Abstract - We will describe 3Dblog technology, which we have proposed. 3Dblog is a system and technology that combine Web3D with the concept and technology of blog. Since we are studying and developing a 3DBlog system, solving the problems of its core system and development is crucial. However, it’s also important to solve the copyright problem of authors and develop applications with useful GUIs for the actual management of a 3DBlog system. In this paper, we explain the fundamental 3DBlog technology and then focus on the problems that arose during system development and testing. We also describe methods to solve these problems.

Keywords: 3DBlog, Blog and Web3D

1 Introduction

Reflecting the progress of CG software, even beginners are easily producing 3D CG works, especially students. If places to publish 3D works are available, works can be gathered through which the creators can communicate. Web3D technology enables CG authors to publish their work on the Web. However, Web3D remains too difficult for non-experts to learn. Therefore, we are concentrating on blog technology because it provides easy spaces to present personal information for many users. We are developing a 3DBlog system that offers places on the Web for communication through 3D CG works, that is, a system to publish and communicate through 3D CG works as usual blog sites. First, we developed a 3DBlog system that consists of fundamental functions.

Next, we concretely explain the contents. It is only possible to display three-dimensional data on a blog by conventional blog. Our 3DBlog system not only displays three-dimensional data but also provides communication places through three-dimensional works by a method in which three-dimensional data possess mutual cooperation. Our system provides places where users can use trackback and linkage with three-dimensional data and easily contribute three-dimensional data. Our system’s remarkable features allow users to systematically deal with three-dimensional works, which we integrated in a system.

We solved the following problems of this system’s development: trackback methods and linkage between three-dimensional objects and inserting copyright information of the three-dimensional data concerning shapes, surface color/textures, and motions, for example, walking and jumping, concerning the whole or parts of works.

Except for our previous research [2], [6] and 3Dblog in [1], [3], [4], [7] and [8], very few studies [5] have combined Web3D and blog technology. The purpose of this study is to propose, develop, and improve 3DBlog technology to provide a superior development environment that effectively allows Web3D works to be reused and created for all users, regardless whether they are professional, amateur, creator, or spectator.

In this paper we describe these fundamental functions for a 3DBlog system and show several examples of such a system. With these results, we show that the 3DBlog system provides a comfortable communication space for 3D CG users.

2 3DBlog technology

Figure 1 is an example of a 3DBlog used by our students to make their works.

2.1 3DBlog technology proposal

3DBlog, the idea and technology that we have proposed, was born by mating Web3D with blog technology and is proposed as follows. It combines Web3D with the blog concept and technology and users who contribute their own Web3D works with a minimum of difficulty. They can also modify and develop works collaboratively and dynamically.
2.2 3DBlog technology features

3DBlog technology has the following features:

- Simple production, control, and management of homepages that deal with Web3D works
- Simple and dynamic production and use of Web3D works
- Trackback function to easily contribute Web3D works and exchange opinions
- Simple development tools for specific, applied fields, including VR education, display of houses, development of 3D characters, 3D block games, and so on
- Additional archive functions to easily reuse previous works

2.3 Registration of Web3D components by creators and component advancement in a community

For our computer classes on Web3D use, we created and accumulated a large number of Web3D works and components. Although we once exhibited these works for reference on an HTML-based homepage, few students reused them; in fact, most have never been reused. If we do attempt to reuse such data, many problems must be solved: the center coordinates, scale values and directions of works, types of lighting, background, sensor and timer settings, and the category and hierarchical file structure of these works and components. Without data file structure information, users of the works must try again to interpret and understand the data files. This defect is fatal. Furthermore, making such Web3D data available is very time-consuming.

One advantageous method for solving these problems is to employ a database of Web3D data. Users register Web3D works and components and add data property information such as file name, key words, classification, and any additional comments. Users can then easily search for desired data by enlisting such powerful search functions as SQL. In this way, the data of previous Web3D works can be reactivated.

2.4 Collaborative development, accumulation, and reuse of Web3D data

Figure 2 shows the concept of our developing environment. First, users need to record the fundamental information in 3DBlog user registration pages, which immediately generates a 3DBlog page. The style of the 3DBlog page reflects the registered fundamental information. Users and creators can then access the archives of past works and contribute their own as well. Since Web3D content on a 3DBlog page is collaboratively and dynamically modified or supplemented, new relationships will be built between content creators and users.

