Artefact generation in second life with case-based reasoning

Software Quality Journal

ISSN 0963-9314
Volume 19
Number 2

Software Qual J (2011)
19:431-446
DOI 10.1007/
s11219-010-9125-4

Springer

Software Qual J (2011) 19:431–446
DOI 10.1007/s11219-010-9125-4

SQM 2009

Artefact generation in second life with case-based reasoning

Ahmad Shubati · Christian W. Dawson · Ray Dawson

Published online: 23 February 2011

Abstract Launched in 2003, Second Life is a computer-based pseudo-environment accessed via the Internet. Although a number of individuals and companies have developed a presence (lands) in Second Life, there is no appropriate methodology in place for undertaking such developments. While some existing methods have been adapted by users to their individual needs, this paper explores the development of a method for developing lands specifically within Second Life. This method is based on case-based reasoning (CBR) as this method has a number of similarities with Second Life itself. A system was designed based on CBR with some modifications to work in accordance with Second Life. In this paper, the system and its modifications are discussed and its application to the development of space within Second Life is evaluated. From tracking its progress against previous specifications and future activity, an updated version of the CBR web tool component covering the latest changes and improvements in the tool is introduced here.

Keywords Second life · Development methodology · Case-based reasoning

1 Introduction

The development of the Internet has allowed individuals from around the globe to meet synchronously, breaking down the distance barrier and interacting together. One of the communicating mediums by which the interaction process between users has been enhanced is by the use of virtual reality (VR) software that allows all people using this software to act together more effectively and more realistically. The idea of VR is that it creates a world where all users become immersed and are represented by avatars that deal with one another in a realistic fashion.

There are over 50 different virtual worlds currently available, of which the most notable is Second Life (Azzara 2007)—see Fig. 1.

A. Shubati (✉) · C. W. Dawson · R. Dawson
Department of Computer Science, Loughborough University, Loughborough, UK
e-mail: S.Ahmad@lboro.ac.uk

Fig. 1 Second Life

Currently, Second Life supports over 13 million users and possesses tens of millions of square metres of virtual lands that can be purchased and used for either leisure or businesses (Educause 2008). From a preliminary study of virtual worlds, and Second Life in particular, it was apparent that there were no common or established software methodologies used in the development of spaces within such worlds. This paper describes how this gap is filled through the development and application of a tailored software methodology.

The remainder of this paper is structured as follows. Sect. 2 describes the Second Life development environment; Sect. 3 provides a brief introduction to case-based reasoning; Sect. 4 discusses the concepts behind the proposed CBR web tool; Sect. 5 discusses the tool itself; Sect. 6 describes the second version of the CBR web tool and the new feedback platforms; Sect. 7 discusses the initial results of the CBR web tool; and Sect. 8 presents the final results and conclusions.

2 Existing Second Life Environment Development Methods

A survey was conducted inside Second Life to establish whether any of the major companies involved applied recognised software methodologies in developing their own lands.

The research was semi-structured interview-based and involved more than thirty companies including IBM, Mercedes Benz, Orange, Coca Cola and Sky. Each was contacted by locating their lands on Second Life and booking appointments with their representatives. All conferences took place in Second Life according to the company's schedule. Questions included asking whether the developers used standard modelling and design methods used in software development, such as whether the developers used UML. After obtaining all of the relevant information, it was established that only IBM had used UML, but then only simple use cases.

Following this preliminary study, more interviews were carried out using a random selection of more than one hundred Second Life users that came from different backgrounds

and ages. All interviews took place in Second Life by visiting the most popular lands there. The feedback from individuals implied that they preferred to have existing ready-made objects rather than having to create their own. The new users generally used past experience of previous users in order to find the objects that match their requirements.

Overall, the research clearly showed that most of the users within the virtual world were not familiar with how to use the built-in Second Life programming language that is available to construct new objects there. Furthermore, a supplementary search of the Internet and Second Life was carried out to find whether there were any tools designed specifically for Second Life. The only result found was a trade website offering objects for sale to Second Life users. This clear gap in the market indicates a general lack of understanding of how Second Life developers work and the tools they would need to support their development of Second Life lands. This finding led to research to develop a tool specifically for second life development as described in this paper.

