

Social Games

Distributing Games with Gluon and Open Collaboration Services

Dan Leinir Turthra Jensen
Dept. of Computer Science
Aalborg University
Selma Lagerlöfs Vej 300
9220 Aalborg Øst
Denmark
admin@leinir.dk

ABSTRACT

The Gluon game creation framework is designed to make the creation of games simpler. But it is also about making the distribution of those games simpler, and enabling those who play the games to report back on the quality and any other things they may have to say on the topic. As such, this paper describes how the Gluon software packages uses the Open Collaboration Services framework to allow for this multi-way communication: From creator to player, from player back to creator, and from player to player.

Categories and Subject Descriptors

H.3.5 [On-line Information Services]: Data sharing

General Terms

Distribution of Gluon-based computer games using Open Collaboration Services

Keywords

open collaboration services, gluon, games

1. INTRODUCTION

While video game development has become a very large business over the last few years, there is still a grass roots movement, known as indie games. This is a wide community of small game makers, which operate in a much more direct and communicative manner. An example of this is Wolfire Games¹, the creators of the widely acclaimed (and recently open sourced) rabbit fighting game Lugaru HD².

Indie games are often somewhat quirky and experimental, and target a much smaller audience than the AAA games that tend to get most of the press. However, lately the

¹<http://www.wolfire.com/>

²<http://www.wolfire.com/lugaru/>

indie games have been getting a lot of press - even so far as to getting their own, dedicated section on Microsoft's game console, called Xbox Live Indie.

Many of the indie game creators do not, as a direct result of being very small teams, have enough resources to build an engine for their games. As such, they often look elsewhere for help with this part of the development. The rising star on the game engine sky is Unity³, who by Game Developer Magazine were named one of the top 5 gaming companies[Nutt 2009] of 2009.

Now, this is the state of the art of game creation. However, it falls short in one very important aspect: The distribution chain for Unity and its ilk (which counts engines such as the Unreal Engine 3 Dev Kit and the Source Engine) stops on the desktop of the game creator. It is our intention in this paper to show how the Open Collaboration Services⁴ combined with the Gluon game engine⁵ can provide the game creation community with a new and alternative way of creating and distributing games.

2. PREVIOUS WORK

Distributing games is traditionally done either via classic media such as CDs or DVDs, and more recently through digital distribution systems such as Steam Powered⁶, Impulse Driven⁷ and the new player on that particular field, ModDB's Desura⁸. These are all at the opposite end of the distribution chain to the game engines described above.

However, in none of these engines exist any integration with the distribution systems, analogue or digital, and as such, the barrier is somewhat higher than it really needs to be. A game creator either needs to know how to build websites (to provide their own download site for the game) or manually upload the game to one of the distribution services.

In KDE 3, a service was introduced by the name of GetHotNewStuff. This service originally started as a way of fetching things from a website; basically a simple, one-way dis-

³<http://www.unity3d.com/>

⁴<http://socialdesktop.org/ocs/>

⁵<http://gluon.gamingfreedom.org/>

⁶<http://www.steampowered.com/>

⁷<http://www.impulsedriven.com/>

⁸<http://www.moddb.com/groups/desura/>

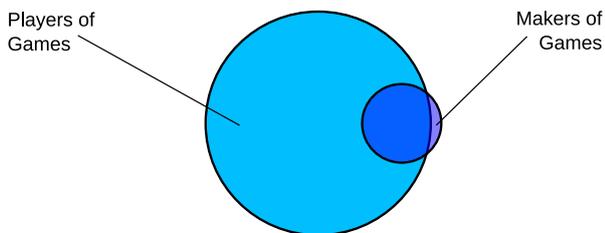


Figure 1: The Players and Makers of Games

tribution system for various content. In KDE 4 this was greatly expanded upon, and it was eventually ratified by Freedesktop.org under the name Open Collaboration Services [Karlitschek et al. 2010], currently at version 1.5, with the next version being worked on.

One of the important aspects of the Gluon game engine is that there are no compiled binaries that must be shipped with a game. Rather, the games are collections of assets, such as graphics (png, tga...), sounds (wav, ogg/vorbis...), brains for the SMARTS AI system, and game logic is created as scripts using the powerful scripting system QtScript. This allows the same game to be shipped on any platform that the Gluon engine runs on, without any changes from the game authors' point of view.

