Web Version of *IntelligentBox* (*WebIB*) and Its Integration with Webble World

Yoshihiro Okada

Innovation Center for Educational Resource (ICER), Kyushu University Library,
Motooka 744, Nishi-ku, Fukuoka, 819-0395, Japan
okada@inf.kyushu-u.ac.jp

Abstract. This paper treats a 3D graphics software development system called *IntelligentBox* known as a 3D meme media system and its web version (*WebIB*). *IntelligentBox* provides various 3D software components called *boxes* each of which has a unique functionality and a 3D visible shape. *IntelligentBox* also provides a dynamic data linkage mechanism that allows users to develop interactive 3D graphics applications only by combining already existing *boxes* through direct manipulations on a computer screen. Recently, the author extended *IntelligentBox* system to make it possible to develop interactive web 3D contents. This is called the web version of *IntelligentBox* (*WebIB*). This extended mechanism is also regarded to be available for the integration of *IntelligentBox* with Webble World, an open web-based meme media environment. This paper explains essential mechanisms of *IntelligentBox* and *WebIB*, introduces some interactive web 3D contents realized using *WebIB*, and also discusses about the possibility of the integration of *IntelligentBox* with Webble World.

Keywords: 3D graphics, Toolkit system, Component ware, Web contents, 3D Meme Media.

1 Introduction

Advances in recent computer hardware technology have made possible 3D rendering images in real time. Consequently, 3D software has become in great demand although its development is more laborious work than 2D software development. For this reason, we proposed a 3D graphics software development system called *IntelligentBox* known as a 3D meme media system [1, 2]. Application fields of *IntelligentBox* include 3D-CG animation creation, virtual reality software development, interactive simulator development, and so on. However, the developed 3D graphics applications could not be available on the web. If they were available on the web, the usefulness of *IntelligentBox* would become higher. Therefore, we extended *IntelligentBox* system to make it possible to develop interactive web 3D contents [3, 4]. This is called *WebIB* as the web version of *IntelligentBox*. In this paper, we explain essential mechanisms of *IntelligentBox* and *WebIB*, and introduce some interactive web 3D contents including educational contents and information visualization tools.

O. Arnold et al. (Eds.): WWS 2013, CCIS 372, pp. 11–20, 2013.

On the other hand, recently, an open web-based meme media environment called Webble World was proposed [5] and its components are called webbles. If *IntelligentBox* system is possible to integrate with Webble World, its usefulness would become higher than ever. Fortunately, the same mechanism as that of *WebIB* seems available for the integration. So, in this paper, we also discuss about the possibility of the integration of *IntelligentBox* system with Webble World.

The remainder of this paper is organized as follows: Section 2 introduces related work. In Section 3, we explain essential mechanisms of *IntelligentBox* and its extended mechanisms for *WebIB*. Section 4 introduces several interactive web 3D contents realized using *WebIB*. In Section 5, we discuss about the possibility of the integration of *IntelligentBox* system with Webble World. Furthermore, in Section 6, we discuss about development costs of web 3D contents, their performances and significance of *IntelligentBox* system and *WebIB* as development environments for web 3D contents. Finally, we conclude the paper in Section 7.

2 Related Work

Our research purpose is to propose a software architecture that makes it easier to develop 3D graphics applications including interactive web 3D contents. Its related systems are 3D graphics toolkit systems and programming libraries like Open Inventor [6], Coin3D [7] and 3D Widget [8]. Open Inventor is an OpenGL based object oriented programming library. Coin3D is also library very similar to Open Inventor. 3D Widget is a Widget-based toolkit system for the 3D GUI development. Some of them provide an authoring tool that enables to design 3D graphics contents. Even using such authoring tools, it is not easy to develop 3D graphics applications because developers have to write text-based programs for that. As for development tools for interactive web 3D contents, there are library systems like Java3D [9], Jogl [10], Papervision3D [11] and WebGL [12]. Java3D and Jogl is Java-based 3D graphics library that works as a plug-in virtual machine running on a web browser for 3D graphics contents. Papervision3D is Flash-based 3D graphics library that also enables to develop web-based 3D graphics contents. WebGL (Web Graphics Library) is JavaScript API for rendering interactive 3D graphics and 2D graphics within any compatible web browser without the use of plug-ins. These are library systems so that the user has to write text-based programs for developing web-based 3D graphics contents.

