MASHUPS IN SECOND LIFE @ VU

Anton Eliëns, Frans Feldberg, Elly Konijn, Egon Compter

VU University
Amsterdam

KEYWORDS
Second Life, Web Services, Mashups, Virtual Economy, Serious Games, User Tracking

ABSTRACT

In this paper we explore how to enhance our presence in Second Life by utilizing Web Services in meaningful compositions (mashups). After discussing the technical requirements that must be met, we discuss possible applications of mashups in Second Life, including serious games, and delineate a behavioral model that allows for tracking the behavior of visitors of our world. Taking our requirements analysis and envisioned design, which essentially includes the 3D nature of Second Life worlds, as a starting point, the paper provides an overview of research and development(s) that may contribute to the realization of meaningful mashups and serious games in Second Life.

INTRODUCTION

Second Life seems to be overtaking the world. In the whole range of cummunity-building platforms, Second Life stands out as an immersive 3D world with an almost stunning adoption, by both individuals, companies and institutions, followed attentively by the Press. Not entirely without an understanding of the value of press coverage, the VU University Amsterdam decided to create presence in Second Life, by creating a virtual campus, to realize a (virtual) community of learners, Eliens et al. (2007). And, indeed, we succeeded in being the first university in The Netherlands with presence in Second Life and, as hoped, this was covered in the 8 o'clock nation-wide TV news.

More substantial than getting into a nation-wide television broadcast, however, is our aim to communicate our institutional goals, *creating a community of learners*, by creating a virtual campus in Second Life, offering *an information portal* as well as *a meeting point*, in a media platform that is widely adopted by our target community. Virtual presence in Second Life, obviously, is not enough. The relatively long history of virtual worlds has shown that lack of interesting content and functionality easily leads to boredom, desinterest, and hence *churn*, users dropping off. As a consequence, there is a need for sustainable functionality, that both motivates people to come back and participate, and, otherwise why choose Second Life, makes essential use of the 3D immersive environment offered by Second Life. In this paper, we will explore how to use web services in meaningful compositions or mashups to enhance our presence in Second Life, and create a community where visitors actively participate in both education and research,

structure The structure of this paper is as follows. First, we will briefly describe the construction of our virtual campus and discuss the likely success factors of Second Life. and indicate how to use Second Life as a platform for interaction and serious games. We will investigate what technological support is needed to create mashups in Second Life and, after that, whether Second Life offers the functionality needed to incorporate web services. Further, we will characterize applications based on web services that make essential use of the 3D environment and fit within the virtual economy as it exists in Second Life, in particular serious games and (corporate) awareness systems. And, finally, in before giving our conclusions, we will sketch a behavioral model to characterize user interaction in Second Life, that may contribute to the realization of meaningful mashups and serious games in Second Life.

VU @ SECOND LIFE

What has been characterized as a shift of culture, from a media consumer culture to a participatory culture, Jenkins (2006), where users also actively contribute content, is for our institution one of the decisive reasons to create a presence in Second Life, to build a virtual platform that may embody our so-called *community of learners*, where both staff and students cooperate in contributing content, content related to our sciences, that is. In December 2006, we discussed the idea of creating presence in Second Life. Our initial targets were to build a first prototype, to explore content creation in Second Life, to create tutorials for further content creation, and to analyze technical requirements and opportunities for deployment in education and research. Two and a half months later, we were online, with a virtual campus, that contains a lecture room, a telehub from which teleports are possible to other places in the building, billboards containing snapshots of our university's website from which the visitors can access the actual website, as well as a botanical garden mimicking the VU Hortus, and even a white-walled experimentation room suggesting a 'real' scientific laboratory. All building and scripting were done by a group of four students, from all faculties involved, with a weekly walkthrough in our 'builders-meeting' to re-assess our goals and solve technical and design issues. The overall style is realistic, although not in all detail. Most important was to create a visual impression of resemblance and to offer the opportunity to present relevant infomation in easily accessible, yet immersive, ways. Cf. Bolter & Grusin (2000).

