# The SimCon Generator:

## An Interactive Context Simulator for Rapid Evaluation of Smart Building Applications using Virtual Reality

Kris McGlinn

Knowledge and Data Engineering Group
Trinity College Dublin
Dublin, Ireland
Kris.McGlinn@cs.tcd.ie

Rob Brennan, Dave Lewis

Knowledge and Data Engineering Group
Trinity College Dublin
Dublin, Ireland
Rob.Brennan@cs.tcd.ie, Dave.Lewis@cs.tcd.ie

*Abstract*—**This paper describes the SimCon (Simulated Context) Generator which combines data on the state of a Virtual Reality building with the SimCon Model to generate interactive location context for the rapid evaluation of Smart Building Applications. The paper evaluates the simulated context against physical readings. The SimCon Model does not set out to replace deterministic models, but rather allow for rapid configuration of simulated context early in the building design cycle to evaluate the impact of uncertainty in context on application behavior.**

*Keywords- Context, Simulation, Smart Buildings, Location Context.*

## I. INTRODUCTION

Smart Buildings (SBs) aim to do the right thing at the right time automatically for users [1]. To meet this requirement, a SB Application (SBA) must match its own view of the current context of use with that of the users [2]. That is, if a user perceives a change to their situation (a situation is a higher abstraction of context [3]) differently than the SBA perceives this change (due to delays in the process of context delivery to the application, sensor inaccuracy or incorrect inference) then the SBA may adapt its function at an inappropriate time or in an inappropriate way.

As SBs contain within them many interconnected and interacting systems, user-centred evaluation of SBAs is non trivial [4]. SBAs must work within a range of potential buildings (e.g. offices, homes, public buildings). SBAs may also require context gathered from sensor networks - the acquisition, installation and maintenance (cabling, power and configuration) of which presents a considerable financial risk to SBA developers, especially if the desired level of context-awareness is not achieved for the user. In-situ evaluation may also require coordination of participants moving and interacting with sensor networks in large scale buildings [4]. The opportunity to conduct repeated, user-centred studies may be prohibitively expensive [6].

To reduce costs, Virtual Reality (VR) simulation platforms have been identified as useful for developing and evaluating such systems early in the development life cycle [10], [12], [13]. The Platform for User Centric Design and Evaluation of Context-Aware Systems (PUDECAS) [14] is distinguished from existing approaches by the focus on providing a flexible,

extensible and easy to configure platform for integrating SBA's seamlessly into the rapid prototyping process. The VR games engine maintains a high fidelity view of the environment. As SBAs must often function with unreliable context (due to uncertainty introduced by the measuring process) the simulated context generated using the VR SB (e.g. location context) should reflect the uncertainty in context (random errors, delays) that is generated from physical sensors in an equivalent SB.

The SimCon Generator uses the SimCon Model, which is based upon standard models for describing buildings and sensors, and real time data on the state of the VR building to drive context simulation [1]. These models support the modelling of the behavioural characteristics of sensor deployments through the use of response curves (e.g. output over distance, or output over time or a combination of these factors). Here we present an evaluation of the SimCon (Simulated Context) Generator to assess whether Gaussian Noise is a suitable candidate for introducing uncertainty into indoor location context, for the purpose of early rapid evaluation of SBAs, by analyzing outputs from a physical sensor system.

This paper begins with a review of context simulators, concentrating on interactive context simulators which use VR. Then a review of SBAs identifies a suitable type of context for evaluating the simulation. The next sections detail the design of the SimCon Generator and the evaluation of a location based sensor deployment to determine whether the SimCon Model can be employed to simulate context. Finally we give our conclusions and future work.

## II. STATE OF THE ART

SBAs require sensitivity to "context". Within ubicomp, sensory data is generally used to infer context [1]. A number of research efforts have looked into developing simulation suites that simulate context. Bylund and Espinoza classify these into two specific categories: those that simulate a set of values as a part of a test suite and those which allow interactive testing of services in semi realistic circumstances [2]. We begin by exploring research efforts which take the former approach to determine how accurately they model context.

