

Alternative extension UI - “Sidebar”

Proposal Date

February 3, 2015

Who are the primary Eng and PM contacts for this API?

Eng: Benjamin Kalman - kalman@chromium.org

Ben McCann (Co-founder @ Connectifier, ex-Google) - benjamin.j.mccann@gmail.com

What are the relevant bug(s)?

crbug.com/477424

Which team will be responsible for this API? (team email address)

chrome-extensions-team

Connectifier, Inc. and Ben McCann - benjamin.j.mccann@gmail.com

What’s a reasonable Chrome milestone target for landing most of the code on Trunk?

The code has been authored and submitted for review. The code could be landed in M46.

Overview

The proposed sidebar would be made available for use by extensions. An extension could show the sidebar when the toolbar button is clicked.

Please see the [Sidebar PRD](#) for more details

Do you know anyone else, internal or external, that is also interested in this API?

Please see the [Sidebar PRD](#) for details

How would you implement your desired features if this API didn't exist?

Please see the [Sidebar PRD](#) for details

Is this something that could/should be part of the web platform, or an existing chrome.* API?

This makes sense as an extension API and not as part of the web platform. There is some small possibility that you could have a web standard API that allows sites to modify their preferred display surface, for example by a new [display-mode via the web manifest](#), and/or a new [sidebar option to window.open](#). However, it would be extremely challenging to allow the open web to modify browser UI in a manner which is an overall positive for users, and we are not in support of the idea.

Does this API expose any functionality to the web?

No

Do you expect this API to be fairly stable? How might it be extended or changed in the future?

It may be extended, but the proposed part is not expected to change.

If multiple apps/extensions used this API at the same time, could they conflict with each other? If so, how do you propose to mitigate this problem?

Only one extension at a time will be allowed to show a sidebar. If a user click the browser action toolbar button to open a sidebar for a different extension then any existing sidebar in the current window will be closed.

List every UI surface belonging to or potentially affected by your API:

The proposed API allows Chrome extensions to access and control a sidebar panel - a per-window split-pane HTML container to the right (to the left in RTL environment) of the main page content with the ability to resize horizontally.

Actions taken with app/extension APIs should be obviously attributable to an app/extension. Will users be able to tell when this new API is being used? How? Can it be spoofed?

The sidebar will be shown only when the toolbar button is clicked. This will make attribution very clear.

If additional attribution is required, it would also be possible to add a header to the sidebar with the extension title on it. We do not believe this is necessary and so a simpler UI should be preferred.

Does this API impose any requirements on the Chrome Web Store ?

No.

Does this API have any interaction with other Chrome APIs? Does it impose any restrictions on other APIs that can be used in conjunction?

No.

How could this API be abused?

One concern would be allowing Chrome's UI to become cluttered. However, sidebars will only be shown for users that have requested them by installing an extension. Additionally, only a single sidebar will be shown at a time. Firefox added [sidebar support](#) to their JavaScript API and has long had sidebar support in their XUL API. There are no cases we are aware of where sidebars have become a problem.

Imagine you're Dr. Evil Extension Writer, list the three worst evil deeds you could commit with your API (if you've got good ones, feel free to add more):

1. Show a sidebar filled with ads

- a. The sidebar will only be shown when the toolbar button is clicked. It will be easy to close the sidebar and obvious as to the low-value extension which can then be uninstalled and given poor rating on the app store
2. Show an oversized sidebar
 - a. We may want to limit max width to some percentage of the screen, and again it will be clear which extension created the sidebar
3. Show a message in the sidebar purporting to be from the browser. E.g. “Your Flash player is out-of-date. Click here to update”, which links to malicious binary
 - a. The sidebar will only be shown when the toolbar button is clicked. This means the sidebar is a less attractive target for these types of tricks than popups (where Chrome legitimately shows messages) or injecting a script to edit a page’s HTML

What security UI or other mitigations do you propose to limit evilness made possible by this new API?

The sidebar will be shown only when the toolbar button is clicked. We should make it clear which extension created a sidebar, so that bad actor extensions can easily be removed and given a bad rating on the app store.

We plan to only allow one sidebar to be shown at a time. This provides better usability and limits the amount of UI space that can be used by sidebars. We may want to limit max width to some percentage of the screen.

Finally, we’d like to reiterate that this API would protect users from the privacy issues discussed earlier that are created when extensions try to create their own sidebar by injecting an iframe into the page.

Could a consumer of your API cause any permanent change to the user’s system using your API that would not be reversed when that consumer is removed from the system?
No.

Draft Manifest Changes

```
"browser_action": {
  "open_in_sidebar": true    // optional
},
```

Open questions

The main questions raised in this proposal:

- Is there a certain width of the browser window we’d like to limit a sidebar to? 25%? 33%? 50%?
- Should we allow the location of the sidebar to be configurable? (e.g. left vs. right side of page)