



paulirish commented on 17 Dec 2013

Owner

- workspaces. how do they work
- WS & Sass
- LiveEdit is a gateway drug into workspaces, but its also a complement when you're using them
 - its not clear how many people start with which feature
- include links to Remy's stuff.
- what "[VM] file (322)" means
- When changing a file, why it adds * to the end of filename and what that means
- Explain saving: Ctrl/Cmd + S vs right click and Save/Save as...



dr4b was assigned on 17 Dec 2013



jaredwilli commented on 17 Dec 2013

Owner

Perhaps explain the other available shortcuts too:

Cmd+/
Opt+Delete

Also, something I've noticed since I almost exclusively use workspaces now. When I have a file open, say main.js. If i have a breakpoint on a line that is hit when I hover over something, there's another instance of main.js that opens up, which if I try to make changes in and save, it's not actually saving. I have to switch back to the other main.js file I was editing and make the change there. I'm not sure what's up with that. Probably has something to do with what VM is. Although, sometimes a file has VM next to it and sometimes it doesn't. Curious about this.



SchizoDuckie commented on 19 Dec 2013

I've experienced this same thing. It seems like the mapping is not 100% perfectly matched in some cases (when recompiling and dynamically swapping the function contents?

I've made what @jaredwilli mentions happen by adding a breakpoint, breaking on it, altering code, and then saving and kicking off the auto-recompilation. A new editor tab opens and it has a [VM] tag in the title.



paulirish commented on 19 Dec 2013

Owner

@SchizoDuckie hah thanks for coming around these parts! :)

that's a good bug report. I'll speak to our workspaces engineers to look into it.



addyosmani commented on 20 Dec 2013

Owner

Suggestions:

- Features: add folder, refresh, create new file, delete, ignore lists.

- Document scenarios where it makes sense to author an app directly inside of DevTools Workspaces vs where it doesn't. Currently not really talked about anywhere.
- If linking/embedding video, verify that the features shown/UI haven't massively changed. If they have it may be worth adding a note.
- Link to DevTools Terminal article? discusses a workflow using Workspaces + terminal extension



paulirish commented on 20 Dec 2013

Owner

@dr4b here's my ideas for how to best cover the feature

- Why workspaces? Because copy pasting your final changes back into another editor is slow. And more benefits come when we merge the the editor with the devtools.
- What having a mapped workspace is like. What development feels like and why everything is instant and persisted (styles pane, sources (css & JS, html) and augmenting new files etc)
- Concede that you may have editor familiarity or editor features you can't give up for 100% of the time. point to the remote debugging clients examples page.
 - Devtools doesnt need to be your primary editor to use workspaces. But it will likely make sense to use as your editor during a step debugging session or iterating quickly via Live JS recompilation
- How to use it
 - dev server only. not really usable for production files your filesystem can't change. works for file:/// but you'll need to do the adding work anyway
 - add the workspace. MAP IT. always map it, imo.
 - persistence. it all works
 - ignoring files. managing files (add/create/delete)
 - cmd-o and search all files
- If you change a file but dont hit save, (or if you change a JS file that's not mapped), you may see a [vm] file that looks very similar come up. That is the version of the file that the JS engine is processing. Until you hit Save workspaces will not merge the two.. so do that.
- Live JS Recompilation (née liveedit)
 - what it means
 - it doesnt work for everything, but function redefinitions are its break and butter. also you dont get visual feedback if it totally worked or not, so... be understanding :)
 - a dirty buffer gets you a * in the tab bar. hitting save (cmd-s) does two things:
 - a. patches it to the JS engine for recompilation
 - b. saves it to disk if you've set up a workspace. if you havent' set up a workspace, the background may change colors to indicate the change wasn't persisted but it was however sent to v8.
 - You can always right click and save any file in Sources to save to disk, but it's a file-by-file operation. Workspaces allows you to do this for an entire project folder, which is probably what you meant.
- Workspaces with CSS Preprocessor support
 - we have css sourcemaps which means you can map from styles pane CSS back into what you wrote as sass/less/etc
 - now when you map it back you can change it and save to disk via workspaces
 - of course you'll have to have a daemon on --watch that recompiles for you, but devtools will pick back up the new .css files and update them.
 - we should have a video for this. i'll do it.

