Attribution helper browser extension

Abstract

Despite the spread of Creative Commons-licensed content over the internet, people face problems reusing such content because it's often not clear whether the attribution information is present and what permissions, requirements and prohibitions it applies to the content. I propose to create an extension for a web browser that would solve problems described above by signaling that a viewed page contains licensed objects, show license terms, and provide attribution information for those objects.

Content

Problem statement

Nowadays there is an increasing number of projects and authors using Creative Commons licenses for their content. It includes resources of different scope: from wide-spread Wikimedia projects or MSDN library [1.] to many private projects [2., 3.]. The problem is that a user of these resources does not have a proper tool to:

- Find out whether a viewed page or objects it embeds and links contains licensed content;
- Know where on a viewed page an object's license information is;
- Find out the name and terms of an object's license;
- Clarify the meaning of those terms (what permissions, requirements and prohibitions they apply to an object's usage);
- Generate correct attribution information for an object's re-usage.

It's evidently clear that the functionality described above should be provided by a web-browser, the main application used to access licensed content. Current web-browsers lack such functionality, so the creation of an extension for them should be considered.

Browser choice

According to the statistics [4.], there are only two web-browsers with usage share more than 6 %: Microsoft Internet Explorer (more than 54 %) and Mozilla Firefox (more than 30 %). These two browsers taken together cover more than 85 % of users, which makes them priority ones for extension of functionality.

Further looking at the distribution of each browser's versions gives valuable info: Fx's share consists primarily of 3.5 and 3.6 versions (more than 81 % of its whole share [5.]), while MSIE's share is divided into 3 major parts (34 % for version 6, 22 % for 7, and 36 % for 8 [6.]). This means that the extension of MSIE's functionality would possibly meet a number of compatibility problems, or would lead to creation of 3 separate extensions.

On the other hand, Fx requires less effort to create a functional extension that would cover as much user share as possible, brings cross-platform support, and provides a natural way of extension distribution via a built-in mechanism [7.]. So, Mozilla Firefox is chosen to be the subject for extension.

Past & existing solutions

The review shows that there already is an extension called MozCC, created by Nathan Yergler and Asheesh Laroia. MozCC provided a way to examine meta data (including Creative Commons licenses) embedded in web pages [8.]. Although, it had a number of issues:

- Users reported its poor performance, slowing page loading process [9.];
- It probably needed an another third-party RDF parser integration as a refinement;
- It was primarily focused on parsing RDF, not on providing a complete information about licensing issues [10.];
- It lacked the ability to detect attribution information in embedded objects;
- It lacked the ability to provide assistance on proper attribution information creation;

- It lacked internationalization;
- The last version was published on Mozilla Add-ons site on February 15, 2007 and supported 3.0.* as the maximum version of Fx;
- The last commit to the repository was on December 20, 2008 and the development seems to be stalled.

So, as the development process of MozCC stopped, it needs to be carefully examined and its pros and cons must be taken in consideration while developing a new extension.

Development goals

The following goals are considered the most important while developing a new extension:

- Be user-oriented: the proposed extension is focused on an end-user, as it would be intentionally installed and used to interact with, rather than to “install-and-forget” like Adblock Plus;
- A great effort should be made to create a nice-looking (“sexy”) UI, as it's the only way the extension interacts with a user;
- The performance should not be affected by this extension, as well as it shouldn't have any other side effects;
- There should be a variety of versions in different languages provided;
- Not only the viewed page should be inspected to contain licensed objects, but also a set of embedded and (possibly) linked objects of various types (audio/video types, images, and so on).

UI mockups

Being one of the goals, the value of a nice UI cannot be overrated. Not only should it look nicely, it should bring a minimum of new things to existing browser interface, being as functional as possible at the time.

Prior to glancing at MozCC I tried to implement my own view of UI. I fully understand it's imperfect and needs discussion and a lot of refinements, though. The motivation is the following:

— The extension must show its UI only if it has something to show: no empty fields, greyed out icons and so on;

— Oh the other hand it should be clearly expressed if the extension has some information for a user;

— Among three possible places to be integrated with (a browser's location bar, status bar and context menu) the location bar should be chosen as it's already used to display page “capabilities” as icons: whether it has an RSS feed associated, whether it's bookmarked and so on; the status bar is considered to be unrelated to the currently viewed page and the context menu is not normally displayed, that's why they cannot be used.

A location bar icon should appear if a page has licensed objects (see [11.]). If a user places its mouse over this icon, the number of licensed objects on this page should be shown (see [12.]).

