Journal of Universal Computer Science, vol. 16, no. 15 (2010), 1992-2018
submitted: 31/1/10, accepted: 28/7/10, appeared: 1/8/10 © *J.UCS*

SimCon: A Tool to Support Rapid Evaluation of Smart Building Application Design using Context Simulation and Virtual Reality

Kris McGlinn
(Trinity College Dublin, Dublin, Ireland
Kris.McGlinn@cs.tcd.ie)

Eleanor O'Neill
(Trinity College Dublin, Dublin, Ireland
Eleanor.O'Neill@cs.tcd.ie)

Alan Gibney
(Cork Institute of Technology, Cork, Ireland)

Declan O'Sullivan
(College Dublin, Dublin, Ireland
DeclanO'Sullivan@cs.tcd.ie)

Dave Lewis
(College Dublin, Dublin, Ireland
Dave.Lewis@cs.tcd.ie)

Abstract: The promise of smart buildings (SBs) is a safer more productive environment for users and a more operationally efficient building for owners. The automation of building function is highly dependent on sensing devices and Smart Building Applications(SBAs), which are often only evaluated in situ post deployment, making re-development costly. In this paper we explore our experiences developing a Simulated Context (SimCon) Model which currently supports taking information from a Virtual Reality (VR) SB and converting it into three types of location context to conduct early rapid evaluation of location based SBAs. This model is expressed using the Sensor Modelling Language (sensorML). It also explores the integration of this model into the Industry Foundation Classes (IFC) for modelling and simulating SBs. It also details usability evaluations of the SimConfig and SimConViz Tool for improving evaluation during the design phase of smart building development life cycle.

Keywords: Modelling, simulation, smart building applications, evaluation
Categories: I.6.5, I.6.3, J.7, L.0.0

1 Introduction

The promise of Smart Buildings (or intelligent buildings) is safer, more operationally efficient living and working environments [Finley Jr, et al. 1991, Flax 1991, Clements-Croome 1997, Snoonian 2003, Wong, et al. 2005]. Smart buildings (SBs) are a subset of *smart environments* which, according to Weiser are physical worlds that are "richly and invisibly interwoven with sensors, actuators, displays, and computational elements" [Cook and Das 2004]. This embedded technology enables Smart Building (context dependent) Applications (SBAs) which build up views of the environment using a combination of context data and the building model in order to react proactively to changes in context for the benefit of the user [Lieberman and Selker 2000, Carter and Mankoff 2005]. Designing and evaluating SBAs however is a non trivial matter [Carter and Mankoff 2005]. Often specialised or multipurpose building types are needed e.g. offices, hospitals, homes etc. This leads to large scales of situations (activities which take place in the aforementioned buildings) and infrastructures that quickly become costly and difficult to manage during the design cycle [Papamichael, et al. 1999].

Ideally, developers should be able to evaluate their prototypes early, repeatedly and cost-effectively during development. Modelling tools combined with simulation techniques have repeatedly been adopted to overcome these challenges. To successfully simulate and evaluate SBAs there is a need for standard models to describe buildings and sensors; *smart buildings*. A model is required which exposes all the relevant aspects of the SBs heterogeneity and complexity [Salber, et al. 1999] while still providing feasible, cost effective, rapid evaluation of the SBA [Weis, et al. 2007]. Industry Foundation Classes (IFC) is an open specification to support building information modelling throughout the lifecycle of a building, accommodating both traditional and intelligent building objects.

In this paper experimental extensions to IFC are proposed which move beyond the existing basic *IFCSensorType* definition to provide real time context simulation within virtual smart buildings. Context simulation frees developers and evaluators from existing technological constraints and low-level concerns allowing them to make assertions about the success or failure of their application in a range of contexts [Broens and Halteren 2006], [Hightower and Borriello 2001], [McGlinn, et al. 2008]. Virtual reality (VR) and simulation provide scales of environments, often unobtainable in living test-beds, to support flexible, repeatable experiments with SBAs [O Neill, et al. 2007]. Simulation also supports testing which involves future or concept technologies and accuracies of context which are not realisable through motes and sensors currently on the market. The model forms the basis of a toolset designed to minimise the need for domain experts to step outside the bounds of their professional role and skills as part of the design process. As a result, usability of the toolset is a key issue for a range of users (e.g. SBA developers, sensor specialists, civil engineers).

This paper evaluates the usability of the SimConfig and SimConViz tool, for modelling simulated context sources and placing them in a VR SB to support real time evaluation of SBAs. It begins, in section 2, by exploring the background of SB modelling and simulation. Section 3 discusses the use of IFC models as the central data model for the support of visualisation and simulation of SBs for rapid evaluation of context dependant SBAs. It discusses modelling techniques for buildings, and gives particular attention to SimConfig and SimConViz. It also looks at preliminary attempts at the integration of sensor modelling language (sensorML) concepts with Industry Foundation Classes (IFC). Section 4 discusses the implementation of tools for the simulation and visualisation of SBs to support the evaluation process. Section 5 details the evaluation of the usability of the SimConfig and SimConViz Tool. Section 6 finally gives our conclusion.

2 Background and State of the Art

In this section we discuss the background of SB modelling, context, context modelling and sensor modelling, and the simulation and visualisation of SBs for evaluation of context dependant applications.

2.1 Building Modelling

As we are looking specifically at the design phase of the life cycle of SBs, it is important to look at how existing buildings have traditionally been modelled. Since the advent of Computer Aided Design (CAD) Tools, building modelling has involved the use of vendor specific data models to represent different but primarily geometric views of the Building Life Cycle (which defines the entire life of a building from design, through conception, occupancy and on to eventual demolition) [O'Sullivan and Keane 2005]. This has hindered interoperability between applications as each application has its own specific model. The Building Information Model (BIM) has been developed as a direct response to this issue. A BIM describes an integrated data model for storing all the information relevant to the building life cycle.

In order to realise the acceptance of BIMs, the International Alliance for Interoperability (IAI) is developing the Industry Foundation Classes (IFC) standard (latest version IFC2x3). IFC has the potential of enabling service engineers to collaborate between heterogeneous disciplines, improving interoperability, reducing costs and overall design quality and is currently the only data model that is an accepted ISO standard. Any piece of IFC compliant software has access to the same building information and can add new data to the model. The Open Geospatial Consortium test-bed successfully demonstrated seamless data transfer between architects, project managers, quantity surveyors and building performance analysts[1].

2.2 Context and Context Modelling

SBAs require sensitivity to "context". Within the scientific community, in differing fields, the exact definition of context is still open for discussion due to its subjective nature [Bazire and Brézillon 2005] and remains an avenue of research [Dey 2000]. Within the smart environment domain context is very often determined by what technologies are being used to gather it [Schmidt 2003]. In practice, sensory data is used to infer context [Scholtz and Consolvo 2004]. Building upon Dey and Abowd's definition of context[Dey and Abowd 2000], we define context as any information gathered through sensing technologies which supports adaptation of an entities function (through the application of technology) for the benefit of a user, whilst remaining unobtrusive.

In order to simulate context to evaluate SBAs, it is also important to understand how context is modelled. Early models addressed the modelling of context with respect to a particular application or application class [Strang and Linnhoff-Popien 2004]. Generic context models are of particular interest for providing simulation, as many applications can benefit from them. Various models have been proposed for representing context, among them logic based models, object oriented based models

[1] http://www.opengeospatial.org/pub/www/aecoo-1/index.html

and ontology based models[Strang and Linnhoff-Popien 2004], [Chen, et al. 2004], [Wang, et al. 2004], [Masuoka, et al. 2003]. A common method when modelling context is to employ a two tier approach [Wang, et al. 2004, Lee, et al. 2007], having a core or upper ontology to define generic concepts (like space/location) and a lower extensible ontology for adding to these generic concepts (like room/coordinate).

There seems to be agreement on location, time, identity and environmental factors as being key variables of context [Bisgaard, et al. 2004]. Some models further abstract these types of data into situations, e.g. "in a meeting". The ontology for mobile device sensor-based context awareness looks at abstracting raw sensor data into semantic context in order to develop mobile applications which are more usable [Korpipää and Mäntyjärvi 2003] .

2.3 Sensor Modelling

In practice, it is sensory data that is used to infer context [Scholtz and Consolvo 2004], therefore it is important to understand how sensors are modelled in order to provide accurate simulation. A sensor (which may also be referred to as a detector or transducer) is a piece of technology that can measure some physical phenomena over a discrete unit of time and convert this into a signal which may then be interpreted by an observer or instrument [Botts 2007, López, et al. 2009]. There is no common standard to the design and implementation of sensor systems. As a result the state of the art is of heterogeneous networks of disparate sensors [Botts 2007]. The exact number of sensor types in existence is therefore hard to determine. NASA's Semantic Web for Earth and Environmental Terminology (SWEET) list 415 types of sensor in their sensor ontology [URL].

