Team Edross

Product Deliverable

Contributors:

Marco Nicoletti

Jesus Torres

Taylor Woehrle

Michael Palme

Joe Wedley

#### CS 4260 - Software Engineering Practices

#### Dr. Jody Paul

#### 25 April, 2018

[**Issue #393: If no threads are hidden, "hiddenThreads" doesn't really need to be part of the URL**](#_yp4bbi4pk922) **3**

[Issue Description](#_bennlun1rfl4) 3

[Code Fix](#_kwk64goz3ife) 3

[Visual Fix](#_1pnjs3xol4bg) 4

[**Issue #422: Display Duration of Time Range in the Breadcrumbs Bar**](#_qco6hosw8uaa) **5**

[Issue Description](#_c4yaxlkcu7f2) 5

[Prototype](#_cj00l2ita4gc) 6

[Code Fix](#_y5mq34qk0zac) 7

[Visual Fix](#_mjfnrijy7j2v) 8

[**Issue #736: Hiding Threads Does Not Resize Stack Chart Canvas**](#_ylme7df7ya9l) **10**

[Issue Description](#_l8yo3qo1p10l) 10

[Code Fix](#_g97a50pcfoks) 12

[Visual Fix](#_waetftkiv4mo) 12

[Update](#_6oua4884p38x) 14

[**Issue #737: `Only Show X Thread` Causes Jumping Menu Positions**](#_avsjghq6bat4) **15**

[Issue Description](#_rmcs58vcg15k) 15

[Code Fix](#_356h6mql42dl) 15

[Visual Fix](#_b3ckwsq8haw0) 16

[**Issue #895: Do Not Show the “Only Show” Item in the Context Menu If the Profile Only has One Thread**](#_fd1bxen1agv5) **17**

[Issue Description](#_x1ssfs2x9v7y) 17

[Code Fix](#_n84iqyh86wy3) 17

# Issue #393: If no threads are hidden, "hiddenThreads" doesn't really need to be part of the URL

### Issue Description

In Mozilla Firefox's Performance tools for the web, Dev-tools/Perf.html, there was an issue of excess, irrelevant information in the URL. When users have a profile loaded with no hidden threads, they see in the URL “hiddenThreads=”. Unless if the user hides a thread, this information isn’t useful to them.

Issue report #393 was submitted that asked for this string in the URL to be taken out when it isn’t needed. Firefox Devtools Engineer Greg Tatum, GitHub handle @gregtatum, commented a file path and a line number to an object that constructs the string for the URL.

### Code Fix

Greg showed team Edross where the URL string was being made. The object that was used to generate the URL string had a property “hiddenThreads” which always got a value, even if there were no hidden threads. We removed the permanent property and added a conditional statement to only add the property when it was needed. No other changes needed to be done because the function that turns the JavaScript object into the URL string already handled the case when the “hiddenThreads” property didn’t exist.



Artifact 393-A above shows the difference in code that team Edross contributed to the performance developer tools source code. One value was taken out of the object and it is conditionally put back into the object.

### Visual Fix

Artifact 393-B above shows how the URL used to look with no hidden threads before our fix. Notice that there is nothing after the “=” in “hiddenThreads=” making this unneeded.



Artifact 393-C above shows how the URL looks like with no hidden threads after our fix. Notice that there is no longer a “hiddenThreads=” section of the URL query



Artifact 393-D above shows how the URL looks like with thread 0 and thread 3 hidden after our fix. This is the same behavior with hidden threads before our fix.

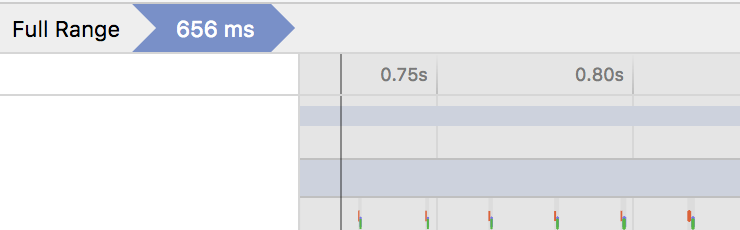
# Issue #422: Display Duration of Time Range in the Breadcrumbs Bar

