Context Nuggets: A Smart-Its Game

Michael Beigl*, Albert Krohn*, Christian Decker*, Philip Robinson*, Tobias Zimmer*, Hans Gellersen+, Albrecht Schmidt-

*TecO, University Karlsruhe + Lancaster University - Universität München

*{michael, krohn, cdecker, philip, zimmer}@teco.edu

+ hwg@comp.lancs.ac.uk

- albrecht.schmidt@informatik.uni-muenchen.de

ABSTRACT

Small, embedded, sensing and communicating computer systems continue to show their applicability in various settings. The Smart-Its platform, which we present, testifies to this. We have developed a game called "Context-Nuggets", in order to test and demonstrate the extremities of this platform when subjected to a setting with multiple, ad-hoc users, discovering each other and exchanging context data. Attendees simply attach a Smart-It to their body and they can join in. The gaming strategy entails collecting as much "context" as possible, through altering interactive behavior with other players. Context sources include light, audio and movement sensors. Context is traded via short-range wireless communications.

A tool that manages the on-site gaming statistics is also used for analyzing run-time behavior and system status of the Smart-Its.

Keywords

Ubicomp Platform, Games, (usability, technology) tests

INTRODUCTION

In-Situ context generation, processing and communication has advantages in many application areas. This demo shows a platform for application scenarios, consisting of tiny computing devices that are embedded into everyday objects, on people or clothing, or in the environment. Further demonstrated are development libraries and tools for building applications and services for supervising applications and experiments. The demonstration presents this technology platform through one example application. The central component of such a platform is a tiny device called the Smart-Its, which is used to retrieve context information from the environment, run applications and communicate via a wireless network. The first part of the demonstration is a Ubicomp game application, secondly we give more details on the technology involved. Attendees are invited to take part or observe the game, and subsequently have a closer look at the enabling software and hardware design of Smart-Its.

Ubicomp games [1] stimulate use of Ubicomp technology, as has been shown in previous Ubicomp conferences - e.g. Pirates [2]. In our game, "Context Nuggets", attendees of the conference are invited to be players by configuring and using a small device. The device can be worn on the waist or adhered to the shirt (figure 1). Analysis tools can be used to actively retrieve the status of the game and hardware, such that the attributes displayed are either game related or retrieved from monitoring the technology.

THE PLATFORM: SMART-ITS

The basic idea behind the Smart-Its platform is a move towards simplifying the embedding of computing, perception and wireless communication into the physical world, along with an integrated toolset to make use of the information collected. The Smart-Its platform enables the development of Ubicomp environments, applications and tests.

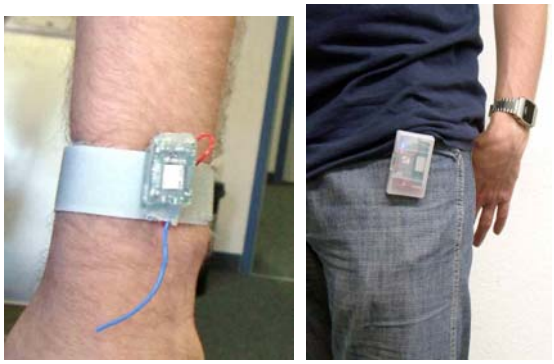

Figure 1: Small Smart-It with sensors attached to body or clothing

A major part of the Smart-Its platform [3] is the hardware device, which comes in various forms (e.g. figure 1). It builds the embedded hardware toolset and contains RF based wireless communication, on-board processing, memory, sensors and actuators. It can produce sensor information from up to 12 sensors, process context information within a local processor, provide adequate storage of context and general information, and host applications such as the game described below. It works independently of any external infrastructure and allows spontaneous, short-range peer-to-peer and ad-hoc exchange of processed data. Smart-Its are tiny, lightweight and have low energy consumption, such that the extent of objects to which they can be embedded ranges from very small or the human body (figure 1). The Smart-Its software can be rapidly developed based on a simple-to-use library providing a high-level access to communication, sensing and actuating functionality. Furthermore, generic programs are available for certain application areas as usability tests.