3 Case-based reasoning

Knowledge can be defined as "understanding acquired through experience" It is this definition of knowledge that is the basis for expert systems. Case-based reasoning (CBR) is an approach that allows a user to solve a new problem using or adapting a solution of a similar well-known problem (Amodt and Plaza 1994).

CBR is a promising form of artificial intelligence that is especially helpful when a problem domain lacks a strong set of heuristics but possesses a set of successfully and unsuccessfully solved cases. It is a computerised method that attempts to study solutions that were used to solve problems in the past (Hobson 2007).

CBR technology has produced a number of successfully deployed systems, the earliest being Lockheed's CLAVIER, a system for laying out composite parts to be baked in an industrial convection oven. CBR has been used extensively in help desk applications such as the Compaq SMART system (Cheetham and Goebel 2007).

At the highest level of generality, a general CBR cycle may be described by the following four processes (Amodt and Plaza 1994)—shown in Fig. 2:

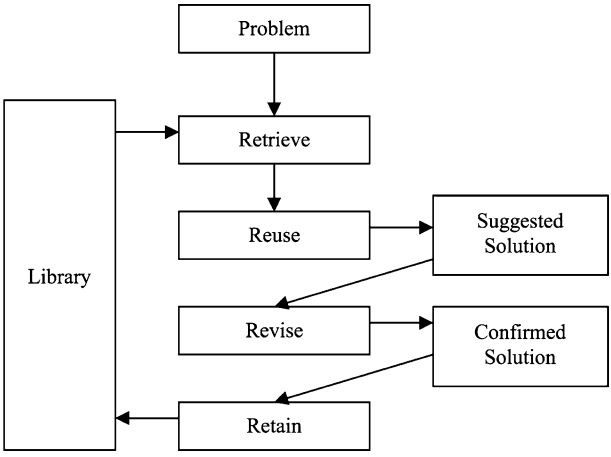

Fig. 2 The CBR Cycle

1. RETRIEVE the most similar case or cases
2. REUSE the information and knowledge in that case to solve the problem
3. REVISE the proposed solution
4. RETAIN the parts of this experience likely to be useful for future problem solving

One of the main attributes of CBR is that it continuously updates its database and creates knowledge depending on feedback from its users. This is similar to the way Second Life users work. Second Life has its knowledge base created by its own users and this is constantly updated. This implies that the CBR system will be compatible with SL for the purpose of providing a tool for SL users.

In the CBR knowledge-building process, dynamic weights are assigned to descriptors to assist in the locating and matching with other cases, where the users assign the weights according to their own requirements. This is again similar to Second Life, where the whole virtual world was built around the creations of its own users and their requirements.

With the vast availability of objects stored in SL, users can find and copy the most appropriate objects they wish to build in SL. This enables the user to find the best possible match to his/her requirements without having to build the object from scratch, and then, if necessary, they can adapt the object to exactly match their requirement. Similarly, the main process in CBR is that it adapts old solutions to meet new demands, using old cases to satisfy the requirements of new situations.

Second Life contains many users from different backgrounds and ages and therefore includes some users within the virtual world who are not familiar with how to use programming languages. In order to build in SL, the preliminary investigations found that it is easier for the users to use existing objects. CBR uses a database where the previous cases are all available and the new case entered is directly compared to these previous cases, assisting the user to find the cases that best match his/her requirement. This would be ideal for Second Life users as, although the store of SL artefacts is vast, locating the best possible match to the user's requirements is difficult due to the unstructured nature of the virtual world.

Having determined the similarities between SL and CBR, it was concluded that CBR is an appropriate software methodology to be used with Second Life.

4 The concept of the CBR web tool

Web-based systems have become popular around the world due to the worldwide expansion of networks as well as the ubiquity of the clients using them. The ability to update and maintain web-based systems without distributing and installing software on potentially thousands of client computers is a key reason for their popularity. In the previous section, case-based reasoning was identified as potentially the best system to be used with virtual worlds to provide a tool for SL developers to use to locate the cases they need in the SL environment. In this section, the steps performed by a CBR tool for SL users are explained. The system was further modified to serve SL practices better, and this will be explained later in this paper.

4.1 Data management

The database contains relational models in which the information of a new problem is stored. Based on the relational data model, useful information can be retrieved. This information is necessary to retrieve the desired objects present within the database.

