WebSticker handbook

Overview

WebSticker is an extension for the Google Chrome web browser. It allows the user to add ‘sticky notes’ to web pages, featuring configurable position, text size and of course text content. They behave much like ‘real’ sticky notes - they can be pinned to any position and they can be dragged around.

In addition, WebSticker allows for a complete XML export, furthermore providing a XML schema file for validation of the created document.

This document is structured in two major parts: The first one describes WebSticker’s technical architecture and how it works internally, including some known issues; the second part describes the installation process.
Requirements and supported browser versions

WebSticker is only tested in combination with the so-called ‘dev channel‘ version of Google Chrome, more precisely, 6.0.472.14 dev. See http://dev.chromium.org/getting-involved/dev-channel for further details and how to get this channel. Although it is not tested and not supported in any way, WebSticker should also work with the beta and even the stable channel versions.

Known issues

WebSticker is a student project still in beta status. This is a list of some known issues:

- Web sites using HTML frames
  - There is no support for web sites using HTML frames in WebStickers. Stickers can be added, but their behavior is undefined, usually, they are not even displayed correctly. Because nowadays frames are considered to be legacy code/questionable design, WebSticker will probably never support sites which (still) use them.

- Various style/display problems on some pages
  - Due to the way WebSticker works internally (more on that later on), the stickers and the sites they are ‘pinned to’ share the same CSS, which can lead to interference. For example, sites using universal selectors (e.g. ‘html *’) regularly influence the rendering of stickers. This kind of bug is usually fixed easily - and WebSticker has of course been tested for compatibility with various popular web sites - but it is likely that some examples will occur during further testing.

- AJAX/URI change without site reload can result in duplicated stickers
  - WebSticker uses the URI of a site (window.location.href) to determine which stickers belong to which site. Usually, when the URI of a site changes, Chrome notices a reload, and our extension does the necessary management work. However, sometimes (most noticeable on google.com, after searching for something), the URI changes, including the displayed page, but Chrome does not notice the changes. Since WebSticker now saves the (still existing) stickers under the new URI, stickers are duplicated.

- Reload necessary after installing/updating the extension
  - After installing or updating the extension, the browser should be restarted or each tab with WebStickers reloaded manually, otherwise ‘strange things happen’.

- Browser action icon popup sometimes ignores clicks
  - Sometimes, Chrome simply ignores mouse clicks on the browser action icon.
Technical description / architecture

WebSticker is roughly divided into two parts, which are rather loosely coupled.

Actual sticker rendering/management

First, there is the actual sticker part, which is responsible for managing multiple stickers, rendering the stickers (by creating elements in the DOM of the currently visited web site), including necessary JavaScript and CSS.

Browser-specific part

The second part, also consisting mainly of HTML, JS and CSS is the extension part, which handles interaction with Google Chrome and is specifically written for that browser. It enables injection of JS and CSS if a site wishes to use WebSticker and also is in charge of storing and loading stickers to persistent storage, in this case, HTML5 local storage.

The following picture should make it clear how everything fits together:
HTML5 local storage and how WebSticker uses it

What is local storage?

Local storage is part of the HTML5 specification and is already implemented by almost any modern browser, including Google Chrome. It allows web applications to access a persistent, local storage system based on key/value pairs. The storage quota is usually 5MB, although the specification describes this limit as ‘arbitrary’ and advises browser vendors to prompt users if the quota is in danger of getting exceeded, to allow for an increase.

How does WebSticker use it?

In theory, there are two ways for a browser extension to use this local storage: It could use the site-local (injected) JS to load and save to/from the site-local storage (i.e. from local storage belonging to orf.at, apple.com, etc.) or it can manage everything in the extension-local storage. It becomes obvious pretty quickly that the second variant is the better one, considering that it is the only way to allow an export-all-stickers functionality to work. Therefore, WebSticker uses it, saving all stickers as JSON-encoded strings with a key following the pattern ‘webSticker_[uri_of_page]’, without brackets.

Installation

Packed extension

The WebSticker distribution archive includes a WebSticker_extension.crx file. This is a packed and signed, ready-to-use Google Chrome extension. It can be installed by simple dragging it to the browser. After confirming the installation, all tabs should be reloaded or the browser restarted (see list of known issues); after that, WebSticker is ready.

Unpacked source files

Alternatively, especially interesting for developers, the ‘source’ of the extension (everything in the src/extension directory) can be loaded manually by enabling developer mode from the extension-managing page in Chrome (chrome://extensions/) and using the ‘Load unpacked extension’ button. This enables ‘reload’ functionality, which is necessary for reasonably rapid development.
How to use

After installing the application, Google Chrome will offer a new ‘yellow note’ icon right next to the address bar. A click on it offers all options WebSticker has to offer, they are pretty self-explanatory and are not covered here in greater detail.

Have fun exploring :)

About this project / document information

This project (the WebSticker Google Chrome browser extension and all documentation) was implemented as student project for the Vienna University of Technology (http://www.tuwien.ac.at), more specifically for the course ‘Informatikpraktikum 1’ under the supervision of Martin Treiber, MSc (http://www.infosys.tuwien.ac.at/staff/treiber/).

Sites tested

The following list comprises a small part of the collection of sites which were tested with WebSticker:

- www.orf.at
- www.derstandard.at
- www.cnn.com
- www.youtube.com/watch?v=k9ez7iNjoVg
- www.apple.com/mac/
- tv.ign.com
- www.gmx.at

Document info

Version: 1.0

Author: Martin Leonhartsberger-Schrott, 0505128