## A. Context Simulators

Morla and Davies present a simulation environment for location-based systems and evaluate a remote health-monitoring application [3]. To provide location, Morla and Davies make use of the standalone Generic Location Event Simulator (GLS). Different models can be plugged in and out of the system as required (e.g. sensor and environment models to simulate the unique and dynamically changing physics present in a room). The models are not based on any existing standards and while they mention an environmental error model based upon pluggable statistical distributions, this is not part of "version 1" and no mention of its design or implementation is given.

SENS is a sensor, environment and network simulator [4]. To enable realistic simulations, they use values from real sensors to represent the behaviour of component implementations. Users can assemble application-specific environments with different signal propagation characteristics for sensors. The evaluation seems to mainly address the issue of performance, examining how many simulated seconds can be executed in real time. SimuContext is a context simulator for providing a range of contexts to help with context-aware application evaluation [5]. It looks at specifically simulating Quality of Context (QoC) issues like quality indicators, such as precision and decay. While SimuContext allows the behaviour of context sources to be modelled, there is no evaluation of how realistic these models are. DiaSim is a simulator for pervasive applications [6]. DiaSim employs stimulus producers which mimic environmental phenomena and/or behavioural patterns of people and vehicles. These in turn trigger simulated sensor events. While mathematical models are discussed to drive the simulation, there is no indication of how these are executed or whether they have been evaluated against real physical stimuli.

While the previous approaches support context simulation and also provide some support for application designers to evaluate context-aware applications, none of them provide the means for interactive testing with real users using applications *in situ,* which has been identified as necessary for good design [7]. In the next section we will look specifically at context simulation platforms which use VR to provide these types of interactive environments and how accurately they model context.

## B. Interactive Context Simulators

QuakeSim makes use of the Quake III Arena to simulate a 3D environment [2]. The Quake engine allows multiple participant avatars to connect and become actors within the environment each with first person views on the environment and the ability to interact with objects. The Quake III engine was modified to extract context in the form of position and altitude to simulate different types of sensors and used to evaluate the GeoNotes [8] application. While QuakeSim demonstrates the usefulness of using interactive VR to provide real time location to a live system, there is no mention of how sensors are actually modelled. UbiWise [9] builds on the work of Espinoza by also making use Quake III. Designers can use UbiWise to emulate an application which corresponds to objects in Quake III. These can then be tested inside the 3D

simulator. Sensors, networks of sensors, and location sensing technologies can all be simulated and various aspects of the technologies can be studied one at a time, but no explicit explanation of how this is done is given or whether they deal with uncertainty in sensor data.

Shirehjini and Klar have developed 3DSim, a 3D-based rapid prototyping and simulation environment that allow development of Ambient Intelligence building blocks (e.g., situation recognition, goal-based interaction) [10]. 3DSim offers open and standardized interfaces allowing (a) to integrate new devices and sensor components as well as (b) to interact with those devices and sensors. During a simulation, sensor data is triggered by humans using GUI elements, e.g. an avatar can point at devices. Little information is given on how sensors are actually simulated and whether they deal with uncertainty in sensor data. As can be seen from the literature, no interactive context simulator system specifically looks at introducing uncertainty into simulated context. Before the SimCon Generator is introduced, location context is identified as a suitable subset for evaluation due to its unique properties.

## C. Identifying Context for Simulation and Evaluation

From examination of the existing literature, concentrating on SBAs (indoor context-aware applications), it becomes apparent that location context is used in a majority of applications [11-13]. There are numerous types of location based sensing technologies available. Hightower et al. give a taxonomy and survey of 15 location based sensing technologies, but concede there are many more [14]. Location-sensing systems have varying levels of accuracy, and various ways of representing location. These can be categorised as location that indicates presence at a particular sensor (with no recognition), proximity, which indicates proximity to a particular sensor (may or may not provide recognition) and coordinate, which defines a position relative to a coordinate system (and may or may not provide recognition).

Hightower et al. conclude that modelling uncertainty in location sensing systems is a considerable challenge. They suggest the future quantitative evaluations of location-sensing systems include error distributions, summarising the system's accuracy and any relevant dependencies. These error distributions could then be used to simulate the location system's behaviour. Therefore, and for the following reasons, we choose to concentrate our evaluation on location-based context:

- The prevalence of location context use within the indoor context-aware community

- The risk associated with installing and maintaining indoor location systems, and hence the need for cost effective rapid prototyping of these types of system.