Our research system *IntelligentBox* and its web version (*WebIB*) provide various 3D software components called *boxes* represented as visible, manually operable, and reusable functional objects. Furthermore, they provide a dynamic data linkage mechanism called slot connection. These features make it easier for even end-users to develop 3D graphics applications including web 3D contents. This is the main difference of *IntelligentBox* and *WebIB* from others. We also have a plan to integrate *IntelligentBox* system with Webble World using the same mechanism as that of *WebIB*. This is new feature mainly shown in this paper.

3 Essential Mechanisms of IntelligentBox and WebIB

WebIB employs the same essential mechanisms of *IntelligentBox*. This section explains those mechanisms in the following two subsections. They are very simple but very useful. After that, we describe extended mechanisms for *WebIB*.

3.1 Model-Display Object (MD) Structure

As shown in Figure 1, each *box* consists of two objects, a model and a display object. This structure is called MD (Model-Display object) structure. A model holds state values of a *box*. They are stored in variables called slots. A display object defines how the *box* appears on a computer screen and also defines how the *box* reacts to user operations. An example of *RotationBox* in Figure 1 has a slot named 'ratio' that holds a double precision number used as a rotation angle. Through direct manipulation on the *box* using a mouse device, its slot value is changed. Furthermore, its visual image changes simultaneously with the change of the slot value. In this way, a *box* reacts to the user's manipulation according to its functionality.

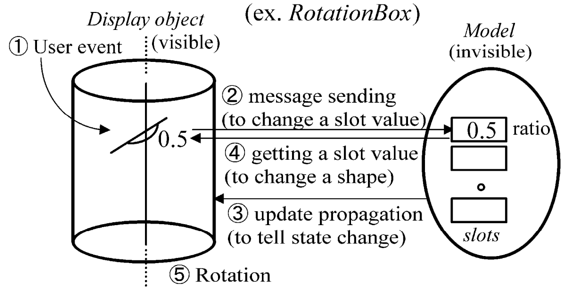

Fig. 1. An *MD* structure of a *box* and its internal messages

3.2 Dynamic Data-Linkage Mechanism Called Slot Connection

Figure 2 illustrates a data linkage concept among *boxes*. As shown in the figure, each *box* has multiple slots. Its one slot can be connected to one of the slots of other *box*. This connection is called a slot connection. The slot connection is carried out by three standard messages, i.e., a set message, a gimme (give me) message and an update message, when there is a parent-child relationship between two *boxes*. These messages have the following formats:

(1) Parent *box* set <slotname> <value>.
(2) Parent *box* gimme <slotname>.
(3) Child *box* update.

A <value> in a format (1) represents any value, and a <slotname> in formats (1) and (2) represents a user-selected slot of the parent *box* that receives these two messages. A set message writes a child *box* slot value into its parent *box* slot. A gimme message reads a parent *box* slot value and sets it into its child *box* slot. Update messages are issued from a parent *box* to all of its child *boxes* to tell them that the parent *box* slot value has changed. By these three messages, the two slots of a child *box* and its parent *box* are connected and their two functionalities are combined.

