An Easy-to-Use 3D Visualization System for Planning Context-aware Applications in Smart Buildings

Jun-Ming Su

Abstract—**To design the context-aware applications of smart buildings is a time-consuming and labor-intensive work. Much research currently proposed the design approaches based on low-level concerns and the real-world deployments based on high-level programming abstraction are rare. The pressing issue is how to offer the easy-to-use tools with acceptable performance to easily design the application scenarios for non-technical application users. This study focuses on the application-layer simulation to propose a Visualization System of Context-aware Application Scenario Planning (VS-CaSP) for assisting non-technical developers in easily designing the application scenario of smart buildings and in performing the acceptable and predictable simulation and evaluation. VS-CaSp applies rule-based and 3D visualization techniques to offer a 3D authoring environment integrated with real Zigbee sensor devices, where designers can rapidly construct immersive 3D buildings and easily plan the context-aware application scenario via the GUI tool based on proposed three-tier rule hierarchy, to visually simulate and verify the planned scenario via virtual and real sensor devices, and to repeatedly modify the control strategies to enhance the deployment effectiveness. The experimental results show that VS-CaSP is easy to use for the support of quickly design smart building applications, but not professional enough for developments of 3D modeling, animation, and rule expression. Accordingly, it is workable and expected to prove beneficial to non-technical users.**

Index Terms—**smart buildings, context-aware, application scenario planning, 3D visualization, wireless sensor network**

I. Introduction

With the proliferation of Wireless Sensor Network (WSN) technologies, the context-aware smart environment [1-3] is becoming an emerging topic in the recent years. However, deploying the context-aware applications in smart environments is quite time-consuming and labor-intensive, because the processes including plan, deployment, test, and modification are required to be performed in the actual target environments [6,7]. Mottola and Picco [7] pointed out that application developers need the proper software platforms to efficiently design the applications for promoting the WSN technology although hardware advances play an important role. However, much research currently proposed design approaches based on low-level concerns, including data collection, sensor deployment, routing algorithm, and communication simulation, to assist the design process in WSN [8-12], but the real-world deployments based on high-level programming abstraction are rare. Moreover, low-level programming-based methods not only shift the focus of application domain experts away from the application scenario, but also require a technical background rarely found among them. Therefore, to efficiently promote the technology of smart building applications, the pressing issue is how to offer the easy-to-use programming platforms with acceptable and predictable performance and reliability to easily design the application scenarios for non-technical application domain experts [7]. *T*his study aims to develop an easy-to-use 3D Visualization System of Context-aware Application Scenario Planning (VS-CaSP) to assist non-technical developers in easily designing the application scenario of smart buildings and in performing the acceptable and predictable simulation and evaluation. Based on high-level programming abstraction and application-layer simulation, VS-CaSp applies rule-based and 3D techniques to offer a 3D authoring environment integrated with real Zigbee sensor devices, where designers *can* rapidly construct immersive 3D buildings and easily plan the context-aware application scenario via GUI tool based on proposed three-tier rule hierarchy, to visually simulate and verify the planned scenario via virtual and real sensor devices, and to repeatedly modify the control strategies to enhance the deployment effectiveness and reduce the cost of time and work.

II. Related Works

Much research focused on the low-level context, i.e., network-layer or physical-layer simulation of WSN. The high-level context and application-layer simulation of context-aware application scenario based on WSN to support the context-aware smart environment [1] are not taken into account yet. Thus, Mocofan et al. [13] and Jouve et al. [14] developed the systems based on high-level application concerns to assist users in configuring application scenarios of smart buildings. However, the application scenario configuration methods using parameters adjustment of predefined scenarios confined the variety and flexibility of applications. Furthermore, to improve this issue, Bischoff et al. [15] and Guo et al. [16] applied rich rule-based languages to edit application scenarios of the smart homes, but the

Manuscript received June 24, 2012; revised August 8, 2012. This work was supported by National Science Council of Republic of China under the number of NSC 101-2511-S-024-004-MY3.

J. M. Su is with the Department of Information and Learning Technology, National University of Tainan, Tainan, Taiwan, R.O.C (phone: +886-6-2133111 ext. 775; fax: +886-6-307137; e-mail: junming.su@ gmail.com).

ISBN: 978-988-19251-6-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

text-based editing method is still difficult to edit for the non-technical end-users and designers. However, aforementioned approaches either support the 2D simulation only [14-17], or cannot support the 3D model authoring [13]. Therefore, the immersive and flexible degrees may be insufficient for the early-stage deployment of context-aware applications. Afterwards, Kaldeli et al. [18] and O'Neill et al. [19] developed systems employed with professional rule engines and 3D modeling tools to support the applications deployment of smart buildings. Similarly, rule-based scenarios also need to be edited by text-based writing. Besides, professional 3D modeling tools may not be easy enough to be learned and used for the non-technical end-users.

