GSTF Journal on Computing (JOC)
DOI: 10.5176/2251-3043_5.1.349
ISSN:2251 - 3043 ; Volume 5, Issue 1; 2016 pp. 29-34

A Game Engine based Networked Infrastructure to Create and Share 3D Abstract Art

Bal Singh
School of Computing, Creative Technologies and Engineering
Leeds Beckett University
Leeds, United Kingdom
B.Singh@leedsbeckett.ac.uk

Pauline de Souza
School of Art and Digital Industries
University of East London
London, United Kingdom
P.Desouza@uel.ac.uk

Abstract— **Online communities have been proactive in producing collaborative creative content such as music, games and other social interactions. Online collaboration has enabled contributors to peer produce and share masses of creative content. Examples range from information sharing such as Wikipedia to open source software and other specific art projects. Software vendors have recently introduced low cost 2D and 3D content authoring tools allowing user communities to generate and share creative content. Emerging networking programming interfaces available inside modern game engines allow contributors to implement multiplayer or multiuser interaction relatively easily. This paper presents a 3D art creation framework to be used over networked infrastructure in a multiuser environment. Contributors will be able to create 3D sculptures at runtime, share with other users in a common networked working environment and critique each other's work. Experimental work also involved evaluating procedurally generated meshes versus instantiation of primitive mesh objects. Saving and loading mesh information in an optimum way is also explored.**

Keywords – 3D Art; online collaboration; interactive art; networking; game engine; procedural meshes; performance analysis;

I. INTRODUCTION

With the proliferation of IT devices and networking infrastructures, online content creation and sharing has become extremely popular. Online communities have produced huge quantities of shared content in disciplines ranging from written articles to music and other interactive applications such as computer games. A good example of such content is the Scratch online project [1]. The contributors of Scratch have produced in excess of 10 million interactive projects. Recent developments in information technology such as smart phones and improved networking infrastructure has enabled contributors to contribute on the go [2]. Millions of users have shared creative content in an online collaboration called Newgrounds [3]. In this project, we have created a framework that implements the concept of multiplayer using the Unity Game Engine [4]. Users can create 3D abstract sculpture in a dedicated space and share with other collaborating users.

II. BACKGROUND AND MOTIVATION

This work is motivated by the fact that people who work as a community are likely to perform better than working as individuals. In a study conducted by Burke and Burr on song writing, it was discovered that people who took advantage the social features of the collaborative website performed much better than the ones did not [5]. The participants were encouraged to interact and share song writing with others. Some users did not treat the website as a community and kept their work private. These individuals wrote half as many songs.

It is not only that the direct collaboration that stimulates creativity and productivity, stigmergic aspect of online collaboration works equally well [6]. In this study, an online collaborative tool called 'Picbreeder' was used. The applet randomly generates images which are modified by users and published online. The other users can download these images and then continue to further evolve. This way, users collaboratively creative very complex art without direct communication. In a sense, Picbreeder stimulates creativity but in a stigmergic sense.

There is a strong evidence of collaborative work in many diverse fields and professions. Cross discipline online collaboration enables designers and engineers with different backgrounds work together to produce innovative products [7]. An online collaboration tool enables designers and engineers to evaluate the design process and contribute to enhance product design. The outcome is a much-refined product since the contributors can visualise the product from different perspectives.

Creative thinking is becoming absolutely essential in all areas including engineering, medicine, art, entrepreneurship and education [8]. In this study, a web-based application was used for contributors to share ideas, comment and document the process. This shared area was seen as social gathering space, which positively influenced social and spatial psychological distance. A self-contained app also provided a framework to comment on other people's work and vote if necessary. The results showed an increase in personal and collaborative creativity.

Collaborative practice enhances user experience and exists in many fields. One example is a transmedia application that combines mobile augmented reality and with television content [9]. A typical television show encourages viewing children to interact with the broadcast and reveal missing words using the mobile app. The evaluation of this study was carried out by monitoring 6 to 10 year old boys and girls. It was reported that interactive technology enabled participants to provide immediate feedback to close the loop quicker than any other means. Players collaborated with each other, discussing which word or image to seek and encouraged each other finding them. This resulted in more richer and challenging engagement with an educational television programme.