- The variance in accuracy and reliability of location context.

Using the Hightower et al. survey we also identify three types of context which cover a wide range of location based contexts. These are:

- **Presence**: context which alerts of presence, but does not identify the cause, e.g. a pressure mat.

- **Proximity**: context which alerts of presence in a particular area from a receiver, and which also provides identity, e.g. an RFID tag.

- **Coordinate**: context which alert of the coordinate and identification.

### III. SIMCON GENERATOR DESIGN AND IMPLEMENTATION

In this paper we are specifically interested at simulation of location context which is directly affected by the interactions of users with the environment and sensors (i.e. location based sensor systems), as these types of interactions impact on the user centric evaluation of SBA behavior. This simulation is handled by the SimCon Generator which takes data on the state of the PudecasVR games engine.

Pudecas maintains a global view on the position of each avatar in relation to the environment's origin [15]. XML messages can be extracted giving the precise location of the user's avatar within the virtual smart building at a specific time, tied to a Cartesian coordinate system. The SimCon Generator extracts data on the state of the VR environment (avatar location and identity). Combining this data with the SimCon Model, the SimCon Generator can generate discrete units of simulated context called "contums" [1] Figure 1 (left). The SimCon Generator is configurable through the SimConfig tool which supports importing standard models in the building and sensor domains, ensuring interoperability within these communities community [1].
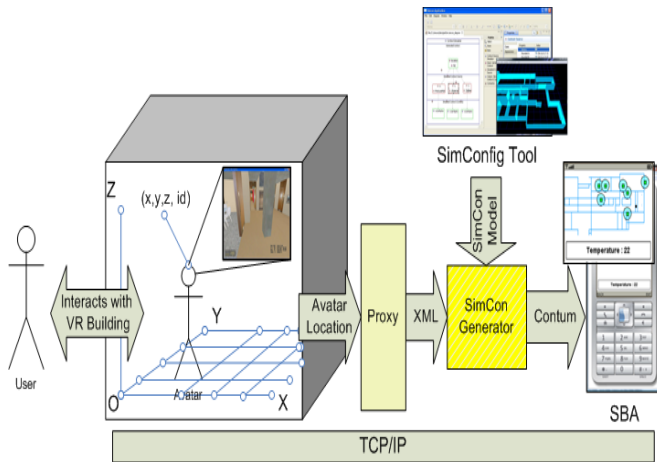


Figure 1 The SimCon Generator using data on the state of the VR environment

### A. Simulating Interactive Location Context

The extracted data from the VR building can be used to simulate the effect of changes in that location on the simulated context produced by the SimCon Generator. It should be noted that the VR engine places limits on what context can be realistically simulated for SBA evaluation. We will briefly describe those limits here.

A VR Games Engine updates the game state at the same speed as the frame rate and this can be anything between 24 and 72 frames per second (fps). The games engine Pudecas employs, Half Life 2 (HL2), updates at 30 fps. However, the performance over head of encoding and transmitting the avatars location may result in the avatar location update being configured to a slower rate. Human recognition times have a limit of approx 190 ms to visual stimuli [16] this means that any time below this is sufficient for VR to give the impression to a user of a changing environment. HL2 is currently configured to extract this data at a rate of approximately one per tenth of a second.

The avatar has been configured to move at a maximum speed of 2.8 meters per second (walk). Therefore within a tenth of a second the position of the avatars location can fall anywhere within 0.28 meters of its original position. This unknown must be taken into consideration when attempting to simulate uncertainty as the SimCon Generator is dependant on this data for interactive context. Migration to faster VR systems is possible though if required, as long as the VR system provides avatar location.

The data on avatar location is high fidelity, i.e. it is a very accurate representation of the state within the bounds of the system. SimCon Sources must support simulation of error in order to produce realistic contums, which unlike the uncertainty introduced through movement (i.e. as the avatar travels along a given path at a given height) can result in locations being offset anywhere in 3D space (as a result of random error [17]). This is necessary to evaluate whether an SBA responds appropriately to likely levels of context sensing error. In a Real Time Location System (RTLS) like Ubisense [18], which provides coordinate location using Ultra Wide Band (UWB), uncertainty can be the result of interference in the UWB signal from various building materials, temperature, humidity, background radiation, its position relative to other objects and occupancy which collectively contribute to reflection, attenuation, multi-path issues, other electronic devices which may interfere with wireless signals and line of sight issues [19, 20]. Simulation must take into account these behavioural characteristics.