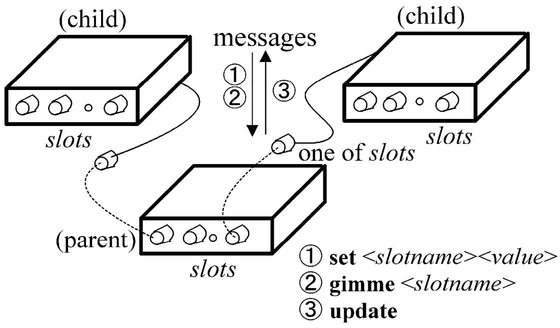

Fig. 2. Standard messages between *boxes*

3.3 Extended Mechanisms for *WebIB*

IntelligentBox system uses OpenGL 3D graphics library which provides an off-screen rendering functionality. As shown in Figure 3 and 4, using this functionality, a rendered image of a 3D scene can be generated on a web-server and transferred to a web-browser through the Internet. On the web-browser, besides a HTML program, a JavaScript program runs to manage user operation events, i.e., a mouse move, a mouse button click and so on. Such user operation events will be transferred to the web-server using XMLHTTPrequest message through a CGI-program of the web-server. The CGI-program (Perl script) once receives the user operation events and applies them to *IntelligentBox* system running on the web-server. And then, *IntelligentBox* system will generate next off-screen rendering image of the 3D scene to be updated by the received operation event. In this way, the user can interactively manipulate 3D contents of *IntelligentBox* that runs on the web-server through his/her web-browser. Since *WebIB* supports most web-browsers, e.g., Internet Explorer (IE), Mozilla Firefox, Opera, Google Chrome and Safari, *WebIB* is available on any mobile device like iOS device and Android OS device. However, when using iOS device or Android OS device, the touch interface should be used. Therefore, *WebIB* also supports the touch interface for iOS devices and Android OS devices.

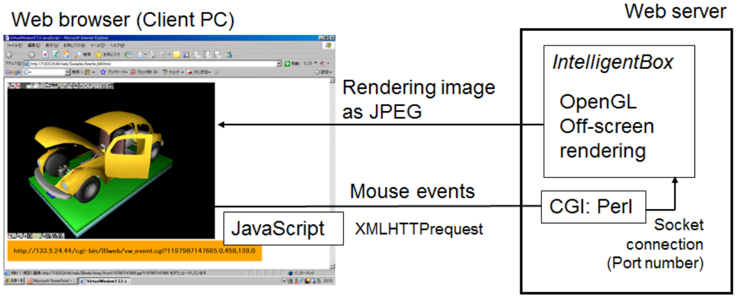

Fig. 3. Extended mechanisms of *WebIB*

3.4 Mechanisms of *WebIB* for Multiple Users

As shown in the left figure of Figure 4, *WebIB* can provide multiple users with a web-based collaborative virtual environment based on the same mechanism of Figure 3. *WebIB* has a System ID No. (SID No.), and by specifying it, each client user can access its corresponding *WebIB* that has the same SID No. through his/her web-browser. Using this data-linkage, communications between a teacher and a learner become possible in the cases of educational 3D contents. The right figure of Figure 4 shows another case that each of multiple users individually uses his/her own *WebIB*. As shown in the figure, each of multiple *WebIB*s can run on the same web server, and using different System ID No.(SID No.), each user can access his/her own *WebIB* through his/her web-browser.

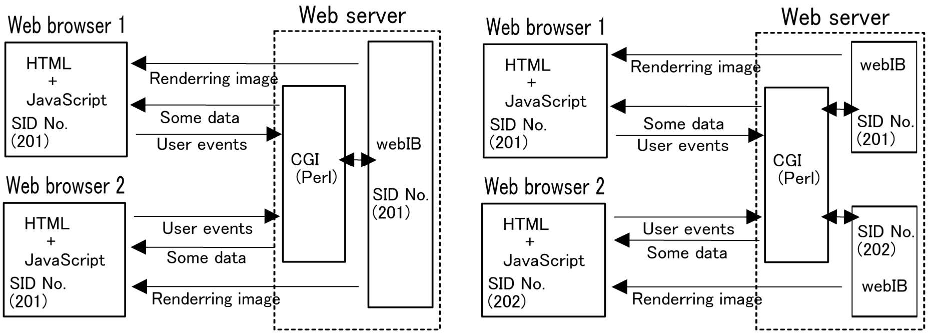

Fig. 4. Data-linkage among one *WebIB* and its two clients (Left), and data-linkage among two individual *WebIBs* and two dedicated clients (Right)

4 Web Contents Realized Using *WebIB*

This section presents actual examples of interactive web contents realized using *WebIB*. In this paper, we especially introduce educational contents because they are regarded to be practical contents. We also introduce information visualization contents related to our one project about cyber physical systems.