### Issue Description

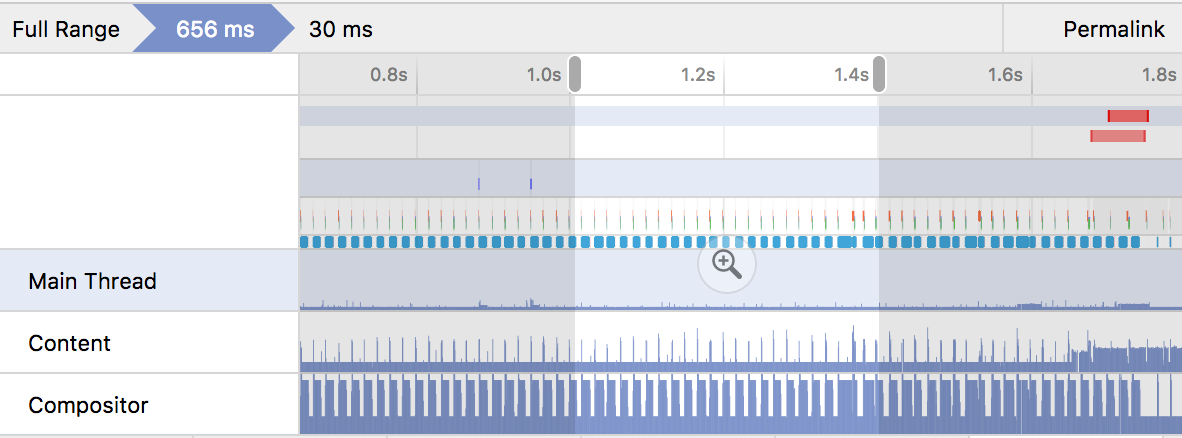
In Mozilla Firefox's Performance tools for the web, Dev-tools/Perf.html, there was an issue of missing information of the time duration. When users would select a time duration on the timeline, a blue box with the time duration would be seen over the timeline information. That time duration is useful information to the user and also the timeline information it covers is useful to the user. The duration time is only shown when the selection is in the process of being made, and not being shown when a time duration is selected by clicking an element in the timeline or after the selection was made. After selection is made, the time duration is replaced by the “zoom in” icon that, when pressed, would render the timeline with the selected new start and ending bounds.

Issue report #422 was submitted that asked for the time duration to be displayed in the breadcrumbs bar above the timeline. Firefox Devtools Engineer Greg Tatum, GitHub handle @gregtatum, commented a proposed design for how to implement a solution to issue report #422. Greg's’ comment contained two pictures of interest, artifact 422-A and artifact 422-B.

Greg's’ comment was posted on July 5, 2017 and the style of the performance tool on April 25, 2018 has several key differences to the pictures Greg showed. As of April 25, 2018, all times in the top bar have the same background color and they’re separated by a black, right facing arrow. The rightmost time is the current time span render of the timeline, this text color is blue while all times to the left of it are black. The “Permalink” button has been split into two buttons, a green one with the text “Share...” and a white one with the text “Save as file...”.



