INCLUSION OF TEMPORAL DATABASES WITH INDUSTRY FOUNDATION CLASSES

A Basis for Adaptable Intelligent Buildings

Hubert Grzybek*, Stephen Gulliver and Zhuotao Huang

Informatics Research Centre, University of Reading, Ground floor, Building 42, Whiteknights, Reading,RG6 6WB, U.K.

Keywords: Intelligent buildings, Industry Foundation Classes, Temporal database, Dynamic Building Model.

Abstract: Current 'intelligent' buildings (IB) lack the crucial requirement of being adaptable. Without being dynamic and adaptable, buildings can only be called 'responsive' or 'automated', and therefore continue to perform sub-optimally due to their reliance on manual calibration and inability to automatically recognise patterns in building usage. It is proposed that the development of an open, dynamic and temporal building model will alleviate many of these problems. This will facilitate IB software to automatically and continuously re-calibrate itself in order to achieve personalised occupant, environmental and energy consumption goals. The aim of this paper is to examine the main standards and technologies that are currently used in the field of intelligent buildings, and highlight deficiencies in their ability to support adaptable intelligence. In addition we propose a novel solution and prototype, combining IFC and temporal database theory, in order to discuss the implications and application of practical deployment.

1 INTRODUCTION

There is currently no universally accepted definition for 'intelligent buildings' (IB). Various definitions differ in focus and emphasis, however most IB authors share the belief that an IB should use modern technology to improve resource efficiency and provide a comfortable environment for occupants. The concept of 'intelligent' systems, however, stipulates that such systems should be capable of 'learning' and automatically calibrating itself to react to changes in its environment, in order to continuously ensure optimal running (Fritz, 2010). This notion contrasts, however, with current IB software, which focuses entirely on building automation and visualisation of building system states by responding to a given set of conditions with pre-programmed actions. Current IB software is subsequently unable to dynamically respond to changes in its environment and adapt its behaviour to varying usage patterns over time. For the purpose of this paper, adaptability and flexibility relate to a building's ability to automatically recognise and respond to changes in its environment, including the 'learning' of recognised patterns in recorded data. Within a dynamic environment, responsive systems will therefore require continual reconfiguration if they are not to become ineffectual or obsolete.

If dynamic adaptability is essential within the 'intelligent building' definition, we would have to agree that the first IB is yet to be designed. Current IB software is incapable of supporting such intelligent buildings, due to their inability to create and make use of historical data relating to the changing state of the building, its lack of information concerning objects with the space, and a lack of a machine-readable semantic building model to allow information to be placed in context of the building space. Historical data is crucial to supporting 'learning', as it supports data mining and the recognition of patterns in use of resources and space (Cheng, 2006). This lack of historical data and its link with a semantic building model is currently a huge obstacle preventing buildings from responding dynamically to the changing patterns of building usage and dynamic occupant behaviour.

Whilst current intelligent building technology is capable of improving building performance in terms of resource efficiency and the provision of a comfortable environment for occupants, they do not meet the requirements of adaptability and flexibility. Changes in business process, organisational

Grzybek H., Gulliver S. and Huang Z.
INCLUSION OF TEMPORAL DATABASES WITH INDUSTRY FOUNDATION CLASSES - A Basis for Adaptable Intelligent Buildings.
DOI: 10.5220/0003259000240031
In *Proceedings of the Twelfth International Conference on Informatics and Semiotics in Organisations (ICISO 2010)*, page
ISBN: 978-989-8425-26-3

structure and/or building use risks impacting building performance, yet can only be optimised via constant manual reconfiguration. To automatically recognise and respond to change, buildings must be capable of benchmarking critical performance metrics, in relation to a set of building goals, and have the ability to recognise when the building's performance is sub-optimal as a result of changes in its environment. The authors believe that, to understand and adapt to such changes, IB software must maintain an accurate temporal model of the building to provide a complete current and historical representation of: the building's structure; the location and state of its contained spaces; occupants or objects of interest; as well as the changing relationships between these occupants and objects. Such records support data mining and the recognition of patterns in occupant behaviour and space usage, which is critical to supporting building adaptability and learning.