As we argue in Eliens et al. (2007), the surprising success and appeal of Second Life may be attributed to an optimal combination of avatar modification options, gesture animations, in-game construction tools, and facilities for communication and social networking, such as chatting and instant messaging. Incorporating elements of community formation, and very likely also the built-in physics and the inclusion of elementary economic principles, seem to be the prime distinguishing factors responsible for the success of Second Life. In addition, the possibility of recording collaborative enacted stories, using built-in *machinima*[1] certainly may contribute to its appeal.

The goal of this paper is to explore the use of web services for, among others, the creation of serious games, and to provide a behavioral model that allows us to give an interpreation to users' behavior patterns, and may perhaps even help to guide users' behavior in Second Life, by providing appropriate recommendations.

WEB SERVICES & MASHUPS

By now the phrase *Web 2.0* as well as applications representing it, such as Flickr and YouTube, are well established, and enjoyed by a wide community. Each day new items are added to the growing list of mashups[2], and the number of web services that constitute the building blocks of mashups also shows a steady growth. Mashups seem to be the easy way to start up a company, since the technology is relatively easy and, making use of appropriate services, initial investment costs can be low. Cf. Shanahan (2007).

What Web 2.0 stands for, from a technical perspective, is succinctly expressed in Dorai's Learnlog[3] *XML Is The Fabric Of Web 2.0 Applications*:

- client side is AJAX (Asynch. Javascript and XML)
- server application typically exposes data through XML
- the interaction model is web services
- mashups combine multiple webservices

And eventhough many alternative representations, such as JSON[4] (Javascript Object Notation) are increasingly being used, all in all XML may be regarded as the *interlingua* of the Web 2.0.

Before taking a closer look at the communication protocol(s) underlying Web 2.0 and de-construct the tight link of AJAX to HTML in-page formatting, it is worthwhile, following Shanahan (2007), to give an overview of a selected number of services, that may be used to create mashups:

services

- google – code.google.com/
- yahoo – developer.yahoo.com/
- del.icio.us – del.icio.us/help/api/
- flickr – www.flickr.com/services/
- bbc – www0.rdthdo.bbc.co.uk/services/
- youtube – www.youtube.com/dev

Although mashups featuring google maps seem to be the dominant mashup type, other services such as offered by del.ici.us, Flickr and BBC might prove to be more worthwhile for 'serious' applications. For example, for developing e-commerce applications Amazon[5] offers services for *product operations*, such as item search and similarity lookup, *remote shopping carts*, to create and manage purchase collections, *customer content*, to access information contributed by customers, and *third party listings*, to find related resellers. It is important to note that many of these services, as for example the *shopping cart* services, may be used independently of the commercial offerings of Amazon!

Most of the service providers and services mentioned above are accessible using a choice of protocols, including WSDL, SOAP, XML-RPC and the REST protocol. The REST protocol seems to be most widespread and as we will discuss in the next section, it seems to be tho most appropriate protocol in Second Life.

REST stands for *Representational State Transfer*. In essence, the REST protocol uses the url as a command-line for stateless RPC invocations, which allows for services to be executed by typing in the address box of a web browser. A great tutorial about the REST protocol can be found in Joe Gregorio's column[6]: *The Restful Web*. As fully explained in Van der Vlist et al. (2007), the phrases *representation*, *state* and *transfer*, respectively, stand for:

REST

- representation – encoding in a particular format
- state – data encapsulated in an object

[1] www.machinima.org

[2] www.programmableweb.com/mashuplist/

[3] dorai.wordpress.com/tag/mashups/

[4] www.json.org/

[5] aws.amazon.com

[6] www.xml.com/pub/a/2004/12/01/restful-web.html

- transfer – using HTTP methods

In practice, the use of REST means that the state associated with a resource or service must be managed by the client. Together with mechanisms such as content-negotiation and URL-rewriting, REST provides a simple, yet powerful method to invoke services using HTTP requests.

The Web 2.0 offers a lively arena for consumers and developers alike, with a multitude of blogs discussing the future of the web. For example, in Dion Hinchcliffe rebuttal[7] of Jeffrey Zeldman's *Web 3.0 Web 1.0 = Web 2.0* blog, entitled *Is Web 2.0 Entering "The Trough of Disillusionment"?* it is suggested that *our services could even be more powerful* by creating *semantic mashups*[8].