Mobile devices such as tablets are being used more and more in school learning environment. In many cases, students only refer to pre-prepared content. However, collaborative content generation and sharing makes learning more engaging and fun. In a pilot study, an application called Media Enhanced Learning (MEL) was developed and employed on Android devices [10]. The application combines tutor generated pedagogical scripts and user generated multimedia content. 81% of the total participants indicated that they enjoyed learning with this app and would use this for their further studies. The most popular module was biology. Students enjoyed taking photographs of plants and shared with others on a common server.

III. IMPLEMENTATION

In our earlier work, we created an application to sketch shared content in the form of 2D graffiti. We used a head-mounted device to provide rich and immersive user experience inside the Tuscany demo scene by Oculus as shown in fig. 1.

Figure 1: Collaborative graffiti using a head-mounted display

In our current project, we have implemented the creation of abstract 3D art in a shared environment. Each user can utilise a dedicated working space to create a 3D sculpture. They will be able to tag their work to claim identity and share with other collaborative users. For implementation, we used the Unity game engine. Its current version has a new networking Application Programming Interface (API) that allows users to creative multiuser environment relatively easily. The application can be extended to mobile devices so that the participants can collaborate on the go. The networking architecture follows the server-client relationship as explained below.

Network Components

The unity networking system consists of a server and multiple clients. In the absence of a dedicated server, any client can act as a 'host'. The host consists of server and client within the same process as shown in fig 2. However, the client within the host is a special type of client called Local Client. The local client communicates with the server directly within the same process whereas the remote clients use the network to talk to the server. The benefit of this type of networking is that the local and remote clients share the same code. Hence in many cases, the developers need to write the code only once for both type of clients.

When new users are created on a network, they are seen by all clients connected to that network as shown in fig 3. Each individual user is capable of existing independently and can interact with other users. Unity's online services are hosted in the Cloud and provide a relay server to implement a server less configuration. This service routes traffic between users and eliminates Network Address Translation (NAT) and firewall issues.

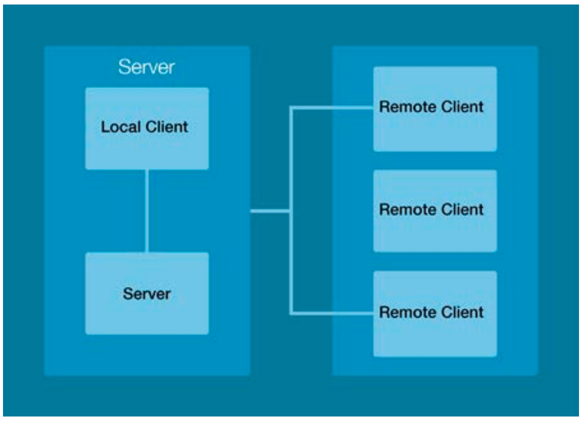

Figure 2: Server Client architecture

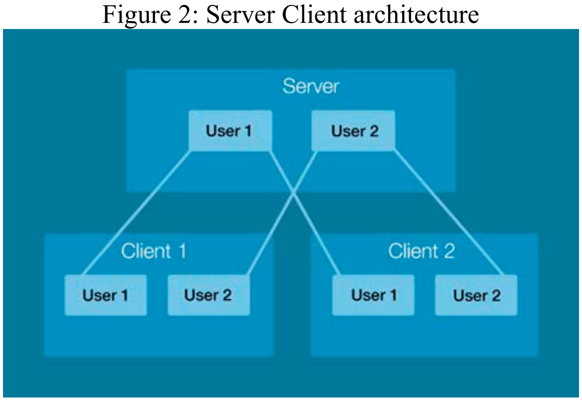

Figure 3: The server and clients see all connected users

In our implementation, each user runs an application on their device. One user acts as a server and has networks address. The subsequent users can all connect to this IP address as clients. On successful connection, they all have their own dedicated player and can also see the other connected users. When the application is launched, the user has the option to connect as a client or act as a server as shown in fig 4.