This paper examines current standards and technologies in the field of IB and explains why the current technology is insufficient. Moreover we propose a novel solution, via the introduction of a temporal building model prototype that we hope will stimulate research ideas in this area. The rest of the paper is divided into the following sections: in section 2 we discuss current building standards and technologies (including BMS, BIM and IFC) and relate them to the requirements of adaptability. In section 3 we give a summary of temporal database theory, its applicability to IB, and a description of the proposed building model standard - including expected advantages and potential drawbacks. Finally, in sections 4 and 5, we present our prototype, which showcases the feasibility of the proposed IB model. We conclude by summarising the key points of the paper and expanding the implications of this technology.

2 CURRENT BUILDING STANDARDS AND TECHNOLOGIES

2.1 Building Management Systems (BMS)

BMS products support the integration of the building's current systems, i.e. HVAC, lighting security, etc. and the automation of specific actions, such as the locking all entrances at a specified time. A BMS can be defined as a system for centralising and optimising the monitoring, operating, and managing of a building. Services may include heating, cooling, ventilation, lighting, security, and energy management.

The functional focus of BMS products is on peripheral connectivity and actuator responses to pre-defined sensor inputs or timers. BMS systems support communication between the software system and the building's sensors and actuators through a communication bus, and provide building managers with the capability of setting automated behaviours related to these peripherals, and viewing the state of the building's systems through a friendly user-interface (Knibbe, 1996).

Whilst standards in communications protocols exist, for example LonWorks and BACnet, BMS vendors still focus on the development of rule-based systems, and do not support self-adapting model based systems. The installation of typical BMS products involves manual calibration, typically involving the setting of building-specific rules and the maintenance of the various parts of the system. Once running, the system is incapable of benchmarking its own performance, recognising patterns in the building's usage and suggesting improvements that will meet the occupant's objectives in terms of resource usage and occupant comfort. As a result, current BMS products do not fulfil the adaptable/flexible requirement of truly intelligent buildings (Clements-Croome, 2004). It is possible, however, that existing BMS system data could be used by a temporal building model to track building usage over time, and store this historical data to support data mining and learning.

2.2 Building Information Modelling (BIM)

BIM aims to shift the construction industry away from 2D CAD (Computer Aided Design) building blueprints to full 3D, object-based models backed by semantic information. It also aims to provide a paradigm shift in the way that building project stakeholders interact, distribute responsibility, share information and collaborate in order to reduce costs, increase value for the customer and remove the current legislative culture (Smith et al, 2009).

BIM tools and techniques aim to support construction project stakeholders in collaboratively developing a complete virtual building model, which includes all details relating to project management, e.g. the building's structure, construction schedule, tasks and deadlines, etc. The benefits include increased time and resource efficiency, as all stakeholders can create and share information that would otherwise lead to repeated data collection. BIM tools typically support the development of 3D visualisations of buildings through the open and object-based IFC format.

By moving away from 2D drawings, which include basic CAD objects such as lines and arcs, towards software objects, which represent physical parts of a building, its contents, and how objects relate to each other, the model is able to carry much more semantic information for use by both people and computers. It is this semantic information, not the 3D visualisation, which we believe to be critical to achieving truly adaptable systems. By allowing computers to understand building models, BIM software is capable of achieving advanced features, such as automated rule-based building design and standards checking. Parametric objects, for example, can automatically re-build themselves according to simple embedded rules, such as requiring a window to be wholly within a wall. As the objects are machine readable, spatial conflicts in a building model can be checked automatically, resulting in greatly reduced quantity of errors and change orders (Eastman, 2009).

Object-based semantic models have great potential in terms of adding dynamic properties to BMS, and support the potential of truly intelligent buildings, however current use of BIM is limited to use by Facilities Management (FM) in tasks such as: scheduling routine maintenance, finding objects within a building, and checking maintenance-related costs.

It is clear that an ‘as-built’ BIM model could provide most of the data required for a dynamic building model. However, to support the adaptable requirement this data would first need to be extracted into software that supports the management of this information in a temporal fashion, i.e. to allow for the information to change whilst recording information relating to all past states to support learning. The structural information for such a model is defined in BIM IFC files, which will be discussed in more depth in section 2.3.