While infrastructure is optional, as Smart-Its communicate ad-hoc, PC based services such as wireless development and maintenance of Smart-Its applications require such integration. Infrastructure equipment enables access to Smart-Its over the Internet and vice versa. Infrastructure – based services may also be a source of additional context information, such as location or a history database. Figure 2 shows a setting with several Smart-Its distributed in a flat or office environment.

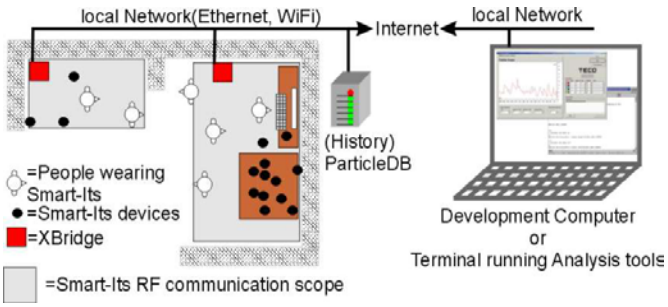

Figure 2: Smart-Its environment

The development toolset is a collection of programs that support programming, configuration and debugging of Smart-Its. With these software tools programming can be done on any PC based computer and over the Internet.

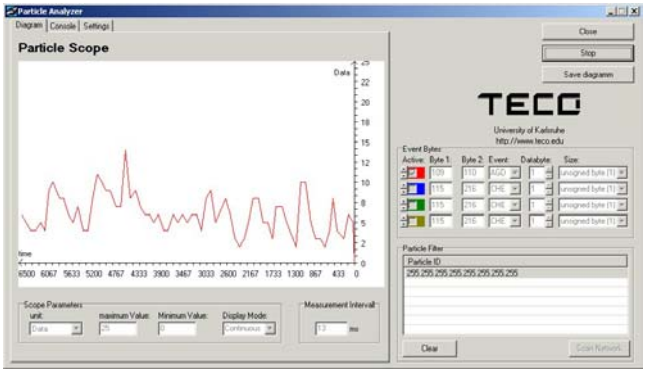

Figure 3: Graphical Visualization via Particle Analyzer

The text-based Smartspy tool and the graphical based Particle Analyzer tool allow the supervision and interpretation of output data from a running Smart-It application (figure 3). The graphical representation of data via the Particle Analyzer provides a quick overview of fast changing context, sensor or network data. This feature is often used for informal verification of Smart-Its behavior or analyzing performance parameters from the network or application. The tool's ability to display raw sensor data is also useful in the application design and debugging phases.

For maintenance or for recording data for test runs, a history database was implemented. This history, stored in the Particle database (ParticleDB), is accessible through a Web-frontend. The tool allows us to export selected data to standard formats for further statistical analysis.

THE GAME: CONTEXT NUGGETS

Imagine being an alchemist in the middle ages trying to produce gold out of mystic ingredients: Lux, spells and magic motions. These ingredients are then used to produce nuggets through a secret formula known only by you. But unfortunately, you cannot use the ingredients you produce yourself. Instead, you have to trade ingredients with other alchemists - one of your lux for one of their magical motions, one of your spells for one of their spells etc.

Based on your formula, "context nuggets" are created and at the end of the day the alchemist with the most nuggets is the winner. Players influence the progress of the game by entering the secret formula to make context nuggets. The secret formula describes how many of the ingredients are needed for creating a nugget and therefore determines the strategy for the player. A total of 10 units from the 3 ingredients are required, and at least one lux, one magic motion and one spell. The 7 remaining units can be allocated arbitrarily by the user, based on a calculated guess of the most available ingredients (figure 4).