Our system deals with the Web3D format to manage three-dimensional works for the following reasons. If different formatted three-dimensional data are intermingled, combining works becomes very difficult. Even when combining Web3D works with the same format, if the specifications greatly differ, many problems can occur, such as heavy parts that slow down a computer. For these reasons, the 3DBlog system has a workbench to deal with Web3D data. Furthermore, to adopt standardization, we easily extracted reusable works and parts of data from the database and contributions.

2.5 3DBlog system’s structure

First, we simply explain the 3DBlog structure. Blog and Web3D functions are included in the 3DBlog workbench that provides users a systematic environment by mutual cooperation. Figure 3 shows a more detailed block diagram.
We commenced this research using the reliable Zope system, but it has a weak point with respect to graphic user interfaces (GUI); therefore, we switched to Perl language. Although Perl is very convenient and simplifies the development of a 3DBlog environment, it also has a shortcoming in regard to GUI: it cannot support a mouse. Because it is imperative to manipulate 3D objects in a 3DBlog, this weakness is fatal. Currently, we are developing and researching 3DBlog technology based on Java and JSP.

The two primary features of 3DBlog are the trackback and RSS functions.

2.6 User control

We applied hierarchical structure to the user control of this system. Four ranks exist: super user, managers, general users, and prohibited users. We adopted MySQL as database software. To control user information, we created the following fields:

- ID number
- login name
- e-mail address
- password
- nickname
- privilege

Super user has all complete permission about the 3DBlog site. For example, users can give and take awareness of the commission for all lower class users. Managers have no privileges over other users. But they control contribution trees. General users can contribute their works to the 3DBlog site. Prohibited users cannot login to the 3DBlog site. To login to the 3DBlog site for the first time, a user must register with his e-mail address and receive a password.

2.7 Data structure of contributions

We applied a hierarchical tree structure to the contributions to this system. After users contribute their works to the 3DBlog site, others can use part of them. These outcomes are also available for other works. This tree structure application provides a convenient environment. We created the following fields to manage the information about contributions:

- ID number
- date time
- subject
- body
- author ID
- parental ID
- URL
- work’s file name
- linkage information

2.8 Trackback and linkage

The trackback function simplifies the formation of communities. When users make trackback with other pages in their own blog pages, a link is established from another blog page. Technically speaking, our 3DBlog sends trackback ping data to another 3DBlog site, which receives

![Figure 3: Block diagram of 3Dblog](image)

Figure 3 shows users entering the top page of a 3DBlog. Although the 3DBlog site manager sets the style sheet data (CSS) when generating this 3DBlog site, users can also easily change it. In practice, the reference servlet, which is the main page of this site, dispatches the Reference & Contribution (JSP) program when users want to refer to or contribute data. In this system we deal with Web3D data as 3D data. Users can produce 3D works with relevant applications and CAD that we have prepared. They can then contribute their 3D works to the contribution servlet, after which it dispatches the reference servlet and submits the data to a database program (data control and data search java program). We adopted MySQL as the database handle program because it offers many benefits: it is very convenient to import into our system, and new versions are very easy to obtain. Users can also search for whatever data they want and reuse old archived data without any problems.
the trackback URL, and then the site automatically adds the link. (See Figure 4)

3 Development of 3DBlog system

We selected Eclipse due to its usefulness for Java, JSP, and Java Servlet programming.

3.1 Problems concerning 3DBlog system development

We previously developed a 3DBlog system that suffered from an unavoidable problem: user control. We tried to attach author IDs to every contribution, but we failed. Therefore, based on the former system, we developed a new 3DBlog system with user control functions. We continue to improve our system through education applications at our university and are addressing problems concerned with 3DBlog system development. We must consider and solve the following problems.

3.2 Trackback methods for 3D objects

Generally speaking, a blog has a trackback method for text base contributions. But we had to add a new trackback function for 3D objects. We inspected two methods, direct linkage and index menu, as shown in Figure 5. The two bears have direct links to other works, which are bears, of course. In the figure, there are also many strung balls. They are menu balls. If a user clicks on a ball, small 3D objects appear as thumbnails for links to other works. If a user clicks on an appeared menu, she moves to the linked object. We must continue to improve these methods.