To conclude this brief overview of web services and mashups we wish to give another quote from Dorai's Learnlog, this time from Jon Udell, in his blog on his move to Microsoft:

> *The most powerful mashups don't just mix code and data, they mix cultures.*

which provides a challenge that trancends all issues of mere technological correctness.

INFRASTRUCTURE

Second Life offers an advanced scripting language with a C-like syntax and an extensive library of built-in functionality. Although is has support for objects, LSL (the Linden Scripting Language) is not object-oriented. Cf. Eliens (2000). Scripts in Second Life are server-based, that is all scripts are executed at the server, to allow sharing between visitors. Characteristic for LSL are the notions of *state* and *eventhandler*, which react to events in the environments.

Among the built-in functions there are functions to connect to a (web) server, and obtain a response, in particular (with reference to their wiki page):

built-in(s)

- request – wiki.secondlife.com/wiki/LlHTTPRequest
- escape – wiki.secondlife.com/wiki/LlEscapeURL
- response – wiki.secondlife.com/wiki/Http_response

Other functions to connect to the world include *sensors*, for example to detect the presence of (visitors') avatars, and chat and instant messaging functions to communicate with other avatars using scripts. In addition, LSL offers functions to control the behavior and appearance of objects, including functions to make objects react to physical laws, to apply force to objects, to activate objects attached to an avatar (as for example phantom Mario sprites), and functions to animate textures, that can be used to present slide shows in Second Life.

On the Mashable[9] *Social Networking News* site a brief overview is given of the use of web services in Second Life, entitled *Second Life + Web 2.0 = Virtual World Mashups.* To access Second Life from outside-in (that is from a web browser), so-called *slurls* may be used, for example to reach VU[10] @ Second Life, and all slurls listed in del.icio.us under *slurlmarker*[11] may be used, also to activate in-world teleporting using scraping techniques.

As remarked in the *hackdiary*[12] by Matt Biddulph, Second Life (currently) lacks the ability to parse XML or JSON, so the best way to incorporate web services is to set up a web server with adequate resources. As Matt Biddulph indicates, to access *flickr* photographs for a particular user (avatar), a web server may contain the following resources:

web server

- /seen?user=SomeAvatar – records the presence of SomeAvatar
- /touched?user=SomeAvatar – invokes flickr API with users tag
- /set_tag?user=SomeAvatar&tag=FavoriteTag – records SomeAvatar's favourite tag

For example, in response to a 'touch' event, invoking *touch* results in consulting the database for the user's tag and asking the Flickr API for a random photo with that tag. It then returns a string containing the url for a particular photograph. LSL functions used in this application include *sensors*, to check for presence, *listen* functions, to respond to spoken commands, and *touch* events, for the physical interface. In addition to supporting strings and lists, LSL provides a perl-like split function to convert a string into a list of strings, thus allowing for processing multiple items in response to a server request.

Another example of using web services in Second Life is writing blogs[13] from within Second Life using the BlogHUD[14] developed by Koz Farina who also is reported to have found a flash hack that allows for reading RSS feeds.

The RSS display uses the ability to stream Quicktime video in Second Life, and again the mashup is not created in Second Life but by appropriate server support.

In a similar vein we may incorporate live streaming video[15], for example by using WireCast[16] to capture and organize live camera input, possibly together with screen output of other applications such as *powerpoint*, which must then be sent to a streaming server supporting

[7] web2.sys-con.com/read/172417.htm

[8] www.web2journal.com/read/361294.htm

[9] mashable.com/2006/05/30/second-life-web-20-virtual-world-mashups/

[10] slurl.com/secondlife/VU%20University%20NL/29/151

[11] del.icio.us/tag/slurlmarker

[12] www.hackdiary.com/archives/000085.html

[13] nwn.blogs.com/nwn/2006/10/really_simple_s.html

[14] bloghud.com/

[15] blogs.electricsheepcompany.com/chris/?p=206

[16] www.varasoftware.com/products/wirecast/

Quicktime, such as Apple's Darwin[17], which may then be accessed from Second Life to texture a display object. Finally, as another *Web 2.0 to Web 3D* phenomenon, announced in New World Notes[18], we may mention the used of Twitter[19] messages, that allow residents to send and receive message about ongoing activities. A similar service is reported to exist for *jaiku*[20] messages.