3. TERMINOLOGY

Before continuing, some terminology needs to be defined. In the paper, two important terms are used to describe groups of Gluon users: The Players of Games and the Makers of Games. As the names indicate, a Player of Games would be a person who uses Gluon as a way of entertaining themselves by playing games (and, we hope to prove, interact with others with the same interest). However, a Maker of Games is not necessarily uninterested in playing games. As such, an Euler diagram of these (see Figure 1) would show a very large overlap between the two, with a large number of Players of Games being only that, but with only few Makers of Games not also being Players of Games.

This fact is important to remember through the rest of the paper - when the term Maker of Games is used, it thus is implicit that it also means Player of Games. The few Makers of Games who do not also take part in the playing of those games (or at least the discussion thereof) are specifically not a part of the target audience. If they are covered anyway, this is of course a pleasant happenstance, and while no effort will be put into discouraging the behavior, no effort will be put into supporting the behavioral pattern of such users either.

4. THE GLUON DISTRIBUTION SYSTEM

Enter the Gluon distribution system: An end-to-end solution for creating, distributing games, and receiving feedback on them. Figure 2 describes this flow; the game creators use the Gluon Creator application to create the games, which then are uploaded to the OCS based Gluon distribution website, and finally downloaded by the players of games using a Gluon Player on any one of the many supported platforms: Windows, MacOS X, Linux/X11 (either the stand alone Qt-



Figure 2: Infographic showing the flow of Gluon games from idea to player and back

only client or the Gluon Player Plasmoid), Maemo, MeeGo and so on.

4.1 Creation and Distributing

Creating games in Gluon Creator is a matter of dragging items into place. The Game Object/Component system shall not be explained in depth here, as it is not relevant to the distribution system, but this is what allows the game creators to create games in this manner. Drag items into place in the scene, add functionality to the items, and set values on these through the Properties dock. Much more information about this part of the process can be found in the Gluon documentation⁹.

Distribution of the games, once created, is the simple matter of using the Distribution dialogue in Gluon Creator, a design for which is pictured in Figure 3. As the mockup shows, this dialogue not only allows for the uploading of new content to the distribution website, which is GamingFreedom.org by default. It also allows for interaction with the players of games, in that a list of new items are shown, both messages and player-provided screenshots, and the maker of the game is thus able to comment on the game as well, and thus interact directly with the players of their game.

Importantly, while only one person can be the owner of a game, multiple people can be contributors. What this means is that the messages they post about the game are marked as author supplied. The owner furthermore has granular control of what the various contributors are allowed to do. The list of permissions is as follows:

Upload (default: allowed) This will allow the contributor to upload new versions of the game. To encourage the inclusive development model, all contributors are allowed by default to upload new versions of a game. This, of course, may not be the optimal solution for larger projects with many contributors, and as such this option can be disabled on a per-contributor basis.

Moderate (default: allowed) The contributor will, when allowed to moderate, be able to edit comments made by non-contributors for the specific game. Editing a message will cause it to be marked, and the moderator's name will be listed alongside the edit, with an optional reason for the moderation.

Heavy Moderate (default: not allowed) The contributor will with this option enabled be allowed to delete comments made by non-contributors for the game.

⁹<http://gluon.gamingfreedom.org/node/10>

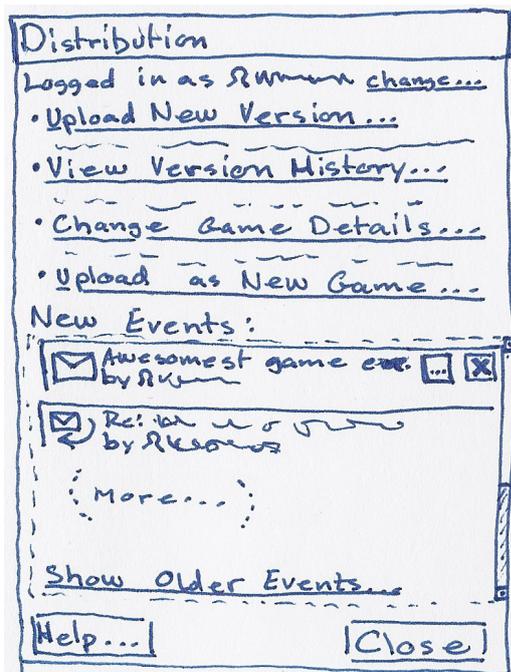


Figure 3: A mockup of the distribution dialog for Gluon Creator

4.1.1 Work Flow

There are two reasons for uploading a game to the distribution site: Either you have a new game you wish to publish, or you have a game which you wish to update.

