

# *World of WebCraft* — Mashing up World of Warcraft and the Web

Knud Möller

Digital Enterprise Research Institute, National University of Ireland, Galway  
`knud.moeller@deri.org`

**Abstract.** This short paper presents *World of WebCraft*, a set of tools which together allow players of the MMORPG World of Warcraft to generate photoblog-like Web representations of their in-game avatars. This is achieved by periodically logging information of the location of the avatar during the game, matching this information with in-game screenshots and then uploading them to Flickr, using machine-tags as annotations. Finally, an additional Web application uses the machine-tagged pictures to generate the photoblog. The tools are implemented using a combination of Lua and Ruby (two scripting languages), as well as Objective-C.

## 1 Introduction

So-called MMORPGs (Massively Multiplayer Online Role-playing Game) such as World of Warcraft (WoW) are online games in which players take the role of a fictitious, often mythical character (their *avatar*), explore a vast imaginary world, solve quests, battle enemies, and thus gain riches, knowledge and new skills within the game. Unlike traditional single-player games, an important factor in MMORPGs is the fact that players don't just interact with computer-generated and -controlled characters, but instead with the avatars of other human players who are online at the same time. In this way, players can form groups and tackle situations in which a single player would have failed. It is the interaction with others that lets these online games transcend simple computer games and move into the realm of online social networks (OSN), an area that is more commonly associated with platforms such as FaceBook, LinkedIn or Flickr: "*At its essence WoW is a social network; like minded people come together online to share a common experience and make connections*" [!] [6].

Users often choose to establish online identities in a number of different online social networking sites. Unfortunately, these sites tend to be closed, and their data not interlinked. E.g., this author has accounts on Flickr, FaceBook, LinkedIn, Xing, Twine, YouTube and probably others he has already forgotten about. Semantic Web technologies like FOAF<sup>1</sup> and SIOC<sup>2</sup>, as well as the very

---

<sup>1</sup> <http://www.foaf-project.org/>

<sup>2</sup> <http://www.sioc-project.org/>

recent and prominent Social Graph API<sup>3</sup> by Google have been proposed to cure this problem and connect the various online identities of a person into a whole [4,3]. Ignoring all potential problems which may occur when attempting to integrate data from all these different services together, their nature as *Web* services at least defines a general strategy for doing so: a person's identity on each site can be referenced by their profile identity URL, which can then be related using a vocabulary such as FOAF. In the case of MMORPGs however, this is not the case. While they do take place on the internet, they are not located on the Web. Consequently, a player's avatar also doesn't have a natural Web URL which could function as a reference.

This paper presents a strategy and (partial) implementation to overcome the barrier between the online, but non-Web world of MMORPGs and traditional OSNs. Taking the popular game World of Warcraft and the photo sharing site Flickr as an example, we will show how various technologies play together to generate a *World of WebCraft* — a mashup of World of Warcraft and the Web.

## 2 Architecture

Figure 1 shows a high-level overview of the approach taken for World of WebCraft. In this section, we will first describe the overall idea, and then focus each of the individual components in turn.