2.3 Industry Foundation Classes (IFC)

To reduce data duplication and inconsistency, BIM proponents use a single building model per project, which is then distributed amongst multiple specialised applications which are designed to collaborate. BIM applications typically store this building data using the open IFC object model. The IFC model provides an object-oriented description of the building and related services, enabling interoperability between different vendors of Architectural, Engineering and Construction / Facilities Management software (Spearman, 2007).

The model supports both step formatted text and XML, and therefore it can be used by software tools across all platforms. The format is supported by numerous CAD software vendors, including Graphisoft , AutoDesk and Bentley Systems . These CAD tools allow for the development of 3D building models, which are semantically marked-up, i.e. window objects are added to the model instead of abstract block references made up of individual “polylines”.

An excellent example of the potential of semantic models within the construction industry is described by Wu et al (2004) who developed a system for generating space model graphs based on data from IFC files. This was possible primarily because the IFC format describes the relationships between objects, e.g. doors and spaces – a function that is not possible without semantic modelling. This functionality allows for quicker and easier analysis of building space, but the same technique can be extended to support automated path creation, facilitating user navigation support of complex buildings. Whilst the design of the IFC format clearly provides potential for sophisticated and powerful functionality, it is currently severely hampered by the support of a static building model, i.e. it is not currently possible to include the dimension of time, thus facilitating the storage and usage of historical data. Spearpoint (2003) identified that the IFC model only provides a static view of a building and that this is insufficient for the purpose of running building simulations.

We propose a solution that merges the semantically rich, yet static, building information provided within IFC files with the innately dynamic capabilities of temporal databases and object-oriented software, thus supporting the development of truly intelligent buildings. Such models would allow IB software to mine, learn and dynamically respond to all sensed building state changes as well as user interactions.

3 TEMPORAL DATABASES

Temporal databases differ from typical ‘snapshot’ databases by supporting the management and querying of historical data. Snapshot databases store only the current state of records i.e. each change in an object’s state replacing the previous state through SQL (Structured Query Language) ‘update’ statements. Likewise, SQL delete statements are used in order to remove records that are no longer current, e.g. a customer cancelling their subscription to a particular service which effectively implies that the customer never had a relationship with the company. In contrast, historical databases do not delete records (without administrator intervention) and only update and insert new records in order to maintain an object's historical audit trail. This supports data mining and the recognition of trends as results of object states and aggregate statistics can be compared at different time points.

Temporal databases manage time values in one of two ways: i) valid time and ii) transaction time. Valid time denotes the time period during which a fact is true with respect to the real world. Transaction time is the time period during which a fact is stored in the database. Some databases also support a bi-temporal mode by including both valid time and transaction time support.

In terms of building information, a temporal database could be used, for example, to record temperature changes within defined spaces. By assigning a 'start' and 'end' (valid time) timestamp to each record, it is possible to develop a complete history of temperature fluctuations in different rooms, and these records could be searched in order to find the temperature of a given room at any past point in time, or used to find the average temperature during a period of time. This is one, fairly simplistic example, however, the potential to creating temporal relationships gives rise to much more advanced functionality, including the ability to support mobile sensors. If sensors in a building were able to move between spaces, their readings could be linked to the correct space objects. In this instance it would be possible for the sensor to create records for each temperature reading, but for each reading to be attributed to the correct room. As a result of the potential to create temporary relationships between objects, and track the full history of all object interactions, IB software would be provided with a wealth of knowledge that could be mined to recognise patterns to support resource efficiency and user well-being optimisation.

There are, however, several difficulties to be overcome in the development of a temporal database based IB model, including: the complexity of design, implementation and mining of data, as well as the potential performance problems due to the complex requirements of managing records. Perhaps the greatest obstacle to the adoption of temporal databases, however, is the lack of support from database vendors.

