Smart Vacuum Cleaner –
An Autonomous Location-Aware Cleaning Device

Svetlana Domnitcheva
Institute for Pervasive Computing
Swiss Federal Institute of Technology (ETH) Zurich
8092 Zurich, Switzerland
+41 1 63 20636
domnitch@inf.ethz.ch

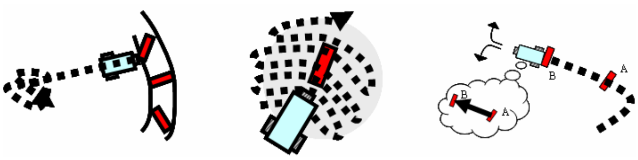
ABSTRACT
For many devices in a ubiquitous computing environment the ability to recognize their present position is an essential feature. Nowadays, stationary or mobile Radio Frequency Identification (RFID) [1] antennas are often used to detect the position of mobile RFID-tagged objects (e.g., RFID gates in logistics; RFID combined with motion trackers in *Magic Touch* [2]). In this poster, we present a prototype of an autonomous mobile cleaning robot which implements an opposite, *self-positioning* approach. Enhanced with an RFID antenna, this device learns its present location by detecting small RFID tags spread on the floor and adjusts its behavior according to this information.

Keywords
Ubiquitous computing, self-positioning, context, RFID, BTnode, Lego Mindstorms

INTRODUCTION
While ubiquitous computing technology is expected to become very small and reasonably cheap in the future, so that large amounts of electronic items can be deployed almost everywhere, such perspectives make it interesting to take a look at scenarios where small chips would become so plentiful that this will not only allow identification of relatively valuable single objects but could also be simply distributed over surfaces, like in scenarios emerging from sensor networks research. Identification of areas and spots on a writable surface using visual recognition of unique patterns printed on paper has been made in *Anoto* [3] technology. Yet, for instance, RFID tags spread over or weaved into a passable surface (e.g., carpet on the floor) could also provide us with interesting new applications, like allowing mobile devices to obtain their current location or context information on the spot, without rather expensive positioning calculations. The aim of the presented work was to examine the possibilities and potential problems of such an *RFID-based self-positioning method*.

Scenario
As a scenario for this prototype, we considered efficient and, as far as possible, autonomous cleaning of a previously identified area on the floor (similar to functionality of *Trilobite* [4] or *Roomba* [5] vacuum cleaners, yet considering RFID technology instead of ultrasound or infrared). Regarding the positioning problem, the setting for this vacuum cleaning task included a special test surface with RFID-tags distributed on it, and a mobile cleaning robot, which had to be able (by learning its location from the detected tags) to stay in the area meant for cleaning, recognize already cleaned spots and avoid already taken paths. To provide higher autonomy, additional constraints, such as mobile power supply and independent data processing, were considered.

Behavioral Patterns
The designed *Smart Vacuum Cleaner* (SVC) can be put into one of the following states. First, so-called *Marking Mode,* during which said device is moved along the border of the required area to mark all seen tag IDs as “border tags”. After the border has been defined this way, SVC changes into *Cleaning Mode* and chooses the actions to follow in accordance to the last tag that was detected. Originally all tags are assumed to be “dirty”. If such a tag is seen, SVC switches on the vacuum cleaner, performs the cleaning movement and marks the tag ID as “clean”. When it approaches a border tag, special movements are performed in order to stay in the area and to avoid already taken paths. In addition, for every detected tag a *déjà-vu* check makes sure that in case the same sequence of two tags had occurred before, an alternative direction of movement will be chosen this time.



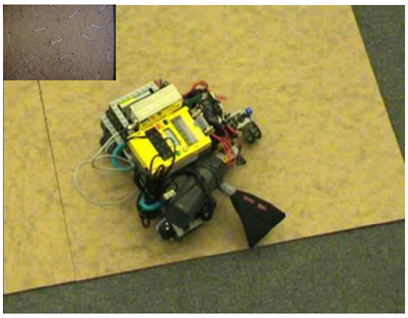
border case *cleaning mode* *déjà-vu*

SYSTEM COMPONENTS
To perform its task with no external support, the system is equipped with an RFID reading device and provided with an additional power source. The RFID antenna, placed in the lower part of the device, enables recognition of the tags

on the floor. All data is processed in the device itself, so that no external computational support is needed.