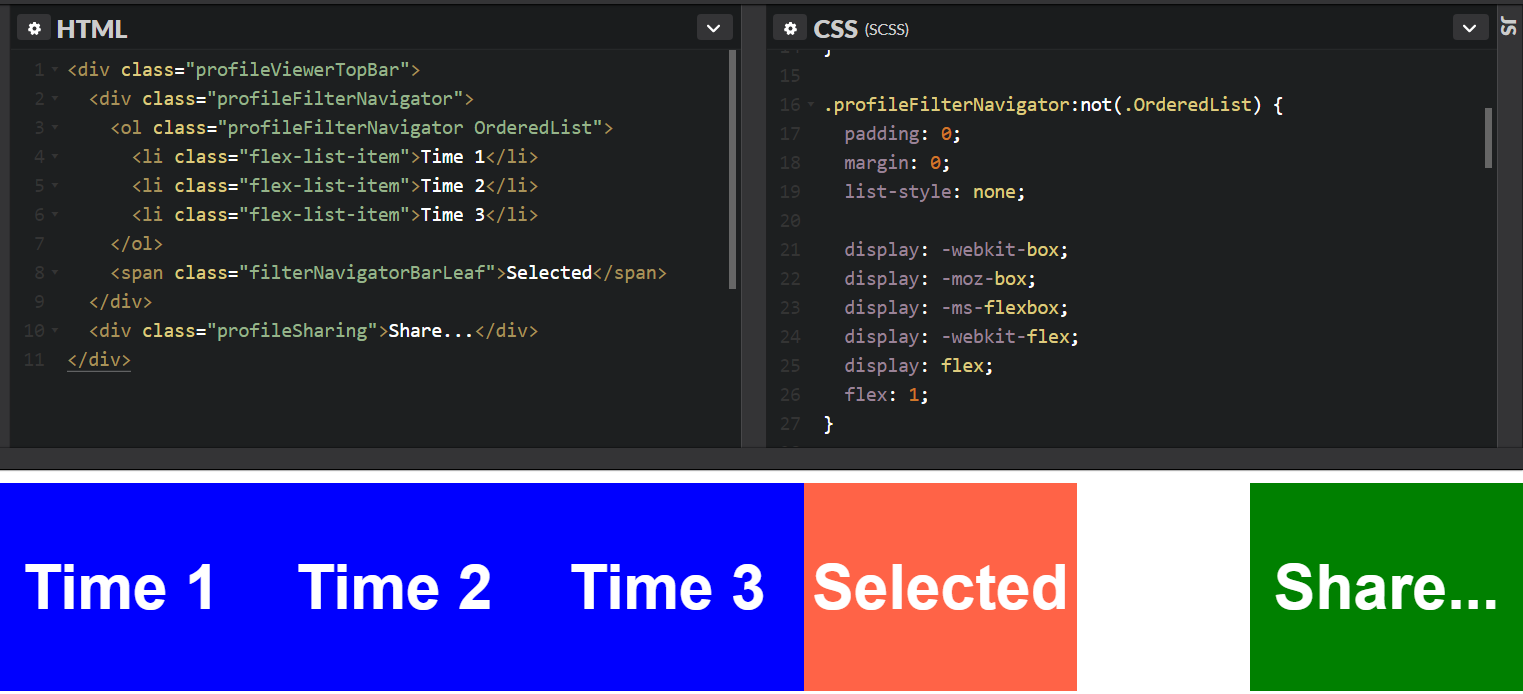
Artifact 422-A above shows the leftmost part of the top bar and the upper leftmost part of the timeline canvas at the time of Gregs’ comment.



Artifact 422-B above shows the proposed solution by Greg. It contains the whole top bar, the whole timeline canvas and the whole profile header list at the time of Gregs’ comment with the addition of the duration time in the breadcrumbs bar. Note: The “30 ms” time in the breadcrumbs bar is the selected time, it can be seen when the selection isn’t changing, and it isn’t covering anything in the timeline below it.

### Prototype

Issue report #422 is one of the more challenging issues team Edross chose to fix. So for this issue in particular, team Edross created a prototype of what the product should look like.