Fig. 3 The available categories within SL (snapshot of the first step of the system)

The SL environment was examined to identify all categories available. These categories are shown in Fig. 3. Within these categories are subcategories that possess within them all available objects that can be accessed or used within the whole SL environment.

The system database is divided into three tables: a huge number of various SL objects were stored in the first one, including their attributes of name, description category, subcategory and the price of each object with the weight of each attribute. The second table contains the cases stored in the system by the users. It is a dynamic table that updates its data automatically from the users' input. The third table stores temporary data used in the calculation of the similarity values to derive the final result. Every time a new user builds a new case, the temporary values are cleared as new calculations are made. The data management concept makes the whole process of obtaining the results faster and more accurate.

5 CBR web tool phases

5.1 The initial phase

The data retrieval phase is the initial phase that is performed after defining the attribute values of the objects required. Similar previous objects are retrieved by searching for objects that match the desired object's attributes. A set of steps takes place until the best matching case is defined. The steps are as follows:

1. The set of problem descriptors are defined.
2. These descriptors are then compared with those of objects within the database and the most appropriate existing objects are selected.
3. The final step is for the best case to be chosen from the closest matching objects already defined in the database.

5.2 Matching function

The matching function is performed to assess the most similar cases to the one being studied. This is shown in Eq. 1, in which n is the number of descriptors, w is the

summation of the weights, x is the weight of the new case, and y is the weight of the existing case. The cases that obtain higher scores are more similar to the new case and will be retrieved before the cases with lower scores. There are several methods of comparison used. The k-nearest-neighbour technique (KNN) is used here. In this method, each attribute is allocated the same weight. For the SL development tool, the KNN method is modified so each descriptor is given a weight and the system acts dynamically to change the weights used.

$$\text{Similarity} = \frac{\sum_{i-1}^{n} W_i * Z_i}{\sum_{i-1}^{n} W_i} \quad \text{if } x_i < y_i \quad \text{then} \quad Z_i = \frac{x_i}{y_i} \quad else\, Z_i = \frac{y_i}{x_i} \tag{1}$$

The retrieval of the previous cases was performed using a similarity equation whereby the results of the new problem are obtained. This equation works by the summation of the weights of each of the descriptors of the new case multiplied by the ratio between the new case and the stored one. This summation is then divided by the total for each of the attributes of the new case. The method used in obtaining the ratio is as follows: If the value of the new case is bigger, it is placed in the numerator and is divided by the value of the stored case in the denominator. If the value of the stored case is bigger, then the opposite occurs. The value that results from this equation is then multiplied by 100, and the result is the percentage of similarity between both cases (see Fig. 4).

The weights were chosen according to six descriptors which are:

- Categories: There are seventeen categories covering the SL environment.
- Subcategories: There are various subcategories for each category and the values depend on the main category.
- Modifiable: This simply identifies whether the SL object can be modified.
- Copiable: This means that the users have the ability to create copies of the same object.
- Transferable: This attribute means that one user is able to send this object to another user.
- Price: This has a range of five values related to the price of the object.

For these descriptors, the first two, category and subcategory, are not allocated a weight but the other four are allocated dynamically by allocating a priority option for each one, being either High, Normal or Low. The choice of priority chosen changes the weight of the

Fig. 4 Case Similarity (snapshot of the similarity window)

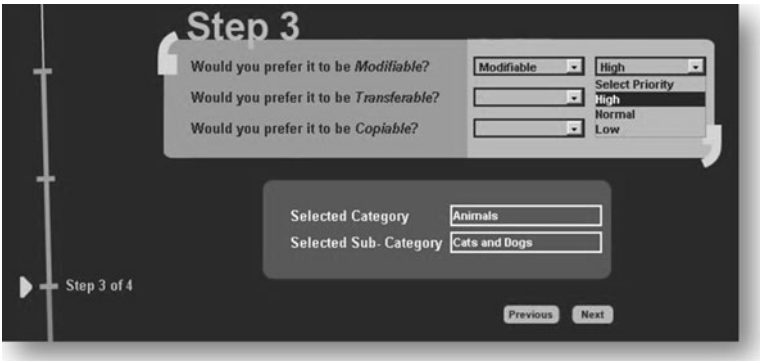

Fig. 5 The priorities are shown next to the descriptors to be chosen (snapshot of the 3rd step of the system)

descriptor. This section will be further explained in the next phase (Data adoption/reuse phase).