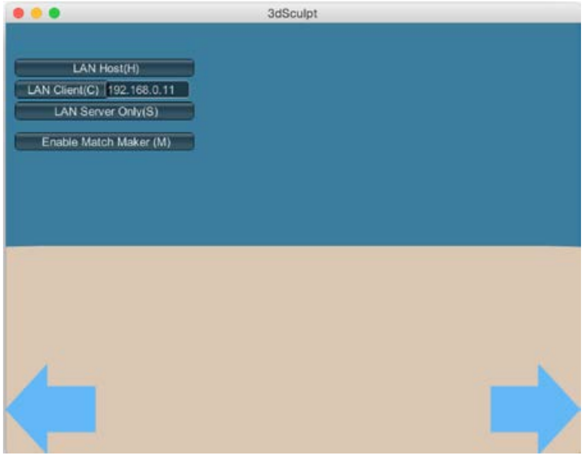

Figure 4: Application on launch

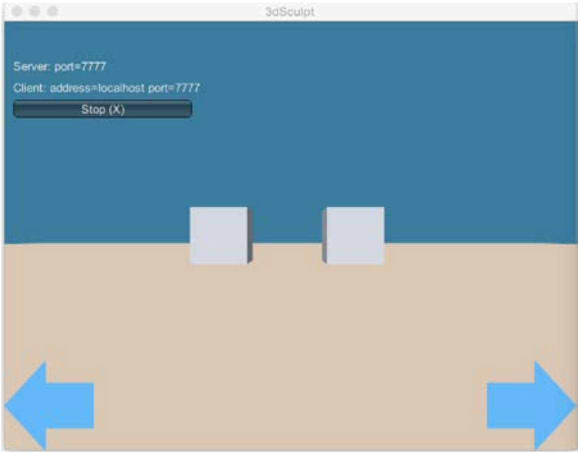

Figure 5: Server instance

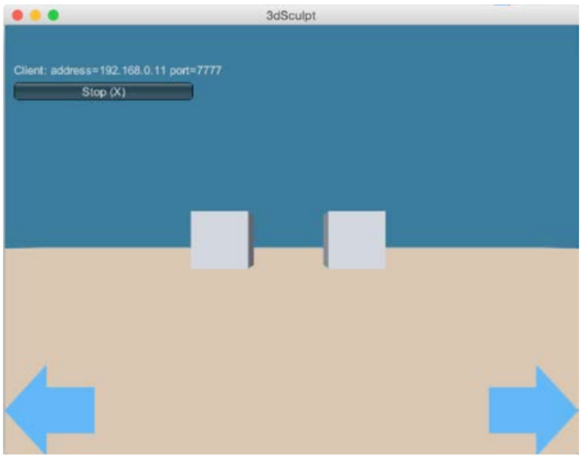

Figure 6: Client instance

In this example, only two users are connected as shown by two grey cubes. O Computing (JOC) Vol.5 No.1, August 2016nce instantiated, each user can click and create a series of blobs to construct a 3d Sculpture as shown in fig 7.

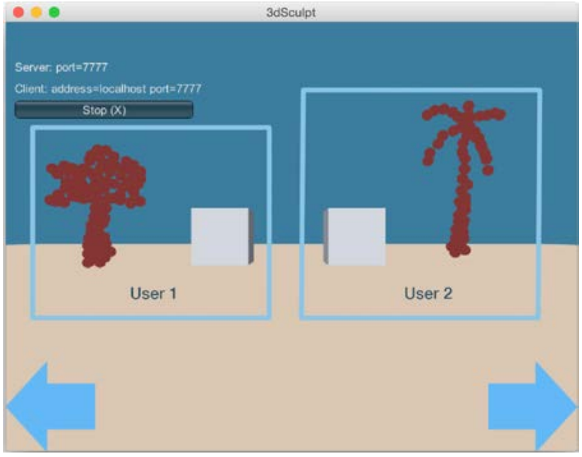

Figure 7: Two connected users

The client window also sees the exact same content as the host as shown in fig 8.

