

Second Life

A model for applications - Generic Web support for serious games in Second Life and beyond.

Tom Bestebreurtje (1203932)

FEW

VU University
Amsterdam

November 2007

Abstract

Second Life⁽¹⁾, an online 3D virtual community developed by Linden Research, Inc.⁽²⁾, came to international mainstream media attention in 2006. Since then its user base has roughly doubled every six months. With a greater number of users, the number of applications of a community with a varied user base also increases. These applied uses of Second Life can be analyzed and placed in a two dimensional model where the axes correspond to the amount of rules and the measure of entertainment involved in the application. We will find that Second Life is still mostly used for entertainment purposes and less for constructive ends.

Furthermore, the recent distribution of the client software under the GNU GPL⁽³⁾ allows for greater customization of the client, like communication with third party packages. We will explore some of the possibilities of using modified Second Life clients in combination with a separate back-end to facilitate communications not supported by the Second Life servers. The implications of an important future step, open source server software, are also worth looking into.

We will conclude with a brief overview of the future of Second Life. An appendix on developing the open source viewer software is also presented.

Keywords: Second Life, Open Source, Web Services Architecture, Model of Second Life Applications, Mash-ups, developing Second Life software.

Abstract	1
Introduction.....	2
Creating a model: Categorising the applications of Second Life	3
Open source client code	6
Open source server code	7
Integrating Web 2.0 services into Second Life: Mashups	8
Minimal requirements for web services in Second Life	10
Conclusion	12
Appendix: developing the Second Life viewer.....	14
References.....	16

Introduction

Second Life (abbreviated 'SL') is an online 3D virtual world that was launched in 2003 by Linden Research, Inc., also known as "Linden Lab." The program came to international mainstream media attention in mid-2006. This was largely due to the fact that numerous large companies decided to present themselves in this virtual world that seemed to be all the rage in order to gain more attention from their target audiences and to show that their company was a modern one that kept up with the latest social habits of the mainstream^{(4, 'Leading a double life') (5, 'Get a Second Life')}.

Second Life users (residents) can create objects when they are logged on. They retain property rights to their creations and are free to copy them, give them to other residents. Objects that are transferred to other residents can be modified by new owners, but only if the original creator has set the permissions of the object accordingly. They can also sell their creations or allow next-owners to sell them.

Second Life has its own economy based on its own currency, the Linden Dollar (L\$). L\$ that are earned by residents by selling property or exchanging US Dollars for Linden Dollars. The exchange rate is fairly stable around 265 L\$ per US\$1.00^(6, LindeX Market Data). In 2006 the GDP of Second Life was 64 million US\$. Estimates for 2007 are between 500 and 600 million US\$, almost tenfold last year's GDP.

As with the rest of the Internet, Second Life users come from many different stations and belong to a wide range of subcultures, almost all of which are in some way represented in Second Life. Different users have different needs and thus use Second Life each for their own purposes. These user groups and their purposes with Second Life will be described later.

The world of Second Life, also called "the grid," is hosted entirely on servers run by Linden Lab. By default, users only have to download and install client software locally. Since the entire grid (all avatar data, objects, landscapes, textures and texts) is hosted by Linden Lab, this means that the content of Second Life can only be modified through the Second Life client software. The recent move to distribution of the client software under the GNU GPL has made it easier for third parties to develop their own modified versions of the client software and special tools for Second Life. Another big step that goes even further is that in the near future, Linden Lab plans to make the server code open source as well. The implications of such a decision are widespread and will have a profound impact on Second Life as it is currently known. The issues surrounding open source client and server software for Second Life will be discussed.

VU @ Second Life: joining a hype?

In December 2006 I was asked to participate in creating a presence in Second Life for the VU. The aim of this project was to make the VU the first Dutch university present in Second Life. On February 28th the virtual VU campus went live^{(7, 'VU @ Second Life -- creating a (virtual) community of learners')}.

The project was a good way for myself to get more involved with Second Life; the possible applications are numerous and building a virtual campus is a great project to see what is and is not possible with building and scripting in Second Life.

The project can definitely be considered a success, but after spending time in Second Life, I prefer to watch the evolution of Second Life from the sidelines. I do not consider myself an active user of the software; for now it suffices to follow the news surrounding the 'hype,' what Second Life is often called, and wait eagerly for big

developments, like the rolling out of the new physics engine^(8, Havok 4 Beta), and the release of open source server code. I find the observing and studying of the users and uses of Second Life an interesting subject, and important in order to assess its usability for various purposes.

Creating a model: Categorising the applications of Second Life

Just like the group of Internet-users is a subset of the human population, the group of Second Life users is a subset of the users of the Internet. Users on the Internet are of different stations and belong to one or more subcultures depending on their activities on the Internet. This is also the case with Second Life. Though the amount of uses differs between the Internet and Second Life, they can still be distinguished in each environment.