5.3 Adaptation phase

The way adaptation is performed is in the form of the priorities allocated to the four descriptors in the new case. These are given weights according to the priority chosen for each descriptor. The user chooses one of three priorities for each descriptor. These rank from High to Normal to Low as shown in Fig. 5. In the case of Normal, the weight assigned to the descriptor remains unaltered and the descriptor weight is unchanged. In the case of High or Low priorities, a constant value is added to or subtracted from the original weight in order to allow the priority to be taken into account.

Having conducted several tests, the best constant increment or decrement to change the weight of the descriptor was found. This value was chosen by trial and error until the best possible value that would reasonably change the weight was reached such that the value made the retrieval of matching objects closer to that desired.

The option of searching with alternative search criteria was introduced. This enables the user to change any weight assigned to a descriptor. This will change the similarities of the retrieved cases by changing the results of the similarity equation so that the ranks of the ten cases received will change to better fit the user's needs.

6 Second version of the CBR web tool

6.1 The new version

Collecting and analysing the feedback for the first version defined the adjustments and improvements needed. By performing these adjustments and testing them, the system became more and more useful for users wishing to access items and objects within SL.

In addition to this, new functionalities have been added in the improved version to further refine and develop the tool to make it highly effective for SL development. Other changes include creating a dynamic database, applying new feedback platforms, implementing the adjusting phase and changing the system design.

6.2 Reasons for changing

The second version of the CBR web tool has been implemented for the reason below:

1. Collecting and analysing the feedback for the first version defined the need of improvements. This feedback came from SL users, colleagues, friends and conferences attendees.
2. The need of new functionalities to increase the validity and reliability of the CBR web tool.
3. Considering the research side of this project, improvements have been implemented in this version to enhance the research value of the project.
4. The need of new interface and design to make the CBR web tool more presentable and understandable for the users.

6.3 New functionalities

The new functionalities added in the improved version of the system are given in Table 1.

6.4 The new continuous feedback approaches

Feedback obtained from users substantially enriches the design process, thus leading to innovative ideas and unexpected results (Rosen and Salomon 2007). This gives the developers the necessary information for improvements towards high-quality learning materials matching the needs in different learning situations (Meisalo 2005). Continuous feedback enables improvements to be made to the system such that it can be adjusted according to the users demand. The mechanisms for continuous feedback provided by the system were based on four approaches—dynamic weights; advertising; feedback services; and case studies, described in the next sections.

Table 1 Additional functionality

Name	Comment
Adjusting phase	The adjust phase has been implemented and dynamically connected to the matching function by allocating a new weight for an attribute in the similarity equation
Dynamic database	The most important element of the CBR method is to be able to find solutions similar to the case being solved in the existing cases database (Kowalski et al. 2005). A new section has been added that permits users to add objects in the database to enlarge the availability of objects and continuously update the database, so the system will become more and more useful for users
Feedback platforms	New feedback platforms have been provided, and these are further discussed in the next section of this paper. This will facilitate the use of the system and obtain results which are more closely related to the users' needs
System design	The design of a system is important for it to be complete and usable. In a process of continual improvement, a new design has been implemented to make it more presentable for users

6.5 The use of dynamic weights

The dynamic weights can change the relevance of the search parameters, tuning the search results to become more similar to the user's needs (de Hugo and Hoeschl 2003).

The k-nearest-neighbour technique is used here. In this method, each attribute is normally allocated the same weight. For the SL development tool, the KNN method was modified so each descriptor is given a weight and the system acts dynamically to change the weights used.

The equation used was adjusted with a method of dynamic weighting to be more accurate in the results obtained after the keywords were entered.

The way dynamic weights are used was based on three phases:

1. The first phase is performed in the form of the priorities allocated to the four descriptors in the new case. These are given weights according to the priority chosen for each descriptor. The user chooses one of three priorities for each descriptor. These rank from High to Normal to Low. In the case of Normal, the weight assigned to the descriptor remains unaltered and the descriptor weight is unchanged. In the case of High or Low priorities, a constant value is added to or subtracted from the original weight in order to allow the priority to be taken into account. Having conducted several tests, the best constant increment or decrement to change the weight of the descriptor was found. This value was chosen by trial and error until the best possible value that would reasonably change the weight was reached such that the value made the retrieval of matching objects closer to that desired.