Figure 8: Client window

IV. Evaluation

A focus group comprising of students studying on a games design course was held to evaluate the effectiveness of this application. These students are familiar with the concept of 3D art and networked multiplayer implementation. The following criteria was used for evaluation:

Table 1: Evaluation criteria

Criteria	Response
This will allow me to share my ideas with others	93%
This will allow me to learn from other users	87%
This will stimulate creativity amongst participants	87%
I'd like to tag my creation to claim ownership	67%
I'd like to critique others' work	93%
I'd like to receive feedback and criticism on my own work	87%
	80%
I'd like to use this app on A mobile device/desktop	33%/87%

Although the size of the focus group is fairly small but the responses are very positive. The results of the focus group evaluation are shown in chart 1. The categories 'this will allow me to share my ideas with others' and 'I'd like to critique others' work' scored favourably well. The category that deals with the ownership did not score that well. Perhaps they would like to stay anonymous and still be part of a large community. Surprisingly, most people preferred to use the app on a desktop computer. Some general comments included:

- I'd like to see a refined version in conjunction with a VR device and a gesture-sensing controller.
- A cool idea perhaps can be a plugin for a 3D authoring package

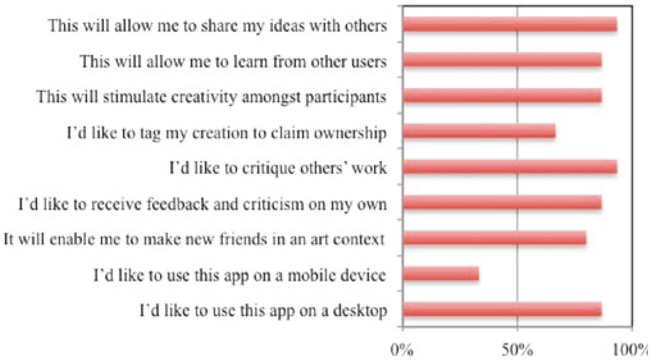

Chart 1: Focus group evaluation results

V. Procedural versus primitive geometry creation

Procedural modelling (PM) has recently gained a lot of popularity and has become a very hot research topic. PM is applied in a wide range of areas such as creating plants, buildings, terrains, roads and 3D art to name a few [11]. It is possible to generate beautiful terrains and other artefacts by adding some form of noise to procedurally generated geometry [12]. Fig 9 shows a deformable terrain generated by creating procedural cubes where perlin noise is added to introduce randomness.

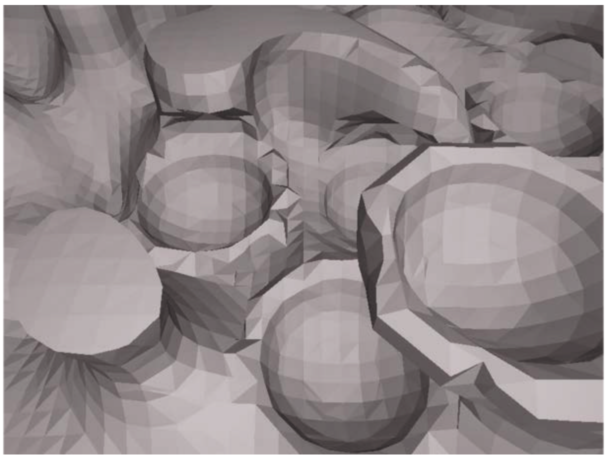

Figure 9: A deformable procedural terrain

This approach involves generation of cubes by first creating six faces of a cube. It is a very complex process, which involves creation of vertices, UVs, triangles, tangents, bi-tangents and normal as shown in fig 10.