Most of the Second Life users begin their Second Life experience by just walking around, exploring the world and enjoying the 3D scenery. These are the casual users who, for now, walk about with no real goal in Second Life other than to enjoy themselves. Users that want to achieve more in Second Life, if only to earn more L\$, start building their own objects and selling them. This can become more than just a hobby if a resident wants to make a considerable amount of money. Statistics show that out of the 8 million residents a number around 40.000 residents and rising has a Positive Monthly Linden Dollar Flow. This profit ranges from 0-50 US\$ for 30.000 residents to more than 2.000 US\$ for a select group of users of around 400^(9, Second Life's first millionaire).

Real-world companies that want to reach their target audiences through Second Life invest in virtual property and, for instance, sell virtual versions of their sports shoes to residents.

Second Life is also a place where scholars of different disciplines come to study. Students that are interested in 3D graphics use the built-in creation tools to create objects. Students of social sciences conduct social or psychological experiments similar to those in the real world. On the same note, economic students study the workings of the Second Life economy and how it applies to the real-world economy. The basis is that Second Life functions as a sandbox for experiments that are not quite manageable in the real world.

Second Life is often called a game, and its residents have disputed this because there are essentially no rules imposed on the residents except the rules that make the virtual world function. These rules include physics (though a house can be floating in mid-air if its creator so chooses, or can fall and slide down a hill for that matter) or restrictions on areas that are not open to the general public. Second Life in itself has no competitive element, no hall of fame, though the amount of fortune in L\$ can make people famous in the virtual world. An avatar's looks come from the clothes, hairstyles and accessories that avatar wears and uses. These give a resident some sort of status, but money is less of a factor here than in real life, since all these items can be easily created or obtained if a user has enough experience with the program and knows the locations in which to find the good stuff. The fact that gathering these items is not a common goal in Second Life means that users can do absolutely nothing and still enjoy the world.

Some areas of Second Life have extra rules that visitors must adhere to. These areas are made by residents who want to use Second Life as a game, a 3D shooter for instance, where statistics like health and ammunition have been scripted and where residents 'die,' only to find that nothing has really happened once they choose to exit the game area. So in essence, if Second Life is not technically a game, these areas are^(10, Second Life Games). These Second life Game engines run a separate PHP backend with an SQL database on an external WWW server for character data and statistics^(11, Weapons of Choice).



Figure 1: Lost City of Angels, a Second Life game

Continuing, some areas have rules not centered on the 'physical' interaction of bullets and bodies, but rather on the social interaction of residents. Role Playing Games (RPG's) exist in many different forms and places, likewise they exist in Second Life^(12, Samarkand RP kicks off). Here residents use their self-created or L\$-bought clothes and indulge in all kinds of role play, from medieval fantasy to science fiction to more niche-like fantasy settings, out of which virtual erotica is perhaps not so niche-like, but well, no one wants to admit to having had cyber-sex.

The diversity of the Second Life users gives way to a wide range of applications: the reasons or goals for which Second Life is used. There are casual residents who just hang out and make friends in the virtual world, or meet existing friends in the virtual world, and who maybe build some simple objects for their own uses (because nobody wants to pay for a hat that looks like a box). Then there are students who spend time in Second Life working with the building tools or conducting social or economic experiments. In other cases, employees work on the virtual presentation of their company in Second Life in order to gain the attention of the masses. Gamers who either engage in action-like games with guns that they designed themselves or gamers who act in RPG's. People also use Second Life to act out whatever fantasy they have that cannot easily be acted out in real life.

When looked at, the different activities within Second Life can be categorised in different ways. It depends on the viewpoint. One viewpoint can be obtained by looking at the number of rules that are involved in the activity. Is the resident free to

do whatever he wants, or are there restrictions in conduct and/or movement? A casual user can walk go wherever he wants to, talk to whomever he wants, and build whatever he wants. There a no rules whatsoever that restrict him. A resident that participates in an RPG on Second Life is often a member of a group of some kind. He is expected to wear clothes that match the setting of the game and adhere to a certain code of conduct. These restrictions make the RPG what it is, and failure to follow these rules makes the experience less enjoyable for the other players.

Another viewpoint is the measure in which the activity has a goal that extends beyond Second Life. When the casual resident explores the world of Second Life this does not affect his activities in the real world. It only costs him the time that he uses to be entertained by logging on to Second Life. Likewise, the RPG gamers may have rules to enhance their experiences in Second Life, but in the end it is still a form of entertainment that does not grant any reward outside the world of Second Life. A resident who logs on and learns to create 3D objects can use these skills outside of Second Life in 3D modeling tools. If this user uses the learned skills to create objects that other users actually want to buy he can even earn L\$ and make a profit. The skills learned and the profit, which can be converted from L\$ to US\$ are useful outside of Second Life.

To get an overview of the different applications of Second Life by its users these two viewpoints can be used make a 2-dimensional model where the two axes correspond to the two viewpoints:

Lighthearted vs. Serious:

- A lighthearted, playful application imposes no rules over the user's behaviour.
- A serious application typically has rules of conduct and an expected behaviour by the user.

Entertaining vs. Rewarding:

- While entertainment can be a reward in itself, here we mean that an entertaining application does not yield any rewards other than entertainment of the user.
- Rewarding applications teach the user skills or earn him a profit that can be used outside of Second Life. Whether or not an activity is rewarding sometimes depends on the user's intent. If a user spends time learning to build amazing objects in Second Life but does not want to sell them to make a profit, then the activity was more entertaining that rewarding. Also, some rewarding tasks can be very fun/entertaining to perform. In the case of this model we will primarily measure the reward that is associated with the application.