2. In the second phase, the option of searching with alternative search criteria was introduced. This option gives the user the chance to adjust the results by changing the value of the ratio between the cases. This will change the similarities of the retrieved cases by changing the results of the similarity equation so that the ranks of the ten cases received will change to better fit the user's needs.

3. The third phase is done by asking the user direct questions when he or she uses a specific case, so the case solutions retrieved are evaluated. As a result the database can be adjusted to introduce a new case or to adjust the attributes of existing cases to give a better match on future use. The detail of the method of adjustment is given in Sect. 7.2.

6.5.1 Advertising inside SL

Advertising is the structured and composed non-personal communication of information, usually paid for and usually persuasive in nature, about products (goods, services and ideas) by identified sponsors through various media (Arens 2007). Two types of advertising connected with Second Life must be separated. There is advertising with Second Life and advertising in Second Life. The difference is that advertising with Second Life relates to using Second Life for marketing. Advertising in Second Life relates to all measures and activities within the Second Life world to promote and support an offer, a product, service or corporation.

In advertising, feedback can take many forms: redeemed coupons, phone inquiries, visits to a store, requests for more information, increased sales, responses to a survey or e-mail enquiries. Dramatically low responses to an advertisement indicate a poor communication process (Arens 2007).

The advertising in SL for the system was based on three approaches: person-to-person advertising, distributed advertising and presence.

6.5.2 Person-to-person advertising

Traditionally, advertising has been principally a one-way process, but with today's new interactive technology, consumers can give feedback to advertising messages in real time using the same channels as the sender (Arens 2007).

Second Life has over seven million registered users, of which over 1.5 million have logged in at some point during the last 30 days (Educause 2008). These figures suggested going inside SL to directly communicate with the users using an avatar to give an overview of the purpose of the CBR web tool and to gain a direct feedback from them.

The person-to-person advertising approach is performed in the form of semi-structured interviews of a random selection of Second Life users that came from different backgrounds and ages. All interviews took place in Second Life by visiting the most popular lands there, so the results should accurately represent the majority of the SL users.

6.5.3 Distributed advertising in second life

There are several companies supporting distributed advertising networks in Second Life. The distributed network usually means that company has partners or publishers who display advertisements on billboards as shown in Fig. 6. There are many popular venues in Second Life, from malls to clubs. Many of them will put up posters or kiosks that can distribute information, links, objects, or provide teleports to the sponsoring location. (Polischuk 2007).

The way the distributed advertising approach was performed was based on two phases:

1. The first phase was performed inside SL by publishing posters in the most popular lands there, explaining the ideas of the CBR web tool in a presentable way to encourage the users to try it.
2. The second phase used the social networks which are connected to SL, such as Facebook. These have defined groups for SL users to exchange ideas and information.

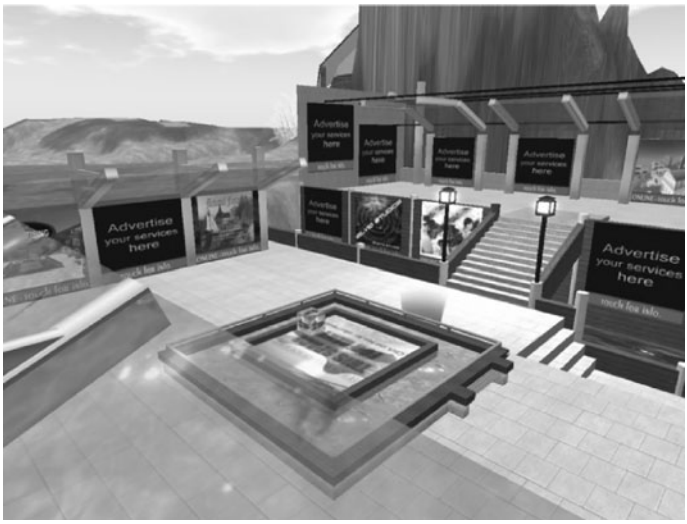

Fig. 6 Distributed Advertising

6.5.4 Presence in Second Life

As with real-world properties (whether web or physical), to be successful you will need to promote your presence and activity to the Second Life community through group notices, event notices, articles and advertisements in Second Life–related periodicals, or by any other means appropriate (Azzara 2007).