Doing more

- devtools terminal
- other things that augment the "devtools as an editor" workflow.
- how a mapped workspace is actually great while using an external editor. basically like Emmet LiveStyle but just on every devtools focus it brings in the new version of your file. See internal email from John J Barton talking about the Editor->Workspaces->recompilation flow. Also works great for CSS.

caveats:

- can't go from DOM back into HTML. you probably wouldnt want to. but editing the HTML file is easy..

<http://remysharp.com/2013/07/18/my-workflow-v3-full-coding-stack/> video shows an unmapped workspace and how you can edit backend files easily. In the comments i point out the advantages to mapped workspace.

<http://gregrickaby.com/turn-chrome-developer-tools-into-an-ide/> this kinda shows what the story is for workspaces in the larger scheme of things.



paulirish commented on 24 Apr 2014

Owner

Initial docs published: <https://developers.google.com/chrome-developer-tools/docs/workspaces>



Protector1 commented on 29 Apr 2014

This all sounds wonderful, but are pure HTML files still used in contemporary web development? In these times of Dru pals, PHPs and Node.js, I can't imagine how many use cases there are for having static HTML pages. Or is this feature solely intended for editing client-side script and css?



hoschi commented on 29 Apr 2014

Already a good document. In my opinion the "Refresh" section should give more details what happens and when, when files get changed outside of Chrome. What's really unexpected is that only the file of the open tab (sources view, not browser tab) gets recompiled. When I change more files I must visit each tab in dev tools to recompile them. Is it possible to recompile all changed files at once, not only the file of current tab?



Garbee commented on 25 Jun 2014

Owner

@pavelmaximov Sounds like something to bring up on the [Issue tracker](#) for the project. This is just a documentation repository.



paulirish referenced this issue on 9 Aug 2014

[Proposal] New section - Workflow #125

5 of 22 tasks complete

Closed



thomasvs commented on 12 Aug 2014

I'm trying to understand how the mapping from network source to local file system works. As soon as I right-click to map a single network source to a local file, it tells me I should reload. When I do, the source for the webserver disappears forever (presumably because it has been mapped). That implies that chrome figures out some kind of relative mapping between two files, and applies that mapping to all files. Is that correct?

However, there are a lot of cases where the server's hierarchy of resources does not match the file system's; in my case for example, /style is served by a node server from a different filesystem location than other parts. Hence, it looks like the styles pane cannot show me the source file name anymore for something I'm inspecting.

Is there a way around this, or am I supposed to make sure the filesystem hierarchy completely matches the server's?



breck7 commented on 28 Aug 2014

The normal scenario works for me, but what about the scenario where I am editing a remote file and don't have the remote file on my local disk? I expect to be able to map any arbitrary url to a local file.

For example:

- Imagine I am loading jQuery from "<http://somecdn.com/jquery.js>".
- I create a file "C:\myjquery.js" and put my own Javascript in there.
- Now when the page requests "<http://somecdn.com/jquery.js>" I want to load "C:\myjquery.js" instead

This does not work. Instead I get this error:

Workspace mapping mismatch

The content of this file on the file system: [path]
does not match the loaded script: [path]



Garbee commented on 28 Aug 2014

Owner

@breck7 That is the exact thing issue [#151](#) is open for documenting. Currently what you want to do isn't possible with the DevTools. Additional extensions are required.

In the other issue is a link to a bug report. You can star that issue to let the developers know you are also interested in the feature addition. Please do remember to not comment with simply +1 or other noise. Stars are sufficient to get things prioritized.



breck7 commented on 28 Aug 2014

Done! Thanks @Garbee!



tereshkovych commented on 2 Oct 2014

Hi.

Can someone explain how to map one file into two workspaces

One workspace - web server folder path

Second - source code

I want to update both files when i edit source in chrome.

Is it possible?



Garbee commented on 3 Oct 2014

Owner

@tereshkovych Not by default. You need to have the remote folder mapped locally via the OS then add that to workspaces. That way the workspace updates the "local" file that is actually the remote mapping. Or have some kind of auto-sync operation going on. Either way, it can get pretty involved depending on your server configuration and operating system on each, so that is pretty far out of the DevTools documentation scope.



tereshkovych commented on 3 Oct 2014

@Garbee Thanks for clarification!



skube commented on 29 Oct 2014

So, editing while in Elements tab automatically saves, while editing within Sources tab does not? And modification history does not show after a file/folder has been mapped? Am I understanding correctly?