III. 3D VISUALIZATION SYSTEM OF CONTEXT-AWARE APPLICATION SCENARIO PLANNING

A. Problem Definition

Planning a context-aware scenario to support the applications in the smart building is necessary to consider the interaction among sensors, e.g. ZigBee and RFID, and actuator devices, e.g. controls of lights and doors, in the target environment, and further design the control strategy (a.k.a control rule) of devices for achieving the desired scenario of the application. For example, in the context-aware application scenario of Good Sleep Service proposed by Kang et al. [6], the air conditioner should be turned on if the temperature is higher than the preferred setting and the light should be turned off when sleeping. However, a context-aware application scenario usually consists of many sensor-actuator rules to control these included sensor and actuator devices. Hence, the problems concerning the actual deployment of context-aware application scenario to support the smart buildings includes: (1) difficult deployment: hard to control and realize the resultant context-aware application scenario, and (2) difficult verification: hard to debug and modify the planned context-aware application scenario.

B. Architecture of VS-CaSP

A 3D Visualization System of Context-aware Application Scenario Planning, called VS-CaSP, consists of three modules (Fig. 2).

(1) Context Resources Management Module: manages context resources, including 1) 3D objects, models, and scenes: managed by 3D Model Manager, and 2) user-constructed scenarios consisting of control rules and relevant 3D models: packaged by Scenario Packager.

(2) 3D Application Scenario Authoring Module: offers designers to rapidly and easily plan the desired context-aware application scenario, in which the 3D scene can be constructed by the 3D Scene Editor; the virtual operable action object, i.e. animated model, controlled by the actuator, e.g., door and window, can be constructed by the Action Object Editing to support the animation effect; the required sensing functionalities of virtual sensor devices, e.g., ZigBee and RFID, can be defined by the Sensor Object Editing; the control strategy among sensors and actuator devices can be planned by the Context-aware Application Scenario Editor to achieve the desired scenario in the virtual 3D scene.

(3) 3D Simulation & Soft/Hardware Integration Module: offers designers to easily debug and simulate the application scenario by the Context-aware Application Scenario Simulation Viewer for the verification and modification. Moreover, the sensing data of sensor nodes in the 3D scene

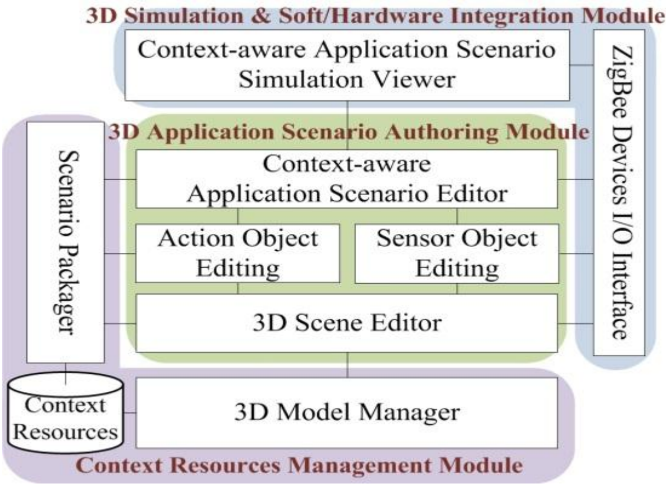

Fig. 1. Architecture of VS-CaSP.

can be inputted through both virtual and real sensors by ZigBee Devices I/O Interface for enhancing actual deployment performance.

C. Definition of Context Resources

Context resources are defined to support the construction and package of an application scenario in smart buildings.

Definition 1: Context Resources, $\mathbf{R_{res}=\{M_{3D}, S_{sen}, AO_{set}\}}$, where:

- $\mathbf{M_{3D}}$ = {P, T}: denotes the 3D models, where:
  - P=(x, y ,z) denotes coordinate information of a 3D model associated with T={(u, v), t_{tex}}, which is Texture Coordinates, $0 \le u, v \le 1$. t_{tex} is physical file of texture in 3D model.
- $\mathbf{S_{sen}}$ = {Z, R}: denotes the sensor devices, where:
  - Z={s_1 , s_2, …, s_n} denotes the set of ZigBee sensor devices, each of which is represented by s_i={att_1, att_2,…, att_k} to perform Context-aware Functionalities (att_i), such as the luminance, occupancy, or temperature sensing of sensor devices.
  - R=($RFID_{tag}$, Reader, Writer) denotes RFID devices including Tag, Reader, and Writer.
- $\mathbf{AO_{set}}$ = {ao_1, ao_2,…, ao_m}: denotes the set of action objects controlled by actuators, where ao={act_1, act_2,…, act_p} has action types (act_i) to perform the specific animation effect in 3D virtual scene. For example, light object (ao) can be turned on/off (act), and virtual door can be opened/closed.