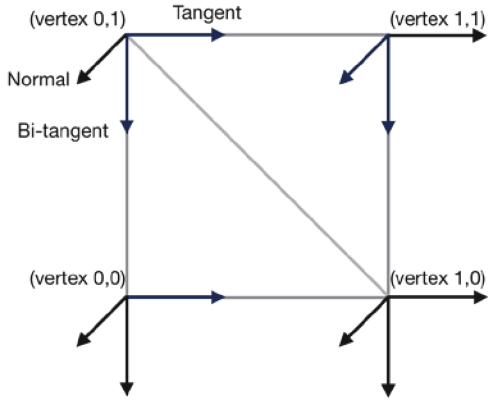

Figure 10: A procedurally generated face

As it can be seen, this process is very complex and requires generation of faces as shown in fig 10 and then combining 6 of such faces to generate one cube. The object creation phase hence is very CPU intensive. Fig 11 shows part of the profiler window focused on procedurally creation of a cube. Examining the method to implement this indicates that the CPU time taken to create a cube is 22.1%, which seems to be rather high.

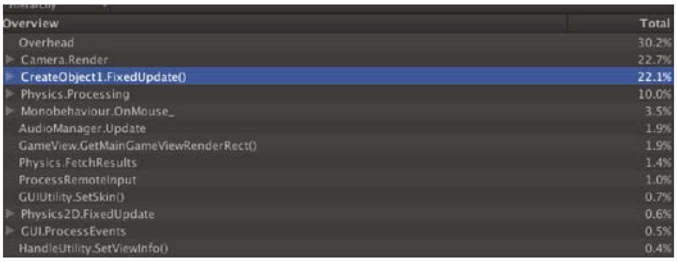

Figure 11: Performance analysis of procedural mesh creation

Although it is a short duration spike in CPU usage, it can add up to a large overhead if a user is creating a large number of objects. It will also result in longer network delays when drawing the same geometry on other connected client machines.

Alternative approach is to use a primitive object such as a cube and create cloned instances of it to create a desired structure. This process incurs much smaller overhead as compared to a procedural approach. Fig 12 shows part of the profiler window highlighting the performance analysis when instantiating a primitive. It is notable that the CPU usage to instantiate a primitive is only 1.5%. This analysis demonstrates that a procedural mesh creation results in over 1400% CPU usage as compared to its primitive instantiation counterpart. The compromise is that the meshes instantiated from primitives are not inherently editable. Additional methods can be developed should a user require implementing an editable mesh. This, however, will improve performance significantly since we are only modifying fewer meshes at runtime. This can be a valuable improvement especially if a client is connected using a mobile device.

Figure 12: Performance analysis of a primitive instantiation

VI. Saving and Loading Meshes

Whether the geometry is procedurally generated or instantiated, saving 3D objects is not possible. Since the transform coordinates of each object and other mesh parameters are known, these can be saved in a text file. This data can be saved and recalled using the Binary Formatter class. The data can be serialized and deserialized so that it is not human readable and hence not open to abuse. Since this data is primarily text, saving and loading does not incur heavy penalties. Fig 13 shows listing of such file where it is not possible to interpret data in any sensible way.