There are two main general approaches to modelling signal propagation: empirical and deterministic [21, 22]. Empirical models are based on simple formulas where empirical parameters obtained from measurements describe the environment. Deterministic models attempt to follow the physical principles of electromagnetic wave propagation, the most popular methods being ray tracing and ray launching. A combination of these models can also be used, for example the motif model [21]. The advantage of deterministic models is that they are very accurate, the disadvantage is that they require complex models of the environment (including wall and object materials) in order to work, expertise of the tools to build the models and they require large amounts of processing, making them slow to run, which does not make them good candidate for real time simulations.

Empirical models do not provide the same kind of accuracy, but they are easier to apply and fast to run. While the motif model attempts to address some of the difficulties, it still requires knowledge of walls and materials, which may not always be available with surface models like those for VR.

Therefore the SimCon Generator takes the former approach, using empirical models. It does not directly set out to understand the cause of sensors potentially complex error functions; rather it sets out to emulate them through the use of either numerically generated or stereotypical error distributions, which summarises the systems accuracy and any relevant dependencies (distance, occupancy, etc.). A response curve can be used to model these types of dependencies. The Gaussian (or normal) distribution has been suggested as an appropriate means to describe the error distribution of location sensing systems [14] where detailed deterministic models are not available, or where the time to implement this models is prohibitively expensive. While the SimCon Generator makes use of Gaussian distribution to introduce uncertainty, we do not limit it to this approach and leave it open to be extended to use other statistical distributions (like Poisson distributions) if required, or to improve the accuracy using deterministic models where these models are available and do not introduce additional delays into the process of simulation.

The model works by assigning SimCon sources response curves. Each response curve defines a set of conditions which influence the contum. For example, the time of day or number of occupants in a room may have an impact on temperature contums, or the position of a transmitter relative to a receiver, or to some area of increased interference (be it a person or object), on a location contum. Another response curve is defined which specifies how the data will distribute around that mean value by applying a Gaussian function, i.e. Ubisense specifications state that in optimum conditions a transmitter's measured location should have an accuracy of 0.15 cm's (i.e. it should fall within at least 0.15cm of its actual location). The application of each response curve increases the complexity of the simulation, and in theory should improve the realism of the simulated output. Response models may also be used in a similar fashion to define delays in measurement rates, as these can have a significant impact on location contums. For example, when location is a coordinate the inaccuracy is increased proportional to the amount of distance the avatar has moved before the contum is generated (due to delay).

The level of granularity in the simulation process is limited by the extra cost in processing power, which must not introduce delays greater than those of a physical system, and also the extra levels of expertise required to model. The trade off therefore is between the realism of the simulation, the time to configure the simulation (for the purpose of rapid prototyping) and also the processing requirements of the SimCon Generator. At minimum, the SimCon Generator must generate contums to a level of detail sufficient to support the evaluation of the SBA without introducing perceptible overhead delay in processing of simulated context.

A Java proxy has been implemented which connects to the games server [23]. The SimCon Generator is implemented in Java and connects to the proxy over a TCP/IP connection Figure 1 (left). The process of configuration is detailed in greater detail here and has been shown to provide rapid configuration of Smart Building for the purpose of SBA evaluation [1]. The evaluation of the performance is going work and a technical paper is available here [24].

## IV. EVALUATING PHYSICAL SENSOR READINGS FOR THE PURPOSE OF SIMULATION.

This section examines a Ubisense RTLS which provides coordinate context [18] to determine how effectively a sample sensor system can be modelled using the SimCon Model approach.