IV. APPLICATION SCENARIO CREATION

A. Definition of Context-aware Application Scenario

To easily design the desired context-aware application scenario for non-technical users, the Scenario Representation based on the proposed Three-Tier Rule Hierarchy (TTRH) in VS-CaSP is required to be defined in advance for the processes of editing, management, package, and simulation.

Definition 2: Scenario Representation, $\mathbf{S=\{TC, RC_{set}\}}$, where:

- **S** denotes the 1st tier of the Scenario Representation,

ISBN: 978-988-19251-6-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

consisting of Rule Classes (RCs) in the 2nd tiers, each of which contains Rules (R) in the 3rd tier.

- **TC** is the Trigger Condition (TC). If TC is satisfied, the associated RCs will be triggered for executing the setting of a scenario.
- **RC_{set}** = $\{(RC_1,CM_1), (RC_2,CM_2),\ldots, (RC_n,CM_n)\}$: denotes the set of associated RCs. Every RC in a Scenario has the associated Control Mode (CM), where:
  - **RC**=$\{R_1, R_2,\ldots, R_m\}$: denotes a RC which can perform the specific action based on included R, where R is defined as rule-based format, i.e., *If <conditions> Then <actions>*. For instance, **If** <Sensor.att> 30℃> **Then** <ao.act> denotes that if the temperature of sensor is higher than 30℃, the action object (ao), e.g. air conditioner, will be turned on (act).
- **CM**=(*Enabling*, *Firing*): denotes the control approach for a RC. *Enabling* mode will trigger the included R in RC if it is satisfied. In *Firing* mode, each R of the RC will be triggered directly, but if TC has the condition value, the rule must be checked in advance according to this condition value. For example, "User ID" is the condition value in TC, the Rule with "User ID" in condition setting needs to be checked.

In the definition 2, a scenario representation consists of a Trigger Condition (TC) and several associated Rule Classes (RCs) in 2nd tier, each of which includes several associated Rules (Rs) in 3rd tier, i.e., sensor-actuator rule described in Section III.A. Therefore, as illustrated in Fig. 2.a, the purpose of scenario representation, as a three-tier rule hierarchical structure, aims to support the easy configuration and control process of a context-aware application scenario. Similar to the form filling interface proposed by Guo et al. [16], the confined rule structure can help non-technical users to easily plan a convergent application scenario. Based on this definition, a Scenario Packaging Scheme in Fig. 2.b is also proposed to package the application scenario into a Scenario Package for the easy management, reuse, and share.