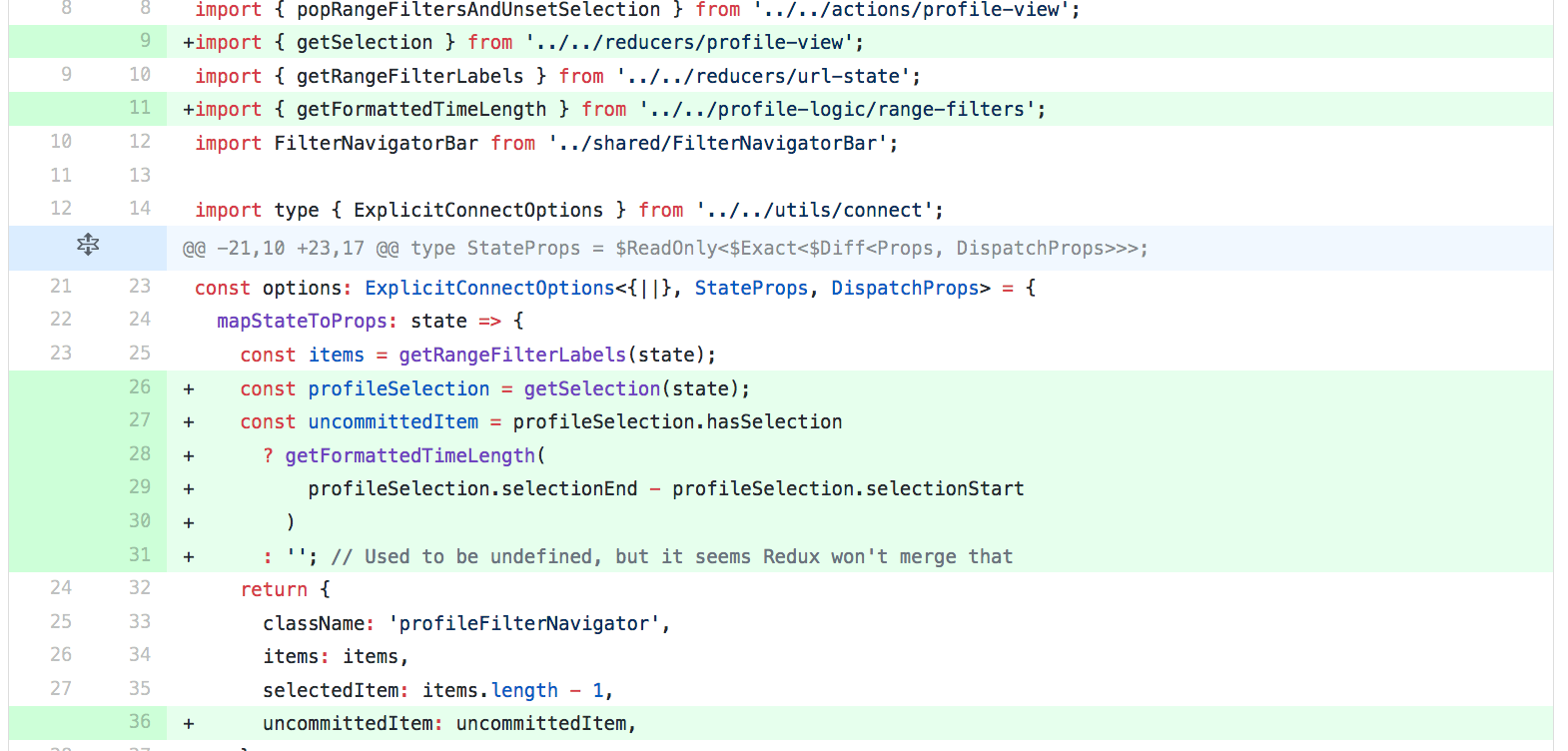
Artifact 422-C above shows the prototype HTML code and part of the CSS code used to generate the blue, red and green squares beneath it. This example was made on the website <https://codepen.io> at a Fork team Edross made at <https://codepen.io/mojac/pen/XELEaX>.

### Code Fix

When Greg added the mock up images (see artifacts 422-A and 422-B) he also included a suggestion for how to add the item needed in code. We took his suggested differences and adapted them to the current, updated, codebase as best as we could. We added a new optional property to the “FilterNavigatorBar” component. We then conditionally add a new HTML list item to the list of other items if this optional property not “undefined”. The wording “not undefined” is awkward, but intentional because “undefined” is a JavaScript keyword.

Artifact 422-D above shows the difference in code that team Edross contributed to the performance developer tools source code. This is the FilterNavigatorBar component. Line 17 shows the addition of the optional property and lines 60-73 show the extra list item getting conditionally added when the new property is not undefined.