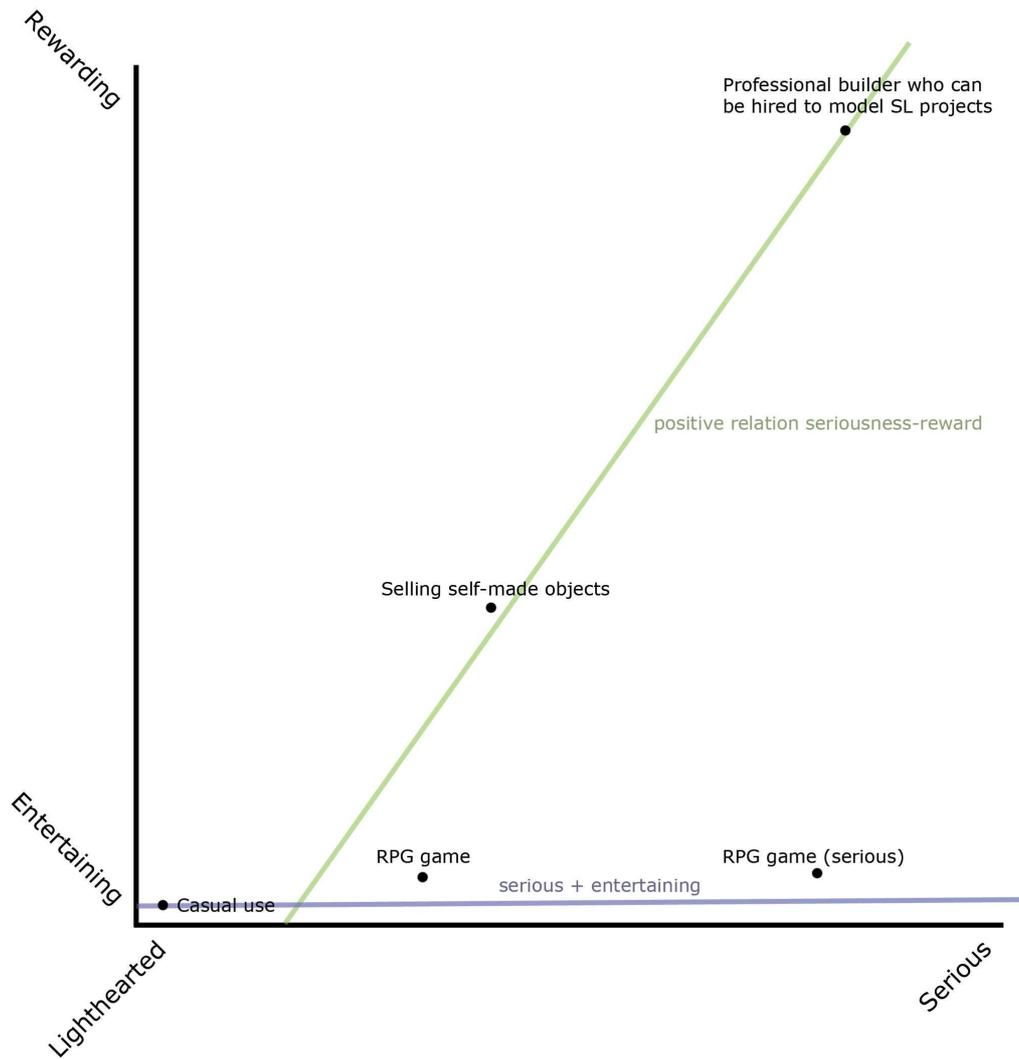


Figure 2: A model for applications of Second Life

This model (figure 1) is more relative than absolute. The reason is that one very important value, the reward, cannot be measured exactly, since some rewarding tasks, like the educational aspect or virtual decision making processes cannot be expressed in terms of money. There is of course a kind of diagonal upward line, shown in green, which shows that more work means greater rewards.

Some gamers that use Second Life freely choose to make Second Life as a game more serious to them. This does not in itself grant any reward, thus that line, shown in blue, remains horizontal. The opposite, a trend that rewards greatly for the mere entertainment derived from the Second Life experience does not exist, because there's just no such thing as a free lunch.

Open source client code

In January 2007, Linden Lab made the Second Life client source code available to the public for use within the guidelines of the GNU GPL version 2. This makes it possible for third parties to start developing and distributing their own versions of the client

software. If Linden Lab approves the added features of these modified versions they are incorporated in the next official build.

One such project, called libsecondlife⁽¹³⁾, started half a year before the code was made open source, but was already supported by Linden Lab. Libsecondlife is a library that was made to be used in third party applications which communicates with the Second Life simulation servers. It gained much attention for its use in the controversial CopyBot, originally developed by the libsecondlife developers as a debugging tool⁽¹⁴⁾. Using libsecondlife, CopyBot can duplicate any object, texture or even a complete avatar from the world, which is problem in a world where objects remain the intellectual property of their original creators and are sold for L\$.