The first then is where the initial setup of a game as presented to the players of games is provided. This information is:

Owner The username of the user who will be listed as the owner of the game.

Contributors A list of usernames of the users who will be marked as contributors, along with checkboxes for the state of the options described above.

Name The name of the game, simple as that.

Icon A square image used in the lists. This image can be provided in several standard icon resolutions (16x16, 22x22, 32x32, 64x64, 128x128 and 256x256, as well as a scalable vector graphics (svg) version whose minimum and maximum resolutions can then be set, inspired by the freedesktop.org icon specification[Dawes 2006]), for use in various places (some clients may wish to use larger or smaller icons depending on how they present the list of games to the user).

Blurb A very short description of the game, shown in the list of games and at the top of the description of the game on both website and clients. See for example the button "Read Author's Description" in Figure 4 later in the paper.

Description A longer description of the game, formatted using the same simplistic HTML-like rich text tags

that Qt's labels support (in stead of reinventing the wheel).

Initial Screenshots A list of screenshots provided by the makers of the game, so the screenshots list can be populated immediately upon game release, so the makers of games do not have to wait until users have provided screenshots. As the internet saying goes: "Pics, or it didn't happen".

Initial Tags Tags which the makers of the game believe apply to this game. This is not a canonical set of tags, as any user is able to apply their own tags¹⁰.

A notable absence here is the achievements which are so popular with so many players of games (witness the many people who are hunting achievements on the various other more closed game platforms such as XBox 360, Steam and the like). The reason for this is that achievements in Gluon are handled by a Component (see the documentation), which allows the distribution site to simply look through the game and pull out this list directly, without further interaction by the makers of games.

The implementation of the work flow for the second upload (that is, uploading a new version of an existing game) requires the makers of games to supply their login information (that is, the login of the game project's owner, or an upload-allowed contributor, as defined above), after which they will be provided with a list of existing games owned or upload-enabled for this user. Once the user has selected the item they wish to upload a new version for they are presented with the information they have already provided for the game (name, initial tags and the like), and a field in which they can describe what is new for this updated version of the game.

Games are thus stored in a versioned manner. That is, all the data for each version of the game is stored, from the very first version and ahead. Whether old versions are allowed to be seen by users who are not contributors to the game can be set by those contributors with upload permissions.

4.2 Submission Control

As it would be possible, at least to some extent, to write malware and various other things of dubious legality using a powerful system such as the Gluon engine, it would not do to simply allow people to upload items and then have them pushed to the players of games immediately, without at least some vetting. As such, the distribution website provides a UI which allows administrators to view the queue of new submissions, and mark them as accepted, or mark them as non-accepted, and at the same time inform the creator of why the game was not accepted.

To assist these administrators in their work, users can allow themselves to be exposed to these potentially dangerous

¹⁰This is according to the broad folksonomy model which implies that the weight of a tag is relative to how many people apply that tag to the item, as opposed to the narrow folksonomy applied by e.g. flickr where only one person can set the tags and thus there is no weighting of the tags[Wal 2005]

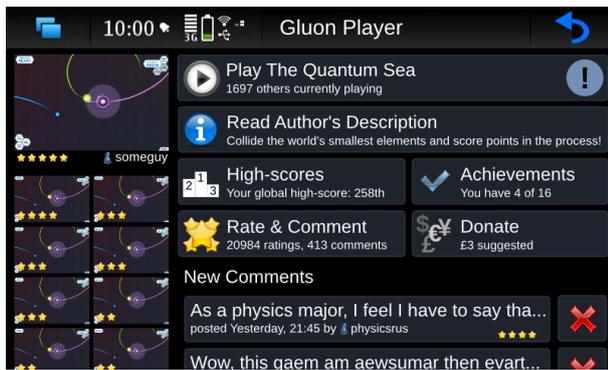


Figure 4: Mockup of the game details view of the Maemo/MeeGo version of Gluon Player

games, by taking the red pill - inspired by the system found in Maemo 4, by which users can accept the responsibility of taking care of themselves of trying out experimental software. This then entitles these users to take part in the review process by playing the games and then rating and commenting on them before they are officially launched by the administrator team. The red-pill function is deliberately not available in the game clients. This is only possible from the website's profile editing, to reduce the discoverability of the function. The reasoning behind this is that we do not wish for people to accidentally turn on this feature.