297179121 commented on 4 Nov 2014

chrome version 38+

I like workspace of chrome, This is very cool! but I have some questions

- 1、 My project is GBK not UTF-8
 - 2、 My javascript file like this `<script type="text/javascript" src="x.js" charset="GBK"></script>`
 - 3、 The file show normal before I use [add folder to workspace] in 【source】 panel
 - 4、 The new create file show garbled after I use [add folder to workspace] in 【source】 panel
- Because of it is garbled so... Prompt Workspace mapping mismatch why??

I need you help! Thanks!



Garbee commented on 4 Nov 2014

Owner

@297179121 Sounds like an issue that needs to be addressed in the [bug tracker](#).

@skube Sources panel does not automatically save while Elements does. This is because Elements is

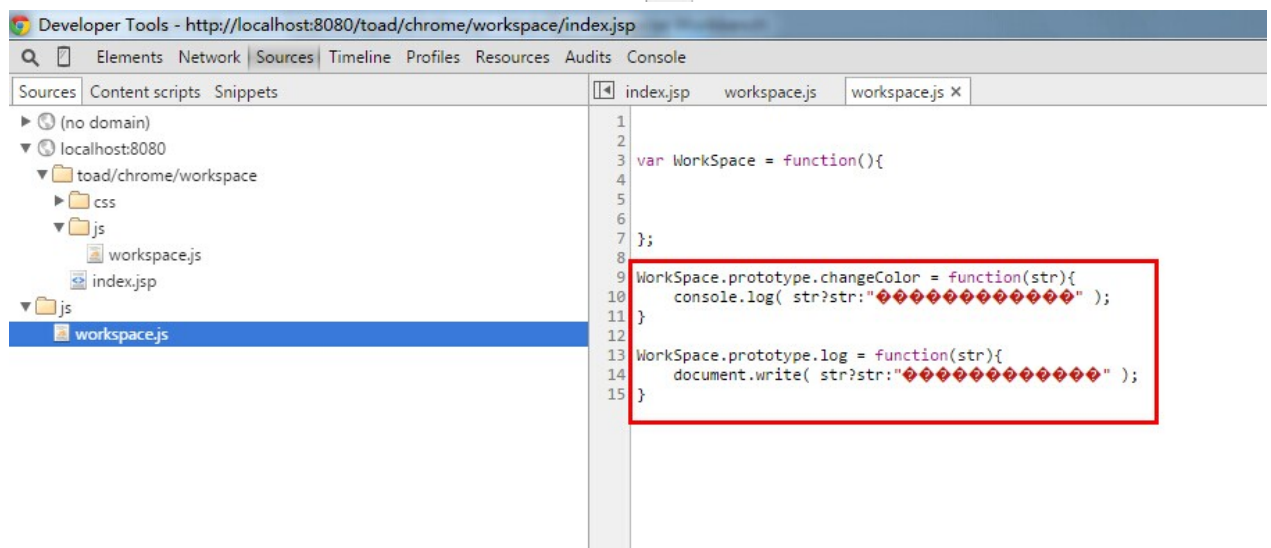
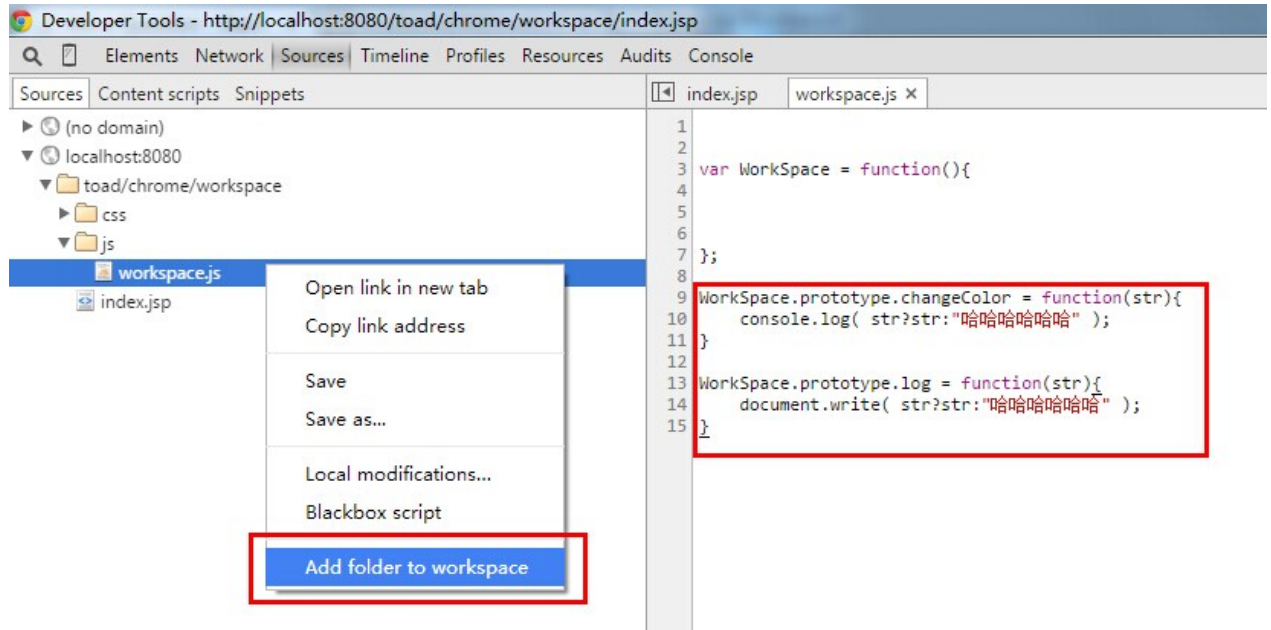
editing the DOM live as the page runs while Sources is editing the actual source file. Modification history should show after mapping (If it doesn't a bug should be filed for this to get added if it doesn't exist already.) But it only remembers history for the current editing session.