Suffice to say that the transition to open source software is a big step, and that this copyright infringement is one of the biggest problems Linden Lab has to face, because in the end all object data that is processed by the client to be viewed on-screen can be duplicated.

Open source server code

Making the client code open source has been a big step for Linden Lab, but a necessary one, because they plan on making the simulator server code open source in the future as well.

In the current situation, where Linden Lab manages the Second Life servers, all object data is stored on their servers, making up the Second Life grid. Open source server code would allow users to start hosting their own private simulator servers and, more importantly, give them control over whoever is present on their private server, leaving owners free to duplicate, store and modify content that is created or passed through their private servers.

Some of the comments made by Linden Lab at the 'Platforms and Technologies Panel' for the Virtual Worlds Conference 2007^(15, VW07 panel) about the future of Second Life clearly indicate that the growing user base is beginning to become too big a load for on single server grid:

"We'll be delivering assets in a totally different method that won't be such a burden on the simulators. Very soon we'll be updating simulators to support multiple versions so that we don't have to update the entire Grid at once."

The obvious solution to this scalability issue is a distributed server architecture. This will ultimately lead to Second Life being a fully distributed 3D virtual world^(16, 'Second Life to open-source grid; Will Google bite?'), much like the Internet in it's current form, with servers all over the (real) world.

"We'll be open-sourcing the back end so sims can run anywhere on any machine whether trusted by us or not."

The issue of trust and security is important, but Linden Lab states that they are not to be seen as a kind of police force for Second Life, but rather as providers of services. In the end, as with the rest of the Internet, responsibility always lies with the user.

“Second Life cannot truly succeed as long as one company controls the Grid.”

--Joe Miller, VP for platform and technology development at Linden Lab

Linden Lab will no longer be the only company that sells virtual land. Selling land is of course merely another way of saying they rent out server processing power. With an open standard for Second Life Simulation Servers, other companies will have the possibility to compete. These other companies will all sell grid-hosting services, much like one can buy web hosting for the World Wide Web. Linden Lab is likely to become part of the development of standards for the whole of Second Life, much like what the W3C is for the World Wide Web.

Integrating Web 2.0 services into Second Life: Mash-ups

A popular trend with Second Life is the merging of existing web services with the virtual world. These mergings are called mash-ups and they are generally seen as the latest possibility to use Web 2.0 content in Second Life⁽¹⁷⁾, ‘Web 2.0 + Second Life = Virtual World mash-ups!’). Mash-ups have become possible since as of version 1.10, the Second Life client’s Linden Scripting Language (LSL) got its own HTTP request feature⁽¹⁸⁾, which can be used to request information via HTTP. So while mash-ups generally indicate World Wide Web content that is displayed in Second Life, some work the other way around, displaying Second Life content on the web, without using the Second Life client software. In general it could mean any communication of information from the World Wide Web to Second Life or vice versa. What is presented next is a list of mash-up applications currently developed. The flow of information is indicated in the figures. Some of these tools were written by Linden Lab itself.

Displaying photos from Flickr in Second Life

Using the LSL llHTTPRequest feature, Matt Biddulph from *Hackdiary*⁽¹⁹⁾ has succeeded in displaying photos from photo sharing website Flickr⁽²⁰⁾ in Second Life, Albeit through an external web server that makes the invoking of the Flickr API more manageable.

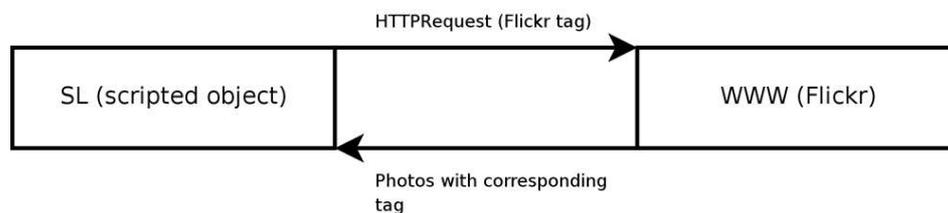


Figure 3: data flow (LSL script - flickr)

SLurl

The SLurl⁽²¹⁾ (Second Life url) was developed by Linden Lab and allows teleportation to in-world locations from the World Wide Web. A SLurl points to a location within Second Life. It consists of a region name and x-y-z-coordinates:

`http://slurl.com/secondlife/<region>/<x-coordinate>/<y-coordinate>/<z-coordinate>`

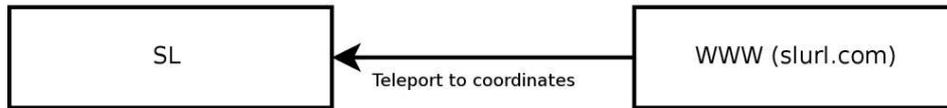


Figure 4: data flow (Second Life - slurl)

Slurlmarker

Slurlmarker combines the functionality of del.icio.us⁽²²⁾ and Second Life. It is a Second Life tool that enables its user to bookmark locations as SLurls. These bookmarks (SLurls) are stored in del.icio.us as URLs with the tag Slurlmarker and can be teleported to if the Second Life client is installed.