### A. Evaluation of Contums against Real Sensor Output

The ubisense cell consists of four sensors at each corner of the common room, at the centre of these a set of thirty five points were laid out on a table surface at a distance of 0.6 meters each (making a grid). A tag was set to publish messages at its highest rate (approximately every tenth of a second) for a period of 30 seconds (although this rate may vary due to the visibility of the tag to the four sensors). A second set of readings were taken for a tag held by a user standing at each point. The tag was at elbow length from their body and kept at midriff (the same height as the table surface).
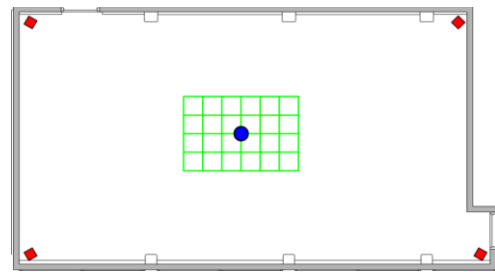


Figure 2 Lloyd Common Room, Sensors and Grid.
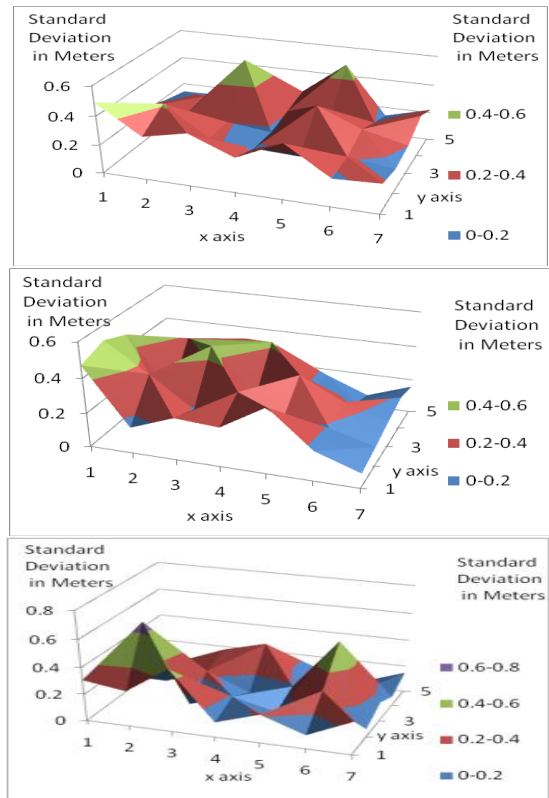
### B. Findings



Figure 3 Standard Deviation in Ubisense Readings when Tag not held (Top z-axis, centre y-axis, bottom x-axis)
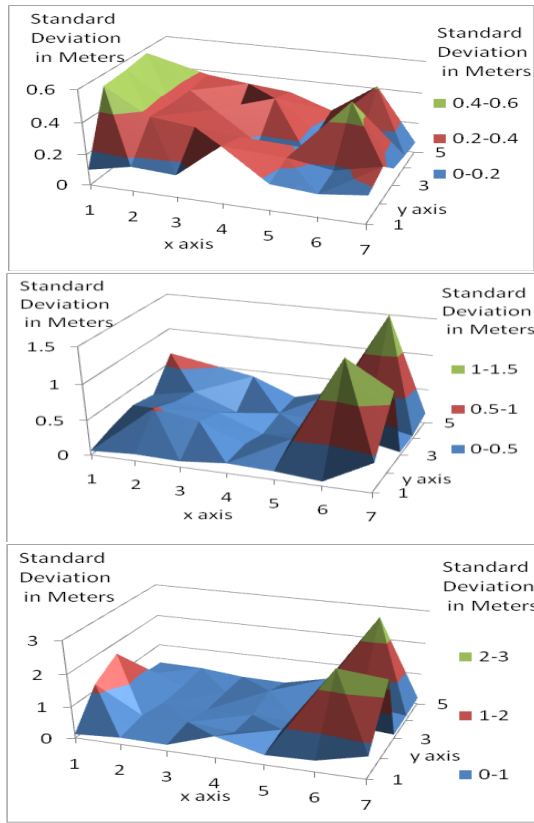
Figure 4 Standard Deviation in Ubisense Readings when Tag is held (Top z-axis, centre y-axis, bottom x-axis)

Figure 3 shows the seven by five points on the table (x and y axis in diagrams) and for each point on that grid the standard deviation (SD) of the z, y and x coordinates is given when a tag is placed at that point. As can be seen the majority of the coordinates returned were off their actual location by between 0.1 and 0.4 meters. Figure 4 shows the same grid system when the tag is held by a person at elbow length from their body at the same height as the table. The majority of readings here lie between 0 and 0.5 meters.