The sensor modelling language (sensorML) [Botts 2007] has been developed to address this issue. SensorML is an approved Open Geospatial Consortium standard which provides standard models and an XML encoding for describing the process of measurement by sensors and instructions for deriving higher-level information from observations. OntoSensor builds on the sensorML model with the aim of to providing a "knowledge repository" which allows for more comprehensive inference than sensorML [Russomanno, et al. 2005]. This is achieved through the use of the Web Ontology Language (OWL)[McGuinness and Van Harmelen 2004], which allows for complex relationship definitions.

2.4 Context Simulation

A number of research efforts have looked into developing simulation suites that simulate context values. The standalone Generic Location Event Simulator (GLS) is designed for the visualisation, scalability testing and evaluation of location-aware event driven middleware and applications [Sanmugalingam and Coulouris 2002]. They define "locatables" as objects whose location can be sensed and have developed a simulator that models the behaviour of locatables in a simple model of a physical space. The simulated location context is provided in a format matching existing sensor deployments. These outputs are also fed into visualisation and analysis tools. Different models can be plugged in and out of the system as required. These include sensor and environment models to simulate the unique and dynamically changing physics present in a room, and a world model to model the buildings geometry.

SENS is a sensor, environment and network simulator [Sundresh, et al. 2004]. It features a modular architecture to permit simulation of a range of different Wireless Sensor Network (WSN) scenarios. In particular, they have implemented components to support sensor nodes communicating via wireless broadcast in an environment represented by tiles which modulate sound and radio propagation. The simulator would benefit from automatically generated timing information. Users are also forced to use a specific SENS API for any applications which run on top of these sensor networks. SimuContext looks at specifically simulating Quality of Context (QoC) issues [Broens and Halteren 2006]. As context information represents real-world situations, QoC gives certain quality indicators, such as precision and decay [Krause and Hochstatter 2005, Sheikh, et al. 2007] The SimuContext framework abstract from the complexity of interfacing with physical context sources and facilitates testing and demonstrating context-aware applications in a controlled way.

2.5 Evaluation using Virtual Reality

A number of research efforts exist which have specifically used VR simulation test beds in order to test context-aware applications. Bylund [Bylund and Espinoza 2002] introduced a tool called QuakeSim which makes use of the Quake III Arena to simulate a 3D environment. The environment is semi realistic and allows multiple participants to connect and become avatars within the environment. It provides tools for building new environments, modelling avatars and objects. They modified the Quake III engine to extract context in the form of position and altitude to simulate different types of sensors. They then used the context toolkit for gathering, aggregating, interpreting and publishing sensor and context information. They used this to evaluate the GeoNotes application.

UbiWise [Barton and Vijayaraghavan 2002]also makes use of Quake III games engine in order to simulate 3D environments. They simulate prototypes of new devices and protocols with a Java program. A 3D environment view is maintained by UbiSim and a 2D application view by Wise, collectively called UbiWise. The simulator focuses mainly on computation and communication devices. Shirehjini and Klar have developed 3DSim [Shirehjini and Klar 2005]. This is a tool for rapidly prototyping Ambient Intelligence building blocks (e.g., situation recognition, goal-based interaction). 3DSim currently runs on a single meeting room featuring smart projectors and shutters, and including avatars to represent the human element of the environment.

3 Modelling Smart Buildings

What becomes apparent from the state of the art is that while research has looked at specific views of smart buildings, currently there is no integrated model to capture all aspects necessary for SB simulation and visualisation. This paper takes a data centric approach to SB modelling in order to conduct simulation and visualisation which provides early rapid evaluation of the smart building during the design phase. This requires integration of sensor models with detailed building models.

The IFC core model describes the building and its elements. Extensions (property sets) to this model (based upon sensorML) describe more specific types and properties of sensors. This forms the capability to build tools to conduct simulation i.e. testing and verifying design choices before implementation/construction. For new buildings, generally the design process begins with an architect modeling building geometry (for example using ArchiCAD[2]) (Figure 1). This model supports a range of tools, for example, tools for modelling the VR building, for determining radio propagation and for context simulation. The resulting outputs drive interactive simulations and visualisation for conducting evaluation of the SB design and the design of context dependant Smart Building Applications (SBAs).

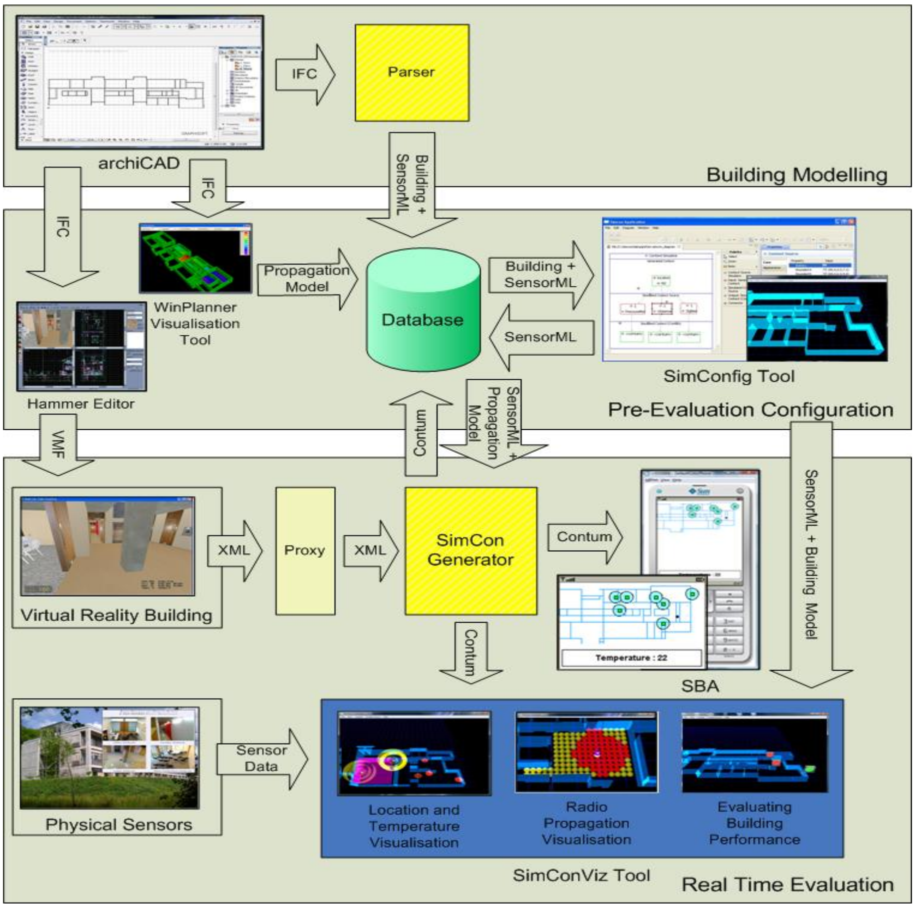

Figure 1: Smart Building Modelling to Support Smart Building Application Evaluation

Interdependencies and shared information across all the models support the variety of stakeholders involved in the lifecycle (design, management and maintenance) of a smart building e.g. Architects, Facility Managers and Application

[2] http://www.graphisoft.com/products/archicad/

Designers. Findings at every stage feed into other stages allowing each specialist to conduct meaningful evaluation early and repeatedly in the design phase, adjusting only the parameters of interest to them. In this paper we concentrate on evaluating tools which support the evaluation of the SBAs using VR and location based context simulation and visualisation.

3.1 Modelling SimCon (Simulated Context) Sources

We begin by describing our approach to sensor and context modelling. A sensor which provides context to a SBA is a "Context Source" [6] or SimCon Source (SCS). We call a discrete unit of simulated context produced by a low level context source a "Contum". A contum (a contraction of context quantum) is the smallest indivisible value of context possible and has an associated level of uncertainty, introduced through the simulation process in order to reflect the uncertainty caused by inherently unpredictable fluctuations in the readings of a measurement apparatus.

In order to provide meaningful context to a wide range of smart applications, we have identified the following requirements for our context source simulation model.