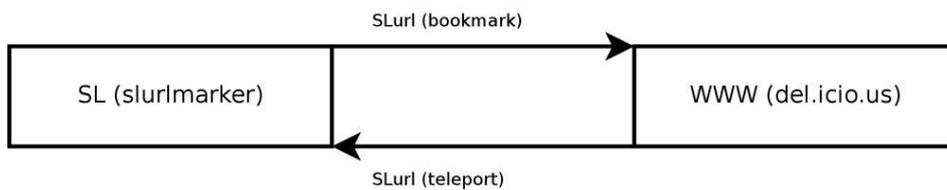


Figure 5: data flow (slurlmarker object - del-icio-us)

Webmap API

The Webmap API⁽²³⁾ is a project of Linden Lab that is currently still in beta stage. Using JavaScript, an interactive map of the grid can be viewed. This map can contain markers with an attached image of the location in a window.

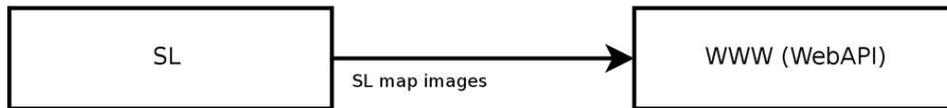


Figure 6: data flow (Second Life servers - Webmap API)

Regapi

Like the Webmap API, the RegAPI⁽²⁴⁾ is another one of Linden Lab's beta projects with the goal to allow users to implement Second Life account creation on their own website. The site's owner can track these accounts' avatars. RegAPI essentially creates a unique Second Life name and password, along with an optional starting location for the avatar.

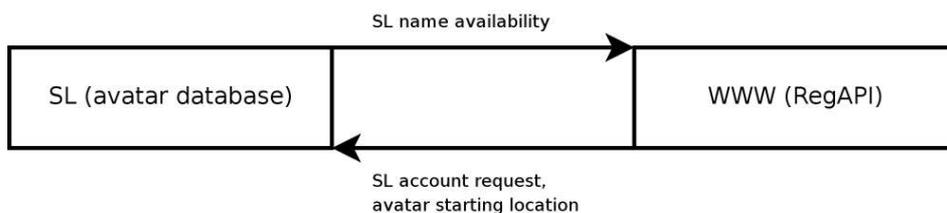


Figure 7: data flow (Second Life login server - Registration API)

Mash-ups are not just seen in Second Life. Google maps is often used in mash-ups on the World Wide Web. Websites like Wikipedia and Panoramio are integrated in Google Earth. The integration of Panoramio and Google Earth was so successful that Google acquired Panoramio in May 2007⁽²⁵⁾.

Several other web services could produce mash-ups in Second Life^(26, Eliens et al, 2007). Mash-ups will be playing a major part in expanding the functionalities of Second Life in its current state. The Second Life grid still lacks the necessary computing power for possible high-end applications like multiple streaming videos and real-time action game environments, but with the future move to an open standard for Second Life server code, specific demands from these applications could be met with dedicated private servers. This would mean that, as with the World Wide Web, certain areas would have lower production values and thus less visual appeal and performance.

Minimal requirements for web services in Second Life

With open source client software, every functionality possible can in principle be implemented. Whatever the goal, one just has to modify the client software to achieve it. A modified client can communicate with external servers, it can communicate with other instances of that client in a P2P fashion, anything is possible and this is, after all, the primary goal of these modifications: more extensive Web services.

However, this also means that with every update of the official client software that Linden Lab releases the modified client is no longer compatible and has to be updated. The modifications would have to be re-implemented after every update. Linden Lab implements only a small number of user-made small-scale modifications in the official Second Life client.

Second Life supports XML data transfer from the Internet, but imposes some constraints on the processing of XML data, namely that an LSL scripted object sleeps for three seconds after replying to an XML Remote Procedure Call^(27, XML-RPC). These speed penalties do not occur in HTTP requests. Two other methods of communication in Second Life are chatting and instant messages, and e-mail. An XML RPC has to be initiated by the external server. Some kind of XML-based message passing is needed for web services to be efficiently integrated into an application such as Second Life. Because of the internal constraints on XML data processing on the Second Life servers, a separate web server that regulates XML traffic is needed.

A minimal architecture

The following architecture can be the basis for any kind of system that needs to interface with the grid. It consists of a number of components that communicate with each other. The two most important components that enable communication outside of the grid are an object containing various LSL scripts, and an external server to process events generated by the object.

The scripted object has to be activated by the client in order to register client actions and communicate them to an external server that processes the actions. The server provides feedback that is sent back to the object. The object can then apply the server's instructions to the client. The most important thing is that all data processing is done transparent to the client and takes place outside the grid, and that by using a web services architecture, Second Life client modifications are no longer integrated

into the client. This means that each regular update of the client software will not require re-implementation of the modifications.