4.1 Educational Contents

As visual contents for educations, there are two types of contents, i.e., 3D contents and 2D contents. *WebIB* is used as a 3D model viewer for 3D contents and as an image viewer for 2D contents. In the following, we introduce these examples.

1) 3D Model Viewer

The left figure of Figure 5 shows an example 3D model for students to learn a brain and its some parts represented as web 3D contents using *WebIB* on IE browser. A student can operate interactively for the change of his/her viewpoint, i.e., rotation, zoom and pan, and can point out any parts by a mouse device click to display their part names of the brain. As described in the previous section, using the same SID No., these operations are shared among other users. If one user is a teacher and other users are students, the all students can see the teacher's operations and they can understand part names pointed out by the teacher. The right figure of Figure 5 is another example, a hurt model. Similarly to the left figure, each student can learn the structure of a hurt and its part names individually on the web at any time and at any location.

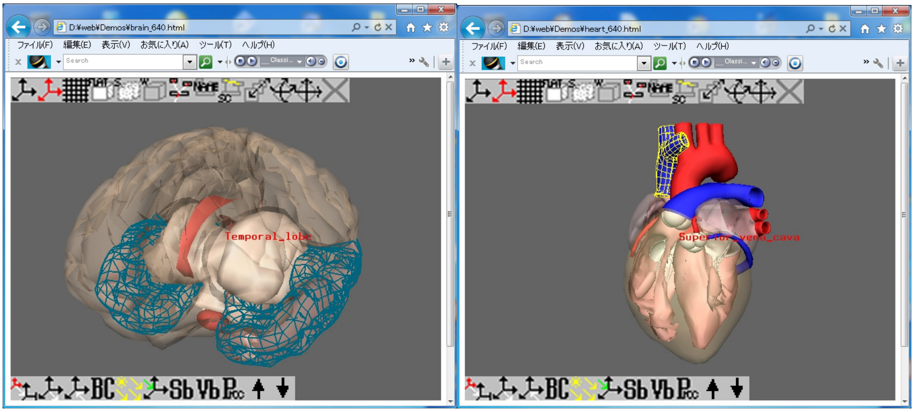

Fig. 5. Web 3D content examples of medical education

2) Image Viewer

As educational contents, images are common and often used. In this case, *WebIB* can be used as an image viewer. Although *WebIB* contents are 3D graphics contents, using the texture mapping mechanism, any image data can be displayed in a 3D space. The left figure of Figure 6 shows an image of a certain typhoon [13] provided by NASA and the right figure of Figure 6 shows the same typhoon image looks from the very far viewpoint. It is possible to manipulate the image for the pan and the zoom like Google Maps operations. If a teacher have high resolution images as educational materials, he/she can provides such images with his/her students on the web like Google Maps service. Students can interactively make a pan or a zoom towards any part of an image that they want to look. Figure 7 shows another examples, CT images as educational contents of medicine.