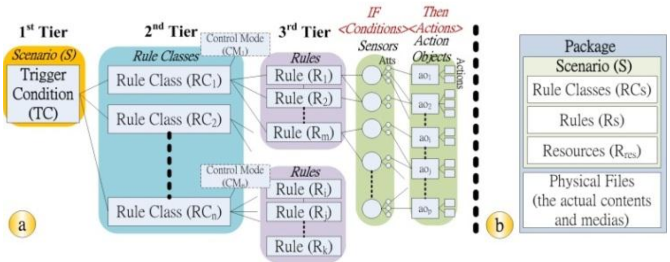

Fig. 2. (a) Scenario representation and (b) Scenario packaging scheme

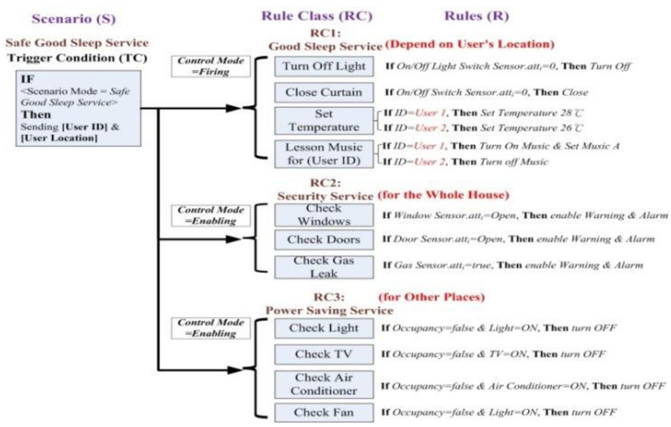

Fig. 3. Example of a context-aware scenario based on three-tier rule hierarchy

Example 1: In Fig. 3, based on the Scenario Representation, a Scenario (S) can includes several RCs, so designers can define various RCs to support various service scenarios in advance, such as RC1: Good Sleep Service, RC2: Security Service, and RC3: Power Saving Service. RC1 can offer the good sleep service according to the location and identification (ID) of a user. For example, the light and curtain will be turned off and closed respectively, and the temperature and music will be set according to the preference of the user 1 if the RC1 is enabled. RC2 collects the status of security sensors to decide whether enabling the alarm or not, and RC3 manages the electric appliances for the Power Saving service. According to the requirement of service scenario, RC1 needs to execute the rules, set the status of sensors, and perform the corresponding actions directly, so its control mode (CM) is defined as *Firing* mode. On the contrary, the RC2 and RC3 need to check statuses of sensors before rules are fired, so their control modes are defined as *Enabling* mode. The service scenario can be easily recombined and reused to support various service scenarios in smart home environments. For example, RC1, RC2, and RC3 can be combined to construct a more complex scenario, such as "Safe Good Sleep Service". Its Trigger Condition (TC) can be defined as **IF** <Scenario Mode = *Safe Good Sleep Service*> **Then** Sending **[User ID]** & **[User Location]**. Therefore, the RC1, RC2, and RC3 services will be controlled and performed according to the location and ID of a user, as shown in Fig. 3.

B. Authoring Process of VS-CaSP

A context-aware application scenario can be easily constructed based on definitions of the Context Resource, Scenario Representation, and Scenario Packaging Scheme. Authoring Process of a context-aware application scenario includes three phases (Fig. 4). 3D Scene Construction Phase edits and manages required context resources, and constructs the desired virtual 3D scene to meet the requirement of deployment. Then, Context-aware Environment Deployment Phase deploys required sensors and action objects in the 3D scene. Finally, Scenario Planning Phase visually plans control strategy based on three-tier rule hierarchy to construct desired application scenario for supporting smart home services.

C. Simulation Process

Fig. 5 according to the Example 1 shows the simulation process, whose left-hand side is the simulation environment and the right-hand side is the simulation process. Two types of sensor devices based on ZigBee Home Automation specification [4] are deployed in this 3D scene. Type 1 is equipped with many sensing functionalities. For instance, sensor 1 can emulate to turn on/off air conditioner, light, and music. Type 2, as the IAS Zone in ZigBee, is able to detect the alarm conditions, such as intrusion and fire. During the simulation process, the VS-CaSP will collect environmental sensing data from deployed sensors and employ the rule-based inference process to determine how to offer the service and response according to the user's behavior during walkthroughing in the 3D scene. For example, while a user goes to the main bedroom and s/he chooses and enables the "Safe Good Sleep Service", VS-CaSP will use the [User ID]

ISBN: 978-988-19251-6-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

and [User Location] to simulate the planning of context-aware application scenario described in Example 1. Simulation effects of air conditioner, lighting, curtain, and music in the main bedroom will be controlled by sensor 1 and triggered according to RC1 (Good Sleep Service). RC2 (Power Saving Service) will enable sensor 2 to turn off the TV and light if nobody is at the living room. VS-CaSP can emulate the application scenario integrated with virtual and real sensor devices. Input data of virtual sensors 3 and 4 are respectively defined as real sensors, i.e. ZigBee Node 2 and 1 in the real environment. Thus, input data of sensor 3 will be collected from the ZigBee Node 2 to infer relevant rules. For instance, in RC3 (Security Service), if nobody is at kitchen detected by ZigBee Node 2, the effect of turning off light in the kitchen will be simulated. The alarm service will be triggered by Sensor 4 as soon as ZigBee Node 1 detects the intrusion, such as alarm running in 3D scene.

V. Implementation

A. Prototypical VS-CaSP System

Prototypical VS-CaSP integrated with real ZigBee hardware based on ZigBee Home Automation Public Application Profile [5] has been developed by Visual C++ and DirectX SDK. Fig. 6 illustrates screenshots and the constructed 3D scene in Fig. 5 based on the authoring process in Section IV.B. VS-CaSP provides non-technical users with three types of authoring methods to rapidly build the **(1) 3D virtual building:** "drags and drops" objects in 2D mode and then views scene in 3D mode (Fig. 6), quickly create the **(2) Animated object:** uses "eight steps" only to create simple animation effects (Fig. 7.a), and easily plan the **(3) Application Scenario:** uses GUI tool to design based on three-tier rule hierarchy (Fig. 7.b) and easily simulate it (Fig. 8.a).