Some companies have found success with an ongoing presence in Second Life and driving visitors though related or unrelated events—technical talks (Intel), or concerts, for example. The key, as with a web site, is to provide fresh content that gives residents a reason to visit and interact with your land. Others have created places with activities to engage residents (e.g. Vodaphone and Weather Channel).

The way the presence approach was performed was based on two phases:

1. The first phase was performed inside SL by developing a land there to perform as a small venue or place that encourages SL residents to visit and use the CBR web tool. It is also used as a platform to communicate with SL residents to exchange ideas and information.
2. The second phase was performed by attending the main events and conferences held inside SL to promote and exchange ideas with the attendees there.

6.5.5 Feedback services

In the words of the Second Life creators, Second Life "is a 3-D virtual world entirely built and owned by its residents." (Polischuk 2007). In this spirit, a user feedback service has been provided within the CBR tool where users can give their comments and suggestions for improvement.

Feedback services are provided in two sections:

1. The first section gives the user the opportunity to provide his or her feedback about a specific case and to evaluate the case solution retrieved by the CBR tool as shown in Fig. 7.
2. The second section gives the user the opportunity to provide general feedback about the CBR web tool and any ideas or suggestions for future improvement.

6.5.6 Case studies inside SL

Case studies typically examine the interplay of all variables in order to provide as complete an understanding of an event or situation as possible (Becker et al. 2005). A series of case studies were conducted inside Second Life to establish more feedback from the users. The case studies involved semi-structured, interview-based questioning of a random selection of Second Life users. All interviews took place in Second Life.

These studies involved three stages:

1. Giving the user an overview about how the system works for the user.
2. Sending the system's link to the user and inviting him or her to use it.
3. Following up the case study with the user, after he or she has finished, to gain feedback by asking him or her some specific questions. These questions have been chosen to cover all aspects of this research by clearly identifying the purpose of each question and its influence to the evaluation phase.

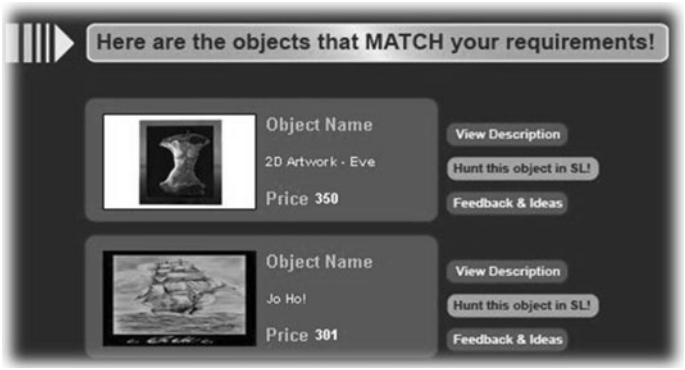

Fig. 7 Feedback service (snapshot of the system)

These case studies enabled the advertising of the CBR web tool ideas and gained direct feedback from SL users.

The first two stages have been carried out for more than fifty users at the time of writing, but these case studies are still progressing, so it is not yet possible to analyse the users' responses. This continuous study will enable the system to be upgraded to make it more effective in assisting SL development. The research aim is also to show how Second Life developments can be improved with the use of these software techniques and methodologies.

7 Initial results

7.1 System evaluation

System evaluation is the process of assessing the performance of a complete system to discover how it is likely to perform in live market conditions. It is the process of reviewing information gathered and using it to form judgments, which, in turn, are used in further decision making. The close monitoring of the use of the tool will enable the benefits of using the system, including cost savings, to be recorded and any disadvantages or obstacles that the system may encounter will be observed.

The first system evaluation was subjected to three types of evaluation as it neared completion. The first form of evaluation was a system test, in which test cases were input to see whether the system's recommendations matched the cases chosen by the users. The second form of evaluation was a user feedback service where users can report problems and suggest improvements. This enables adjustments to the system to be made to further improve the results. The third form of evaluation is analysing the case studies inside second life.

7.2 System test

When full cases become available, the users can be questioned to determine how to add the new cases to the case base to improve system coverage. The evaluation and action taken is based on recommendations made by users.