Figure 4: Entry page for the Context Nugget game

Producing ingredients, trading and processing the nuggets appear to be "magical" as no explicit user intervention is required. After entering the formula Smart-It devices are distributed to the players and the game starts. At this point, players can influence the game by generating ingredients and meeting other players for trading. Ingredients are produced through a sensor perception on the Smart-it device. Sensing light level results - after a certain time - in producing a lux unit; likewise, sensing movement (acceleration sensor) and sound (microphone) result in magic movements and spells respectively. Following the rules of the game and the secret formula, the device also calculates and stores the nuggets for the player. It is not the intention of the game to encourage players to adapt their behavior to win the game by creating more ingredients etc. - which is also very difficult to perform during the game. In spite of this, creation of nuggets, trading and generation of ingredients can be perceived by the users through short flashes of light in different colors on the Smart-Its device.

The game ends after some hours at a fixed time where all devices have to be given back for final download of

buffered data. The "alchemist" with the device containing the most nuggets is the winner.

Technical Setting

To run the game, players are required to wear a tiny electronic Smart-It device and to attach this device to their clothing (figure 1). The device works independently of other computers and networks, holds the game rules and all information concerning the devices' player. The device constantly detects physical attributes such as light, sound and movement.

The device is able to communicate wirelessly and spontaneously in order to exchange ingredients information between participants of the game. Game communication is uses the Smart-Its ad-hoc and infrastructure-less networking. Collection of physical data, semantic aggregation and the ad-hoc communication is done implicitly without any user interaction. Any internal processing like trading and generation of ingredients or nuggets is stored together with a time stamp in the Smart-Its memory buffer. When the device enters a special marked area with connection to the Internet, buffered data is transferred to the Particle DataBase for immediate or later use. Conference attendees are able to monitor the current game status using either their own WiFi enabled device or using a Game Terminal during the game.

Game rules

1) The goal of the game is to create more nuggets than other players

2) You create nuggets by collecting ingredients and processing them to nuggets according to the formula you entered at the start of the game. Converting ingredients to nuggets is done automatically when the necessary type and amount of ingredients are available. Only traded materials can be used to produce nuggets, so you can't use your own ingredients.

3) You are able to generate 3 ingredients: Lux (created from a light level sensor), magic motions (created from movement sensor) and spells (created from an audio sensor) through wearing a tiny magic device. Ingredients are produced automatically without any user interaction.

4) Ingredients that are not traded are perishable - their maximum usability period is about 2 minutes.

5) To create nuggets, you consume 10 ingredients from others alchemists. The ingredients you need are part of the secret formula you enter at the start of the game.

6) You can trade with other participants on a 1-to-1 basis by standing within 5 meters of them or even passing by. The longer you stand next to other wizards the more you trade.

7) Trading and creating can be done everywhere as no extra infrastructure is needed. Simply wear your device correctly at the belt or shirt, not inside a bag etc. Otherwise your ingredient production stops and you are not able to trade.

What it Demonstrates

The Game shows some specific strengths of the Smart-Its platform. First of all, it shows that the Smart-Its are complete self-contained and independent devices. They do not need infrastructure and are able to generate higher-level information, represented here by the context (nuggets) and ingredients, from physical sensors.

The second strength is the ability of a Smart-It to work unobtrusively as factor of its small dimensions and long time operation. Furthermore, the device does not require any administration, maintenance or other explicit interaction to fulfill its task.

The third strength is the de-centralized communication. No master device or access point is necessary. Nevertheless, when connected to a backbone network, additional data analysis and statistic functionality is enabled.

GAME EVALUATION

The evaluation part of the demonstration uses information generated by the "Context Nuggets" game and shows one application of the infrastructure toolset and services. These toolset and services can be used for a variety of application areas including supervision of field tests or living lab [4] tests. In our demo setting, it monitors the progress of the game application, computes game data (e.g. the score) and observes technical parameters.

Observation with Smart-Its

In the Smart-Its set-up, simple sensors are the basis for the supervision in contrast to complex video surveillance often used in controlled laboratory user studies. An advantage of the use of directly attached simple sensor systems is that they can collect data automatically, with fine granularity and independent of the location, e.g. while on the move. Additionally, they are able to measure data constantly without being disturbed by occlusion. For many situations, ad-hoc embedding of the proposed technology is easy to handle and cheap, making the Smart-Its based evaluation suitable for small ad-hoc tests. Due to the lack of video surveillance cameras, the entire user behavior is not supervised. This suggests a move towards privacy-sensitive user monitoring.