3.1.1 Requirements

Here the specific qualities which are required of the SimCon model are defined:

i. Generality: The core model must be sufficiently generic to support simulating a range of heterogeneous contexts providing usable context for SBAs.
ii. Extensibility: Where new context simulation is required (for specific requirements of an SBA), it should be possible to extend the core model to support this.
iii. Interoperability: By basing the SimCon conceptual model on existing context and sensor models, interoperability will be maintained in the model-set.
iv. Scalability: The simulation process should scale to meet the requirements of existing and future smart building application evaluations.

That the generated simulated context be:

i. Dynamic and Accurate: Context represents data sensed from the environment. Simulated context must reflect the dynamic temporal and spatial nature of the virtual environment and interactions within it. The dynamic simulated context should also attempt to approximate the behaviour of real context as closely as feasible including accurately reflecting the uncertainty inherent in real world context.

From previous analysis of SBs, context and SBAs, it is seen that context is an abstraction of sensed data. In order to simulate these types of abstraction and provide support for rapid evaluation of SBAs design using VR, the following requirements for SimConfig and SimConViz have been identified:

i. Configurability: The SCS's must be configurable in order to support the rapid design of SBAs (e.g. data rates and accuracy).
ii. Usability: The process of deploying and configuring a SCS within the virtual environment should be usable for a range of disciplines (involved in SBA design, e.g. civil engineers, computer scientists) and support rapid evaluation.

In this paper we evaluate the configurability and usability of SimCon.

3.1.3 Types of Context

There has been a heavy emphasis on using location based context within the context-dependent application domain [Want, et al. 1992, Schilit, et al. 1994, Abowd, et al. 1997, Fels, et al. 1998, Bardram 2004, Mara, et al. 2009]. This combined with the tendency towards error and the particular nature of error within location-based systems, due to the mobility of transmitters and changing environmental effects on radio wave propagation [Zyren and Petrick 1998], has motivated the focus of this paper on modelling location based context. We have identified 3 types of contum which cover a wide range of potential location based systems.

- 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 (for example an RFID tag)
- Coordinate: context which alert of the coordinate (tied to some reference frame) (e.g. a real time location system).

3.1.4 SimCon Model

In order to successfully simulate these types of context, we begin by capturing a set of basic properties which are essential to modelling simulated context sources, based on the above requirements. (Table 1) shows these properties.

Property	**Description**
G.U.I.D.	Global Unique Identifier
Type	Type of Context Source
Position	This can be a geodetic coordinate or related to some local frame (for instance a Cartesian frame within a building).
Envelope	The bounds of a context sources' range.
Response Curve	Response characteristics are specifications for accuracy and precision, delays, and behaviour (including reliability) under certain environmental conditions.
Output/ Contum	A value representing a measurement (coordinate, received signal strength etc.) An output is derived using the response characteristics of the context source.
Measurement Interval/ Rate	The optimum rate at which a sensor produces measurements (before delays are introduced).
Phenomena	The type of phenomena the sensor detects (temperature, touch, radio waves).

Table 1: Context Source Properties for SimCon Sources

Every context source has a unique id and type (e.g. coordinate). As a context source represents a physical sensor in the environment, it also has a position and an envelope (which represents the bounds beyond which a sensor is not capable of measuring phenomena). A phenomena represents what the context source is measuring (e.g. the received signal strength between a transmitter and receiver). By defining response curves, it is possible to extend the model to represent ever more fine grained filters which may 'modify' the measurements in some way [Reynolds, et al. 2006]. These are only restricted by the computational limits of the system in which the simulation is being run. The output (or contum) encodes the format of the output as well as the type of data output (determined by context source type). This can be a generic piece of location context (e.g. location) or represent a specific location based system (e.g. a ZigBee transceiver). Figure 2 shows two example contums from two simulated contexts sources.

```
<contum>
 <type>coordinate</type>
  <cell>0003</cell>
     <tagID>0004</tagID>
  <coordinate >
     <x>56.6314</x>
     <y>4.28185</y>
     <z>8.23979</z>
  </coordinate >
  <time>257.1221</time>
```

```
<contum>
<type>ZigBee</type>
<id>0001</id>
   <rss>-45.8009</rss>
   <transmitterID>0002</transmitterID>
</contum>
```

Figure 2: Example Contums

3.2 Integrating SimCon into sensorML

To improve interoperability the SimCon Tool Set defines context sources using sensorML, an XML-based syntax for portable sensor description. Figure 3 shows a response curve described in sensorML using UML notation for clarity. The "zigbeeTransceiverSteadyState" and "zigbeeAccuracy", together give an expected received signal strength of 8 within 0.6 meters of the origin of the sensor with a standard deviation of a maximum of 5 (using a Gaussian distribution) (see section 4.3 SimCon Generator).

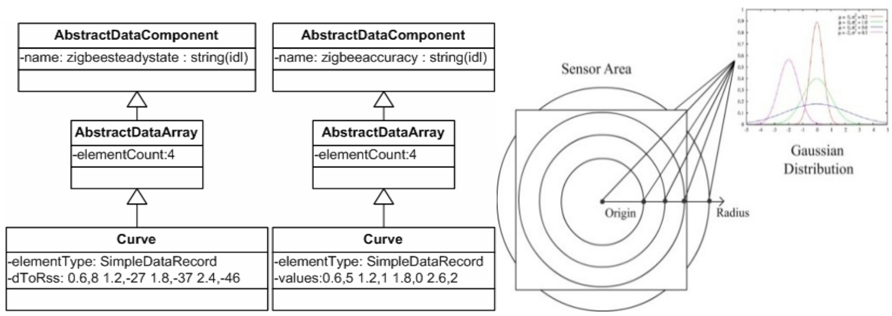

Figure 3: SensorML Description of a ZigBee Reciever Steady State Response Curve and associated Error Response Curve.

3.3 Integrating SimCon into IFC

While IFC2x3 has descriptions of basic sensor properties and a set of property sets for describing a range of other sensor types (heat/humidity etc.), IFC does not yet have the kind of rich data descriptions of sensor systems that are required to describe the properties of complex sensor types. The IFC model does however support extensibility and flexibility through the use of property sets. An IFC property set is comprised of a simple attribute-value pair, where a value can be defined as one of a number of predefined types e.g. scalar, pair of scalars or table among others. This extensibility of IFC property sets supports the interoperable approach adopted for this context source simulation platform in order to meet the objectives of a data centric approach to SB modelling. Table 2 displays our extensions to an existing IFC entity called IfcSensorType which defines a particular type of sensor using property sets to specify the parameters of that sensor.

Type of Information	Property Type	Functional Part
G.U.I.D	IfcPropertySingleValue	IfcGloballyUniqueId
Type	IfcPropertySingleValue	IfcLabel
Placement	IfcPropertySingleValue	IfcObjectPlacement
Envelope	IfcPropertySingleValue	IfcBoundingBox
Response Curve	IfcPropertyListValue	IfcTable
Output/Contum	IfcPropertySingleValue	IfcValue
Measurement Interval/Rate	IfcPropertySingleValue	IfcTimeMeasure

Table 2: Simulated Context Source Information Requirements

These additional encodings have been defined to correspond with the context source properties defined already for SCSs. Existing IFC descriptions make provisions for an entities position and physical properties within IfcProduct and so these have not been transferred from the definitions in Table 1.

4 Smart Building Simulation and Visualisation

The simulation and visualisation toolset presented here has been developed to address the novel concerns associated with SBA design such as investment required, scale of environments and heterogeneity of data. The toolset has also been designed with the multitude of potential end-users involved in the design of smart buildings e.g. architects, sensor and SBA developers. The information models described in section 3 are used to drive the simulation and visualisation. The tools designed around this model are designed to minimise the need for domain experts to step outside the bounds of their professional role and skills as part of the design process.

Simulation in this work focuses on two aspects of the environment. Firstly it provides for generation of simulated sensor data at runtime, based on the activity of real users and non-player characters. Secondly it allows actuations of entities in the virtual world e.g. lights automated doors. These actuations happen when signalled by the System Under Test (SUT). Visualisation is used to provide two distinct views of the environment (Figure 1). A first person view provided by 3D immersive simulation allows test users to interact with an application in virtual surroundings. This is built upon existing BIM models like IFC which form blueprints for VR building modelling using the hammer editor (part of Half Life 2 (HL2) games engine [14]). A 2.5D person visualisation element, implemented using OpenGL, allows an overview of the SB and context generation within it and can provide a designer/tester with feedback about the status of the environment at runtime. This requires a conversion of the BIM into coordinates for each wall section in the building.

In this section we discuss the current implementation of this simulation and visualisation toolset for supporting SBA design.