Left-clicking on the icon brings a pop-up menu (see [13.]) that allows to highlight any of licensed objects on the page (see [14.]; the exact algorithm and view of this highlight feature must be thought about). Also, this pop-up menu may be used to open the “Page Info” window with the “Licensing” tab shown (see [15.]). The usual way of opening this window by right-clicking on the page is also supported.

The whole number of licensed objects found is shown at the top. If a user selects a licensed object in the list it's highlighted on the currently viewed page. This behavior is controlled by the “Highlight objects on this page” checkbox.

An entry in the list of licensed objects consists of the name of the object (the exact algorithm of its creation must be thought about), the block of information related to rights holder and the block of information related to the license terms. Blocks content must adapt the available information (here “Creator” and “Right holder” fields are joined together, as it's the same person). Possible links are automatically created, a set of icons related to the license type is shown.

A button to get the attribution information in different formats is present at the bottom. It may bring a context menu with “Copy to clipboard in HTML format” and "Copy to clipboard in plain text format" items, as well as some more complicated options may be provided, like opening a window that allows attribution information customization (see the second step of [16.]).

Planned results

This project is intended to bring the following results:

- A fast, unobtrusive Mozilla Firefox extension that perfectly detects and shows the information for any of Creative Commons licenses; additional licenses may be also added;

- The support for license data detection in embedded object of at least two different types (like image and audio/video formats);

- As many language versions as possible; I'm able to provide English and Russian versions natively and I would collaborate to get the others versions done;

- As many popular web-sites containing licensed objects as possible tested to be compatible with CC REL/RDF and this extension; an effort would be made to contact webmasters of those site which are incompatible;

- Zero count of open bugs in the tracker;

- An every-week memo/blog post describing the results of that week; a document describing the whole development process; commented and documented code complying the coding standards.

Schedule

The whole available time of GSoC is about 16 weeks. My planned goals on a per-week basis are:

- Week 1-2: A full functionality and UI specification; reference UI mockups; a list of reference sites for further extension testing;

- Week 3-4: An extension that detects license information on the viewed page; first sight of UI;
- Week 5-6: Ability to detect license information in embedded/linked objects; UI improvements;
- Week 7-8: Generating of attribution information support; fully functional complete UI (milestone for mid-term evaluation);
- Week 9-10: Testing extension to comply the whole specification and reference UI mockups; testing it to behave as intended on reference sites;
- Week 11-12: Internationalization support by adding a number of supported languages; testing and fixing bugs; beta release;
- Week 13-14: Documentation development; user support; testing and fixing bugs;
- Week 15-16: Testing and fixing bugs; getting everything finished; final release (goal for final evaluation).

Links

1. http://en.wikipedia.org/wiki/List_of_projects_using_Creative_Commons_licenses
2. http://xkcd.com/
3. http://p.yusukekamiyamane.com/
4. http://en.wikipedia.org/wiki/Usage_share_of_web_browsers
5. http://en.wikipedia.org/wiki/Template:Firefox_usage_share
6. http://en.wikipedia.org/wiki/Template:Msieshare1
7. https://addons.mozilla.org/
8. http://wiki.creativecommons.org/MozCC
9. https://addons.mozilla.org/en-US/firefox/addon/363#reviews

10. http://simplest-image-hosting.net/i0-mozcc-png.png

11. http://simplest-image-hosting.net/i0-icon-png.png

12. http://simplest-image-hosting.net/i0-mouseover-png.png

13. http://simplest-image-hosting.net/i0-mouseclick-png.png

14. http://simplest-image-hosting.net/i0-highlight-png.png

15. http://simplest-image-hosting.net/i0-info-png-0.png

16. http://creativecommons.org/choose/

Additional information

Contact details

Name: Igor Lukanin

Email: mail@igor.lukanin.name

Jabber: jabber@igor.lukanin.name

Academic & development background

I'm a student of the Ural State Technical University, situated in Yekaterinburg, Russia. My major subject is the informational security of telecommunicational systems which gives me a perfect knowledge of computer systems and their interaction, as well as strong software development skills. All this is affected by a constant consideration of security requirements.

Although I had software development courses (x86 assembly, C and Java) in the university, I'm personally familiar with a much wider range of technologies (OSes, programming languages and so on). This allows me to participate in various projects, from scripting a specialized Linux distribution to create an automated wireless network detection, measurement and report tool to programming a plugin for a Jabber-server for corporate data stealing prevention.

Why Creative Commons?

I should mention I've always payed much attention to the open source world. Using in my everyday life tools provided by the open source community, I got familiar with licensing issues and I can fully estimate the importance of them.

This project is so much challenging for me, because I understand what advantages it would bring to people putting their works to the public. Just as Creative Commons licensing initiative made it easy for authors and recipients to communicate on rights reservation and waiving, this project would make it easy for web-content consumers to deal with those licensing issues.