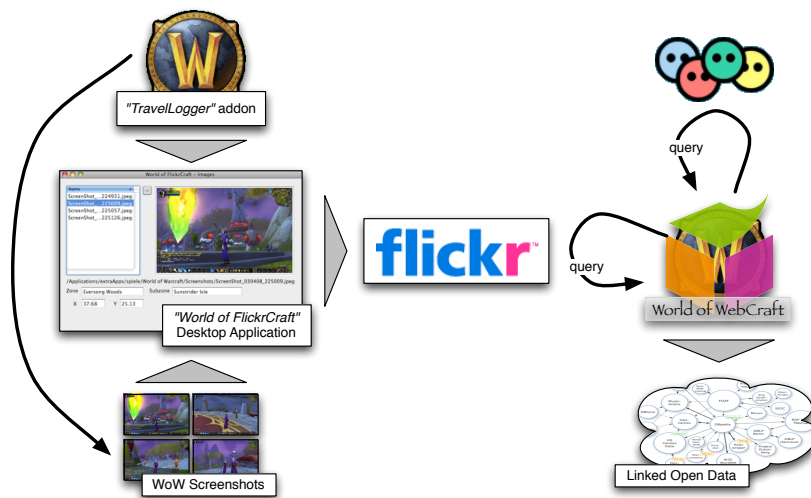


Fig. 1: World of WebCraft High-level Architecture

<sup>3</sup> <http://code.google.com/apis/socialgraph/>

In our setup, players can take screenshots of gameplay, which will automatically be tagged with information such as who the player is, when the screenshot was taken and where in the game world. This is possible by exploiting the dynamic nature of WoW's game engine, which allows it to write plugins (usually referred to as *addons*) to extend the game's functionality and user interface. A plugin called "TravelLogger" will periodically record the player's position in the world. After finishing a gaming session, a player can use the *World of FlickrCraft* desktop application to match up screenshots and log entries, and then upload the tagged screenshots to Flickr. While automatic tagging is useful in itself — it alleviates the player of the tedious task of manual tagging —, the tagged screenshots can now be used to generate a Web presence of the player's avatar through the *World of WebCraft* Web application<sup>4</sup>. By simply supplying their Flickr profile URL, the player enables World of WebCraft to generate a photographic online diary for their avatar — a photoblog for an orc! Alternatively, the player can also provide a link to their FOAF profile, given that it in turn points to their Flickr profile. Finally, World of WebCraft will make the avatar's photoblog accessible on a stable, meaningful URL and provide the diary data in a variety of formats (e.g., as RDF to contribute to the Web of Linked Open Data [1,2]).

## 2.1 *TravelLogger* — Using the World of Warcraft API with Lua

Almost all user interface elements in World of Warcraft (with the obvious exception of the 3D graphics) — windows, buttons, chat panes, etc. — are implemented on top of an engine which is based on the dynamic scripting language *Lua*<sup>5</sup>. Interaction between the UI and the game itself is handled through an events system. Almost anything that can happen in the game world — the player or any other object moves, monsters attack, spells are being cast, items sold and chat messages sent, etc. — will fire an event, which can be registered and acted upon by the Lua engine. WoW producer Blizzard have decided to open this system up for external developers. This makes it possible to write plugins (or *addons*) for WoW which can interact with the game events, extend the user interface, etc. A comprehensive documentation for the WoW API is available through [http://www.wowwiki.com/World\\_of\\_Warcraft\\_API](http://www.wowwiki.com/World_of_Warcraft_API).

The *TravelLogger* addon, which was developed as part of the World of WebCraft project, is a simple, light-weight plugin which makes use of the extensible WoW-engine to create a log of the avatar's movements through the game world. The plugin can be started and stopped from within the game using command `\tlog`. While running, the plugin will call the methods `GetPlayerMapPosition`, `GetZoneText` and `GetSubZoneText` in intervals of five seconds, to query both the precise geographical coordinates of the avatar, as well as a human-readable description of the approximate location (the *zone* and *subzone*). This data will

---

<sup>4</sup> <http://www.kantenwerk.org/wowc/>

<sup>5</sup> <http://www.lua.org/>

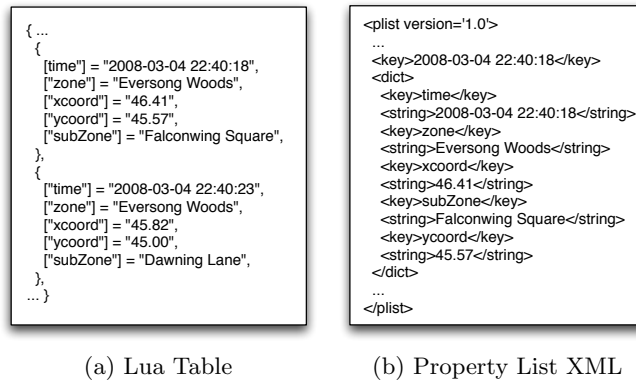


Fig. 2: The Travel Log in Lua and XML format

then be written out to a log file as a Lua table (see Fig. 2a). For security reasons, WoW only allows plugins to write in very specific locations of the file system, which follow the pattern below. Also, it is only possible to write Lua data structures (and not, e.g., a more common format such as XML).

```

WTF/Account/$ACCOUNT_NAME/$SERVER_NAME/
  $AVATAR_NAME/SavedVariables/$PLUGIN_NAME.lua