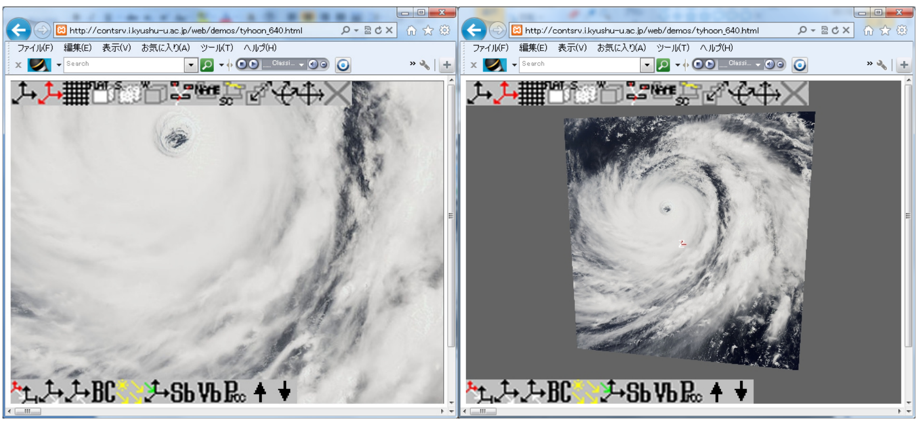

Fig. 6. Typhoon images as educational content examples

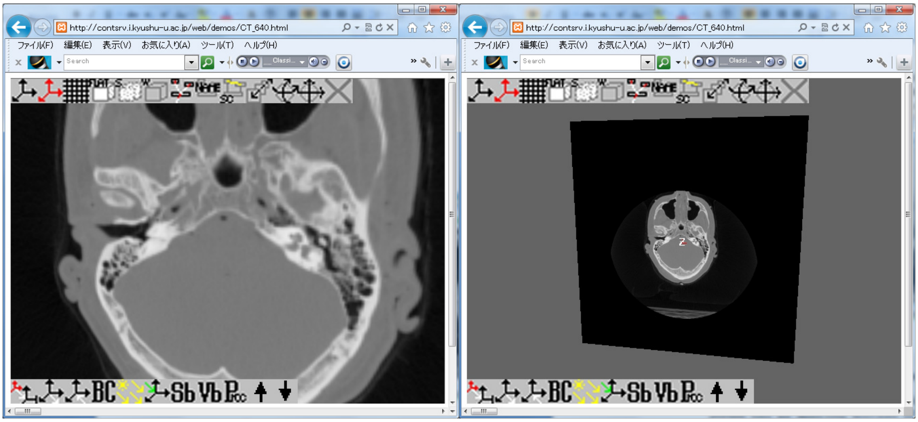

Fig. 7. CT images as educational content examples

4.2 Information Visualization Contents

We have one project about cyber-physical systems. One of the important topics in researches on cyber-physical systems is the analysis of big data to be collected from the physical world. As one of the analysis methods, information visualization is useful. Currently, we have been developing two types of visualization tools using *IntelligentBox* system for our cyber-physical system project. The first one is for the analysis of human movements because human movements are very important cues for the analysis of human activities. The left figure of Figure 8 shows a screen image of such web content. This is the building of our graduate school consisting of several floors. To simplify a 3D model for the building, we employ 2D floor map images and use the texture mapping mechanism. It is very easy to visualize the building and display human movements. In this case, glyphs have a different color, a different size and a different shape to specify several attribute values of their corresponding persons and move on a floor. So, we can understand persons' activities from glyphs' movements.

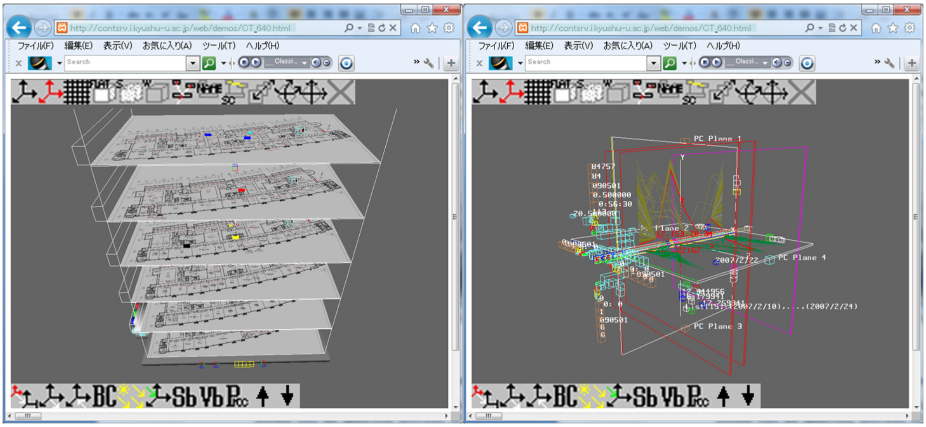

Fig. 8. Information visualization contents as web services

The other one is for the analysis of multi-dimensional/multi-attributes data because data to be collected from the physical world are various types and have various attributes. We have already proposed one information visualization tool called *Time-tunnel* for the analysis of time-series numerical data and extended it by adding a visualization functionality similar to Parallel Coordinates. The right figure of Figure 8 shows a screen image of such web content. See the papers [14, 15] for its detail.