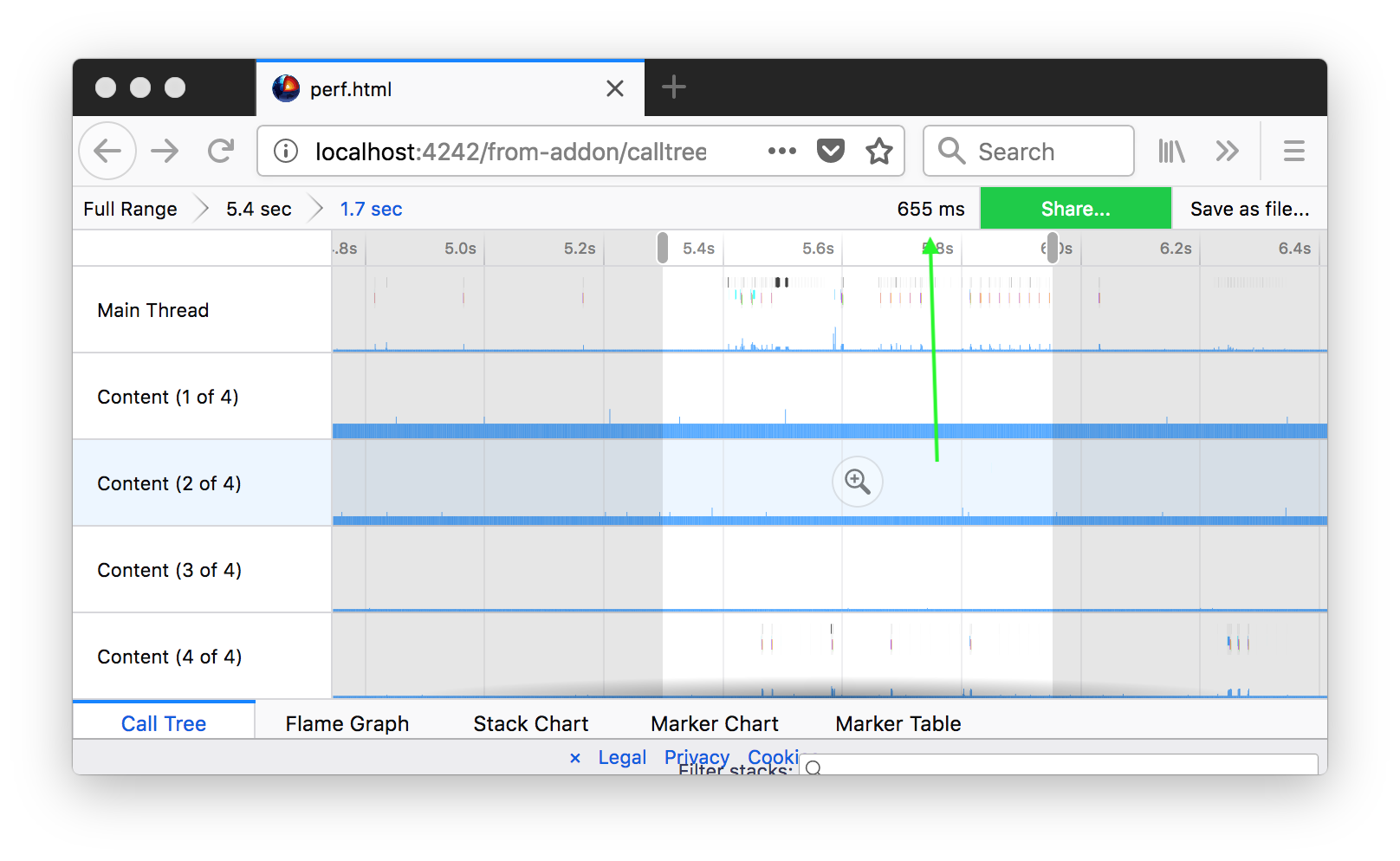
To get it to work we added logic where the FilterNavigatorBar component is used to populate the property if there is a selection or not. We modified the “mapStateToProps” function that is called by the Mozilla team’s frameworks to build the FilterNavigatorBar component to add the optional property when there was a selection. The frameworks manage the state and properties of the component and the item is added when a selection is made and removed when a selection is canceled.



Artifact 422-E above shows the difference in code that team Edross contributed to the performance developer tools source code. This is where the properties for the FilterNavigatorBar are created when state updates. Lines 26-36 show the conditional population of the optional property.

### Visual Fix

The fix that team Edross proposed would include the duration time in the top bar, but on the right side, next to the “share” button instead of being on the end of the breadcrumbs bar. This duration time would be seen at all times when there is a selection and would not be shown when there is no longer a selection. The duration time that covers the timeline when a selection is in the process of being made is still there, to be taken out at a later date when the duration time in the top bar has been finalized.



Artifact 422-F above shows the performance profiler with the addition of the duration time in the top bar. A green arrow points to the addition team Edross made to the profiler.

This change as it was wasn’t accepted. Firefox Devtools Engineer Julien Wajsberg, GitHub handle @julienw, commented that the duration time wasn’t in the breadcrumbs bar on the left as expected by Greg’s suggestion. Julien has suggested further work that we can do to get our fix ultimately solved. This includes the suggestion to create a new pull request with an additional test case covering our code changes. We will be able to use these suggestions in the future to get this pull request finalized and merged in.