There are also some disadvantages. Firstly, only specific parameters can be supervised and have to be known beforehand. Secondly, in the absence of additional surveillance of the user, users may fool the system by not or inappropriately using the devices and so adulterate collected data.

Example scenario: Game supervision

The "Context nuggets" game builds the application scenario for collecting user related data. The behavior of players gives hints to how good the game performs in a given environment. From this data, valuable analyses about the overall performance of the game can be carried out, but also individual player performance data can be shown.

For the proposed game several parameters are of interest:

- How often and when do players generate ingredients

- How often do players meet in general
- How often do players exchange ingredients (combines generation and meeting)

Additionally, for the progress of the game the information about the average "nugget" production per time is of interest.

Technically, the above parameters are retrieved by querying the Particle DB through a Web-Server based script. The ParticleDB holds all events that took place during the game with the correlating timestamp. Using this detailed data, the statistic applications generate graphs and reports. These can then be accessed by a Web-Browser either from the Terminal on the demonstration place or from any other computer connected to the network.

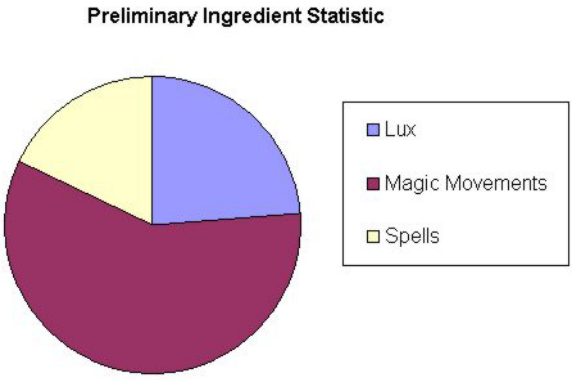

Figure 5: Example Statistic

Example scenario: Technical (network and sensor) set-up evaluation

For technical evaluation the ad-hoc and statistical analysis are also of interest. For the "Context Nuggets" game, network coverage or general network problems could be critical. The percentage and the average time of backbone access of every player and the load distribution among access points are of interest. A low access to one of the access points may indicate the need for a relocation of this access point.

Additionally, the context aggregation behavior of Smart-Its is noteworthy in order to optimize the rule-set of the game. The threshold and generating algorithms for units of ingredients can be adjusted according to the measured movements, noise and light.

All these measured parameters can be graphically displayed through the Particle Analyzer program from one of the terminal PCs on the demonstration site.

ACKNOWLEDGMENTS

Smart-Its project is funded by the Commission of the European Union as part of the research initiative "The Disappearing Computer" (contract IST-2000-25428).

REFERENCES

1. Björk, S., Holopainen, J., Ljungstrand, P. & Åkesson K-P. (2002). Designing Ubiquitous Computing Games - A Report from a Workshop Exploring Ubiquitous Computing Entertainment. In *Personal and Ubiquitous Computing January*, Volume 6, Issue 5-6, pp. 443 - 458
2. Björk, S., Falk, J., Hansson, R., & Ljungstrand, P. *Pirates! - Using the Physical World as a Game Board*. Paper at Interact 2001, IFIP TC.13 Conference on Human-Computer Interaction, July 9-13, Tokyo, Japan.
3. Beigl, M., Zimmer, T., Krohn A., Decker, C., Robinson, P.: Smart-Its - Communication and Sensing Technology for UbiComp Environments. Technical Report ISSN 1432-7864 (2003)
4. Kidd, Cory D., Robert J. Orr, Gregory D. Abowd, Christopher G. Atkeson, Irfan A. Essa, Blair MacIntyre, Elizabeth Mynatt, Thad E. Starner and Wendy Newstetter. The Aware Home: A Living Laboratory for Ubiquitous Computing Research" *Proceedings of the Second International Workshop on Cooperative Buildings*, October 1999.