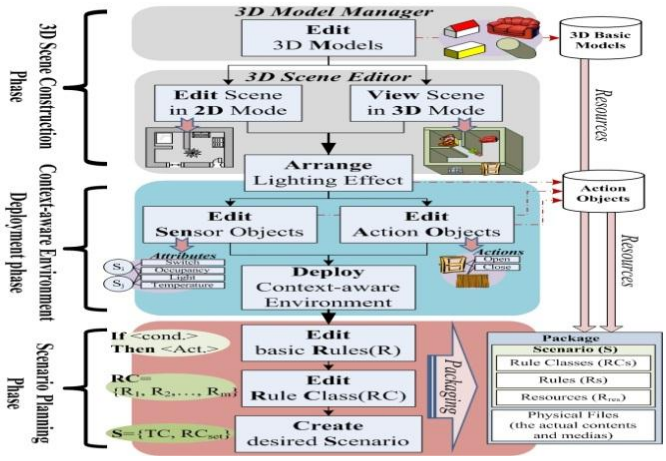

Fig. 4. Authoring process diagram of VS-CaSP

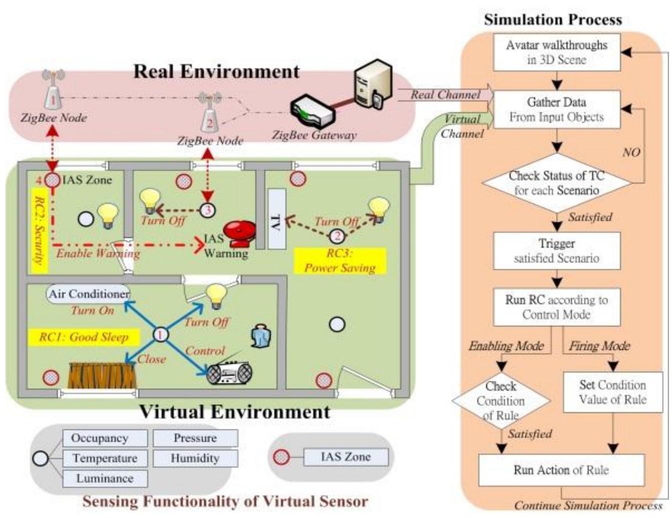

Fig. 5. Simulation process diagram of a context-aware application scenario

B. Experimental Results

Effectiveness of VS-CaSP system

To evaluate effectiveness, the scenario-based evaluation methodology used in [13-15,18,19,21] was adopted. We used the VS-CaSP to plan six smart home applications, i.e. (1) Leave Home, (2) Come back Home, (3) Power Saving, (4) Security, (5) Good Sleep, and (6) Wakeup scenarios. Furthermore, a more complex smart home application reusing above 6 scenarios, which consists of 13 rules classes (RCs) with 28 rules, was constructed to automatically control 4 lights and 16 Action Objects (AOs) according to two users' preferences by collecting the context-aware information from 3 virtual RFID readers, and 6 virtual and 2 real ZigBee sensors in the target 3D environment as shown in Fig. 8.b. Fig. 9 illustrates the simulation results to conclude the supportability of the VS-CaSP. It is able to quickly support the context-aware application scenario planning in a created virtual 3D smart environment.

Analysis of Users' Feedback

We evaluated three aspects of authoring methods in VS-CaSp, i.e., (1) 3D Building, (2) Animated Object, and (3) Application Scenario, in terms of three criteria: (1) Ease-of-Use: the easy-to-use degree of authoring methods, (2) Professionality: the professional degree of authoring tools, and (3) Supportability: the supportable degree of application planning, to analyze the users' feedback. Ten Participants took part in the experiment, where six of them have no experience in 3D authoring tools and eight of them have no knowledge in rule language editing. The experiment had two rounds. In the first round, each participant was briefly taught how to use the VS-CaSp to design the application scenario for one hour. Then, everyone was asked to design an application scenario and fill in the questionnaire (with score from 1 to 5 (highest)) after the design activity. In the second round, we briefly taught them how to use existing tools including Sketchup, and Hammer Map Editor used in [18,19] for 3D scene and animated object editing, and Jess rule for rule editing used in [16]. They expressed their feedbacks by the same questionnaire to compare with existing tools but focusing on the requirements of rapidly planning an application scenario. Average statistical results of questionnaires in Fig. 10 depict that ease-of-use and supportability degrees are satisfied, but professonality degrees are neutral in the first round. After the second round, degrees of ease-of-use in terms of three aspects of authoring methods increase (Fig. 10.b), i.e. 0.7, 0.6, and 1.2, but professonality degrees decrease, i.e. -1.1, -1.5, and -1, and supportability degrees slightly decrease, i.e. -0.6, -0.2, and -0.1 (Fig. 10.c). This shows that VS-CaSp is easy to use and supportable for the requirements of rapidly planning an application scenario but it is not professional enough for the authoring of 3D modeling and animation.