4.1 3D Visualisation

For the designer of adaptive systems, delivering robust applications suitable for the target deployment environment is a complex process. Applications designed to improve an end-user's day-to-day activities can fail in deployment where end-users are unaware or uninformed about the social or interaction paradigms surrounding the technology. End user satisfaction requires a balance where system performance is matched by a user's confidence and at the application level and is an issue central to the success of ubicomp at a commercial level. It is no longer only explicit user instruction which affects these systems; users can implicitly impact a system's exhibited behaviour through their movements and activity, or lack thereof.

Creating realistic virtual environments is complex, expensive and time-consuming. However, the current generation of game engines are sophisticated, tightly engineered for usability and performance and offer realistic graphics and advanced Artificial Intelligence [Trenholme and Smith 2008]. In this instance, the Half-Life 2 game engine [14] provides the 1st person VR environment. Multiplayer simulations allow up to 32 users to interact simultaneously in the context of the virtual world. Bot driven simulations further increase this number as role playing bots roam the virtual world testing defined scenarios.

The availability of realtime avatar location information within a game engine makes it a particularly useful tool for testing location-aware systems. Both user-driven characters and bots generate simulated sensor data at runtime which in turn is supplied to the rest of the toolset to drive experimentation. Bots are particularly useful for experiments of long duration because they can be left unsupervised for prolonged periods or overnight. The toolset builds on this using HL2 to maintain a global state of the world and to generate XML encoded messages containing the precise location of the user's avatar (Figure 1). Combining this approach with the SimCon model allows properties for location based context sources to be captured at runtime.

4.2 SimCon Source Configuration Tool and Generator

The simulated context source and tag modelling is handled through a set of tools collectively called the SimConfig Tool [McGlinn, et al. 2008] (Figure 4). This includes a graphical interface built upon the Graphical Modelling Framework[3] (GMF) for creating and configuring simulated context sources (accuracy, rates, and delays), a visual aid for placing context sources within a SB and providing visualization of context source properties (G.U.I.D., type, position and bounded area).

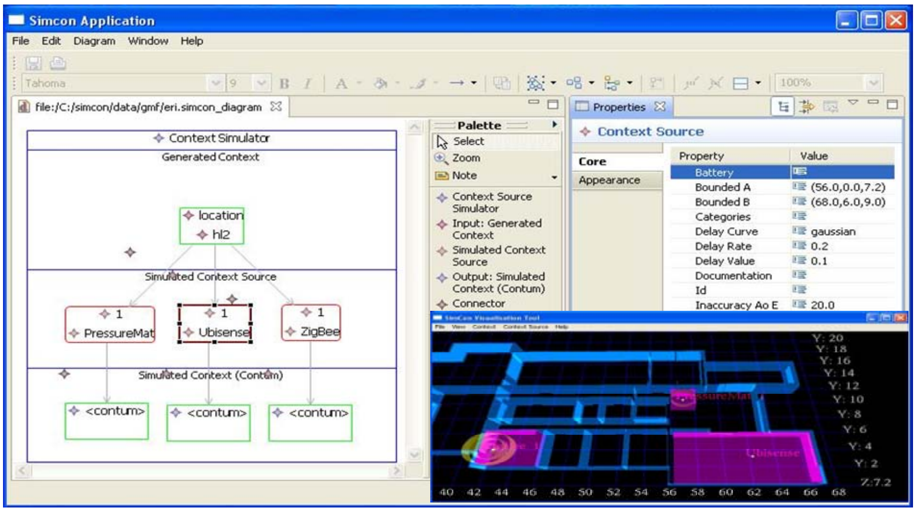

Figure 4: SimConfig Tool

The XML encoded data provided by the HL2 VR environment is accessed by SimCon Generator (written in Java) through a proxy. On start-up, the SimCon Generator loads in all the sensorML sensor descriptions and userXML descriptions (configured using the SimConFig tool) from an eXist database[4]. When a tagged avatars location falls within an appropriate bounded area, a "Contum" (simulated context) for that simulated context source is written back to the database. This can be either a generic context (e.g. coordinate, proximity, presence) or one which is modelled on a real sensor deployment (e.g. a ZigBee Transceiver or ubisense real time location system[5]).

We introduce error using Gaussian noise [Hightower and Borriello 2001], [9]. For instance, ubisense provides location as a coordinate. The SimConfig tool allows an error to be introduced into the accuracy of the reading, so each simulated "reading" can be offset from its actual position a pseudo random amount (for example, within 0.15 centimetres of its location within the virtual SB). A ZigBee transceiver can provide received signal strength between a transmitter and receiver (to determine proximity). This signal strength may fall within a range around the expected steady state value which can be modelled as a Gaussian distribution up to differing levels of granularity. We calibrated our simulation using real world sensor systems. Ubisense simulation was modelled on specifications taken from the documentation provided

---

[3] http://www.eclipse.org/modeling/gmf/
[4] http://exist.sourceforge.net/
[5] http://www.ubisense.net/en

with the Ubisense system and the ZigBee Transceiver mote simulation was modelled on calibration readings taken from within a lab[McGlinn, et al. 2009].

4.3 Visualisation of Context and Sensor Data

To support the evaluation process a 3rd person visualisation tool has been developed (Figure 1). This provides a means to visualize the SB environment and all avatars within the building. The tool also supports visualisation of different types of context in real time: proximity, coordinate locations and temperature, as well as radio wave propagation. The Visualisation Tool is built upon Java's SWT (Standard Widget Toolkit) bindings for Java OpenGL (the Open Graphics Library). By using the SWT bindings, it will be possible to integrate this with the GMF for a more integrated tool set. Currently though the Visualisation Tool is a standalone application. The Visualisation tool uses 2D coordinates for walls (2 x, y points) which can be semi automatically retrieved from IFC models. These can be displayed in either 2D or 3D.

This 3rd person visualisation tool provides feedback to evaluators on how an application perceives simulated location context within the SB. For proximity and presence, location context is represented as a coloured area while coordinates are indicated by a small avatar. The generic context model provides for an extension to allow different context sources be modelled. As a result we have implemented the visualisation of temperature context for a context source within a SB. The current implementation has been using live sensor data, building on work conducted on a live demo [Mara, et al. 2009] which overlaid live sensor data for temperature and humidity from a physical building, the Environmental Research Institute (ERI) in Cork, with the occupancy of a simulation of the same building being run in the Science Gallery in Trinity College Dublin (TCD) (Figure 1).

Sensory data from the ERI was recorded to a Global Sensor Network (GSN)[6] which was then accessed via an http connection in Trinity College Dublin (TCD) and recorded to an eXist database, which is accessible to the SimCon Visualisation Tool. By overlaying temperature context visualisation with occupation, it is possible to get new insights into occupancy and its effect on temperature as users interact with the environment.

4.3.1 Visualising Real Time Radio Propagation

The visualisation of estimated radio propagation from wireless devices provides valuable insight into the influence of a particular building structure on the application being simulated. WinPlanner utilises IFC models as an input is a wireless sensor design tool [McGibney et al. 2007, Guinard et al. 2009] (Figure 1). The physical location of sensor nodes strongly influences the performance of the network from the perspective of accurate data sensing and reliable communication [McGibney, et al. 2008].

The use of the design tool can also reduce the need for labour intensive site surveys and avoid costly redeployments. The design tool improves on current deployment strategies for indoor WSN with particular focus on the application to energy aware buildings. WinPlanner uses IFC to capture application requirements and

[6] http://sourceforge.net/apps/trac/gsn/

also the physical characteristics of the building where the network should be deployed, including the number of walls, their position and material type. Based on these requirements, the design tool automatically optimises the number and more importantly the position of wireless devices to meet user defined application requirements. A key element to this design step is an accurate propagation model.

```
<WinPlannerPropagation floorplan="ERI_FIRST_FLOOR" originx="0"
originy="0">
    <transmitter name="sensor_0" description="Temperature"
    x="55.51" y="10.57" z="0" frequency="2.4GHz" power="0">
        <prediction model="Motif Model"
        data="Signal Level" unit="dBm">
            <gridpt x="17.4918" y="10.2249" z="0">-98.5791</gridpt>
```

Figure 5: WinPlanner Propagation Model

The design tool utilises a 2D ray tracing model known as the Motif Model [Klepal and Pechac]. Although the Motif Model is more computationally demanding than empirical models such as Multi Wall Model [Damosso and Correia 1996], it is much faster to compute than other ray optical based models as it takes advantage of the simple line-drawing techniques by dividing the environment description into a grid.