5 Integration of *IntelligentBox* with Webble World

Webble World is an open web-based meme media environment and its components are webbles developed using Microsoft Silverlight technology. Originally, Silverlight supports XMLHTTPrequest communication mechanism. Therefore, as shown in the upper figure of Figure 9, a communication between a webble existing on a web browser and *IntelligentBox* system runs on a web server is possible through the same mechanism shown in Figure 3. In addition, the latest version of Silverlight supports the standard socket communication mechanism. Therefore, a webble also can communicate with *IntelligentBox* system runs on any server without any CGI-program as shown in the lower figure of Figure 9. In this way, *IntelligentBox* system can be integrated with Webble World technologically although we have not yet implemented such webbles. In the near future, we will develop such wobbles and their applications.

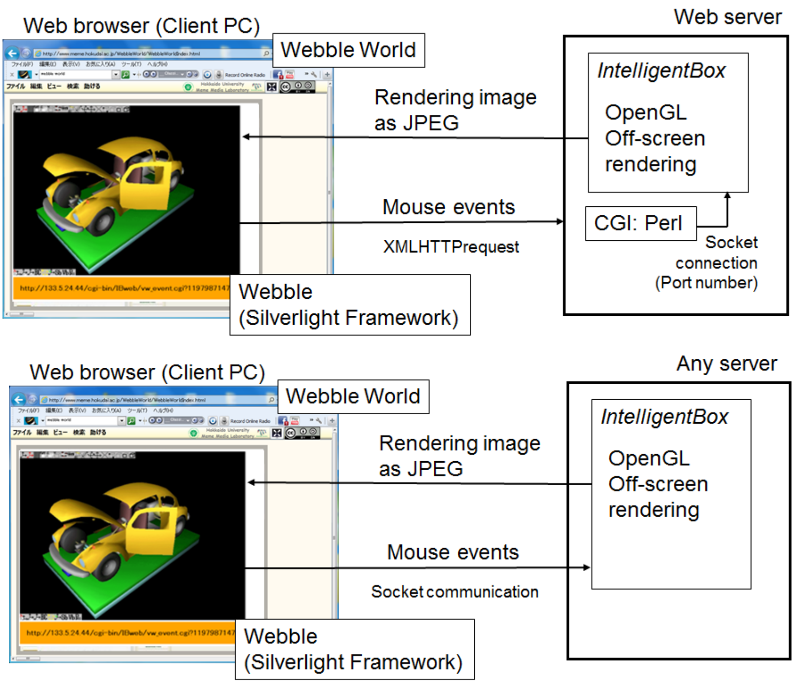

Fig. 9. Integration of *IntelligentBox* with Webble World

6 Discussion

6.1 Development Costs

For preparing the example models of Figure 5, 6 and 7, it was not necessary to write any text-based program. They were constructed only through direct manipulations on a computer screen using a mouse device. Therefore, even teachers who do not have any programming knowledge can construct web 3D contents including 3D educational contents by using *WebIB*.

6.2 Performances

The most important factor should be considered as for the performance of *WebIB* is update rate (refresh rate or frames per second) of off-screen rendering images on a web-browser transferred from a web-server. This value depends on a network environment. However, there is no problem because the system employs JPEG compression and can appropriately reduce the network bandwidth required to transfer an off-screen rendering image.