Equipment Used for the SVC Prototype

The constructed prototype is a battery-driven, mobile, autonomous vacuum cleaner built up with *Lego Mindstorms Robotic Invention* kit [6], equipped with a *Hitachi*[1] *μ-chip* [7] RFID antenna for tag recognition, a *BTnode* [8] control and computation unit, and a small vacuum cleaner. The size of the assembled device is approx. 20 x 20 x 13 cm.



Lego Mindstorms and Lego RCX

As a basic kit for building SVC we used Lego Mindstorms because of plainness and flexibility of Mindstorms construction kits.

The chassis of the vehicle is made of Lego bricks. Each of the two back wheels is driven by separate motors controlled by Lego Mindstorms *Robotic Command Explorer* (RCX), which enables it to carry out special movements.

Hitachi's μ-Chip

On the floor panels, randomly distributed Hitachi μ-chip *inlets,* consisting of a tiny μ-chip (about 0.4 mm x 0.4 mm) with an external antenna of approx. 4 cm in length, create anchor points on the surface, so that every single tag represents a mark for the certain area on the floor.

An integrated μ-chip antenna glides close to the floor and provides the computational unit with information about the currently seen tag.

BTnode

Along with the 4.5V battery for the μ-chip reader, the chassis holds a BTnode, an electronic control unit which consists of a micro controller, 64 KB RAM, 128 KB FLASH ROM, 4 KB EEPROM memory, and a Bluetooth wireless radio. BTnode allows the autonomous operation of SVC, so that a PC is not required for data processing.

Communication and Programming Specification

After the μ-chip reader has recognized a tag, it transfers the tag ID to the BTnode, which performs a simple ID analysis, makes a decision about the next action, and, finally, sends the chosen command to the RCX.

Various interfaces enable communication between the hardware modules. The μ-chip system exchanges data with the BTnode via a serial connection. Programmed in C, the BTnode sends commands and receives acknowledgements to/from RCX via an infrared connection. For programming the RCX we used the *Not Quite C* [9] programming language.

As a further development, a Bluetooth-equipped mobile phone has been integrated into the system as a remote control so that the mode switch commands (Marking Mode, Cleaning Mode) as well as movement control commands can be sent from the phone to BTnode over the Bluetooth link that makes the system completely decoupled from an external PC. The device is also able to recognize RFID-tagged objects accidentally dropped onto the floor, take a photograph of them and request the user for further actions.

CONCLUSION AND FURTHER WORK

The experiments showed that a good surface coverage and reasonable self-positioning for the vacuum cleaning task are possible if the floor is tagged with a sufficient amount of RFID-tags. An investigation of reasonable patterns for tag distribution, as well as integration of other sensory technologies seems to contain challenging research potential.

ACKNOWLEDGMENTS

We would like to thank Julio Perez and Matthias Sala for their creative work, Matthias Ringwald for his assistance with BTnode.

We would also like to thank our project partner Hitachi SDL, Japan for providing the μ-chip RFID equipment and support.

REFERENCES

1. Finkenzeller, K., RFID Handbook: Fundamentals and Applications in Contactless Smart Cards and Identification. John Wiley and Sons, 2003.
2. Pederson, T., Magic Touch: A Simple Object Location Tracking System Enabling the Development of Physical-Virtual Artefacts in Office Environments. Springer-Verlag London Ltd, Personal and Ubiquitous Computing, vol. 5 (2001), 54–57.
3. Anoto. Further information at http://www.anoto.com/
4. Trilobite vacuum cleaner. Further information at http://trilobite.electrolux.com/
5. Roomba. Further information at http://www.irobot.com/
6. Lego Mindstorms. Further information available at http://mindstorms.lego.com/eng/.
7. Hitachi μ-chip. Further information available at http://www.hitachi.co.jp/Prod/mu-chip/.
8. BTnode specification. Available at http://btnode.ethz.ch
9. NQC Not Quite C - Programming Language for Lego RCX. Available at http://bricxcc.sourceforge.net/nqc

[1] This work has been partially funded by Hitachi Systems Development Laboratory (SDL), Japan, as part of the μ-Chip Research Project.