Experimental Data for the A?B*A Pattern in CSS: Inputs and Outputs.

A Dataset Artefact for "The A?B*A Pattern: Undoing Style in CSS and Refactoring Opportunities it Presents"

@leonardpunt
Leonard Punt
University of Amsterdam, The Netherlands
Q42, The Netherlands

@sjoerdvisscher
Sjoerd Visscher
Q42, The Netherlands

@grammarware
Vadim Zaytsev
University of Amsterdam, The Netherlands
Raincode, Belgium

The dataset can be download from the following web location:

- http://leonardpunt.github.io/masterproject/dataset-and-results.zip

It is available under the MIT license.

Facebook	Pinterest
YouTube	Reddit
Twitter	Tumblr.com
YahooMail	Wordpress.org
Outlook.com	Vimeo.com
Gmail	Igloo
Github	Phormer
Amazon.ca	BeckerElectric
Ebay	Equus
About.com	ProToolsExpress
Alibaba	UniqueVanities
Apple.com	ICSE12
BBC	EmployeeSolutions
CNN	SyncCreative
Craiglist	GlobalTVBC
Imgur	Lenovo
Microsoft	MEC
MSN	Staples
Paypal	MSNWeather
9292.nl	Rijksmuseum.nl

TABLE I
SELECTED SUBJECTS.

INSIGHTFULNESS

- **Timely**
  Analysis of web applications is a timely topic both due to the challenges it presents and due to widespead use of its use.
- **Educating**
  Replications on website analysis papers are usually next to impossible since most modern active vendors change their applications continuously and deploying new versions up to 50 times a day [4]. Providing both the dataset and the tool allows to easily retarget them for the next target.

USEFULNESS

- **Purpose**
  Our dataset provides a timed snapshot of a collection of web applications.
- **Awkwardness**
  The dataset is not just complicated to collect, but impossible if anyone wants an honest snapshot from 2015.
- **Cost**
  The dataset is stored as a ZIP archive with an intuitively understood directory structure, the complete sets of DOM states shown as one HTML document, and files that contain experiment results.

USABILITY

- **Understandability**
  The contents of the ZIP file should be easy to understand, but they are also explained on the second page.
- **Tutorial**
  The tutorial on the next page is focused on the artefact creation process and its peculiarities rather than on providing checklists, recipes and examples.
- **Executability**
  Non-executable artefact.

REFERENCES

[1] D. Mazinanian, "Dataset for FSE'14 submission." [Online]. Available: http://users.encs.concordia.ca/~d_mazina/papers/FSE'14/

[2] D. Mazinanian, N. Tsantalis, and A. Mesbah, "Discovering refactoring opportunities in cascading style sheets," in *Proceedings of the ACM SIGSOFT International Symposium on the Foundations of Software Engineering (FSE)*, 2014.

[3] A. Mesbah, A. van Deursen, and S. Lenselink, "Crawling Ajax-Based Web Applications Through Dynamic Analysis of User Interface State Changes," *ACM Transactions on the Web*, vol. 6, no. 1, pp. 3:1–3:30, 2012.

[4] D. Schauenberg, "Development, Deployment & Collaboration at Etsy," in *QCon London*, 2014, https://qconlondon.com/london-2014/london-2014/presentation/Development,%20Deployment%20&%20Collaboration%20at%20Etsy.html.

Artefact Description

This dataset is used to detect undoing style in CSS code. In total, this dataset contains 41 subjects. Each subject has its own folder, which contains:

- The captured states
- A `states.html` file, is used to load all captured states in one document.
- A folder called `results`, which contains the detected undoing styles, the refactored style sheets and the detected semantic changes.

The file `states.html` is used as input for our tool.

I. Selection of subjects

In order to select representative real-world web applications, we used the empirical data that is used in the study conducted by Mazinanian et al. [2]. This data set includes 38 randomly selected, as well as author selected online web applications. It includes a subset of the top-100 visited web sites based on the Alexa ranking. Furthermore, web applications developed by companies considered leaders in web technologies, such as Facebook, Yahoo!, Google, and Microsoft, are added. This data set is available online [1].

Besides the 38 subjects from the study of Mazinanian et al., two web applications developed by Q42 have been studied as well.

The 41st subject is a new version of the subject 'Gmail' from in the original data set. More information on this can be found in section III.

The complete list of the selected systems is shown in Table I.

II. Extraction of CSS styles and DOM states

Mazinanian et al. used the dynamic analysis features of Crawljax [3] to dynamically capture different DOM states of a web application. These DOM states are persisted to HTML files.

The HTML files contain inline and internal style sheets, together with links to external style sheets. In order to extract the external style sheets Mazinanian et al. developed an external CSS file extractor plug-in for Crawljax.

Note that the references to the external style sheets in the HTML documents need to be updated, because the extracted external style sheets are in a different location.

III. Issues with dataset

There are some problems with the dataset of Mazinanian et al. First, there are two sites with an incorrect name: 'Apple.ca' is actually 'Apple.com' and 'MountainEquip' is 'MEC'. In our dataset we renamed these two sites to their correct names.

Next, the DOM state that is captured for the subject 'Gmail' is not usable. The problem is that a cookie is missing, therefore a visitor will be redirected to a non-existing page. We mitigated this problem by removing the redirect from the source code. Besides that we captured the intended state ourselves as well. These two subjects are named 'Gmail original' and 'Gmail fixed'.

Furthermore several external style sheets are captured incorrectly, resulting in an empty style sheet. We chose to include these style sheets as is in our dataset, in order to stay as close to the original dataset as possible. We did investigate why the style sheets were empty, our findings are listed below:

- 'MEC', the file 282a12.css. This is a CSS file in a `<script>` tag, probably an error in the document.
- 'About.com', the files 18b91843bb4bcb07c2ba68a01bbb8a02b9eb4c50.css and 54c660b14dd08ca6b408f07de1f5080d251a4ef2.css. The original URLs for these files do return style sheets, so probably an error occurred while fetching these style sheets.
- 'Apple.com', the file 91cab95bff78fd3800625c0789cd87c0e4180299.css. When we tried to retrieve this file, a File Not Found error was returned. However, we do not know if this is also the error that occurred when the original dataset was collected.
- 'GlobalTVBC', the file 61dc696007ca3c1aeb54a2d0bab8ea932de60e21.css. When we tried to retrieve this file, a Forbidden error was returned. However, we do not know if this is also the error that occurred when the original dataset was collected.
- 'Alibaba', the files d8e76b82abbae61a4a89fb4000324c09b6413719.css and f20bbdc283941382159c1e12d655f54f1bdc68c2.css. The original URLs for these files do return style sheets, so probably an error occurred while fetching these style sheets.
- 'SyncCreative', the file 0e6fedfab56593cd6d0ea0bd8dee80454585b2af.css. When we tried to retrieve this file, a Forbidden error was returned. However, we do not know if this is also the error that occurred when the original dataset was collected.

Finally, some unused style sheets have been captured. We have excluded these style sheets from our dataset. The files are:

- 'ProToolsExpress' the files: 167d8fb47eb42d1f908ba5d4141a34f4333b18c9.css, a0f24af3ff23a278289a1f50a5d6b6598a76415a.css, b5f14f865786216f67ae8a41ab1c1774aa955334.css, c7a368297aab3abe7d74e0ae421fc38dd18a048f.css and e54053a51b1eb8ae62e9bc76ad9351ea4f1d4c89.css. These files are not linked in any document.
- 'SyncCreative' the file style.css. Since this file is a duplicate of b4ad21b4c1ba99451234f5e3da9a501a50dac0fe.css.