```

Even though there are now millions of registered WoW players and even more avatars, the `$SERVER_NAME` (which server is the player currently on) and `$AVATAR_NAME` together uniquely identify an avatar and can therefore later be used by World of WebCraft to mint a URI for this particular avatar.

## 2.2 *World of FlickrCraft* — A Ruby-based Screenshot Uploader

WoW is very restrictive in the actions it allows plugins to perform outside the game. It is for example not possible to write arbitrary files, let alone open a network connection. For this reason, if we want to get any data out of the game and onto the Web, we have to do so from outside the game. In our approach, we use a dedicated desktop application called “World of FlickrCraft” (or just FlickrCraft) for OS X to parse and interpret the output of the TravelLogger plugin described in Sect. 2.1, match the log with a number of in-game screenshots specified by the player and upload the screenshots to Flickr.

The application is implemented using a mixture of the Lua, Ruby and Objective-C (ObjC) languages, each chosen for specific tasks. Native OS X application development is based on the *Cocoa* libraries<sup>6</sup>, which are implemented in large parts in ObjC (an dynamic OO extension to C, heavily inspired by SmallTalk).

<sup>6</sup> <http://developer.apple.com/cocoa/>

However, through language bridges, it is also possible to access the full stack of Cocoa libraries through other languages such as Python or Ruby. For the benefit of fast prototyping, we have chosen Ruby over ObjC for the development of FlickrCraft, except in a few border cases.

In the remainder of the section, we will go through the individual steps of the application.

*Parsing the Log File* Since the output of TravelLogger is a Lua data structure, we have chosen Lua itself to parse and transform it to a format more accessible to the Cocoa libraries. From within the application, a Lua script process will be started, which parses the log and generate an XML structure from it (in the *Property List* format used throughout Cocoa), as shown in Fig. 2b. The output of the script is then piped back into FlickrCraft.

*Matching Screenshots with Log Entries* The player can now select a number of screenshots they have taken during the game (which probably show them accomplishing heroic feats or other noteworthy actions) and drag & drop them onto FlickrCraft. Based on the creation date attributes of the image files, the application will then perform a simple algorithm to select the log entry which is closest in time for each image. The image will be thus be associated with a number of tags, which consist of the `zone`, `subZone`, `xcoord` and `ycoord` attributes of the log entry, as well as the server and avatar name, which are extracted from the path of the log file.

*Uploading to Flickr* As a final step, FlickrCraft then uploads the selected screenshots to Flickr, using then *ObjectiveFlickr*<sup>7</sup> wrapper for the Flickr API. Two things are important during this phase: (i) the file creation dates are injected into each image file as the value of the EXIF<sup>8</sup> `DateTimeOriginal` attribute, in order for these dates to be picked up by Flickr after the upload, (ii) the tags are added both as simple tags (using just the values of the attribute value pairs from the log entries), but also as *machine tags* [5], which are Flickr's light-weight implementation of RDF-like metadata. Fig. 3 shows the uploaded screenshot in Flickr, along with both its simple and machine tags. The property names for the latter consist of a (not further specified) `wowc` namespace and the attribute names taken from the log file.