*C. Interpretation*

The majority of readings had a SD of under 0.4 meters when the tag was left resting on the table surface, the peaks being towards the longer side for the grid, probably due to the angle of the sensors and the length of the room. When a user was carrying the tag it was far less accurate with SD readings falling again well below 0.4 meters. The largest peaks here being to the right of the grid. This area had considerable fluctuation in sensed location, often not updating location at all until the user moved to a new position. This may be a result of the left right corner sensor being slightly blocked by the indented door combined with the subject's body, but this requires further investigation.

## V. CONCLUSION AND FUTURE WORK

Ubisense location context can vary greatly due to the environment and lay out of the sensors and room. The accurate reproduction of these error distributions is a complex issue. Gaussian noise can be employed to simulate the variation

experienced in a physical sensor deployment like Ubisense, but a model which models behaviour for the entire sensor area the same is not sufficient due to the impact of the environment and the resulting impact on SBA behaviour. Within a sensor cell, there are areas of higher uncertainty, the cause of which is not always apparent without detailed analysis of the environment. Therefore we envisage a combination of uniform outputs which capture the normal behaviour and also areas defined for increased uncertainty Figure 5.
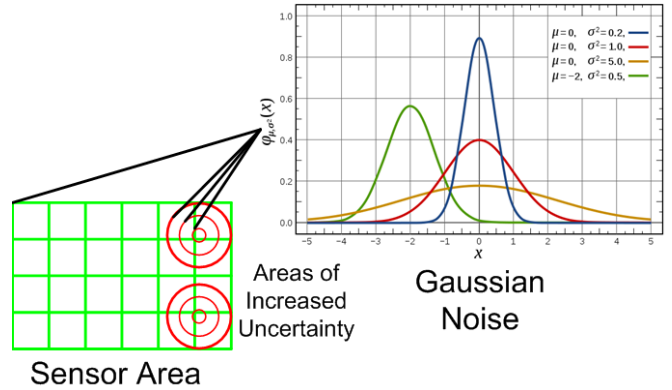


Figure 5 Modelling Uncertainty in SimCon Sources

While it may not be possible to reflect the actual physical system exactly, being able to simulate these issues and reproduce similar effects at different areas of a room can provide insights during early evaluation of an SBA. Extending the model with additional dimensionality (e.g. effect of orientation, interference from other person) or by incorporating more complex response models into the simulation process should also further improve the accuracy of simulated outputs. Future work will look at new types of location context (e.g. received signal strength for proximity) and also different types of context like temperature [22].

## VI. ACKNOWLEDGEMENTS

## VII. REFERENCES

List and number all bibliographical references in 9- point Times, single-spaced, at the end of your paper.

[1]     E. O. N. K. McGlinn, A. Gibney, D. O'Sullivan, D. Lewis, "SimCon : A Tool to Support Rapid Evaluation of Smart Building Application Design using Context Simulation and Virtual Reality," *Journal of Universal Sciences,* p. Yet to be published, 2010.

[2]     M. Bylund and F. Espinoza, "Testing and Demonstratign Context-Aware Applications with Quake 3D," in *Communications of the ACM*. vol. 45, 2002, pp. 46-48.

[3]     R. Morla and N. Davies, "Evaluating a location-based application: a hybrid test and simulation environment," *IEEE Pervasive computing,* vol. 3, pp. 48-56, 2004.

[4]     S. Sundresh, W. Kim, and G. Agha, "SENS: A sensor, environment and network simulator," in *37th Annual Simulation Symposium* Washington, DC, USA: IEEE Computer Society 2004, pp. 221-228.

[5]     T. Broens and A. Halteren, "SimuContext: Simply Simulate Context," in *In ICAS '06: Proceedings of the International Conference on Autonomic and Autonomous Systems*, 2006, p. 45.

