its.jamb2002.pdf

[8] A. Schmidt, "Smart-Its Hardware Tutorial" http://www.comp.lancs.ac.uk/~albrecht/smart-its/platform/

[9] Smart-Its Overview Video http://ubicomp.lancs.ac.uk/~hwg/Smart-Its_640_480.mov