Medical eGadgets

Stephen Bellis
David Murphy, David Harty
John Barton, Brendan O'Flynn
Kieran Delaney
Cian O'Mathuna
NMRC
Lee Maltings, Prospect Row
Cork, Ireland
+353 21 4904078
sbellis@nmrc.ie

Nicholas Drossos
Achilles Kameas
Irene Mavrommatti
CTI
Computer Technology Institute
PO Box 1122 GR 26110
Patras, Greece
+30 210 3416701
ndrossos@cti.gr

Anthony Pounds-Cornish
Arran Holmes
Martin Colley
Vic Callaghan
University of Essex
Wivenhoe Park
Colchester CO43SQ
United Kingdom
+44 1206 872138
apound@essex.ac.uk

ABSTRACT

The main objective of this paper is to assess the feasibility of an ad-hoc wireless network system for use within a hospital. The proposed system would operate as follows: Every bed within a hospital would be transformed into an eBed. These eBeds would monitor and record a patient's details and vitals. In addition every doctor on the ward would possess a Personal Digital Assistant (PDA)/laptop that would function as an eChart. These eCharts would be capable of wirelessly connecting to any eBed on the eWard, thereby allowing a doctor to view manipulate and save a patient's vitals, medical details and history.

Keywords

Disappearing Computer, Ad-hoc wireless sensor networks, EGadgets, Java, Embedded Java Controller.

INTRODUCTION

The main objective of this work is to assess the feasibility of an ad-hoc wireless network system for use within a hospital. The design is based on a recently developed technology known as eGadgets [1]. eGadgets is an innovative technology which seeks to enhance everyday objects with computing, sensing and networking abilities. The wireless ad-hoc network architecture that facilitates interaction and communication between eGadgets is a recently developed technology and as such few systems implementing the technology exist. Previous demonstrators exploiting the technology were relatively simplistic, placing few demands on it. The system proposed in this project represents the most sophisticated eGadget scenario developed to date. Development of the system can be roughly separated into two categories; hardware and software. The bulk of this work is concerned with the development of the software required to provide the eBed and eChart with their necessary functionality. The development of these eGadgets involves rigorous investigation into the operation and structure of eGadget technology. The hardware component of this project relates to the development of eBed sensors and the reduction in the physical size of the required eBed hardware.

eGADGETS

An eGadget is an everyday tangible object that: is enhanced with sensing, acting, processing and communication abilities; possesses an ability to cooperate and communicate with other eGadgets and can be connected up in a user-defined wireless ad hoc network known as a GadgetWorld

eGadgets are realised by adding to the object a processor, memory, sensors, actuators and wireless communication module.

Gadgetware Architectural Style, GAS

The framework that governs the concepts and mechanisms, through which a GadgetWorld is established and maintained, is known as the Gadgetware Architectural Style, or simply GAS. GAS basically sets out the protocols through which eGadgets are defined and interact.

Gadgetware Architectural Style - Operating System, GAS-OS

The GAS-OS is the software that implements GAS. The GAS-OS manages an eGadget's resources, and provides the underlying mechanisms that enable communication among eGadgets. Thus, it can be considered as a mini-operating system. In this analogy, eGadgets are to the GAS-OS what processes are to operating systems [2].

THE MEDICAL SCENARIO – E-WARD

The central aim of the project was to implement a medical application that fully exploited the functionality of eGadget technology. The scenario demanded that every bed in a ward be transformed into an eGadget. These eBeds would monitor and record a patient's details including a patient's current condition, a recent record of patient's vitals (e.g. heart rate, body temperature, etc) and a record of patient's medical details and history.

Appropriate members of the medical staff on the eWard would have a PDA. This PDA would function as an eChart. These eCharts could discover and connect to any eBed in the vicinity. Once discovered the eChart could view and manipulate information specific to a given patient. Previous scenarios were essentially static and didn't exploit the full power of eGadgets. They were themselves unable to 'discover' neighbouring eGadgets and did not possess the capability to dynamically create and destroy synapses. The scenarios required a GWEditor running on a nearby laptop to efficiently setup, alter and destroy synapses between eGadgets.