[6]     W. Jouve, J. Bruneau, and C. Consel, "DiaSim: A parameterized simulator for pervasive computing applications," in *IEEE International Conference on Pervasive Computing and Communications* Seattle, USA.: IEEE Computer Society, 2009, pp. 1-3.

[7]     S. Carter and J. Mankoff, "Prototypes in the wild lessons from three ubicomp systems," *IEEE Pervasive Computing,* vol. 4, pp. 51-57, 2005.

[8]     F. Espinoza, P. Persson, A. Sandin, H. Nystrom, E. Cacciatore, and M. Bylund, "Geonotes: Social and navigational aspects of location-based information systems," *Lecture Notes in Computer Science,* pp. 2-17, 2001.

[9]     J. J. Barton and V. Vijayaraghavan, "UBIWISE, a simulator for ubiquitous computing systems design," *http://www.hpl.hp.com/techreports/2002/HPL-2002-303.html,* 2002.

[10]    A. A. N. Shirehjini and F. Klar, "3DSim: rapid prototyping ambient intelligence," in *Proc. of the 2005 Joint Conf. on Smart objects and ambient intelligence: innovative context-aware services: usages and technologies sOc-EUSAI*, Grenoble, France, 2005, pp. 303-307.

[11]    J. J. Bisgaard, M. Heise, and C. Steffensen, "How is Context and Context-awareness defined and Applied? A survey of Context-awareness," *Aalborg university,* 2004.

[12]    G. Chen and D. Kotz, "A survey of context-aware mobile computing research," Technical Report TR2000-381, Dept. of Computer Science, Dartmouth College 2000.

[13]    M. Baldauf, S. Dustdar, and F. Rosenberg, "A survey on context-aware systems," *International Journal of Ad Hoc and Ubiquitous Computing,* vol. 2, pp. 263-277, 2007.

[14]    J. Hightower and G. Borriello, "Location systems for ubiquitous computing," *Computer,* vol. 34, pp. 57-66, 2001.

[15]    E. O Neill, D. Lewis, K. McGlinn, and S. Dobson, "Rapid user-centred evaluation for context-aware systems," *Lecture Notes in Computer Science,* vol. 4323, p. 220, 2007.

[16]    R. J. Kosinski, "A literature review on reaction time," *Retrieved http://biae.clemson.edu/bpc/bp/Lab/110/reaction.htm 06/04/2010.,* vol. 18., p. 2006, Sept. 2006 2006.

[17]    J. G. Webster, *The measurement, instrumentation and sensors handbook*: Springer, 1999.

[18]    P. Steggles and S. Gschwind, "The Ubisense Smart Space Platform: A Ubisense White Paper," in *http://www.pervasive.ifi.lmu.de/adjunct-proceedings/demo/p073-076.pdf retrieved 06/04/2010*, 2005.

[19]    A. Schmidt, "Ubiquitous computing-computing in context." in *Thesis* Lancaster University, 2003.

[20]    J. Zyren and A. Petrick, "Tutorial on basic link budget analysis," *Application Note AN9804, Harris Semiconductor,* 1998.

[21]    P. Péchai, M. Klepal, and K. Novotny, "Novel Approach to Indoor Propagation Modelling," *Radioengineering,* vol. 9, pp. 21-16, 2000.

[22]    E. C. K. Mcglinn, E.O. Neill, and M. Keane,  D. Lewis, D. O'Sullivan, "Monitoring Smart Building Performance Using Simulation and Visualisation," in *Ubicomp 2010 Workshop: Ubiquitous Computing for Sustainable Energy (UCSE 2010)*, 2010.

[23]    E. O'Neill, M. Klepal, D. Lewis, T. O'Donnell, D. O'Sullivan, and D. Pesch, "A testbed for evaluating human interaction with ubiquitous computing environments," *Proceedings of TRIDENTCOM 2005,* 2005.

[24]    K. McGlinn, R. Brennan, and D. Lewis, "SimCon: Simulated Context Sources for the Rapid Evaluation of Context-Aware Entities using Virtual Reality," *Under Review (Draft copy available as TCD Tech. Report TR0075, https://www.cs.tcd.ie/publications/tech-reports/reports.09/TCD-CS-2009-42.pdf,* 2009.