6.3 Significance

There are many services on the web. However, there are very few services of 3D graphics contents on the web. One of its reasons is that currently most web-browsers natively do not support the 3D graphics rendering. If we want to receive 3D graphics content services on the web, we have to install any plug-in software like Java3D, Jogl [11], etc. This is inconvenient especially for end-users who are not accustomed to doing so. Even if this is not inconvenient, there is another reason that 3D graphics contents need 3D polygonal model data but they are usually not allowed to be distributed through the Internet due to their user-licenses. Even if the distribution of 3D polygonal model data is allowed, there is a problem that the transmission time of them become very long. On the other hand, our web version of *IntelligentBox* works as a SaaS application does not have the above problems because only off-screen rendering images of 3D scenes are transferred from a web-server to a client web-browser as explained in Sec. 3. So, interactive 3D contents already developed as applications of *IntelligentBox* can be reused as web 3D contents those work as SaaS applications. Its MashUp with other SaaS applications are also possible. As a result, the usefulness of *IntelligentBox* has become higher than ever. The integration of *IntelligentBox* with Webble World mainly proposed as new feature in this paper is also regarded as one of such MashUp examples.

7 Conclusion

This paper introduced software architecture that makes it easier to develop 3D graphics applications including interactive web 3D contents. Our research system *IntelligentBox* provides various software components each of which corresponds to each of required functionalities. Combining these software components by direct manipulations on a computer screen enables users to develop 3D graphics applications without writing any text-based program. This feature is important for end-users who do not have any programming knowledge. Furthermore, we enhanced the usefulness of *IntelligentBox* by extending it to make it work as a development system for web 3D graphics applications like web 3D educational contents. This is called *WebIB*, the web version of *IntelligentBox*. In this paper, we showed some interactive web 3D contents to clarify the availability of *WebIB* and also indicated the possibility of the integration between *IntelligentBox* system and Webble World, an open web-based meme media environment.

As future works, we will improve the transfer mechanism of an off-screen rendering image to reduce the required bandwidth and to improve the performance. Also, we will develop more practical web 3D contents to clarify the usefulness of *WebIB*. Moreover, we will develop webbles that can communicate with *IntelligentBox* system runs on a web server and their applications to show the usefulness of the integration of *IntelligentBox* with Webble World.

Acknowledgment. This work was partially supported by FOST(Foundation for the Fusion Of Science and Technology), Japan.

References

1. Okada, Y., Tanaka, Y.: IntelligentBox: A Constructive Visual Software Development System for Interactive 3D Graphic Applications. In: Proc. of Computer Animation 1995, pp. 114–125 (1995)
2. Okada, Y., Tanaka, Y.: Collaborative Environments in IntelligentBox for Distributed 3D Graphic Applications. The Visual Computer 14(4), 140–152 (1998)
3. Okada, Y.: IntelligentBox as Development System for SaaS Applications Including Web-based 3D Games. In: Proc. of the 9th Annual European GAMEON Conf., pp. 22–26 (2008)
4. Okada, Y.: Web version of IntelligentBox (WebIB) for Development of Web 3D Educational Contents. In: Proc. of IADIS Int. Conf. Mobile Learning 2011, pp. 251–255 (2011)
5. Webble World, http://cow.meme.hokudai.ac.jp/WebbleWorldPortal/
6. Open Inventor, http://oss.sgi.com/projects/inventor/
7. Coin3D, http://www.coin3d.org/
8. 3D Widget, http://www.viewpoint.com/widgets/
9. Jave3D, https://java3d.dev.java.net/
10. Jogl, https://jogl.dev.java.net/
11. Papevision3D, http://blog.papervision3d.org/
12. WebGL, http://www.khronos.org/webgl/
13. http://eoimages.gsfc.nasa.gov/images/imagerecords/15000/15467/nabi_amo_02sep05_250m.jpg
14. Akaishi, M., Okada, Y.: Time-tunnel: Visual Analysis Tool for Time-series Numerical Data and Its Aspects as Multimedia Presentation Tool. In: Proc. of 8th International Conference on Information Visualization (IV 2004), pp. 456–461. IEEE CS Press (2004)
15. Notsu, H., Okada, Y., Akaishi, M., Niijima, K.: Time-tunnel: Visual Analysis Tool for Time-series Numerical Data and Its Extension toward Parallel Coordinates. In: Proc. of CGIV 2005, pp. 167–172. IEEE CS Press (2005)