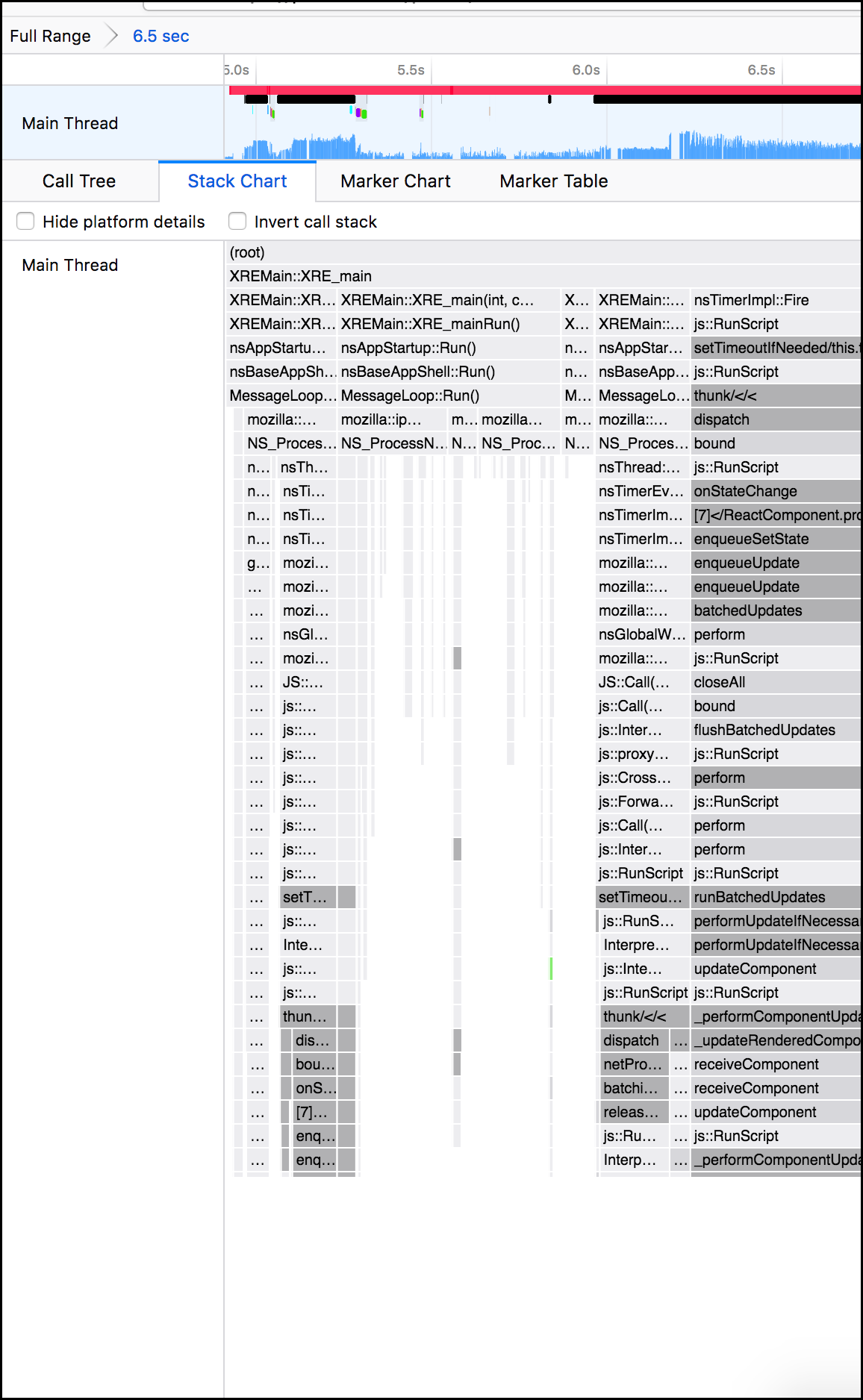
# Issue #736: Hiding Threads Does Not Resize Stack Chart Canvas

### Issue Description

In Mozilla Firefox's Performance tools for the web, Dev-tools/Perf.html, there was an issue of the canvas object not being resized when its dimensions were altered. When a user would isolate a single thread or hide one or more threads in the profile header list part of the timeline, the stack chart canvas object, with a rendering of thread information, would be increased with the vertical length of the newly freed up space left by the timeline. When this increase occurs, the canvas object doesn’t get resized. The canvas moves up the screen into the free space, but then the free space is seen at the bottom of the screen. When scrolling down in the canvas, new information comes from the bottom of the canvas. So instead of the information seemingly coming from the bottom of the browser window, the information comes from somewhere above the bottom of the browser window. This is until the canvas gets resized by many other methods, in which case, the canvas reaches the bottom of the browser window and appears how it should.

