One Way to Select Many (Artifact)*

Jaakko Järvi[1] and Sean Parent[2]

1 Texas A&M University
College Station, TX, USA
jarvi@cse.tamu.edu

2 Adobe Systems Inc.
San Jose, CA, USA
sparent@adobe.com

Abstract

The artifact is built around the *MultiselectJS* library that implements the multi-selection semantics presented in the companion paper. The purpose of this JavaScript library is to help programmers add multi-selection to Web user interfaces with ease. The artifact includes a demo application showcasing the use of the library, documentation of the library API, and a tutorial on how to use the library. The library clearly separates the reusable aspects of multi-selection from the application specific aspects. The latter are collected into an object referred to as the *selection geometry*. The complex task of implementing a full-fledged multi-selection feature reduces to a considerably simpler task of implementing a selection geometry. The demo application implements several different selection geometries. It supports the claims of the companion paper that the library considerably lessens the task of implementing multi-selection. Following the guidance of the tutorial, users can build their own multi-selection features.

1998 ACM Subject Classification D.2.11 Software Architectures: Domain-specific architectures; D.2.13 Reusable Software: Reusable libraries

Keywords and phrases User interfaces, Multi-selection, JavaScript

Digital Object Identifier 10.4230/DARTS.2.1.7

Related Article Jaakko Järvi and Sean Parent, “One Way to Select Many”, in Proceedings of the 30th European Conference on Object-Oriented Programming (ECOOP 2016), LIPIcs, Vol. 56, pp. 14:1–14:26, 2016.
http://dx.doi.org/10.4230/LIPIcs.ECOOP.2016.14

Related Conference 30th European Conference on Object-Oriented Programming (ECOOP 2016), July 18–22, 2016, Rome, Italy

1 Scope

The artifact demonstrates that the multi-selection semantics that the companion paper defines can be directly translated to a practically useful software library. It shows that multi-selection features built using the library behave as expected and support all common aspects of multi-selection, and more. Users can verify this by experimenting with the demo application. Further, each multi-selection feature requires a relatively small amount of code, and the code is not complex. Users can verify this by implementing their own multi-selection features using the library.

* This work was supported in part by NSF grants CCF-0845861 and CCF-1320092.

2 Content

The artifact package includes:

- The MultiselectJS JavaScript library described in Section 4 in the companion paper. The library is distributed as one (non-minified) JavaScript file.
- A demo application built using MultiselectJS. Partial snapshots from the application are shown in the companion paper's Figure 8. The application allows for experimenting with the selection features described in the paper. It also demonstrates how the same selection semantics applies equally in many different selection geometries.
- A tutorial that describes how to implement multi-selection features using MultiselectJS.
- The API reference of MultiselectJS.
- Documentation of the source code of MultiselectJS (in HTML). The library was developed in literate programming style using *Emacs org-mode*; the documentation of the source code is generated from literate programming source files (org-files), which are not included. The documentation of the source code is mainly targeted for someone extending or modifying the library itself. It explains design decision and other details about the implementation.
- An `index.html` file that links to all parts of the artifact.

We do not package the artifact with a virtual machine as the only dependency is a web browser.

3 Getting the artifact

The artifact endorsed by the Artifact Evaluation Committee is available free of charge on the Dagstuhl Research Online Publication Server (DROPS). The latest version of MultiselectJS and its documentation are available on the library's GitHub pages: `https://github.com/HotDrink/multiselectjs`.

4 Tested platforms

The code in the artifact is tested with Google Chrome 50.0, Safari 9.1, and Firefox 35.0 browsers on OS X (El Capitan). It should work on any modern (in 2016) browser. Some ECMAScript 6 features are used. User-defined OS-wide key bindings may conflict with some of the key bindings in the demo application and in applications that are part of the tutorial.

5 License

The MIT License (MIT) (`https://opensource.org/licenses/MIT`)

6 MD5 sum of the artifact

a4efd9d47cbcf17304fc94ae022b827d

7 Size of the artifact

168 kB

Acknowledgements. The authors wish to thank Anirudh Ramanathan for discussions about details of the design of MultiselectJS and for helping out with testing the library.