WinPlanner was integrated with SimCon to provide real time simulation and visualisation of the influence of the building structures influence on radio propagation signal strengths. WinPlanner outputs the radio propagation over a grid of points within a floor of a building (Figure 5). Using a set of these, a scenario in which a simulated receiver is placed in the virtual SB and which detects the changing signal of a user as he moves around with a simulated transmitter was implemented. The integration of WinPlanner with SimCon provides a form of simulation which can evaluate the effect of varying signal strengths on an application's adaptive behaviour. This is particular relevant for SBAs which rely on signal strength to determine proximity information, but requires further evaluation.

5 Evaluation

We set out to design and implement a tool which met the requirements stated in section 3.1. The scalability and accuracy of the SimCon Generator is not covered in this paper, but our initial evaluations of scalability have shown that SimCon can support 800 SimTags triggering 800 SCSs producing 800 contums (the equivalent of 800 avatars each triggering a contum) with delays below a tenth of a second latency, which is below the maximum execution time required for accurate simulation of the three types of context we currently simulate. This number falls well within the requirements of existing indoor context-aware application evaluation scenarios [Anind and Abowd, Want, et al. 1992, Schilit and Theimer 1994, Abowd, et al. 1997, Sumi, et al. 1998, Bardram 2004, Mara, et al. 2009]. Details on the dynamicity and accuracy of the SCSs and calibration can be found here [McGlinn, et al. 2009]. In this section we focus on the usability of SimConfig and SimConViz tools.

5.1 Evaluating SimCon Usability

An iterative approach was taken to the design and evaluation of the usability of the SimConfig tool. This built upon our initial evaluation of the SimCon prototype[McGlinn, et al.] which makes use of the Standard Usability Scale (SUS)[Brooke 1996]. Evaluations took place on a laptop in the TCD and University College Dublin (UCD) campus and ERI building in Cork. Each evaluation looked at specific tasks with regard to the iterative design of the tool. These tasks focused on the process of creating, placing and configuring SimCon sources inside the VR SB.

The first evaluation deals with the SimConfig GMF interface. The second with both SimConfig and the SimConViz tool to support evaluation of a hypothetical SBA. The third looks at evaluation of an emulated SBA behaviour. We measured usability both qualitatively (efficiency, learnability, error, satisfaction) and quantitatively (time to complete tasks). We examined a range of users with varying backgrounds which aligned closely to those of expected users of SimCon (e.g. knowledge of indoor location-aware systems).

5.2 Evaluation 1 To investigate usability of the SimConfig Tool when creating, placing and configuring two heterogeneous SimCon Sources for creating location-based contums.

The first evaluation presented here set out to investigate usability when configuring two SCSs for location-based contums using the SimConfig Tool. The focus was on how the participant interacted with the GMF configuration tool specifically. Five participants (all PhD students) took part in this evaluation, four Ph.D students from the Knowledge and Data Engineering Group (KDEG) in TCD, and one post doctorial researcher from the Distributed Systems Group (DSG) in TCD.

The usability test began with an introduction which set the role of the participants. It explained the two types of location systems which the simulated context sources were based upon; ubisense real time location and ZigBee transceiver motes providing links to details on each of these systems. Material provided consisted of a download of the prototype SimConfig tool. Also, as our original implementation did not include SB visualisation, we provided a ground floor plan to create a cognitive link between the process of configuring the simulated context source and its relation to its position in the virtual environment. There were five tasks in all:

1. Open, execute and familiarise themselves with the SimConfig Tool.
2. Create a SimCon Generator.
3. Create and place a Ubisense Cell SimCon Source and configure a blanket error distribution for the cell, an output rate, and an introduce a delay
4. Create a second Ubisense Cell SimCon Source with an additional error distribution.
5. Configure a Tyndall ZigBee Proximity Transceiver SimCon Source and configure its steady state responses, delays and accuracy.

The tasks themselves had minimal instruction, but a link was provided to a more complete set of instructions.

5.2.1 Findings

The pre-questionnaire found three of the participants had no experience with ubiquitous computing. Two of the participants had experience with context-aware systems, one with "context aware and ubiquitous computing", the other "pervasive computing, and middleware". Only one had experience conducting research using sensor systems.

All participants completed the tasks, varying in times from 27 minutes to 57 minutes. Figure 6 shows the break down of the times to complete each task. Downloading and opening the tool took on average 2 minutes. One participant took 6 minutes as they decided to read ahead to other tasks before completing this task. Placing a SimCon Generator also took on average approx 1 minute. Creating, placing and configuring the first Ubisense SimCon Source took on average 14 minutes. Repeating Task 3 and configuring a more precise granularity for accuracy took on average 10 minutes. Creating, placing and configuring a Tyndall ZigBee Proximity SimCon source and configuring its response curve took on average 12 minutes.

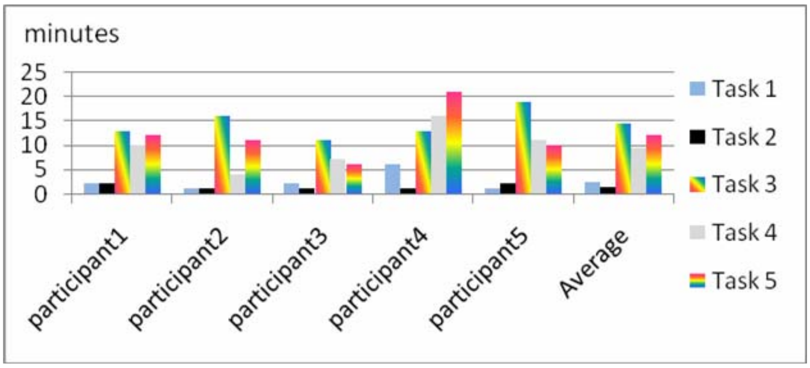

Figure 6: Time to complete tasks

The majority of errors were related to the interface (position of buttons, tab features not being enabled). Also, to access the widget for placing and configuring the SCS required double clicking on free space within a "SimCon" node in the GMF interface. The term fidelity in reference to context source accuracy and delays was not clear to a number of participants. Also, "steady state response" was also not clear and required further explanation for participants who are not familiar with the workings of sensor deployments.

The post questionnaire set out to evaluate participant satisfaction with the tool and consisted of the following questions and results, ranging from very easy, easy, difficult, very difficult (

Figure *7*):

1. Q1: How did you find the prerequisite information to understand?
2. Q2: Downloading and opening the SimConfig Tool was?
3. Q3: Creating a SimCon Generator was?
4. Q4: Familiarising yourself with the SimConfig tool was?
5. Q5: Creating a SimCon Ubisense Cell was?

6. Q6: Setting the SimCon Source area, delay and accuracy was?
7. Q7: Configuring a SimCon Source error distribution was?
8. Q8: Configuring a SimCon Source response curve was?

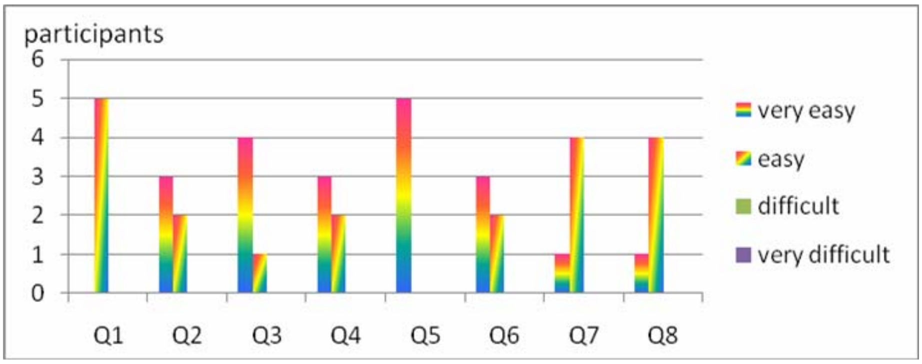

Figure 7: Post Question Answers

5.2.2 Interpretation

The majority of errors were a result of bugs in the prototype. The introduction of extensive instructions reduced errors but also resulted in participants not fully engaging with the usability test, as they could complete tasks by simply following the instructions word for word.

Task 3, 4 and 5 required participants to read lists of numbers from the browser and then enter them into the SimConfig tool. This process was laborious and had an impact on the efficiency of completing tasks. Some of the participants discovered they could copy and paste values from the instructions into SimConfig and this sped up the process again at the expense of engagement. Task 4 required the participant to repeat the process of task 3 (adding a Ubisense Coordinate SimCon Source) with additional configuration of an error distribution curve. There was a marked improvement in times for the majority of participants to complete task 4 over task 3 even with this additional configuration, which demonstrates memorability when repeating similar tasks. In task 5 the configuration of the Tyndall ZigBee Proximity SimCon Source was also an improvement over the configuration in task 4 (as the configuration of error distributions and response curves are very similar).