Whilst some vendors, particularly Oracle, have recently made efforts to include temporal features into their database products, no single product supports all the required features which are required for a true temporal database. As a result the management and querying of temporal data is still a cumbersome and complex process due to the lack of supported temporal features in existing RDBMS (Relational Database Management System) products. This state of affairs is due, at least in part, to the exclusion of temporal features from the SQL ISO standard.

In addition to the technical difficulties, there are also social obstacles to be overcome, especially when attempting to use such a database for the purpose of tracking and recording occupant behaviour, as such a system threatens damaging the trust relationship between staff and management. Moran et al. (2010) investigated this problem and provided suggestions as to how organisations should deal with the issues including: how data is collected and for what purpose, access rights to the data and the length of time temporal data is held by the organisation.

4 PROTOTYPE OF THE PROPOSED SOLUTION

In response to the need for a dynamic building model standard, we propose the use of an adaptive building model based on IFC and temporal databases. The IFC model is capable of providing a complete, yet static, representation of a building's structure and content, whilst the temporal database will handle the recording, management and retrieval of all object state changes. Figure 1 shows a diagram of the proposed solution including the relationship and differences between the proposed building model and the currently available BMS technology.

A prototype system was developed using object-oriented technology in order to support a continuous simulation of the building, where each item has a corresponding software object. Any change to these objects, as reported by building sensors, is immediately updated in the run-time software objects and the related database records. The current version of the prototype focuses on the simulation of sensor instances linked to spaces and supports the recording of sensor readings in a temporal database as well as the visualisation of these readings.

For the sake of simplicity, the prototype currently supports a variable number of temperature sensors that can be set to create a new reading every *n* number of seconds for a specified time period. The readings are recorded in a temporal database that can be visualised with a simple web-based interface. The simulation was developed using an open-source platform including linux, Java and MySQL. The visualisation was developed with Apache, PHP and

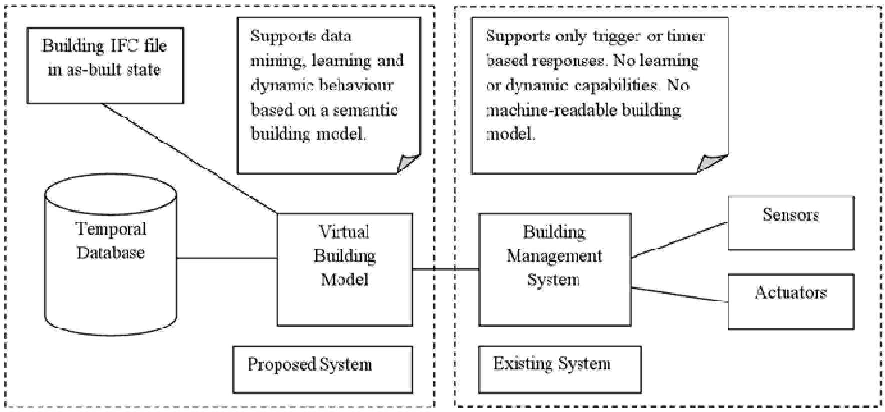

Figure 1: Abstract System Architecture.

the PHP GDGraph library, which are connected to the MySQL database. By using the open-source platform we have promoted the potential for distributing the solution to multiple servers, each of which would be responsible for managing a part of a building, which would likely be prohibitively expensive if commercial technologies were used.

The current version of the database schema is presented in figure 2. The sensor table includes the following fields: id (The unique id key for the sensor), current location (the unique id of the sensor's current space), sensor type (e.g. fire detector/ temperature sensor) and start and end fields, which record the time of the sensor's installation and de-commissioning.

The 'sensor_temp_hist' table records the state changes of temperature sensors. Every time a sensor object reports a different temperature, the database is updated by ending the old state, i.e. updating the 'end time' field and creating a new record which includes the sensor ID, current temperature reading and the new start time. The current state of a sensor always has a 'NULL' value in the 'end time' field. The 'curstate_temp' table contains one tuple per sensor object and records that sensor's current state. This information is also contained by the 'sensor_temp_hist' but has been included in the 'curstate_temp' table in order to prevent length full table scans when searching for current object states. The same pattern has been used with the 'sensor_state_hist' and 'curstate_state' tables, which record the state of the sensors e.g. active/inactive/malfunctioning, as well as the 'location' and 'loc_history' tables, which store the changes in the location of sensors.