VIRTUAL ECONOMY

Mashups on the Web are interesting representatives of what one may call a *virtual economy*, with a business-model that is not grounded in traditional *production* and *trade* values, but rather consists of value-added services with an indirect, albeit substantial, financial spin-off, due to recommendations and referrals. The basic mechanisms in a recommender economy are, according to Kassel et al. (2007):

- cross sale – users who bought A also bought B
- up sale – if you buy A and B together ...

Where the principles underlying this virtual economy have definitely proven their value in first (ordinary) life economy, what are the chances that these principles are also valid in Second Life?

According to the media companies selling their services to assist the creation of presence in Second Life, there are plenty *New Media Opportunities In The Online World Second Life*[21], to a possibly even greater extent, as they boldly claim, as in what they call *the predessor of Second Life, the World Wide Web.*

To assess the role web services, including semantic web services, may play in Second Life, it seems worthwhile to investigate to what extent web services can be deployed to deliver more traditional media, such as *digital TV*. To support the business model of digital TV, which in outline may be summarized as *providing additional information*, *game playing* and *video on demand*, with an appropriate payment scheme, Daskalova & Atanasova (2007) argue in favor of the use of a SOA (Service Oriented Architecture), to allow for a unified, well-maintainable approach in managing collections of audio-visual objects. Such services would include meta-data annotation, water-marking for intellectual property protection, and search facilities for the end-user.

With respect to the application of web services in Second Life, however, a far more modest aim, it seems that nevertheless the business model associated with the delivery of media items through digital TV channels may profitably be used in Second Life, and also the idea of wrapping media items in web services has in some way an immediate appeal.

Leaving the economic issues aside we will briefly consider two applications that we envisage to realize within the virtual campus of VU @ Second Life. The first application is Clima Futura[22], a game meant to give information about climate change.

Technical issues in realizing Clima Futura in Second Life are support for ranking, as well as meta-information with respect to locations where relevant information can be found, which may be realized with the techniques indicated previously. Another issue is giving flexible access to video material related to specific topics in climate change.

The other application we wish to discuss is PANORAMA, Vyas et al. (2007). In developing PANORAMA we proceeded from the assumption that people somehow like to have a feel of what is going on in the workspace, although not in any detail, and also like to see items of personal interest, including (their own) birth-announcements and sport troffees.

Embedding PANORAMA in Second Life would allow us to observe, in more detail than in a previous user study, the behavior of users, that is, to be more precise, the proximity to particular objects of interest, the duration of their presence, and, using the mechanisms of recommendation, their interest in related items. Technically, such monitoring can be achieved using the sensors and listeners described before. To make sense of such data, however, we need some model that allows for an interpretation that is more meaningful than the mere registration of presence.

TRACKING INTERACTION

Our virtual campus in Second Life already allows for performing simple statistics, by recording the presence of users at particular spots in the virtual world, using sensors and listeners installed in 3D objects. Since the LSL script-based counters appear to be rather volatile, tracking data are sent to a web server and stored in a database. This mechanism can easily be extended to a more encompassing form of user tracking, recording for a particular user not only presence at particular spots, but also the duration of presence, the actual proximity to objects, and the proximity to other users, as well as explicitly spoken comments or actions such as the donation of (Linden) money. For example, observing that a user spends a particular amount of time and gives a rating r, we may apply this rating to all features of the item, which will indirectly influence the rating of items with similar features.

This does of course not explain nor how ratings come into existence, nor what features are considered relevant, or even how guided tours should be generated. However, as we have demonstrated in Ballegooij & Eliens (2001), based on a rudimentary tagging scheme, we may in

[17] developer.apple.com/opensource/server/streaming/

[18] nwn.blogs.com/nwn/2007/03/post_1.html

[19] twitter.com/

[20] devku.org/docs

[21] www.youtube.com/watch?v=8NOHRJB9uyI

[22] www.climafutura.nl

response to a query generate a guided tour taking the topographical constraints of the virtual world into account, for example to make a user familiar with the (virtual replica of the) actual workspace. It seems that this approach can be generalized to one that uses alternative descriptive methods, as long as they support feature-based information retrieval[23].