VI. Limitation And Discussion

A. Limitation of VS-CaSP

(1) **Capability of the Rule Expression:** the current definition of the rule expression is still insufficient to efficiently support the more complex and diverse requirements of application scenarios and it is required to be extended.

(2) **Capability of the Application Scenario Planning:** the degree of ease-of-use concerning the authoring method of application scenarios in VS-CaSp is 3.2 only (Fig. 10.a). It depicts that current planning approach is still not easy to use enough and needs to be improved further.

(3) **Capability of the Modeling Tools of 3D Scenes and Animated Objects:** to suport more variety of 3D modeling and animations, its functionalities need to be extended and enhanced according to users' feedbacks.

(4) **Capability of the Virtual-Real Integration:** integrating with ZigBee and RFID devices only will become insufficient when the planning of more complex application scenarios, e.g. the integration of real actuators.

Fig. 6. VS-CaSP system and smart buildings created by 2D/3D methods

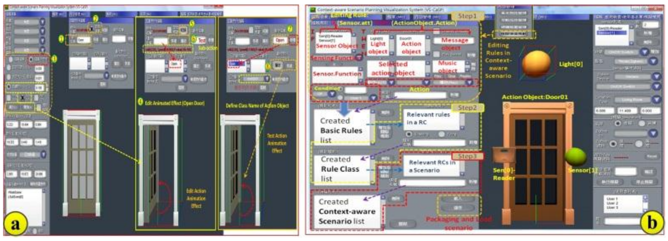
Fig. 7. (a) Authoring action objects with simple animation effects within eight steps, (b) GUI tool of application scenario based on three-tier rule hierarchy within three steps

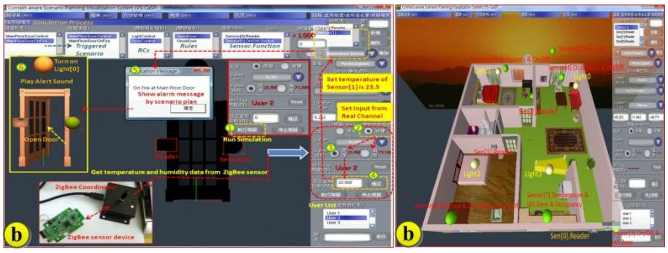
Fig. 8. (a) Simulation process integrated with virtual and real sensors based on application scenario in Fig. 10.b, (b) Deployment of virtual 3D smart building

Fig. 9. Simulation results according to user preferences

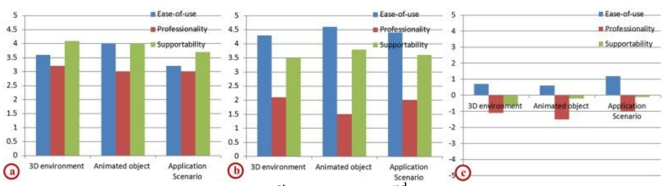

Fig. 10. Average scores of (a) 1st round, (b) 2nd round, (c) difference between 1st and 2nd rounds