4.3 Getting and Playing

As hinted at above, fetching and playing games is not done normally through the website. What that means is that the players of games use a desktop client, inspired in part by the Steam, Impulse and Desura clients mentioned in the introduction. Figure 4 shows one of the early mockups of the Gluon player for Maemo/MeeGo, and Gluon Player for the desktop looks similar to it.

The "Play The Quantum Sea" button at the top will, as indicated, launch the game. If the game has not yet been downloaded to the device currently being used, the player will be told how big the game download is, and whether they wish to continue. This will then happen in the background, and the user allowed to continue browsing while the download is happening.

OCS makes fetching the game easy using the method `CONTENT/download` (and even allows for fallthrough handling of items requiring login), and libgluonengine allows for easy creation of software which will load and play a Gluon Engine based game after completed download. Similar to the description of Gluon Creator above, this particular part is not vital to the understanding of the distribution system, and as such shall not be explored further in this paper.

4.4 Communication and Feedback

Providing your opinion on the games in the Gluon distribution system is part of what makes this interesting to game developers - they are in direct, but still mediated, contact with their public. This means that not only are the players of games able to comment on and rate games, they are also able to get feedback on *their* contribution, in the form of

replies to their comments, both from other users, and potentially from the authors of the games they are commenting on.

Comments are stored both with the time at which they were made, and the version of the game which they had access to at the time. The reason that this is stored is the Red-pill system described above, and as such resolving the versions available to a user at any one time would incur a high penalty, and as such the minor amount of data stored for the version number is considered acceptable.

This community based system is also what enables users to invite each other to play certain games, or to suggest to friends that they should try out a game. This then shows up in the Gluon Player user interface in the form of the blue circles with an exclamation mark in as seen in Figure 4, but also in the application's game lists, where they function as direct launch controls, allowing players to accept these ostensible invitations directly by clicking on them. This system uses a combination of the `FRIEND`, `MESSAGE` and `COMMENTS` sections of the OCS api, as well as a URL handler on the various systems for the `gluon://` URL schema.

4.4.1 The Gluon URL Schema

The introduction of the Gluon URL Schema is based on the concept that any item in the Gluon data structure should be available through a simple, copy-pasteable URL, inspired by the work done by Nikolaj Hald Nielsen in Amarok. Based on this wish, we define a set of URL structures as follows:

- `gluon://game/gameid@provider`
where *provider* is the domain name of the game provider (for example `gf.net` or `gamingfreedom.org`), and *gameid* is the numeric ID of the game as found on the website. Any extra field added by appending a slash (/) after this ID is ignored, and as such this can be used to provide a human-readable URL.
Sample: `gluon://game/321@gf.net/The-Quantum-Sea`
- `gluon://user/id`
where *id* is the user id in the form of the jabber ID provided by the website. This allows for both user and server identification, as well as being human readable.
Sample: `gluon://user/username@gamingfreedom.org`
- `gluon://comment/gameid@provider/id`
where *gameid* and *provider* are as in the game URL, and *id* is the numeric ID of the comment. Similar to the game URLs, adding an extra slash will cause any extra data following this to be ignored. This can be used to provide a small piece of data to the URL, for example encoding a twitter or identi.ca like piece of information from the comment.
Sample: `gluon://comment/321@gf.net/31/I-totally-love-this-game`

4.4.2 In Game Communication

Communication between players while inside a game requires a more real-time approach than that normally provided by OCS, and for this the use of the open standard XMPP[Group 2010] is being considered. Utilizing this standard would allow for both text-based communication in the

style of the text chat found in many games today, but it would also allow for voice and potentially even video chat, as used by gamers world wide through for example Skype and the Ventrillo voice chat system.

One other very important trick that XMPP allows Gluon to perform is provided by the inclusiveness of the distributed network that is XMPP's protocol: When players of games using Gluon are playing some game, they can send invitations to their non-Gluon using friends for games the same way as though they were already players of Gluon games. This is of some considerable importance, as it opens up the possibility of reaching, through viral means, users who are already on for example GMail, which provides every user using it with a Google Talk account, a service also based on XMPP.