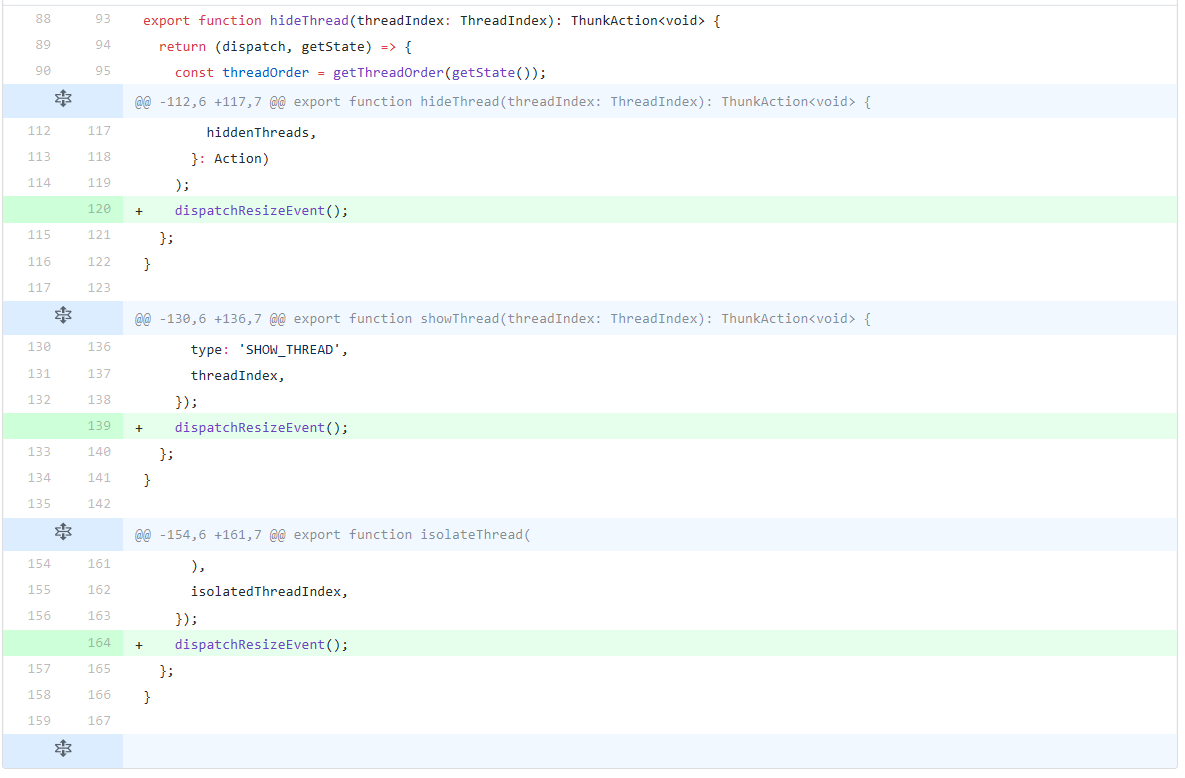
Issue report #736 was submitted that asked for the stack chart canvas to be resized when a few threads were hidden. The issue report contained a picture of interest, artifact 736-A.

Artifact 736-A as seen on the following page shows the profiler with only the main thread being shown. The picture was given a black outline of three pixels on all sides by team Edross to show how far down the canvas displays information. Note: The canvas object isn’t reaching to the bottom of the browser.



### Code Fix

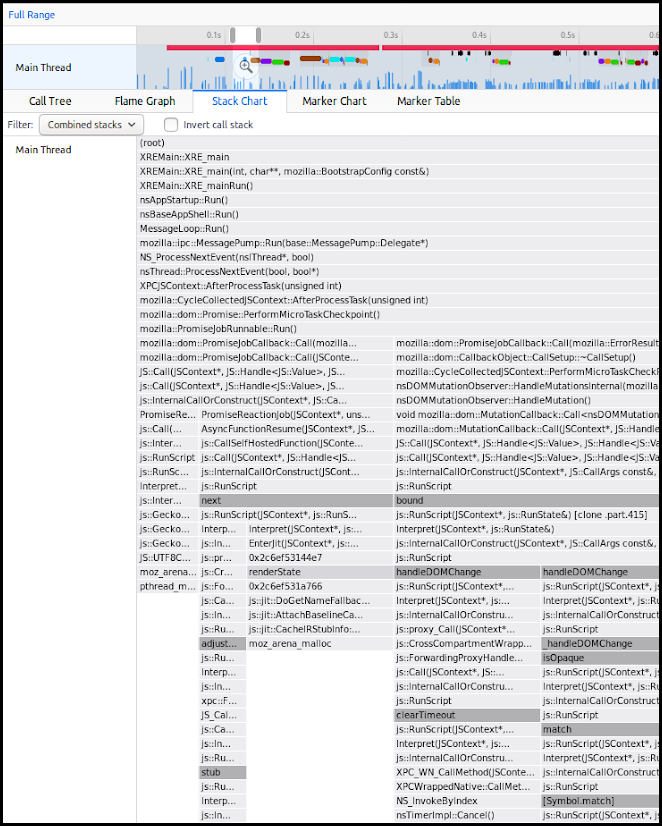
There is a method called “dispatchResizeEvent” that does the resizing of the stack chart canvas object. Team Edross inserted this method call into three places in the file perf.html/src/actions/profile-view.js. Hiding and isolating threads on the profile header list now causes the screen to be rendered which will resize the canvas.