### 2.3 Photoblogs for Avatars with *World of WebCraft*

*World of WebCraft* (or WebCraft) is a Web application, which, by harvesting the machine tags uploaded to Flickr through FlickrCraft, allows users to generate a Web representation of their WoW avatars. The application follows a very simple series of steps: (i) The user is requested to provide their Flickr profile URL. (ii) WebCraft will then query Flickr for any pictures of this user which are tagged

<sup>7</sup> <http://lukhnos.org/objectiveflickr/blog/>

<sup>8</sup> <http://www.exif.org/>

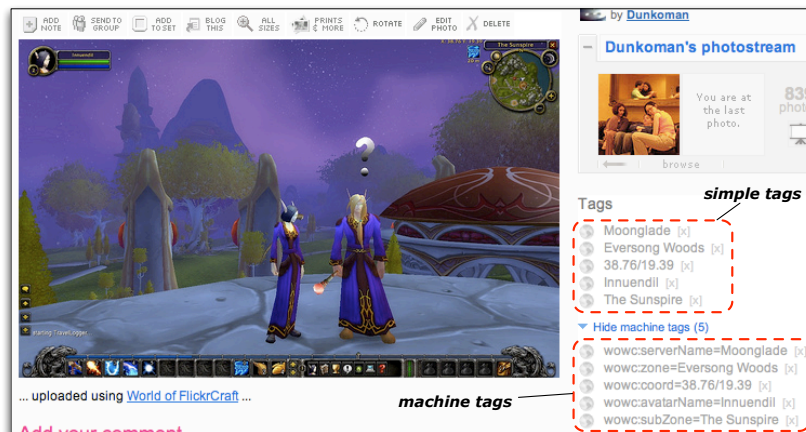


Fig. 3: Simple and Machine Tags in Flickr

with machine tags in the `wowc` namespace. (iii) The information in the tags, as well as the URLs of the images, are stored internally. (iv) For each avatar, a URL following the schema `http://kantenwerk.org/wowc/$SERVERNAME/$AVATARNAME` is minted. (v) Under this URL, the avatar's photoblog will be served in different formats (depending on the request sent to the server): JSON, an Exhibit webpage based on the JSON output, and as an RDF graph.

### 3 Conclusion and Future Work

We have presented an approach for mashing up the popular MMORPG World of Warcraft with the photo sharing service Flickr. Additionally, the World of WebCraft application exposes the data that a player wishes to publicise about their in-game avatars in a variety of formats, thus making it possible to link and mash up WoW with the Web. This opens up WoW's inherent, but largely untapped nature of an online social network.

Using the Exhibit framework, we visualise WoW data and screenshots as a timeline. By making use of Exhibit's map view and the GoogleMaps API, it would be possible to extend World of WebCraft to include a visual overview of where each in-game screenshot has been taken. A good example of how the GoogleMaps API is used on WoW data is `http://mapwow.com/`.

### Acknowledgements

The work presented in this paper was supported (in part) by the L on project supported by Science Foundation Ireland under Grant No. SFI/02/CE1/I131 and (in part) by the European project NEPOMUK No FP6-027705.

## References

1. T. Berners-Lee. Linked data, 2006. <http://www.w3.org/DesignIssues/LinkedData.html>.
2. C. Bizer, R. Cyganiak, and T. Heath. How to publish linked data on the Web, 2007. <http://sites.wiwiw.fu-berlin.de/suhl/bizer/pub/LinkedDataTutorial/>.
3. U. Bojars, J. G. Breslin, A. Finn, and S. Decker. Using the Semantic Web for linking and reusing data across Web 2.0 communities. *Web Semantics*, 6(1):21–28, 2008.
4. J. G. Breslin, A. Harth, U. Bojars, and S. Decker. Towards Semantically-Interlinked Online Communities. In *The 2nd European Semantic Web Conference (ESWC '05), Heraklion, Greece, Proceedings, LNCS 3532*, pages 500–514, May 2005.
5. A. S. Cope. Machine tags, January 2007. <http://www.flickr.com/groups/api/discuss/72157594497877875/>.
6. A. LaFauce. What can social networks learn from World of Warcraft?, March 2008. <http://www.socialtimes.com/2008/01/what-can-social-networks-learn-from-world-of-warcraft/>.