The prototype's functionality is provided by the following classes:

1. DbManager: provides database Application Programming Interface (API) functionality
2. EventReading: represents a sensor reading event
3. ReadingManager: An object which manages received EventReading objects. In future, multiple ReadingManager objects could be useful in buffering or distributing the application among a number of application and/or database servers
4. Sensor: Sensor objects represent real-world sensors which may be of a number of types and create EventReading objects which represent their readings.
5. SensorManager: An object which manages a group of sensors.
6. SensorManagerTask: An object which extends 'TimerTask' allowing the simulation to trigger a task every *n* seconds
7. SensorSim: The main class which starts and controls the simulation

The visualisation was developed using GDGraph, an open source PHP graph library. The script reads the temporal database records and draws a temperature graph as a PNG image (see figure 3).

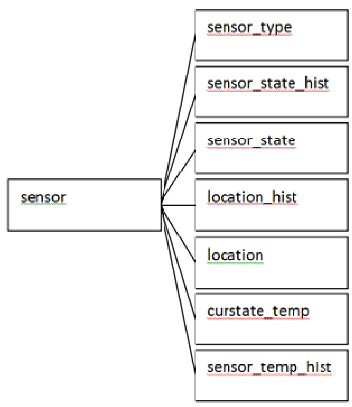

Figure 2: Abstract Database Schema.

5 DYNAMIC BUILDING MODEL: IMPLICATIONS AND APPLICATIONS

The following example demonstrates the applicability of combining temporal databases and the IFC file content to the dynamic building model; assuming that the model were to be continuously updated by accurate sensor data. For this example, we will use an imaginary office building termed Building A, which has a dynamic building model based on a temporal database and semantic information from the building's 'as-built' IFC file.

Building A is occupied during most of every day by Bob who is monitored as he enters, moves around and leaves the building. Bob's movement between spaces (possibly rooms or desks in terms of an open-plan area) as well as the spaces' temperatures are recorded in the temporal database and the relationships between temporal entities, a uniquely identifiable item object or object state, is maintained e.g. during a specified period of time Bob is linked to the record of the space which he occupied during that period, and, temperature readings are linked to each space.

Given sufficient time, the system will build up enough records to support the mining of data in order to recognise Bob's typical pattern of behaviour. For example, the time he normally enters or leaves the building could be calculated by taking the average entry and exit times. Likewise, Bob's main space (his desk or office) could be automatically inferred based on the space in which he spends the highest percentage of his time. The typical temperature of his most-occupied space could also be calculated, and assuming that there are no recorded instances of Bob changing the temperature settings, we can deduce that the recorded temperatures are within the range at which Bob feels comfortable. These simple measurements can be semantically linked to automatically provide the system with accurate information relating to individual occupant behaviour and preferences. This information could then be used to automatically configure spaces to meet occupant comfort settings, in this case: temperature. This principle could, however, be applied to other factors such as: lighting and HVAC. Such knowledge could then be used to plan building ecological strategies, seating plans, etc., to ensure efficient use of building space.

Other recognised patterns can be used to directly affect the building's response to the occupant in order to provide personalised settings to ensure their comfort or to 'intelligently' respond to the user's behaviour in order to save resources (both financial and energy). For example, if the occupant is recognised as frequently leaving and returning to his main space, it would be inefficient for the system to continuously switch his space's lighting on and off. Likewise if the person leaves the building around their expected leaving time, the system can automatically switch off the lighting and electronic appliances in their space in order to reduce energy wastage. Crucially, by continuously re-examining these patterns in the context of recent data, the system can recognise when patterns change and respond accordingly. For example, if Bob moves to a different office, even if uninformed the system will recognise this based on his recent behaviour. Any automated system actions relating to Bob, such as the setting of his heating or lighting preferences, will then take place in the new space.