The post questionnaire gave good indications about overall usability of the tool. The GMF interface made the process of creating, placing and configuring a SimCon Source an easy task for all participants regardless of their background. The questions did not however reveal enough detail about whether the users would use this tool, although, the participant with experience in sensor deployments was enthusiastic about it in conversation and recommended that it could be improved through the use of a tool to place the context sources within the environment. The GMF interface made the process of creating a SimCon Source an easy task for all participants, as indicated by the post questionnaire. The pop out widget also allowed participants to place a SimCon Source and configure its accuracy, delays and response curves with ease, although certain aspects need improvement (how data is entered and presented).

5.3 Evaluation 2 To investigate usability of the SimConfig and SimConViz Tool when creating, placing and configuring three heterogeneous SimCon Sources for creating location-based contums in order to evaluate a hypothetical security system.

The second evaluation investigated usability of the SimConfig and SimConViz Tools. Specifically looking at how the newly implemented SimConViz improved the process of placing a SCS in the VR SB. Three participants in all took part in this evaluation (the number is low due to difficulty finding participants experienced in this area). This consisted of two PhD students studying Civil Engineering in University College Cork (UCC) and one post doctorial researcher from the Cork Institute of Technology (CIT). Pre-requisite information in this usability test consisted of requirements for a hypothetical security application. This security system has two functions:

1. Alert security when an unauthorised user has entered the building.
2. Alert security when a tagged item has been moved by an unauthorised user.

A combination of presences, proximity and coordinate SCSs are required at entry points and corridors to detect between authorised and unauthorised users. There were six tasks:

1. Familiarise yourself with SimConfig.
2. Familiarise yourself with SimConViz (to do this, they were required to move around the VR building and observe the avatars changing location in the visualisation tool).
3. Create, place and configure a simulated context source which provided coordinate context and apply pre set Ubisense properties.
4. Deploy and configure a simulated context source which provided proximity context and apply pre set Tyndall mote transceiver properties.
5. Create, place and configure a simulated context source which provided presence context and apply pre set pressure mat properties.
6. View the new SimCon Sources in SimConfig and analyse the contums using SimConViz (with respect to the security application to determine which types best suits your requirements).

5.3.1 Findings

The pre-questionnaire revealed that all the participants had experience using indoor location tracking systems. And two had knowledge of ubiquitous computing, context-aware computing and sensor deployment. Two also had intermediate experience of indoor tracking systems and indoor location tracking systems.

All the participants completed the tasks in times varying from 37 to 39 minutes. Figure 8 shows a breakdown of time to complete each task. The average time per task was of 4, 3, 12, 6, 3 and 10 minutes respectively.

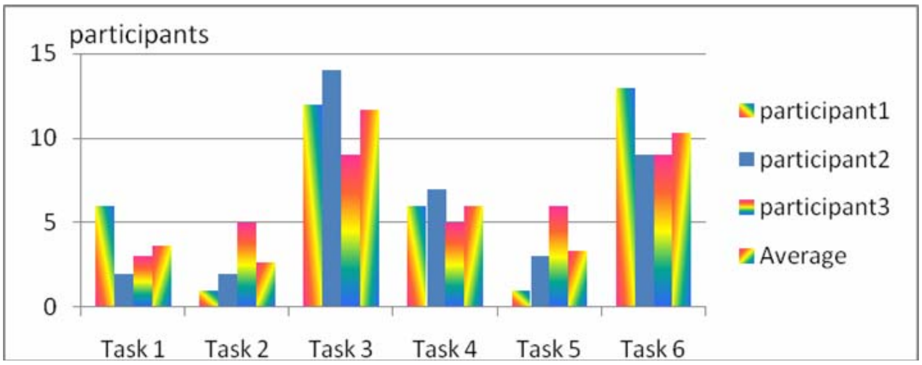

Figure 8: Time taken per task and average time per task.

A typical error which occurred was difficulty seeing the Z coordinate on the SB visualisation (two participants required consultation with the instructor). Also, a number of functions with the user interface still cause problems when they are first encountered. These include double clicking to access the configuration widget and the process by which a "connector" connects the "activities" which involves dragging the connector.

Analysing the Security Application

Each user was required to analyse the contums to determine which SimCon Source best met the requirements for the security application. This task was straight forward as the instructions had given details on the types of contums the application required and the types of SimCon Source which supply those contums. All users agreed that in order to determine whether non authorised users were in the building presence contums were required at strategic locations (doors, hallways). For tracking authorised users coordinate contums provided the most information on the authorised user's location and therefore minimised the amount of uncertainty the application faced. The participants were given discursive questions which were to be answered during the evaluation. These were:

1. Did you feel these context sources met your requirements for the security application?
2. Which context source or combination of context sources best met the security applications requirements and why?
3. Do you think this tool would be useful in evaluating context-aware applications and why?

All participants answered yes for question 1. Question 2 was answered mainly through discussion, and all participants agreed that the use of presence and coordinate locations would suffice for the security application. For question three the most experienced participant (the post doctorate with experience in sensor simulation) answered "It's good to be able to visualise coverage regions" but "auto selecting bounded regions [for a SimCon source]" would be useful. He also would like to see the visualisation tool integrated with the configuration tool (the current implementation has the SimConfig and SimConViz tool as separate applications). Another participant answered "Yes, stable, easy to learn and use".

This questionnaire also evaluated usability using the Standard Usability Scale (SUS). Each question had a range of answers from strongly agree to strongly disagree.

- S1: I think that I would like to use this system frequently.
- S2: I found the system unnecessarily complex.
- S3: I thought the system was easy to use.
- S4: I think that I would need the support of a technical person to be able to use this system.
- S5: I found the various functions in this system were well integrated.
- S6: I thought there was too much inconsistency in this system.
- S7: I would imagine that most people would learn to use this system very quickly.
- S8: I found the system very cumbersome to use.
- S9: I felt very confident using the system.
- S10: I needed to learn a lot of things before I could get going with this system.
- S11: I think this system would meet my specific requirements.

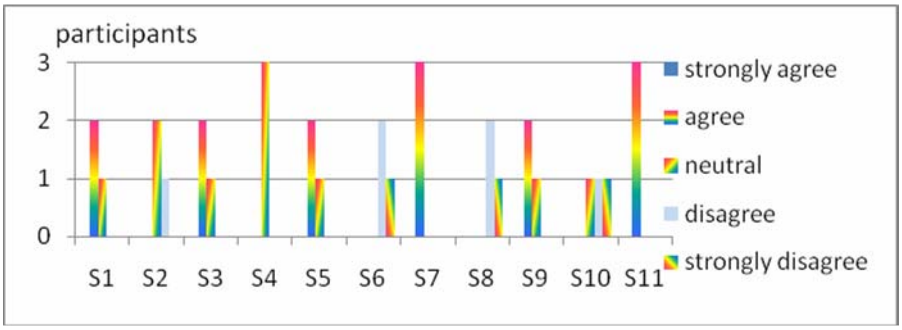

Figure 9: Post- Questionnaire Answers

5.3.2 Interpretation

The findings for placing and configuring the context sources are promising. On average, the time taken to place the second SimCon source compared with the first was 50 percent. This shows that memorability had an impact on the time it took to place each context source. Participants generally only consulted the instructions for the first and second deployment, after which they felt comfortable deploying a SimCon source independently of instruction. One participant did not have strong English; as a result they found the instructions hard to follow. This had an impact on the time they took to complete the tasks.

The analysis of the security application focused the participant's attention to a specific goal. As the task was straightforward, the participants had little difficulty completing it. The visualisation of proximity and coordinate contums was useful in highlighting the different levels of certainty involved in these systems. The use of the

SUS gave good feedback on how participants found using the tool (Figure 10). The majority agreeing that they would use this system frequently, that it was easy to use and that they felt confident using it. In addition, we asked if it met their specific requirements, which they agreed it did.

5.4 Evaluation 3 To investigate usability in the SimConfig and SimConViz tool when creating, placing and configuring a SimCon Source to support participants evaluating the effect of varying accuracy in simulated location context on a location-aware application.