3.3 Two reference methods

Concerning reference problems, we studied two reference methods: inline and embed. The merits and demerits of the inline method are the influence of referred works upon the source work; that is, when some work uses part of another work by the inline method, if the parts are changed or edited by the original author, the original work will be changed by the influence of the changed parts. On the other hand, the embed method never influences the original work.

3.4 Self-reference problem

When some students used our system, it suddenly stopped. The problem was self-reference. In Web3D, there are two methods for referring to other Web3D files: inline method and cutting and pasting. Using the Inline method, we must consider self-reference; no file can refer to itself. We must also consider the stretching of several files. But the Inline method is very useful, because when a work refers to other works as parts and these works are improved, influences will appear (see Figure 7). If we select the second method, such problems will vanish. But file size increases.

![Figure 5: Trackback methods for 3D objects](image)

![Figure 6: The difference of two reference methods](image)

![Figure 7: An example of self-reference](image)
Figure 8: The resolution method of self-reference

To use the inline method, we must develop an inspection tool to avoid self-reference. Figure 8 shows the resolution method of this problem.

3.5 Problem of data structure for contributions

Next, we apply the hierarchical tree structure to this system’s contributions. This tree structure is very favorable for users, but this system must retain the overlapped data (see Figure 9). When two or more works use the same type of data, we must reduce identical parts by common usage. We solved this problem by making an archive.

In addition, we considered the importance of tree management tools, including cut, copy, and paste functions for contribution trees. After making a work in a parent node, we wanted it to be part of another work. In this case, we can use a linkage method, but that isn’t a normal method. So we developed tree management functions.

3.6 Display delay

To display Web3D, we use an “embed” method and paste the Web3D window to the 3DBlog page. When a user wants to use many Web3D windows, display delay will occur. To solve this problem, the system must limit the number of Web3D windows. For example, we can change some useless Web3D windows into thumbnails.

Figure 9: Data structure problem

Figure 10: Addition of author’s copyright information

4 Copyright protection for contributors

We put author IDs on each contribution. But that is not enough to protect copyright because such information will not be “inherited.” Therefore, we added copyright information to the head of Web3D files as a comment to the 3DBlog system, as shown in Figure 10. In this way, even if files are copied, copyright information will be inherited. (See Figure 11) This information can be removed, but not easily. We must continue to study this problem. The next argument addresses copyright contents.

There are considerable items for two-dimensional works, including permission to make copies and to modify. Moreover, for three-dimensional works, considerable items also exist, including problems concerning transformation of object shape and its surface appearance (color, texture, and brightness), and motion. We can deal with motion problems just as we can easily make Web3D files that include motion descriptions. Using this method, we cannot only copyright information of the work itself but also its parts when someone uses these files as parts. For example, for works that have walking bears, we can place a copyright on both the works and parts, such as head, body, foot, and so on. We can also properly copyright motion files that have walking motion. To make a long story short, we are able to deal with some kind of permissions and considerable items as follows.

The kind of permissions

- Usage available / not available
- Change available / not available
- Right abandon / maintain Considerable items.
- Shape Appearance Motion
5 Addition of other functions

We developed other functions. In our previous system, users were able to deal with object data and static texture or images. Figure 12 shows the concept of the motion function we developed. If some user contributes motion data and another user contributes a doll, we can make the combined doll which dances according to the contributed motion. Other new functions are sensor function, sound function and timer.

6 Archive

We developed an archive for this system. Since we have been teaching VR for many years, our students have created many Web3D works. As a result we have also accumulated a great number of Web3D works. Even though some of these works are excellent, we have never reused them. Therefore, we studied an environment that is available to general creators. Users can use the archives to search for the works of others.

7 Conclusions

In this paper, we described the fundamental technology of a 3DBlog system as well as its problems and methods to solve them. We also solved the problems that occurred in operation tests. And we developed new functions to deal with motion, sensor and sounds. Furthermore, we made applications to improve system usefulness and created a 3DBlog system that allows users from amateurs to professionals to systematically deal with three-dimensional works. In the future, we will continue to improve and expand 3DBlog’s existing Web application functions, especially for a proper human interface.

References


