

Supporting Collaborative 3D Editing over Cloud Storage

Yeoun-Ui Ha¹, Jae-Hwan Jin², Myung-Joon Lee³

Department of Electrical/Electronic and Computer Engineering, University of Ulsan,
93, Daehak-ro, Nam-gu, Ulsan 680-749, Republic of Korea,
^{1,2}{gkdus23, jjhok2000}@gmail.com, ³mjlee@ulsan.ac.kr

Abstract. As 3D modeling has been popular in the current social environment, 3D editing has been received much attention. In this paper, we present a 3D editor named Lets3D which supports real-time collaboration in editing and sharing 3D objects. Lets3D is implemented as an extension of the three.js editor, running on Web browsers. For collaborative functionalities, Lets3D utilizes the whiteboard service of C3ware, which is a middleware providing abstract editing operations and concurrency controls on shared objects over cloud storage. Also, to synchronize the editing screens among a group of users, Lets3D relies on the robust messaging facility of the Openfire XMPP messaging server.

Keywords: 3D Editor, Collaborative 3D editing, C3ware, XMPP

1 Introduction

3D Modeling is a process of developing 3D objects and their features. A 3D Editor is a tool for 3D Modeling which enables users to edit 3D objects in a convenient way. As of now, there are lots of 3D editors like Google's sketch up [1], AutoCad [2], sunglass [3] and others. Also, to handle 3D objects in such tools, graphics APIs such as OpenGL [4], direct3d [5], and WebGL [6] are being used popularly. Recently, xml3d [7] is proposed as an extension to HTML5 for supporting interactive 3D graphics in Web browsers.

Along with the emergence of useful 3D editors and rich 3D-related activity, collaboration in 3D modeling is getting much attention. But, the currently existing 3D collaborative environments for 3D modeling [3][8] are provided only as commercial Web services. Besides, the famous open source 3D editors like blender [9] and FreeCAD[10] do not support collaborative environment for now.

In this paper, we present a collaborative 3D editor named *Lets3D*, which supports real-time editing and sharing of 3D objects among users through Web browsers. Lets3D is implemented as an extension of the three.js editor using Three.js [11], which is a typical lightweight cross-browser API of WebGL for rendering of 3D objects. To provide such collaborative functionality in 3D editing, Lets3D utilizes the whiteboard service of C3ware [12] which delivers abstract editing operations and concurrency controls on shared objects over cloud storage. In addition, to create and

maintain users and groups, Lets3D adopts XMPP messaging [13], which is also used to synchronize the editing screens among a group of users.

2 Lets3D Collaborative Editor

In this section, we describe the structure of the Lets3D editing tool and the collaborative features provided by Lets3D.

2.1 Lets3D System Structure

C3ware is a middleware supporting collaborative workspaces over cloud storage and useful high-level collaborative services including the whiteboard service and the concurrency control service. Lets3D uses the whiteboard service of C3ware for sharing 3D objects among group users and collaborative editing. Also, to maintain consistencies of shared 3D Objects which can be edited by a group of users beyond space, Lets3D receives helps from the concurrency control service of C3ware.

In addition, to deliver the edited 3D objects and to synchronize the editing screens among a group of users, Lets3D adopts the Openfire XMPP messaging server. For providing the user registration and group management, Lets3D uses the Strophe.js Library which can be plugged in Web Browsers. Fig. 1 shows the system structure of Lets3D.

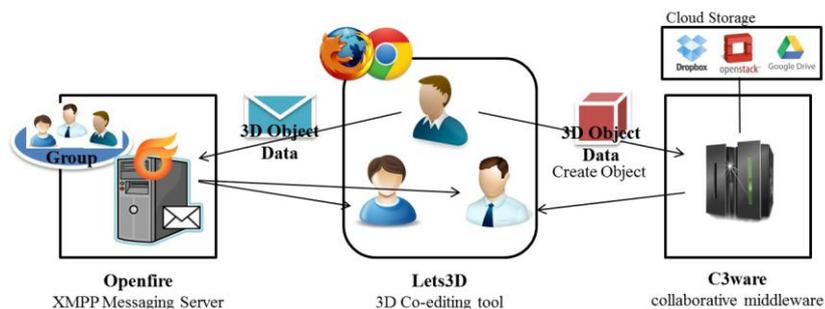


Fig. 1. System Structure of Lets3D

2.2 Functions of Lets3D

For collaboration features, Lets3D provides two main functions: group management and collaborative editing. Every operation associated with group management sends a related message to the XMPP server and/or the related users, whereas every operation associated with collaborative editing sends both a Web service request to C3ware and the related XMPP message to the group members in collaborative editing. C3ware ensures the consistency of the 3D objects through the *Delete Object* Web service

operation and the *Modify Object* operation of the whiteboard service, which uses the concurrency control service internally. Table 1 shows major functions for group management and collaborative editing. Lets3D provides the functions for *import* and *export* of 3D objects, supporting object formats such as *.json*, *.obj* and *interactive xml3D* being used in HTML5.

Table 1. Operations for Collaborative Features in Lets3D .