B. Discussion of VS-CaSP

According to the analysis of users' feedback in Fig. 10, most of participants agreed that the VS-CaSp can assist them in rapidly and easily creating the acceptable 3D environments and animated objects with simple actions because they understood that the qualities of models and animation are not the main purposes of this system. That is why the average score of Professionality measure declines a lot while the average score of Ease-of-Use measure increases contrarily and of Supportability measure decrease a little after they learned the more professional 3D modeling tools (Fig. 10.c). Furthermore, regarding the application scenario editing, the average score of Easy-of-Use increases highly and of Supportability declines slightly only. The reason is that GUI tool based on three-tier rule hierarchy can efficiently confine users to easily plan a convergent application scenario despite the high decline of Professionality measure compared with the text-based rule editing in JESS rule language. However, similarly to the feedbacks in Guo et al. [16], participants suggested that a more intuitive authoring tool with graphical elements is still required for enhancing the degree of Easy-of-Use. Moreover, according to the survey of existing research described in the related work section, the methods employed as the criteria in this paper can be defined. These definitions can be based on that context-aware application scenarios can be designed by what (1) Scenario Planning Approach, deployed by what (2) Visualization and (3) Modeling Approaches, and evaluated by what (4) Simulation Approach. Table I shows the comparison of our approach with existing articles in terms of the aforementioned criteria. According to the comparison in Table I, each approach has respective advantages and disadvantages. To compared with the text-based rule planning approach used in [18,19], the proposed GUI-based approach based on three-tier rule hierarchy in VS-CaSp is more easy to help non-technical users to plan convergent application scenarios. This method is similarly to the web-based form filling interface proposed by [16], but the form filling format needs to be predefined by technical developers using text-based rules editing. However, the shortcoming is that the various and complex application scenarios can be constructed by professional rule languages adopted in [16,18,19], but it will be difficult by means of our method. However, in terms of the main goal of this study, the capability of application scenario planning should be acceptable according to the experimental results. Moreover,

ISBN: 978-988-19251-6-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

the professal 3D modeling and rendering tools used in [18,19,20] are able to create the static and animated models with high qualities. However, for the non-technical users, the advagtages might becomce the leanring barriers to decrease the degree of Easy-of-Use and increase the learning burden. On the contrary, our self-developed tool allow users to rapidly and easily create the acceptable models with simple actions althoguh it is not very professional. Furthermore, similarly to [16,18], the simulation of virtual-real integration approach has also been developed in the VS-CaSp. However, Guo et al. [16] did not offer the 3D simulation, and Barton et al. [20] and O'Neill et al. [19] did not offer the virtual-real integration but they supported the Parallel Testing.

The differences between our proposed approach and the existing studies are: (1) we define the three-tier rule hierarchy to represent the convergent context-aware application scenario in smart buildings and further develop its GUI-based interface to enhance the degree of Easy-of-Use; (2) we develop the easy-to-use 3D modeling tool to allow users to rapidly construct the target virtual environments and animated models; (3) we propose the Simulation Process based on three-tier rule hierarchy and virtual-real integration simulation to evaluate and verify the deployed application scenarios.

TABLE I
COMPARISON WITH EXISTING APPROACHES. (O: YES, ×:NO, Δ: PARTIAL)

Methods		Barton et al. [20]	Guo et al. [16]	Kaldeli et al. [18]	O'Neill et al. [19]	Our Approach
Scenario Planning Approach	Rule-based	×	O	O	O	O
	Scenario Rule Editing	×	Text-based	Text-based	Text-based	GUI-based
Visualization Approach	3D Rendering Engine	Quake III Arena Graphics Engine	2D	Google SketchUp	Half-Life 2	Self-Developed
	Web-based	Δ	O	×	×	×
Model Authoring Approach	Static Scene/model	GtkRadiant	×	Google SketchUp	Hammer Map Editor (SDK)	Existing/Self-Developed Tools
	Animated Model	Milkshape 3D	×	×	Hammer Map Editor (SDK)	Self-Developed Tool
Simulation Approach	Virtual-Real Integration	×	O	O (ZigBee)	×	O (ZigBee)
	Parallel Testing	O	×	×	O	×
	Test Reporting	×	×	×	O (Manual)	×

VII. CONCLUSION

This study aims to propose an easy-to-use 3D visualized planning approach to assist non-technical users in rapidly design, verify, and modify the desired context-aware application scenario in created virtual smart buildings before actual deployment processes. To evaluate the effectiveness, a complex case, consisting of six scenarios, thirteen rule classes, and twenty-eight rules, to support the smart home services was created based on the scenario-based evaluation methodology. To analyze the non-technical users' satisfaction degrees, authoring methods of 3D environment, animated object, and application scenario were evaluated in terms of three criteria, i.e. Ease-of-Use, Professionality, and Supportability. According to experimental results, high Ease-of-Use and satisfied Supportability degrees show that the VS-CaSP is easy to use for the support of quickly design smart building applications, but not professional enough for developments of 3D modeling, animation, and rule expression due to the low degree of Professionality. Accordingly, it is workable and is expected to prove beneficial to non-technical users. In the near future, the functionalities of the VS-CaSP, including the construction of 3D modeling, application scenarios, and virtual-real integration, will be improved and the Web-based version will be developed. Furthermore, the control system or chip using the application scenario outputted by VS-CaSP to directly control the context-aware service in the real buildings will be developed to achieve the concept of "virtual is the real".