To realize the eChart it was necessary to develop a much more powerful eGadget. The eChart needed to be capable of discovering eBeds and forming appropriate synapses with a desired eBed. The eWard could not rely on a GWEditor to form synapses and thus the requirement for it was eliminated. The eWard also greatly reduces the required knowledge of the user. In previous scenarios the user needed a deep understanding of GAS and be familiar with terms such as GadgetWorld, Plugs and Synapse. Otherwise a user would be unable to create working GadgetWorlds. The operation of the eChart demands no knowledge of GAS. Instead the user is presented with much more intuitive operations such as 'Look for eBeds', 'Connect to eBed', etc. The eWard demanded the development of two different eGadgets, namely the eBed and the eChart. Both eGadgets are very different in terms of their hardware architectures, properties and processing capabilities.

While only two different types of eGadgets were developed, multiple numbers of each of these eGadgets exist within the scenario. The only difference between instances of a specific eGadget is their eGadgetID and IP address. Every instance of an eGadget requires a unique eGadgetID in order to participate in a GadgetWorld.

SOFTWARE

The software component of both eGadgets was developed using Java. The eGadget software is designed to be executed directly by the GAS-OS which is itself a Java program. In addition the GAS-OS provides many classes that can be and indeed have to be exploited by the eGadget software to achieve desired functionality. It is therefore fundamental that a complete understanding of the GAS-OS be acquired prior to eGadget development. The eGadget GAS-OS along with the rest of the eGadget specific software is collectively referred to as GAS-related middleware. The functionality of any eGadget is realized by GAS-related middleware running on the CPU of the eGadget. The GAS-related middleware is the software core necessary for the operation of all eGadgets.

HARDWARE

The eChart and the eBed were realised using two very different hardware setups. The required functionality of the two eGadgets heavily influenced the choice of hardware. The decision on the hardware component for the eBed was motivated by two main factors; the need for the eBed to interact with a multitude of sensors and also the lack of a requirement for a standard graphical/command line input or output. The function of the eBed is simply to record a patient's medical history and vitals. This stored data can only be accessed via an eChart. The need for a screen/monitor was thus eliminated. The CPU of the eBed is an Embedded Java Controller (EJC) [3]. The EJC is a small easy to integrate single board computer that possesses a 24 MHz processor and 16Mbytes of memory. Crucially it also possesses a Java Virtual Machine (JVM) that is Personal Java 1.2 compliant; this enables it to interpret the eBed software component. An AirStation provides the wireless link through which the eBed can communicate with other eGadgets. Sensor information is received through a serial port connection. The transfer of sensor information to the EJC is controlled by an FPGA.

The eChart is designed to run on a PDA. The user can interface with the eChart through the PDA's touch-screen, while the wireless link is provided by a WLAN expansion pack. Unlike typical eGadgets, the eChart has no sensors or actuators. The eChart receives data from only the eChart user (via a GUI) and/or from a connected eBed (messages sent through appropriate synapses).

CONCLUSION

This eWard has successfully been implemented using eGadget technology in a lab-based environment. This medical scenario is the most demanding eGadget application to date and reliability issues revealed will drive future improvements in GAS-OS. In discussion with medical professionals, the eWard scenario was received enthusiastically, particularly in light of a grant available to doctors for the purchase of IT equipment including PDA's/laptops. Future work requires the eWard to be implemented in an actual hospital environment.

ACKNOWLEDGMENTS

We thank the European Commission for funding the Extrovert Gadgets project (IST-2000-25240) and the Disappearing Computer community for inspiring this work.

REFERENCES

1. Extrovert Gadgets (eGadgets) website, http://www.extrovert-gadgets.net
2. Kameas, A. An Architecture that Treats Everyday Objects as Communicating Tangible Components, in Proceedings of PerCom03; (Dallas-Fort Worth, Texas USA, March 2003), pp 115-124
3. EJC website, http://www.embedded-web.com