The recording of users in space and the recognition of patterns in this data could also be very beneficial for facilities managers (FM). The role of FM staff includes space analysis, space usage optimisation, support for the cleaning and maintenance of facilities, which is often reliant on accurate information relating to the number of occupants in the building, the changes in such trends and the availability of this information in a timely manner. Currently, many FM managers struggle to keep up with these duties due to the varying number of occupants as a result of the increasing adoption of flexible working (Lindkvist, 2009). This has lead to unreliable predictions of occupant numbers. The system could support these requirements to some degree, but may struggle with highly dynamic environments as historical data may be insufficient to predict occupant behaviour.

Table 1: Sample Data.

Sensor ID	Temp (C)	Start Time	End Time
1	18	17:09:52	17:09:54
2	16	17:09:52	17:09:54
3	18	17:09:52	17:09:54
4	17	17:09:52	17:09:54
1	16	17:09:55	17:09:57
2	15	17:09:55	17:09:57
3	19	17:09:55	17:09:57
4	21	17:09:55	17:09:57

Table 1 displays a sample of test data that was automatically generated by the prototype. The sensor objects have been set to automatically generate a random new temperature reading every 3 seconds. The start and end time fields delimit the various temperature states. This arrangement can be applied to all other temporal relations including location, state (e.g. active/inactive etc) as well as relationships between objects. The visualisation of recorded data (see figure 3) is important as it supports human operators examining changes that take place in the building and for identifying trends. Table 2: Sample Data

The main purpose of the prototype was to demonstrate the applicability of temporal databases to the built environment and intelligent spaces for the purpose of supporting IB adaptability. We also wanted to explore the issues and problems related to their use in building management and learning.

The temporal database was designed in a manner that supports frequent access to both historical records and the current states of objects, as based on temporal database literature (Gregerson, 1999), with an extra table holding only the current states to prevent lengthy full-table-scan database queries when selecting current states.

The database structure includes the concept of ‘spaces’ which can easily be populated from IFC files. By establishing a temporal relationship between spaces and their contained sensors, it is possible for sensors to be moved between spaces and for their subsequent readings to be linked with spaces at the time the readings were taken. This assumes of course that the location of the sensors is updated accurately and in a timely fashion. This prototype functionality, to our knowledge, is currently unavailable within IB software.

The richness of the building model relies heavily on accurate input from a wide range of sensors and sensor types. This has several implications, including the cost of sensors and related technology, data security and occupant trust. We believe that the potential benefits of increased energy efficiency and provision of personalised space settings will over time outweigh the disadvantages related to these factors.

6 FUTURE WORK

Since the demonstrated prototype is still in its infancy, we plan to extend it in several ways. We need to extend the database schema to include more temporal entities, develop test queries that are time-bound e.g. “In which space was Bob at 11am yesterday?” as well as develop IFC import functionality. We are considering the development of a more abstract API, since this would be useful when linked to building simulation software (e.g. fire or lighting). Such an API would support the use of a dynamic building model for running building simulations, which has been commented as a deficiency when using IFC files for such simulations (Spearpoint, 2007). In addition we also plan to link this work with a multi-agent system, which will support various intelligent agents for performing a multitude of tasks, including data mining.

Other proposed areas of application may also include development and testing of the system to support a distributed data structure. This may involve distributing the model among multiple servers, for parts of a building, and required research concerning communication relating to building state changes. Each server might also be responsible for mining it’s own set of recorded data for patterns.

The currently proposed prototype is currently only partly dynamic, as it does not support changes to the building’s physical structure. This structure is fixed based on the content of the imported IFC file. In the future, we plan to allow for the import of updated IFC files which will allow the dynamic building model to remain up-to-date with changes in the building’s structure.

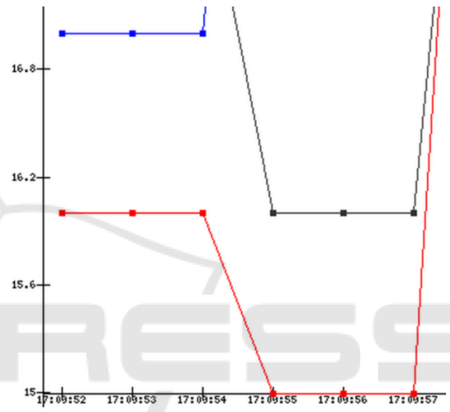

Figure 3: Temperature Graph (X: Time Y: Temperature).