- A **Second Life client** is an avatar that interacts with the grid as all clients do.
- The server is part of the **Second Life grid** and interacts with the client in a normal fashion. Actions that originate at the client are processed by the server. Likewise, the client-side object's scripts are also executed by the server.
- The **LSL scripted object** communicates with an external server. This communication can be done via HTTP requests or via XML-RPC. The client has the scripted object attached, possibly to the viewport, so that it is visible to the user as a sort of HUD (Heads Up Display) for relevant data.
- An **external web server** interfaces with the LSL scripted object. The server and the object can use any protocol, as long as the communication of that protocol uses the Second Life interface. The web application could use PHP to generate XML-RPC's

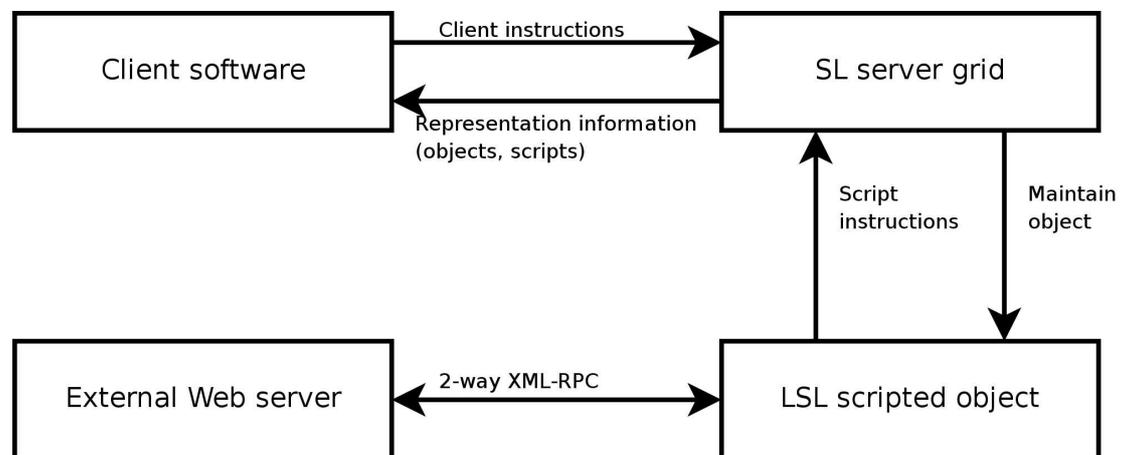


Figure 8: The flow of information from the client software to the external server.

XML-RPC

All communication between the LSL script and the external server is done via XML-RPC. This script-initiated communication allows for the largest amount of data to be transported out of Second Life compared to HTTP requests and E-mail^(27, XML-RPC).

Figure 8 illustrates the information pipeline between client and server via the Second Life server grid. Figure 9 clearly shows the LSL scripted object as the link between the client's actions in Second Life and the external server.