297179121 commented on 5 Nov 2014

@Garbee

I think I did not explain understand, so please look images



paulirish commented on 5 Nov 2014

Owner

@297179121 looks like a bug, yup! Can you file something at crbug.com and attach those images?



297179121 commented on 5 Nov 2014

@paulirish OK! Done!



skube commented on 6 Nov 2014

@Garbee Perhaps there is a technical reason way above my head for how the Elements and Sources panels work. This seems intuitively backwards to me as a user. Mucking about in the Elements panel seems the place to experiment with CSS without repercussion. While Sources seems like the place to make an actual change to the code. Having modifications auto-save while in Elements always trips me up and I have to hunt down all my modified source files and revert each change individually. Having an option not to auto-save while in Elements would be 👍

I guess there is something wrong on my end though since Modification history doesn't appear and thus I can't revert local modifications within Chrome.



SchizoDuckie commented on 6 Nov 2014

I'm with @skube here: I've also experienced (unexpected) auto-saves while fiddling with styles in the elements panel. Perhaps this should be an opt-in?



 **Garbee** commented on 6 Nov 2014

Owner

Oh, you mean edits in Elements are autosaving. Yes with workspaces they probably should do that. It is one of the benefits of using workspaces for your workflow. If you feel there is a problem with the way it works then please file a bug on <http://crbug.com> so a discussion about the functionality can be had with the developers to see if something should be changed.

Think of that this way though. Once the tools have a mapping to source, it is trying to map as much back as possible to help you. This way it doesn't matter where the edit is made, it will persist. There is no history in elements side from undo which is cleaned after a page refresh.

If the docs don't make this functionality clear enough then I should surely look into it soon.

...



Bnaya commented on 6 Nov 2014

I have a suggestion to enhance the workspaces but i'm not sure where to submit it, sorry if this is not the place.

It would be nice to have project file with settings for the workspace(or maybe .rc file) that can be checked in to SCM. that chrome will read automatically when you add folder to workspace. it can contain url mappings, excludes, source maps settings and more settings in the future.



 **Garbee** commented on 6 Nov 2014

Owner

[Http://crbug.com](http://crbug.com) this is the official bug tracker. It is the place to submit bug reports and feature requests.

...



eslachance commented on 4 Mar 2015

Am I correct in thinking that it's still not possible to create local copies of remote files by folder? I have to open each file and hit Save As, one after the other?



Garbee commented on 4 Mar 2015

Owner

Correct. Saving whole folders is not currently supported.



chisholmd commented on 27 Aug 2015

When I added persistence I seem to have lost the ability to set break points



dr4b was unassigned by **chisholmd** on 27 Aug 2015



eslachance commented on 27 Aug 2015

@Garbee Will it be eventually possible? Because that would be an awesome feature.



Garbee commented on 27 Aug 2015

Owner

@eslachance I don't think there are any current plans for the team to add the functionality. Best thing to do is submit a feature request on the [issue tracker](#) against DevTools and see what they say.



Write

Preview

Styling with Markdown is supported

Leave a comment

Attach files by dragging & dropping, [selecting them](#), or pasting from the clipboard.

Comment