Obviously, both user tracking and recommendations may be fruitfully used in the realization of serious (corporate) games, as well as to support exploratory activity in non-serious games and (corporate) awareness systems.

CONCLUSIONS

Based on an overview of research and development(s) in web technologies and our assessment of the technical facilities offered by the Second Life platform, we may conclude that there are ample opportunities to incorporate web services and mashups in Second Life. Our intended applications, moreover, covering (corporate) game playing as well as a system for promoting (corporate) social awareness, indicate that there are clear motivations to deploy web services in Second Life, both for tracking users' behavior and for providing additional information based on recommendations that may be derived from taking record of users' behavior patterns. Although we have sketched a first behavioral model that allows to assign meaning to behavior and interaction, it is clear that this model must be further refined and that we need to gain experience in developing mashups in virtual space to arrive at effective and meaningful compositions of web services, supporting the realization of (serious) games, in Second Life.

Acknowledgement(s) We (again) thank the students involved, Viola van Alphen (FEWEB), Tom Bestebreurtje (FEW), Elbert-Jan Hennipman (FEW), and last but not least Bjorn de Boer (FSW), for their enthusiasm, creativity, and of course their hard work which led to an amazing result.

REFERENCES

Atanasova T., Nern H.J., Dziech A. (2007), Framework Approach for Search and Meta-Data Handling of AV Objects in Digital TV Cycles, Workshop on Digital Television, Proc. EUROMEDIA 2007, Delft, Netherlands

Ballegooij A. van and Eliens A. (2001), *Navigation by Query in Virtual Worlds*, In: *Proc. Web3D 2001 Conference*, Paderborn, Germany, 19-22 Feb 2001

Bolter J.D and Grusin R. (2000), *Remediation – Understanding New Media*, MIT Press

Daskalova H. and Atanasova T. (2007), Web Services and Tools for their Composition considering Aspects of Digital TV Workflow, Workshop on Digital Television, Proc. EUROMEDIA 2007, Delft, Netherlands

Eliens A. (2000), *Principles of Object-Oriented Software Development*, Addison-Wesley Longman, 2nd edn.

Eliens A. and Chang T. (2007), *Let's be serious – ICT is not a (simple) game*, In:*Prc. FUBUTEC 2007*, April 2007, Delft

Eliens A. Feldberg F., Konijn E., Compter E. (2007) , VU @ Second Life – creating a (virtual) community of learners, In *Proc. EUROMEDIA 2007*, Delft, Netherlands

Jenkins H. (2006), *Confronting the Challenges of Participatory Culture: Media Education for the 21th Century*, White Paper, MIT MediaLab

Kassel S., Schumann C-A. and Tittman C. (2007), Intelligent Advertisement for E-Commerce, In *Proc. EUROMEDIA 2007*, Delft, Netherlands

Nern H.J., Dziech A., Dimtchev and Jesdinsky (2007), Modules for an Integrated System Approach for Advanced Processing of AV Objects in Digital TV Workflow, Workshop on Digital Television, Proc. EUROMEDIA 2007, Delft, Netherlands

Rymaszewski M., Au W.J., Wallace M., Winters C., Ondrejka C., Batstone- Cunningham B. (2007). *Second Life – the official guide*, Wiley

Shanahan F. (2007), *Amazan.com Mashups*, Wiley Publishing Inc.

Van der Vlist E,, Ayers D., Bruchez E.,, Fawcett J. and Vernett A. (2007). *Professional Web 2.0 Programming*, Wiley Publishing Inc.

Vyas D., van de Watering M., Eliens A., van der Veer G. (2007), *Engineering Social Awareness in Work Environments*, accepted for *HCI International 2007*, 22-27 July, Beijing, China

[23] www.cs.vu.nl/∼eliens/research/rif.html