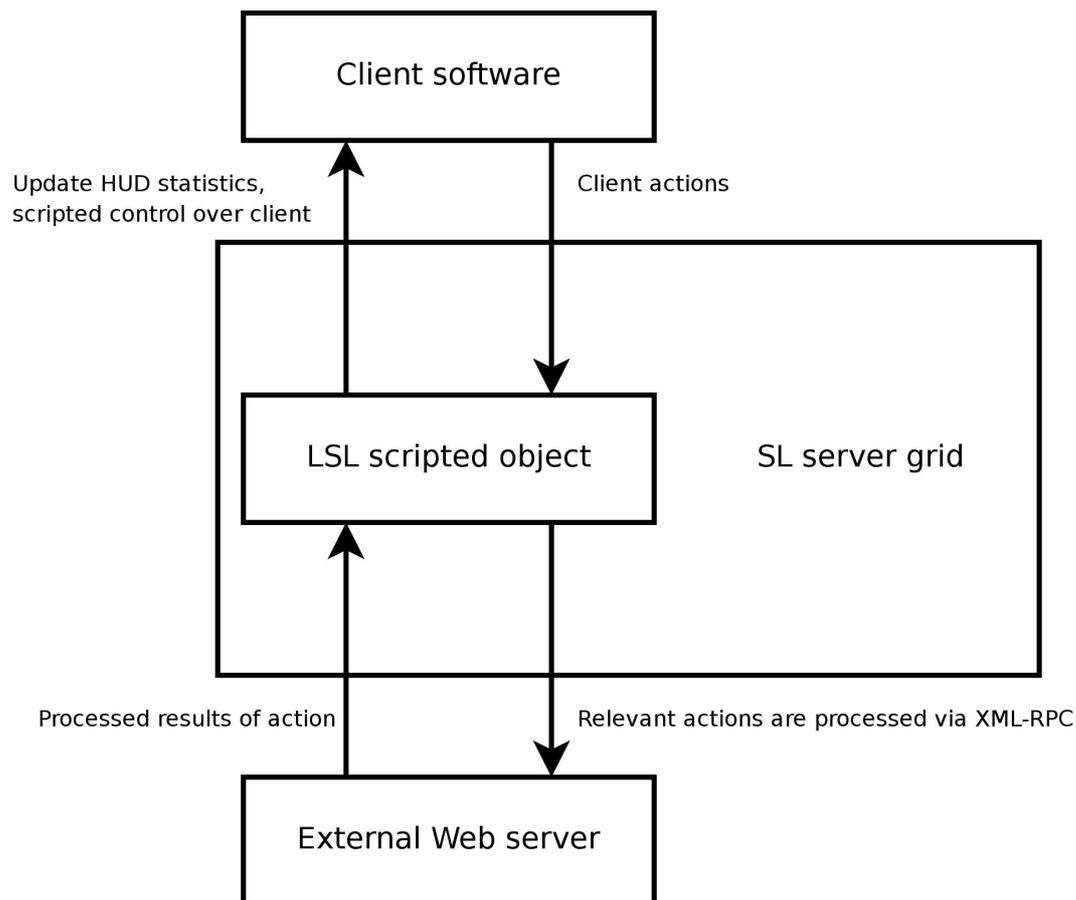


Figure 9: Control feedback loop. The LSL script monitors client actions and sends the application-relevant actions to the external server. Results are sent back to the LSL script and are subsequently fed back to the client.

Conclusion

Among all the virtual worlds that exist today, Second Life has the largest user-base. It's competitors, like Active Worlds, the Croquet Project and other virtual worlds all have different strengths, but none match the popularity with the general public and the commercial companies that Second Life has.

Being a 3D rendered virtual environment, Second Life and other 3D worlds have the advantage over 2D virtual worlds like Habbo Hotel⁽²⁸⁾ and Gaia Online⁽²⁹⁾ because of the immersion that a 3D environment has to offer. 2D virtual worlds are often aimed at a younger audience or have a more limited range of applications. A 2D approach is often chosen because of limited resources, or because of a deliberate choice in art style, perhaps to appeal more to a teenage target audience.

Second Life still needs to rely on the World Wide Web for the display of large amounts of information used in mash-ups. Text will always be 2D, and Second Life does not yet support a user-friendly way of displaying large texts for reading in the client. It might not be as fun as a mash-up where you can view photo's in an interactive virtual poster frame, but a solution is an integrated browser in the Second Life client that displays any information a regular browser can. Even though a regular browser that might be running in the background, such an integrated browser enhances the immersion of the virtual world. Second Life is a stand-alone application, because browser-integrated 3D graphics are not yet developed enough for Second

Life to be a browser application. The environment might be 3D, but in Second Life, the World Wide Web can still be represented in 2D.

General architectures for external data processing in Second Life are an important step towards easier implementation of various types of applications like separate games and other applications that require large amounts of data to be processed. A support architecture, combined with the planned migration to a new physics engine for Second Life could make developing applications for second life a more manageable task.

The Open Source Metaverse Project

The Open Source Metaverse Project (OSMP)⁽³⁰⁾ aims to create a metaverse client that is compatible with all the big virtual worlds like Second Life, There and Active Worlds. The information in each metaverse will be rendered differently depending on the settings of the client, but the key is that a user should be able to travel from one virtual world to the other. The different metaverse servers (Second Life, Active Worlds, etc.) could be compared to separate web servers.

One important factor is that the virtual world server code must be an open standard. Fortunately, Second Life is about to make that step. In the future, there might no longer be different virtual worlds to choose from. There could be one World Wide Web alongside one virtual world.

Appendix: developing the Second Life viewer

This is a short guide with tips about locations with information on how to develop a modified version of the Second Life client with added features. To develop tools that can be used in the viewer you need to understand the architecture of the software, and there's no better place to start than the Second Life Open Source Portal. This portal is a part of the Second Life Wiki⁽³¹⁾, which also contains portals to Web Services for Second Life, a guide to LSL and more Second Life community services.

The Open Source Portal is divided into two sections called 'Learn' and 'Participate.' Participating means downloading and testing beta builds of the Second Life viewer, while learning focuses on the development of the viewer software.

Learn

This section is all about understanding the way Second Life software works. The Second Life viewer is written in C++.

Features

A feature ontology from a user's perspective.

Viewer architecture

Descriptions of every component in the viewer.

Learn by example (step-by-step examples of developing features in different categories.)

Viewer source code

Description of the client's directory structure.

Server architecture

Description of Second Life server components.

Protocol documentation

Illustrates all communication between the client and the servers.

Third party libraries

A list of all known third party libraries that are used with second life. Listed with each library is whether or not they are GPL-compatible and if not if they are covered under the Linden Lab FLOSS License Exception (for software that is free but not GPL-compatible).

Licensing

Information about using licensed software in Second Life, GPL-compatibility and the abovementioned FLOSS exception.

Second Life coding standard

Definitions and formalizations of the style of code produced at Linden Lab.

Roadmap

List of immediate Second Life viewer projects by Linden Lab.

Participate

This section focuses on the official Second Life viewer. Users without programming skills can test the test build and experiment with it. Developers can submit code to fix bugs or fill feature requests.

Communicating with the developers

SLDev@lists.secondlife.com is a high-traffic mailing list. Summaries can be found at http://wiki.secondlife.com/wiki/SLDev_Traffic. Technical issues can be discussed at the technical forums (<http://forums.secondlife.com>). Direct communication is possible via IRC (<irc://irc.efnet.org/#opensl>)

Filing and vetting bugs

Second Life has an issue tracker database that contains all known issues with the viewer, both bugs and feature requests (<http://jira.secondlife.com>). Testers are encouraged to check and reproduce bugs.