The third evaluation set out to investigate how rapidly a user can deploy and configure a simulated context source using SimCon in order to evaluate the effect of varying accuracy on a location-aware SBA. Specifically we set out to determine how the visualisation of context highlighted the issue of uncertainty in location context. Eleven participants in all took part in this evaluation. This consisted of six PhD students from TCD and three PhD students and two post doctorial graduates from University College Dublin (UCD). A J2ME emulator was used to evaluate an SBA which makes use of location context to manoeuvre around a virtual maze. The scenario required the participant to evaluate the impact of changing levels of precision in location context on the application. This exercise only required the use of one type of location context (coordinate). There were five tasks:

1. Download the material and familiarise yourself with the SimConfig and SimConViz tool. (By moving around the VR SB and observing the SimConViz tool).
2. Create, place and configure a SimCon Source and apply pre set Ubisense properties to provide a contum which reflects a Ubisense output.
3. View the SimCon Source in the SimConfig tool, and also view the contum (represented by a green avatar) in real time as it was generated by the SCS.
4. Change the accuracy and delay values on the SCS and test the application once again.

5.4.1 Findings

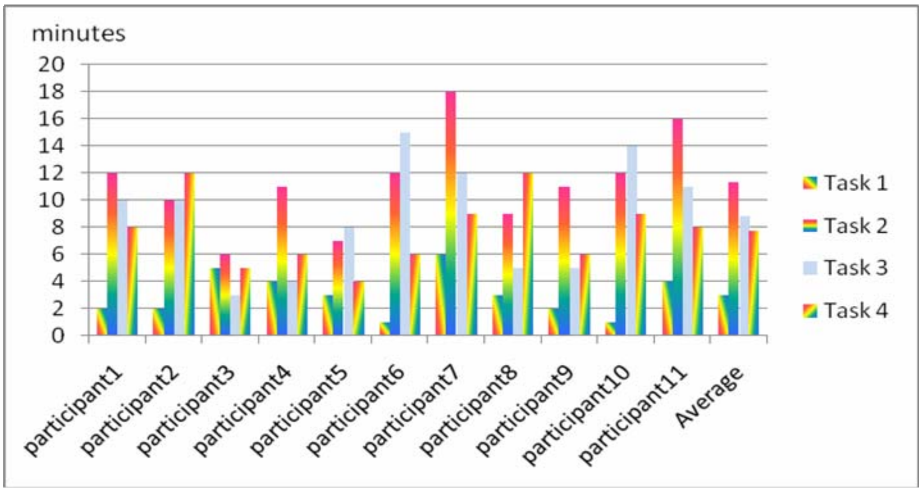

Figure 10: Time taken per task and average time per task.

The pre-questionnaire examined the users experience with regard to indoor location aware systems which consisted of one expert, three with intermediate knowledge, four novices and three with no experience. All participants completed the tasks ranging in times from 19 to 45 minutes. Figure 10 shows how much time was spent on each task. Task 1 took on average 3 minutes. Task 2 took on average 11 minutes. Task 3 took on average 9 minutes. Task 4 took on average 8 minutes.

Noteworthy errors were difficulties due to the number of windows they were expected to navigate, causing some confusion when attempting the tasks. One participant had difficulty navigating the VR SB.

The post questionnaires examined the users on how much they learnt about the issues facing location-aware application design. The SimConViz tool allowed participants to quickly evaluate how introduced delays and changes in accuracy of location had an affect on how this location is "perceived" by the context source in real time. This put the issue of jumpy inaccurate location and its effect on how the application performed quickly into perspective. Additional feedback from participants was generally positive, the majority of which found that the SimCon tool highlights the effects of varying accuracy of data on the application. Example comments include: "Found it a little difficult moving between multiple applications (visualisation tool, gmf tool, phone, instructions), layout + overall parts/application were intuitive -> I could see how they all worked together" and "Gives [the tool] a feel as to how modifying parameters effect the application very quickly".

5.5 Usability Evaluation Conclusion

We set out to evaluate the usability of the SimConfig and SimConViz tool. We have demonstrated that the tools have been designed (through an iterative design process) to provide a usable approach to creating, placing and configuring SCSs within a VR

SB, for a range of participants (civil engineers, location-aware system developers). While some issues with the interface continue to cause errors, these are minor and cannot be wholly designed out of the system as they are often the result of the particular experiences of the user. The contums have been used to evaluate a location-aware SBA, allowing participants to quickly evaluate how varying the accuracy affects the performance of the application. The visualisation of contums has demonstrated itself to be useful in teaching users about these types of issues, which we believe can be applied to a range of mobile applications to highlight these effects for SBA designers.

6 Conclusion and Future Work

In high value capital works like buildings, where conventional prototyping and production runs are not appropriate, it is important to have the capability of simulation i.e. testing and verifying the design while it is still in the design phase. The emergence of smart building applications further adds to the issues which must be considered for modern buildings. In this work we have extended an established building model, Industry Foundation Classes, to include descriptors and parameters relevant to smart building technologies. We have combined this with a simulation and visualisation toolset for supporting smart building (SB) design. IFCs provide an established building standard in which to ground this work. The Simulated Context (SimCon) Model provides a generic method of simulating a range of location based contexts. Extensions to IFC have been based in parameters from the SimCon Model not already catered for by existing IFC property sets. The flexibility of the modelling approach is further improved by extensibility in the SimCon Model through the use of response curves to provide linear based models to simulate problems such as error curves and delays in simulated context. SimCon can also incorporate complex models allowing the tools to benefit from external models not implemented as part of the toolset i.e. in the case of radio propagation. Finally, by sharing the conceptual model with an existing sensor modelling language (sensorML) we have retained interoperability in the resulting model-set.

Using this approach we have developed a platform that provides a run time test environment for simulation and visualisation of SBs. For SBA developers, this platform provides a relatively inexpensive and scalable means to do early, rapid, repeatable user centric evaluations of their SBA over a range of scenarios. We have also developed and evaluated the SimCon toolset which allows developers place and configure SimCon sources to simulate a range of location based contexts. These evaluations have demonstrated that the SimCon tools are usable for a range of users with requirements close to those of expected users and provide capabilities to support SBA evaluation, like modelling "accuracy" of context to determine the minimum accuracy required of a contum for the SBA to meet the user's goals. The SimConViz tool also provides intuitive visualisation of how the SBA perceives the environment.

Findings at this level of evaluation can feed back into the design of the underlying sensor infrastructure. Likewise, sensor specialists can see in real time how the properties of the building effect radio propagation as a transmitter moves around and can evaluate how this will impact on an application which relies on data provided by the receiver (for example, where signal strength is used to determine proximity). This type of data can feed back to civil engineers and the design of the building itself. This approach provides valuable insights into smart building and smart building application design early in development, thus reducing the risk to developers. While this discussion has focused largely on modelling of location information, it should be noted that the model's definition is extensible to allow addition of other sensor systems in the future.

Acknowledgements

This work is supported by the Higher Education Authority (HEA) under the NEMBES project (www.NEMBES.org). We would also like to acknowledge James O'Donnel for his contribution to this paper.

References

[Abowd, et al. 1997] Abowd, G.D., et al. "Cyberguide: A mobile context aware tour guide." Wireless Networks 3(5): 421-433.

[Anind and Abowd 2000] Anind, D. K., Abowd, G. D.: CyberMinder: A Context-aware System for Supporting Reminders. Proceedings of CHI (2000).

[Bardram 2004] Bardram, J. E.. "Applications of context-aware computing in hospital work– Examples and design principles." Symposium on Applied Computing Proceedings of the 2004 ACM symposium on Applied computing Nicosia, Cyprus: 1574 - 1579.

[Barton and Vijayaraghavan 2002] Barton, J. J., Vijayaraghavan, V.: "UBIWISE, a simulator for ubiquitous computing systems design." HP Laboratories, Oct.

[Bazire and Brézillon 2005] Bazire, M., Brézillon, P.: "Understanding context before using it." Lecture Notes in Computer Science **3554**: 29-40.

[Bisgaard, et al. 2004] Bisgaard, J. J., et al. "How is Context and Context-awareness defined and Applied? A survey of Context-awareness." Aalborg university.

[Botts 2007] Botts, M.: "Sensor Model Language (SensorML) for in-situ and remote sensors." Open Geospatial Consortium Inc.

[Broens and Halteren 2006] Broens, T., Halteren, A.: SimuContext: Simply Simulate Context. In ICAS '06: Proceedings of the International Conference on Autonomic and Autonomous Systems, IEEE Computer Society.

[Brooke 1996] Brooke, J.: "SUS-A quick and dirty usability scale." Usability evaluation in industry: 189-194.

[Bylund and Espinoza 2002] Bylund, M., Espinoza, F.: "Testing and demonstrating context-aware services with Quake III Arena."

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

[Chen, et al. 2004] Chen, H., et al. "An ontology for context-aware pervasive computing environments." The Knowledge Engineering Review **18**(03): 197-207.