For a specific case study, the case solution retrieved is evaluated. Then, two results may be produced:

1. If the result is successful, the case is retained and saved in the database as a new case in order to learn from the success. The success of the object cases retrieved in matching the user's requirements is determined by the user. If the user decides the result is successful, this is noted and the case that was most appropriate is stored in the database for future searches as a good case. This case will then be used in any new searches with similar attribute values.
2. If the results are unsuccessful, then the case base may be adjusted to improve the results returned by the matching process. Action is taken as a result of the case studies if several different users give the same response in similar circumstances or if the developer believes the comment is reasonable and agrees with the response given. In such cases, the descriptor attributes for some objects may be adjusted so that future similar object searches will give more satisfactory results on subsequent occasions.

7.2.1 Initial results

Thirty responses were obtained on different cases of the system. These responses have been analysed as initial results in Table 2.

7.3 Feedback sections

A feedback section has been provided where users can give their comments for improvement. From this feedback, the system will be adjusted according to the users'

Table 2 Initial results 1

Do you think this case suits your requirements?
73.3% stated the given case suits their requirements
Eight responses out of 30 said that the solution case did not suit their requirements. Then, they clarified the reasons by giving their feedback about this case
In this case, the descriptor attributes for some objects will be adjusted so that future similar object searches will give more satisfactory results on subsequent occasions
That should mean, with more responses in the future, that a higher percentage returned solutions will be satisfactory in subsequent evaluations
Did you find your desired object here?
66.7% stated the system help them in finding the object they required
The difference between this percentage and the previous one leads to the conclusion that even if the case is considered as a good case, it does not always mean that it will give the user the object they desire
This leads to two conclusions
(1) Case adjustment is not only needed for the bad cases, some of the good cases may also need some adjustment for better results
(2) The system needs more objects to be added to the database
Do you think the system functionality works fine the way it currently is?
70.3% stated the system is working fine
Only two people said that the functionality is not good
This can be considered as a good sign for the system success as the evaluation phase is still in the beginning

Table 3 Initial results 2

Do you think this system is useful?
85% stated the system is useful
This means that the majority of the users believe that the system can be useful. The aim is to further improve the system to match user's expectations
Do you consider the using of previous solutions and learning from them is helpful for you?
35.3% stated that they do not know whether it helps or not
These responses showed that some of the users within the virtual world were not familiar with the idea of using previous solutions
Do you think modelling helps in developing the lands in SL?
50.3% stated that they do not know whether modelling helps in developing within SL
These responses showed that the idea of applying software modelling in SL is not understood

demand. By performing these adjustments, the system should become more and more useful for users wishing to access items and objects within SL. This section is in two parts. The first part asks the users some specific questions about the idea of the system and the system itself. The second part is by giving the user the chance to provide general feedback about the system and any ideas or suggestions.

7.3.1 Initial results

Twenty responses were obtained from different users of the system. And these responses have been analysed as an initial result in this section (see Table 3).

The responses that came from the general feedback implied that the users preferred to have existing ready objects rather than having to create their own. The new users generally used past experience of previous users in order to find the objects that match their requirements.

The responses clearly showed that most of the users within Second Life were not familiar with how to use the built-in Second Life programming language that is available to construct new objects in the virtual world.

8 Future work

This paper describes the development of a methodology for developing artefacts within virtual environments. This first step is the creation of a tool for SL developers capable of generating feedback from the users on how successful the tool is on retrieving desired objects from the SL environment that match the user's needs. From this feedback, improvements that can be made to the system will be identified and the system can then be adjusted according to the users' demand. This will facilitate the use of the system and obtain results which are more closely related to the users' needs.

For future development, a new continuous feedback approach has been implemented in the second version. From this, it will be possible to increase the validity and reliability of the system by constantly tracking its progress. The close monitoring of the use of the tool will enable the benefits of using the system, including cost savings, to be recorded and any disadvantages or obstacles that the system may encounter will be observed. This should

enable the system to be refined and developed to become a highly effective SL development tool.

9 Conclusions

Having researched Second Life, case-based reasoning and other software methodologies, it was concluded that CBR, with its process cycle, is more compatible and better suited than any other software methodology to be used with SL. However, the interactive nature of SL development and the tool developed enabled some of the steps in the CBR system to be adapted to make it yet more applicable to SL. This involved adapting the process by using priorities that changed the weights of the descriptors used for finding a case match, as opposed to using the standard KNN technique where the weights are equal for all descriptors.