4.5 Payment Model

While OCS itself does allow for payment for downloads, the Gluon team proposes a new payment model, based on donations. In the game world, it is a wide spread problem that games, no matter what you do, are pirated immediately after the release of the game¹¹. This, of course, is not a problem in the world of Free Open Source Software, as there really is no reason to pirate software which is provided for free to start with, and as such it is the purpose of Gluon to provide the world of FOSS games with a mass market distribution channel, which at the same time encourages the players of games to provide monetary assistance to the makers of the games they enjoy.

As seen in Figure 4, the button immediately below the button "Achievements" and next to the "Rate & Comment" button has the title "Donate" and the subtitle "£3 suggested". What this suggests is a donation-based payment model, where you are free to download any and all games, but even though both amount and action is optional, the donation of a specific amount is encouraged to the users.

On top of the donation of a specified amount, the use of the Flattr system is also possible¹². This creation by the original founder of The Pirate Bay is all about providing users with the ability to make micropayments in a manner which is much simpler than is currently the case. The system works like so: You pay in a monthly amount to your Flattr account, and this is then shared equally between people you decide to "flatter" over the following month. This of course means that you can spread that amount over as many or as few people as you wish, but you always have a set amount per month to pay to people whose work you enjoy.

A point which is currently undergoing discussion is that of the subscription based payment model for game content. This model would allow users to subscribe to a game, and receive new content at certain intervals, content which is then available to them perpetually or while they pay for the subscription. This discussion, however, is at time of writing

¹¹Ubisoft's Splinter Cell 3 being a notable exception, providing a 422 day period between release and crack. This was achieved using the highly controversial StarForce 3.0 system. Source: <http://blog.wolfire.com/2010/03/DRM-can-be-effective>

¹²<http://flattr.com/>

not concluded, and the Gluon team invites the readers to join them in discussing this topic.

While payment for downloading an item is supported, donations are not currently supported through the OCS interface. While this does mean that work must be done on the API with this in mind, it also means that, based on the support for payment which is currently limited to using PayPal, it has not been implemented without the option of multiple payment suppliers in mind.

5. CONCLUSION

While the Gluon system is still under heavy development, the proof of concept implementations found in the Gluon alpha releases show that the system works in theory as practice.

The missing parts in OCS as identified in the paper are in short: Donation, micro-payment and subscription payment, high-scores and achievements. While the last two are relatively trivial except for ensuring cheating does not happen, the first three are more difficult. However, they are in no way insurmountable, as shown by AmaroK's integration of payment systems for e.g. Magnatune, and as such it is the belief of the author that nothing presented in this paper presents a challenge which is so difficult that it is not possible to implement it in the foreseeable future.

Frank Karlitschek, the owner of the code behind opendesktop.org, has offered to provide the Gluon project with assistance in developing the Gluon distribution system website by providing a test server for this. As such, it is the hope of the author that this development will allow for the more rapid construction of this system. While the framework, as mentioned previously, does not contain all the features needed, it goes a very long way to provide the basis for it. As such, this development is received with much gratitude, and it is the hope of the entire Gluon team that this will allow for a more rapid construction of the distribution chain which has been described in this paper.

6. REFERENCES

- [Dawes 2006] DAWES, R. 2006. Icon naming specification. <http://standards.freedesktop.org/icon-naming-spec/icon-naming-spec-latest.html>.
- [Group 2010] GROUP, N. W. 2010. Extensible messaging and presence protocol (xmpp): Instant messaging and presence. <http://tools.ietf.org/html/draft-ietf-xmpp-3921bis-06>.
- [Karlitschek et al. 2010] KARLITSCHKEK, F. ET AL. 2010. Open collaboration services (v1.5). <http://www.freedesktop.org/wiki/Specifications/open-collaboration-services>.
- [Nutt 2009] NUTT, C. 2009. Gamasutra's best of 2009: Top 5 game companies. http://www.gamasutra.com/view/news/26243/Gamasutras_Best_Of_2009_Top_5_Game_Companies.php.
- [Wal 2005] WAL, T. V. 2005. Explaining and showing broad and narrow folksonomies. http://personalinfocloud.com/2005/02/explaining_and_.html.