Classification	Function	Description
Group Management	Create Group	Create a new group at XMPP server
	Invite Group	Invite user to group
	Delete Group	Delete group at XMPP server
Collaborative Editing	Create Object	Create 3D object at the editor
	Delete Object	Delete 3D object at the editor
	Modify Object	Modify 3D object at the editor
	Clone	Clone 3D object at the editor
	Import/Export	Import/Export file format you want
	Redo/Undo	Redo/Undo object at the editor

2.3 3D Collaborative Editing Scenario

The editor screen of Lets3D is composed of *Viewport*, *Menu bar*, *Side bar* and *Tool bar* as shown in Fig. 2. The *Viewport* is the view screen displaying 3D objects. The *side bar* shows the hierarchical structure of the detailed information on the selected object. The *Menu bar* and the *Tool bar* support editing functions for 3D objects such as creation and movement of objects. Usually, users perform the following procedure for collaborative 3D editing through Lets3D.

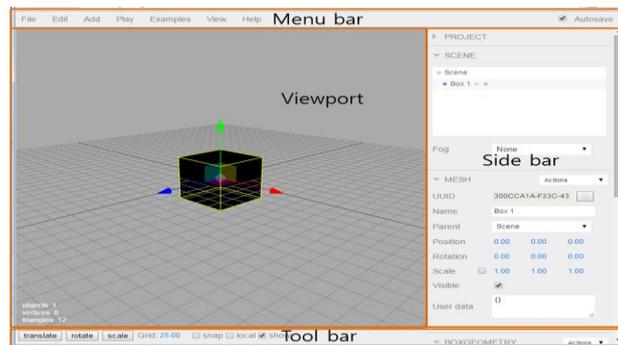


Fig. 2. Editor Screen of Lets3D

1. A user makes a registration to the Openfire XMPP server through the *login* and *join screen* of Lets3D as shown in Fig. 3.
2. The joined user creates a group, inviting other users to the group at the *group management screen* of Lets3D in Fig. 4.

3. Users of the group perform editing operations on 3D objects sharing the same editing screen. With the help of the concurrency control service of C3ware, Lets3D synchronizes the editing screens of the group users through XMPP messaging.
4. Each 3D object created or modified by a group user is stored in the related group workspace over cloud storage.

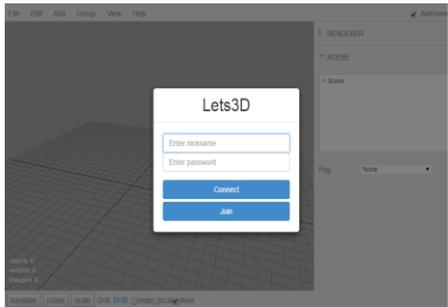


Fig. 3. Login and group management screen of Lets3D

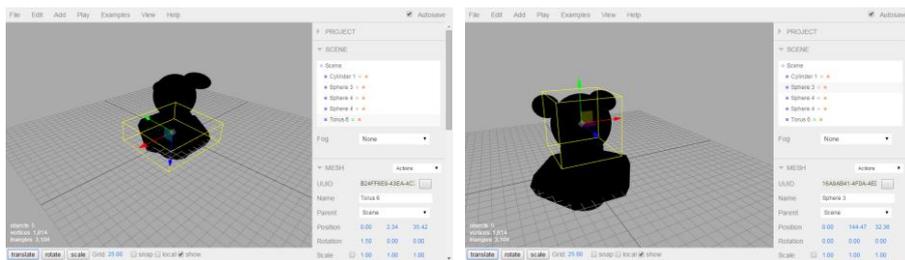


Fig. 4. Collaborative editing through Lets3D

3 Conclusion

In this paper, we described a new collaborative editing tool running on Web browsers. As for the core editing functionality, Lets3D borrows the editing functionality from the open source three.js editor based on the three.js WebGL API. As for the collaborative functionality, Lets3D utilizes the whiteboard service of C3ware, which enables Lets3D to store 3D objects into collaborative workspaces provided by C3ware and to perform synchronized edit operations on those shared 3D objects. For communication among users as well as user/group management, Lets3D uses XMPP messaging which is proven reliable in decades. The easy-to-use interface of Lets3D comes from the proven feature of the simple three.js editor. We believe that Lets3D would be useful especially for non-professional users who want to handle 3D objects because of its easiness for editing and its novelty for collaboration.

Acknowledgement. This research was partially supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (No. 2013R1A1A4A01004459)

References

1. Google's sketch-up, <http://www.sketchup.com>
2. Autocad, Yarwood, Alf, and Bernd S. Palm. Introduction to AutoCAD 2016: 2D and 3D Design. Routledge, 2015.
3. Sunglass, <http://sunglass.io>
4. OpenGL, <https://www.opengl.org/documentation>
5. Direct3D, [https://msdn.microsoft.com/en-us/library/windows/desktop/hh309466\(v=vs.85\).aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/hh309466(v=vs.85).aspx)
6. WebGL, <http://www.webgl-publisher.com/TechInfoEn.html>
7. Xml3D, <http://xml3d.org>
8. collaborative environment for 3d modeling, <https://clara.io>
9. Blender, <https://www.blender.org>
10. FreeCAD, http://www.freecadweb.org/wiki/index.php?title=Main_Page
11. Three.js, Dirksen, Jos. Learning Three.js—the JavaScript 3D Library for WebGL. Packt Publishing Ltd, 2015.
12. C3ware, Lee, Hong-Chang, Jong-Eun Park, and Myung-Joon Lee. "C3ware: A Middleware Supporting Collaborative Services over Cloud Storage." The Computer Journal (2013): bxs168.
13. XMPP messaging, Saint-Andre, Peter. "Extensible messaging and presence protocol (XMPP): Core." (2011).