Artifact 736-B above shows the difference in code that team Edross contributed to the performance developer tools source code. Three lines of code were added.

### Visual Fix

Hiding/isolating threads on the profile header list now causes the screen to be rendered. This resizes the canvas and gives the desired appearance asked of in the issue report.



Artifact 736-C as seen on the previous page shows the profiler with only the main thread being shown. The picture was given a black outline of three pixels on all sides by team Edross to show how far down the canvas displays information. Note: The canvas object now reaches to the bottom of the browser window.

### Update

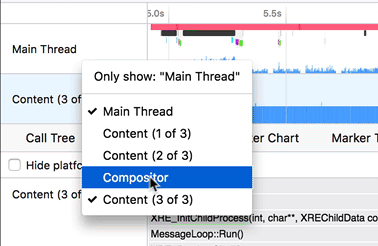
Pull request feedback contradicted the suggested feedback from the repo owner @gregatum. The feedback stated that changes were going to be made later on and our fix would not be compatible with those new fixes. The repo owner apologized for the confusion and gave us useful information on how he would like us to fix the bug. Unfortunately this feedback came after our “code freeze”, so we will not have time to submit a pull request before the submission of our product deliverable. Still all of the information to continue with development is stored in our GitHub page.

# Issue #737: `Only Show X Thread` Causes Jumping Menu Positions

### Issue Description

In Mozilla Firefox’s Performance tools for the web, Dev-tools/Perf.html, there was an issue of a menu conditionally jumping around. When users would right click a thread in the profile header list, a menu appears. If the profile a user looks at shows one thread, then this menu will only show the list of the loaded profile threads. If two or more threads are shown then this menu will show the list of the loaded profile threads along with an extra button at the top. This extra button, when clicked, will hide all threads besides the topmost of the shown threads listed in the menu.

issue report #737 was submitted that asked for the button to be there in the case where only one thread is shown. The issue report contained an animation of interest, artifact 737-A.



Artifact 737-A above shows the profile header list menu that appears when you right click one of the threads. When only one thread is shown, the button “Only show: ...” disappears. When two or more threads are loaded and shown, the button appears.

### Code Fix

Firefox Devtools Engineers Greg Tatum and Julien Wajsberg, GitHub handles @gregtatum and @julienw, commented a proposed way to implement a solution to issue report #737. Greg mentioned that dimming out the menu item would be an easy fix and Julien mentioned that there is a disabled attribute that could be applied to menu items.

# 

Artifact 737-B above shows the changes team Edross made to make the thread context menu stable by keeping the “only show:...” menu item in the menu, switching between a enabled and disabled state.

### Visual Fix

Artifact 737-C on the right shows the context menu that would appear when right clicking a thread in the profile header list and the number of visible threads is one. Note: the first menu item is disabled.

# Issue #895: Do Not Show the “Only Show” Item in the Context Menu If the Profile Only has One Thread

### Issue Description

In Mozilla Firefox’s Performance tools for the web, Dev-tools/Perf.html, there was an issue of a menu containing redundant information. When users would right click a thread in the profile header list, a menu appears. If the profile a user looks at has only loaded one thread into the profiler, the user gets to see a disabled “Only show” menu item. With only one thread loaded, the user would never get to see that menu item be enabled for the profile they are viewing.

Issue report #895 was submitted that asked for the menu item to not be there, even if its disabled, if the profile only contains one thread. After team Edross solved issue report #737, the menu item “Only show” would always appear even if there was only one thread in a loaded profile. Firefox Devtools Engineer Julien Wajsberg, GitHub handle @julienw, made the report and asked if a member of team Edross might make the fix.

### Code Fix

Team Edross agreed to fix the issue. To make the menu item not appear in the case that a loaded profile only contained one thread, team Edross wrote a condition. 

Artifact 895-A above shows the changes team Edross made to make the context menu item “Only show” not appear when the profiler has only one thread.