Submit code

Submitted code can fall in the following categories: bug fix, crash report, implementation of a feature request or application profiling.

References

⁽¹⁾*Second Life: Your World, Your Imagination*

<http://secondlife.com>

⁽²⁾Linden Lab

<http://lindenlab.com/>

⁽³⁾*GNU General Public License*

<http://www.gnu.org/copyleft/gpl.html>

⁽⁴⁾*Leading a double life: In a user-created universe, alter egos bridge the gap between fantasy and reality*

Irene Sege, October 25, 2006

The Boston Globe

http://www.boston.com/news/globe/living/articles/2006/10/25/leading_a_double_life/

⁽⁵⁾*Get a (second) life*

James Harkin, November 17, 2006

Financial Times

<http://www.ft.com/cms/s/cf9b81c2-753a-11db-aea1-0000779e2340.html>

⁽⁶⁾*LindeX Market Data*

<http://www.secondlife.com/whatis/economy-market.php>

⁽⁷⁾*VU @ Second Life -- creating a (virtual) community of learners*

Eliëns A. Feldberg F., Konijn E., Compter E., VU @ Second Life --
creating a (virtual) community of learners

In Proc. EUROMEDIA 2007, L.

Rothkrantz and Ch. van der Mast (eds), Eurosis, pp. 45-52

<http://www.cs.vu.nl/~eliens/media/paper-secondlife.html>

⁽⁸⁾*Second Life Wiki, Havok 4 Beta*

http://wiki.secondlife.com/wiki/Havok_4_Beta_Home

⁽⁹⁾*Second Life's First Millionaire*

Rob Hof, November 26, 2006

http://www.businessweek.com/the_thread/techbeat/archives/2006/11/second_lifes_fi.html

⁽¹⁰⁾*Second Life Games: We LiKe To PLay*

<http://slgames.wordpress.com/>

⁽¹¹⁾*Weapons of choice*

<http://slgames.wordpress.com/2007/07/15/weapons-of-choice/>

⁽¹²⁾*Samarkand RP kicks off*

<http://slgames.wordpress.com/2007/08/04/samarkand-rp-kicks-off/>

⁽¹³⁾*libsecondlife*

http://www.libsecondlife.org/wiki/Main_Page

⁽¹⁴⁾*Outcry as 'Copybot' threatens copyright protection*

Adam Reuters, Nov 14, 2006

<http://secondlife.reuters.com/stories/2006/11/14/outcry-as-copybot-threatens-copyright-protection/>

⁽¹⁵⁾*Platforms and Technologies Panel at VW07*

Mark Wallace, March 28, 2007

<http://www.3pointd.com/20070328/platforms-and-technologies-panel-at-vw07/>

⁽¹⁶⁾*Second Life to open-source grid; will Google bite?*

Steve O'Hear, April 18, 2007

<http://blogs.zdnet.com/social/?p=142>

⁽¹⁷⁾*Second Life + Web 2.0 = Virtual World Mashups!*

Pete Cashmore, May 30, 2006

<http://mashable.com/2006/05/30/second-life-web-20-virtual-world-mashups/>

⁽¹⁸⁾*Linden Scripting Language: lHTTPRequest*

http://wiki.secondlife.com/wiki/Category:LSecond_Life_HTTP

⁽¹⁹⁾*Alas, Second Life! Web 2.0 in a virtual world*

Matt Biddulph, May 29 2006

<http://www.hackdiary.com/archives/000085.html>

Annotated source code for Second Life flickr screen

<http://www.hackdiary.com/archives/000087.html>

⁽²⁰⁾*flickr, photo sharing*

<http://www.flickr.com/>

⁽²¹⁾*SLurl*

<http://slurl.com/>

⁽²²⁾*Del.icio.us: social bookmarking*

<http://del.icio.us>

⁽²³⁾*Second Life Webmap API*

<http://secondlife.com/developers/mapapi/>

⁽²⁴⁾*Second Life Registration API*

<https://wiki.secondlife.com/wiki/RegAPI>

⁽²⁵⁾*Google agrees to acquire Panoramio*

Eduardo Manchón, May 31 2007

<http://www.panoramio.com/blog/google-agrees-to-acquire-panoramio/>

⁽²⁶⁾*Mashups in Second Life @ VU*

Eliens, Feldberg, Konijn, Compter, 2007

<http://www.cs.vu.nl/~eliens/media/paper-mashups.html>

⁽²⁷⁾*Linden Scripting Language: XML-RPC*
http://wiki.secondlife.com/wiki/Category:LSL_XML-RPC

⁽²⁸⁾*Habbo Hotel*
<http://www.habbo.com/hotel>

⁽²⁹⁾*Gaia Online*
<http://www.gaiainline.com/>

⁽³⁰⁾*Open Source Metaverse Project*
<http://metaverse.sourceforge.net/>

⁽³¹⁾*Second Life Wiki, main page*
<http://wiki.secondlife.com/wiki>