[Clements-Croome 1997] Clements-Croome, T. D. J.: "What do we mean by intelligent buildings?" Automation in Construction **6**(5): 395-400.

[Cook and Das 2004] Cook, D., Das, S.: Smart environments: Technology, protocols and applications, Wiley-Interscience. (2004).

[Damosso and Correia 1996] E. Damosso, E., Correia, L. M.: "Digital mobile radio towards future generation systems." Commission of the European Communities (1996).

[Dey 2000] Dey, A. K.: Providing architectural support for building context-aware applications. Thesis, Georgia Institute of Technology (2000).

[Dey and Abowd 2000] Dey, A. K., Abowd, G. D.: Towards a better understanding of context and context-awareness. CHI'2000 Workshop on the What, Who, Where, When, and How of Context-Awareness.

[Guinard et al. 2009] Guinard, A., McGibney, A., Pesch, D.: "A Wireless Sensor Network Design Tool to Support Building Energy Management." First ACM Workshop On Embedded Sensing Systems For Energy-Efficiency In Buildings. (2009).

[Fels, et al. 1998] Fels, S., et al. Progress of C-MAP: A context-aware mobile assistant (1998).

[Finley Jr, et al. 1991] Finley Jr, M.R., et al. "Survey of intelligent building concepts." IEEE Communications Magazine **29**(4): 18-23 (1991).

[Flax 1991] Flax, B. M.: "Intelligent buildings." IEEE Communications Magazine **29**(4): 24-27. (1991)

[Hightower and Borriello 2001] Hightower J., Borriello, G.: . "Location systems for ubiquitous computing." Computer **34**(8): 57-66 (2001).

[Klepal and Pechac 2002] Klepal, M., Pechac P.: "Prediction of Wide-Band Parameters of Mobile Propagation Channel." XXVIIth URSI General Assembly the International Union of Radio Science (2002).

[Korpipää and Mäntyjärvi 2003] Korpipää P., Mäntyjärvi, J.:"An ontology for mobile device sensor-based context awareness." Proceedings of CONTEXT **2680**: 451–458. (2003).

[Krause and Hochstatter 2005] Krause M., Hochstatter, I.: "Challenges in modelling and using quality of context (qoc)." Mobility Aware Technologies and Applications: 324-333. (2005).

[Lee, et al. 2007] Lee, K. C., et al.. Implementation of Ontology Based Context-Awareness Framework for Ubiquitous Environment. Int. Conference on Multimedia and Ubiquitous Engineering (2007).

[Lieberman and Selker 2000] Lieberman, H., Selker, T.: "Out of context: Computer systems that adapt to, and learn from, context." IBM Systems Journal **39**(3-4): 617-632. (2000).

[López, et al. 2009] López, T.S., et al. "Taxonomy, technology and applications of smart objects." Information Systems Frontiers: 1-20. (2009).

[Mara, et al. 2009] Mara, P., et al. "Pervasive Knowledge-Based Networking for Maintenance Inspection in Smart Buildings." International Conference on Autonomic Computing Proceedings of the 6th international workshop on Managing ubiquitous communications and services, Barcelona, Spain: 59-66. (2009).

[Masuoka, et al. 2003] Masuoka, R., et al. "Ontology-enabled pervasive computing applications." IEEE Intelligent Systems: 68-72. (2003).

[McGibney, et al. 2008] Mc Gibney, A., et al. Design of underlying network infrastructure of smart buildings. Proceedings of the 4th International Conference on Intelligent Environments (IE 08), (2008).

[McGibney, et al. 2007] Mc Gibney, A., et al. User Demand based WLAN Design and Optimisation. Proc. IEEE Vehicular Technology Conference, Dublin, Ireland. (2007).

[McGlinn, et al. 2009] McGlinn, K., et al. "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.

[McGlinn, et al. 2007] McGlinn, K., et al. Modelling of Context and Context Aware Services for Simulator Based Evaluation, MUCS (2007).

[McGlinn, et al. 2008] McGlinn, K., et al. SimCon: A Tool for Modelling Context Sources for Rapid Evaluation of Pervasive Applications using Virtual Reality. Proceedings of the 5th Annual International Conference on Mobile and Ubiquitous Systems: Computing, Networking, and Services (2005).

[McGuinness and Harmelen 2004] McGuinness, D. L., Van Harmelen, F. "Owl web ontology language overview." W3C recommendation **10**: 2004-03.

[O'Neill, et al. 2007] Neill, E.O., et al. "Rapid user-centred evaluation for context-aware systems." Lecture Notes in Computer Science **4323**: 220. (2007).

[O'Sullivan and Keane 2005] O'Sullivan, B., Keane, M.: Specification of an IFC based intelligent graphical user interface to support building energy simulation. Proceedings of the 9th International Building Performance Simulation Association Conference (2005), Montreal.

[Papamichael, et al. 1999] Papamichael, K., et al. "Product modeling for computer-aided decision-making." Automation in Construction (1999) **8**(3): 339-350.

[Reynolds, et al. 2006] Reynolds, V., et al. Requirements for an ubiquitous computing simulation and emulation environment, New York, NY, USA, ACM Press.

[Russomanno, et al. 2005] Russomanno, D. J., et al. Building a sensor ontology: A practical approach leveraging ISO and OGC models, Citeseer, (2005).

[Salber, et al. 1999] Salber, D., et al. "The context toolkit: Aiding the development of context-enabled applications." Human-Computer Interaction (1999) **16**(2): 97-166

[Sanmugalingam and Coulouris 2002] Sanmugalingam, K., Coulouris, G.: "A generic location event simulator." Lecture Notes in Computer Science (2002): 308-315.

[Schilit, et al. 1994] Schilit, B., et al. Context-aware computing applications. Mobile Computing Systems and Applications, 1994. WMCSA'08. First Workshop on, IEEE Computer Society.

[Schilit and Theimer 1994] Schilit, B. N., Theimer, M. M.: "Disseminating active map information to mobile hosts." IEEE network (1994) **8**(5): 22-32.

[Schmidt 2003] Schmidt., A.: Ubiquitous computing-computing in context. (2003) Thesis. Lancaster University.

[Scholtz and Consolvo 2004] Scholtz, J., Consolvo, S.: "Toward a framework for evaluating ubiquitous computing applications." IEEE Pervasive Computing (2004) **3**(2): 82-88.

[Sheikh, et al. 2007] Sheikh, K., et al. Middleware support for Quality of Context in pervasive context-aware systems (2007).

[Shirehjini and Klar 2005] Shirehjini, A. A. N., Klar, F.: 3DSim: rapid prototyping ambient intelligence. Proc. of the 2005 Joint Conf. on Smart objects and ambient intelligence:

innovative context-aware services: usages and technologies sOc-EUSAI, Grenoble, France, ScientificCommons.

[Snoonian 2003] Snoonian, D.: "Smart buildings." IEEE Spectrum (2003) **40**(8): 18-23.

[Strang and Linnhoff-Popien 2004] Strang T., Linnhoff-Popien, C.: A context modeling survey (2004).

[Sumi, et al. 1998] Sumi, Y., et al. "C-MAP: Building a context-aware mobile assistant for exhibition tours." Lecture Notes In Computer Science; Vol. 1519: 137-154 (1998).

[Sundresh, et al. 2004] Sundresh, S., et al. (2004). SENS: A sensor, environment and network simulator, IEEE Computer Society Washington, DC, USA.

[Trenholme and S. P. Smith 2008] Trenholme, D., Smith, S.P.: "Computer game engines for developing first-person virtual environments." Virtual Reality **12**(3): 181-187 (2008).

[URL] URL "SWEET Ontology." http://sweet.jpl.nasa.gov/ontology/sensor.owl

[Wang, et al. 2004] Wang, X., et al.. "Semantic Space: an infrastructure for smart spaces." IEEE Pervasive Computing **3**(3): 32-39 (2004).

[Wang, et al. 2004] Wang, X.H., et al. Ontology based context modeling and reasoning using OWL. Proceedings of the Second IEEE Annual Conference on Pervasive Computing and Communications Workshops, Citeseer, (2004).

[Want, et al. 1992] Want, R., et al. "The active badge location system." ACM Transactions on Information Systems **10**(1): 91-102. (1992).

[Weis, et al. 2007] Weis, T., et al.. "Rapid prototyping for pervasive applications." IEEE Pervasive Computing **6**(2): 76-84. (2007).

[Wong, et al. 2005] Wong, J. K. W. et al. "Intelligent building research: a review." Automation in Construction **14**(1): 143-159. (2005).

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