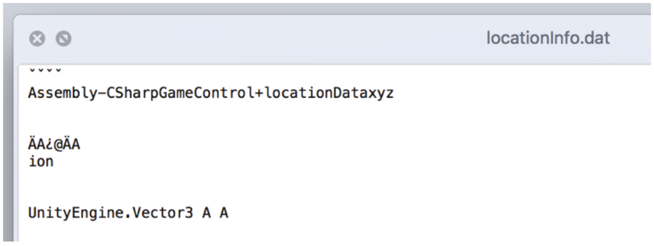

Figure 13: Serialized data file

VII. Conclusion and Further Work

We have created a networked prototype that enables users to create 3D abstract art and share with others. The app is implemented using a modern game engine called 'Unity' that allows us to create 3D art. We tested the app on a corporate networking infrastructure and did not experience any firewall issues. The focus group results show a willingness to use such application in order to share ideas with others, learn from others' experience, give feedback and enhance personal creativity. Procedural mesh creation versus primitive instantiation has also been explored. Results indicate that although procedural mesh creation has its benefits, it causes huge overheads in terms of CPU loading. It is felt that instantiating primitives is far more cost effective and should be used where possible. Mesh editing can be implemented on an ad hoc basis to allow users to modify geometry should the need arises. Saving and loading of sculptures can be implemented using the binary formatter class. We feel that there is a big potential to develop this concept further. Future work will include refining and optimising 3D mesh data. The evaluation will also be carried out on a much larger scale to include participants of varying demographic. Alternative methods of visualisation such as Virtual Reality (VR) displays can also be used to provide rich immersive experience. Emerging gesture sensing devices can be integrated with VR devices to provide innovative ways of interaction.

References

[1] Scratch Programming for All: http://scratch.mit.edu

[2] Benkler, Y. *The Wealth of Networks: How Social Production Transforms Markets and Freedom.* Yale University Press, New Haven and London, 2006.

[3] Newgrounds. http://www.newgrounds.com.

[4] Unity Game Engine - http://www.Unity3d.com

[5] M Burke and B Settles, "Plugged in to the community: Social motivators in online goal-setting groups." In *Proc. Int. Conf. Communities and Technologies (C&T)*, ACM (2011), 1–10.

[6] Jimmy Secretan. 2013. Stigmergic dimensions of online creative interaction. Cogn. Syst. Res. 21 (March 2013), 65-74

[7] T. Magal-Royo, B. Jorda-Albiñana, J. Gonzalez del Rio, O. Ampuero Canellas, J.L. Gimenez-López, Online Collaborative Environments in the Creative Process of Product Development for Engineering Students, Procedia - Social and Behavioral Sciences, Volume 51, 2012, pp 677-681, ISSN 1877-0428

[8] Adina Cocu, Emilia Pecheanu, Ioan Susnea, Stimulating Creativity through Collaboration in an Innovation Laboratory, Procedia - Social and Behavioral Sciences, Volume 182, 13 May 2015, Pages 173-178, ISSN 1877-0428

[9] Glenda Revelle, Emily Reardon, Kristin Cook, Lori Takeuchi, Rafael Ballagas, Koichi Mori, Hiroshi Horii, Hayes Raffle, Maria Sandberg, and Mirjana Spasojevic. 2015. Electric Agents: Combining Collaborative Mobile Augmented Reality and Web-Based Video to Reinvent Interactive Television. Comput. Entertain. 12, 3, Article 1 (February 2015)

[10] Janne Lahti, Erkki Siira, and Vili Törmänen. 2012. Development and evaluation of media-enhanced learning application. In *Proceedings of the 11th International Conference on Mobile and Ubiquitous Multimedia* (MUM '12). ACM, New York, NY, USA

[11] R. M. Smelik, T. Tutenel, R. Bidarra, and B. Benes, "A survey on procedural modelling for virtual worlds," *Computer Graphics Forum*, vol. 33, no. 6, pp. 31–50, 2014.

[12] Lagae et al., "State of the art in procedural noise functions," in *Computer Graphics Forum (Eurographics'10 State of the Art Reports)*, May 2010.

Bal Singh is senior lecturer at Leeds Beckett University in the School of Computing, Creative Technologies and Engineering in the faculty of Arts Environment and Technology. He plays a key role in the running of faculty's App Development Studio and games design course in the school. His skillset includes game design, digital asset creation and game programming. His research interests include interaction with virtual environments using real world signals and machine learning for the classification of neurodegenerative disorders. He is also research active in applying Virtual Reality in the field of art, medical and gameplay environments. He is a senior fellow of the Higher Education Academy.

Pauline de Souza is senior lecturer in the School of Arts and Digital Industries in the University of East London. She was awarded the Fellowship to the Higher Education Academy in 2011. As a programme leader for the Contemporary Art and Visual Media undergraduate degree in the Visual Theories Department based at Docklands campus, she has guided students to explore the historical and contemporary art scene. She has a keen interest in contemporary art.