The first version of the web tool was satisfactory as a proof of concept, but not as a generally useful system for Second Life. Therefore, a number of changes have been implemented in order to address these problems and make it a more practical tool. In addition to this, new functionalities have been added in the improved version in order for it to be refined and developed to become a highly effective SL development tool.

This research has shown that Second Life developments can be improved with the use of software techniques and methodologies. CBR has been found to be particularly appropriate for this. Furthermore, by employing feedback from the users, improvements can be made to enable the system to be adjusted according to the users' demand. Future developments in this research project will aim to use the users' feedback to make the tool yet more effective in assisting SL development.

References

Amodt, A., & Plaza, E. (1994). 'Case-based reasoning: Foundational issues, methodological variations, and system', approaches. *AI Communications, 7*(1), 39–59.

Arens, W. F. (2007). *Contemporary advertising* (10th ed.). New York: McGraw-Hill Irwin.

Azzara, M. (2007). 'Virtual worlds, second life, & live events', A white paper describing the immediate value proposition for B2B media companies, 1 Nov 2007.

Becker, B., Dawson, P., Devine, K., Hannum, C., Hill, S., Leydens, J., Matuskevich, D., Traver, C. & Palmquist, M. (2005). *Case Studies.* Writing@CSU. Colorado state university department of english. http://writing.colostate.edu/guides/research/casestudy/. Retrieved Jan 2010.

Cheetham, W., & Goebel, K. (2007). Appliance call center: A successful mixed-initiative case study'. *Artificial Intelligence Magazine, 28*(2), 89–100.

de Hugo, A. & Hoeschl, C. (2003). E-book lançado pela editora digital Ijuris em 2003 de, Post. Doc.

Educause (2008, June). '7 things you should know about… second life'. www.educause.edu/eli. Retrieved June 2008.

Hobson, N. (2007). 'Should businesses get a second life?'. *Knowledge Management Review, 10*(1), 5.

Kowalski, Z., Meler-Kapcia, M., Zieliński, S. & Drewka, M. (2005). CBR methodology application in an expert system for aided design ship's engine room automation. *Expert Systems with Applications, 29*(2), 256–263.

Meisalo, V. (2005). Quality of modern learning materials—the viewpoint of authors and designers. In Mike Horsley, V. Susanne Knudsen, & Staffan Selander (Eds.), *"Has past passed" Textbooks and educational media for the 21st century. Stockholm library of curriculum studies* (Vol. 15). Stockholm: Stockholm Institute of Education Press.

Polischuk, P. (2007). 'Second life options and opportunities for EPA in a virtual world', Woodrow Wilson International Center for Scholars, May 2007.

Rosen, Y., & Salomon, G. (2007). The differential learning achievements of constructivist technology-intensive learning environments as compared with traditional ones: A meta-analysis. *Journal of Educational Computing Research, 36*(1), 1–14.

Author Biographies

Ahmad Shubati is a PhD student at Loughborough University. He holds a MSc. Degree in Computer Science from the New York Institute of Technology and a BSc. Degree in Computer Information System from the University of Jordan. He had three years' experience in the IT industry and two years in the education sector. His research interests are software development methodologies, artificial intelligence and virtual worlds.

Christian W. Dawson is a Senior Lecturer in the Department of Computer Science at Loughborough University. His research interests include not only software project management, in which he has authored a number of books and journal papers, but also cross-discipline—including artificial intelligence and environmental modelling. He has been involved in a number of projects with the Environment Agency and Environment Canada and is a member of a number of conference committees.

Ray Dawson is Professor of Knowledge Management at Loughborough University, United Kingdom. He obtained a bachelor's and a master's degree from Nottingham University before entering industry with Plessey Telecommunications in 1977. At the company, he developed an interest in the working methods for information systems development as practised in industry. Since 1987, he has continued this interest in industrial working methods at Loughborough University, working with companies to improve their information and knowledge management systems. Ray Dawson is now a Professor and Deputy Head of the Computer Science Department at Loughborough where he leads a multidisciplinary research group on Knowledge Management. He is a Fellow of the British Computer Society, a Chartered Engineer and a Chartered Information Technology Professional.