

# API for enabling/disabling switching tabs with the mouse wheel - chrome.tabs.scrollEventChangesTab

I would like to propose an addition to the chrome.tabs api that allows to disable/enable scrolling through tabs using the mouse wheel.

Currently it is possible to use the mouse wheel to switch tabs in chrome on linux. This feature is disabled for windows and chromeos. There are frequent reports from people who would like to have this feature enabled or disabled on their platform. For this purpose I would like to add a ChromeSetting chrome.tabs.scrollEventChangesTab.

## Proposal Date

January 25, 2017

## Who are the primary Eng and PM contacts for this API? (email addresses)

dullweber@google.com

## What are the relevant bug(s)?

<https://crbug.com/680499>

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

It is only a small addition to an existing api.

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

There are multiple bug reports from users (e.g. <http://crbug.com/383746> and <http://crbug.com/407051>) requesting this functionality.

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

M58

## Overview

This api allows to enable or disable switching tabs using the mouse wheel.

## Use cases

The api is used to give users control over tab scrolling via an extension.

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

Instead of an api this could also be implemented as a setting or flag but that was already declined in several bug reports as the number of settings should be kept low.

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

**API?**

This will be part of the existing chrome.tabs.\* api.

**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?**

Yes, it should be very stable.

**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?**

If multiple extensions use this api at a time they should make sure to use get the current state of the setting before showing it to the user or listen to changes to this setting in case it was changed by a different extension.

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

There will be no UI change. The default behaviour of tab scrolling will stay the same as it is currently on each of the platforms.

**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?**

This won't be possible as there is no UI indication for tab scrolling. But there is also no UI indication e.g. if an extension switches tabs because of mouse gestures on a website.

**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?**

I don't see any concerns of abuse.

**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. Randomly disable and enable tab scrolling and annoy the user? I don't see security issues here.

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

Users will notice this bad behaviour and uninstall/report the misbehaving app. No other damage can be done with this api.

**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?**

The api will be able to change tab switching behaviour. If the state is stored as a preference for the profile.

I could store which extension changed this preference last and revert to the default behaviour when the extension is uninstalled, should this be done?

**Draft Manifest Changes**

No new permissions required.

**Draft API spec**

`chrome.tabs.setTabScrollingEnabled(boolean tabScrollingEnabled, function callback)`

`chrome.tabs.getTabScrollingEnabled(function callback)`

[http://developer.chrome.com/\\_patch/2625113002/extensions/tabs.html](http://developer.chrome.com/_patch/2625113002/extensions/tabs.html)

→ this link doesn't work? What am I doing wrong?

Here is the pending cl:

<https://codereview.chromium.org/2625113002/>

Here is the modified tabs.json in my cl:

<https://codereview.chromium.org/2625113002/diff/40001/chrome/common/extensions/api/tabs.json>