7 CONCLUSIONS

This paper has reviewed a number of state of the art technologies and tools in intelligent buildings in relation to their current inability to support dynamic and historical building models. Such technologies provide a starting point, and we have shown can be modified and extended to support the required functionality. Finally we proposed a novel prototype based on IFC and a temporal database, and demonstrated the feasibility of this approach.

The prototype fulfils a number of roles in our research. These roles include: the development of an ideal temporal database structure for storing building information, a test-bed for the creation, recording, management and mining of temporal data as well as the model’s software objects. The prototype will aid us in discovering potential issues and problems related to the management of temporal building data and will support us in providing recommendations for a dynamic building model standard.

We hope that our development of this model will support future intelligent buildings by allowing them to dynamically monitor and respond to environmental changes and thus meet the requirement of adaptability.

ACKNOWLEDGEMENTS

We wish to thank CDC (Central Data Control) and the EPSRC (Engineering and Physical Sciences Research Council) for supporting and funding our work.

REFERENCES

Cheng T. And Wang J., "Applications of spatio-temporal data mining and knowledge discover (STDMKD) for forest fire prevention", ISPRS Commission VII Mid-term Symposium "Remote Sensing: From Pixels to Processes", Enschede, the Netherlands, 8-11 May 2006, Reed Business Geo. 2006

Clements-Croome, D. (ed.) Intelligent Buildings: Design, Management and Operation. Thomas Telford P. Heron Quay, London, 2004. pp. 39

Eastman C., Aug. 2009, "Building Information Modeling, What is BIM?" (Online) Available: http://bim.arch.gatech.edu/content_view.asp?id=402 (Accessed: 16 May 2010).

Fritz W., "Intelligent Systems and their Societies: Glossary", "27 Jan 2010", (Online) Available: http://www.intelligent-systems.com.ar/intsyst/glossary.htm (Accessed: 16 May 2010).

Gregerson H. And Jensen S.C. "Temporal Entity-Relationship Models – A survey", IEEE Transactions on Knowledge and Data Engineering. Vol 11, No. 3, May/June 1999, pp.484

Knibbe, E.J., "Building Management System" U.S. Patent 5,565,855, Oct. 15, 1996. (Online) Available: http://www.google.co.uk/patents (Accessed: 16 May 2010).

Lindkvist C and Elmualim A, "Pervasive technologies for workspace management", Journal of Facilities Management, Vol. 7, No.2, 2009, pp 98-110.

Moran S. and Nakata K., "Analysing the factors affecting users in intelligent pervasive spaces", Intelligent buildings International 2 (2010), 57-71, doi:10.3763/inbi.2009.0040, 2010 Earthscan ISSN: 1750-8975(print), 1756-6932 (Online) Available: www.earthscan.co.uk/journals/inbi (Accessed: 16 May 2010).

Smith K. Dana and Tardif M., "Building Information Modeling A Strategic Implementation Guide for Architects, Engineers, Constructors, and Real Estate Asset Managers" New Jersey: John Wiley & Sons, 2009 pp. 35.

Snodgrass R., "Temporal Databases Status and Research Directions", Sigmod Record, Vol 19, No. 4, 1990 (Online) Available: http://citeseerx.ist.psu.edu/viewdoc/ (Accessed: 16 May 2010).

Spearpoint, M.J., "Integrating the IFC Building Product Model with Zone Fire Simulation Software", In: Proc. International Conference on Building Fire Safety, QUT, Gardens Point Campus, Brisbane, Australia, 20—21 November 2003, pp. 56—66 (Online) Available: http://ir.canterbury.ac.nz/ (Accessed: 16 May 2010).

Spearpoint M J, Dimyadi J A W (2007) Sharing fire engineering simulation data using the IFC building information model. Christchurch, NZ: MODSIM07, International Congress on Modelling and Simulation, 10-13 December 2007.

Wu S. Et al, "An IFC-based space analysis for building accessibility layout for all users", Construction Innovation 2004; 4: 129–141.