REFERENCES

[1] M. Chan, D. Esteve, C. Escriba, and E. Campo, A review of smart homes—present state and future challenges, Computer Methods and Programs in Biomedicine 91 (2008) 55–81.

[2] C. Mannweiler, J. Simoes, B. Moltchanov, Context-aware smart environment enabling new business models and services, Proceedings of 2010 ITU-T Kaleidoscope Academic Conference, Pune, India, 13-15 December, 2010.

[3] M. Friedewald, O. Raabe, Ubiquitous computing: an overview of technology impacts, Telematics and Informatics 28 (2011) 55–65.

[4] ZigBee Alliance, ZigBee Specification, January, 17, 2008. Retrieved 2012-05-07 form http://www.zigbee.org/

[5] ZigBee Profile, ZigBee Home Automation Public Application Profile, ZigBee Alliance, February, 8, 2010.

[6] L. Shu, M. Hauswirth, H.C. Chao, M. Chen, Y. Zhang, NetTopo: A framework of simulation and visualization for wireless sensor networks, Ad Hoc Networks, 9 (2011) 799-820.

[7] L. Mottola, G.P. Picco, Programming wireless sensor networks: fundamental concepts and state of the art, ACM Computing Surveys, 43 (3) Article 19 (2011) 1-51.

[8] A. Sobeih, W.P. Chen, J.C. Hou, L.C. Kung, N. Li, H. Lim, H.Y. Tyan, H. Zhang, J-Sim: A simulation environment for wireless sensor networks. Proceedings of the 38th Annual Simulation Symposium (ANSS'05), 2005.

[9] S. Kockara, S. Dagtas, Towards 3-dimensional and optimized design of wireless networks for smart environment. Journal of Computing Sciences in Colleges, 21 (5) May (2006) 145-153.

[10] G. Fan, R. Wang, H. Huang, L. Sun, C. Sha, Coverage-guaranteed sensor node deployment strategies for wireless sensor networks, Journal of Sensors, 10 (2010) 2064-2087.

[11] NS2, Network Simulator - Version 2. Retrieved 2012-05-07 form http://nsnam.isi.edu/nsnam/index.php/User_Information.

[12] J-Sim. Retrieved 2012-05-07 form http://sites.google.com/site/jsimofficial/.

[13] M. Mocofan, R. Vasiu, M. Bucos, M. Ionita, I. Ermalai, Multimedia interfaces in the control process of smart buildings using 3D scenes, flash animations and relational databases, Proceedings of the 11th International Conference on Optimization of Electrical and Electronic Equipment (OPTIM 2008), 22-24 May 2008, pp. 213-216.

[14] W. Jouve, J. Bruneau, C. Consel, DiaSim: a parameterized simulator for pervasive computing applications, Proceedings of the IEEE International Conference on Pervasive Computing and Communications (PERCOM), 2009, p.1-3.

[15] U. Bischoff, V. Sundramoorthy, G. Kortuem, Programming the smart home, Proceedings of the 3rd IET International Conference on Intelligent Environments, Ulm, Germany, 24-25, September, 2007.

[16] B. Guo, D. Zhang, M. Imai, Enabling user-oriented management for ubiquitous computing: the meta-design approach, Computer Networks, 54 (2010) 2840-2855.

[17] T.Y. Sohn, A.K. Dey, iCAP: an informal tool for interactive prototyping of context-aware applications, Proceedings of the International conference of Computer-Human Interaction (CHI 2003), April 05-10, 2003, Ft. Lauderdale, Florida, USA.

[18] E. Kaldeli, E.U. Warriach, J. Bresser, A. Lazovik, M. Aiello, Interoperation, composition and simulation of services at home," Proceedings of the Conference on Service- Oriented Computing, ICSOC (2010) LNCS 6470, December 2010, pp. 167-181.

[19] E. O'Neill, O. Conlan, D. Lewis, Modeling and simulation to assist context aware system design, Simulation: Transactions of the Society of Modeling and Simulation International, 87 (1-2) (2010) 149-170.

[20] J.J. Barton, V. Vijayaraghavan, UBIWISE, a simulator for ubiquitous computing systems design. Hewlett Packard, Laboratories, Palo Alto. HPL-2003–93, 2003.

[21] S.Y. Chen, Y.F. Chang, The computer-aided design software for smart home device based on cloud computing service," Proceedings of the IEEE Second WRI World Congress on Software Engineering, 2010, pp. 273-278.

ISBN: 978-988-19251-6-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)