Persisting collapsible sidebar

WIKIMEDIA
FOUNDATION

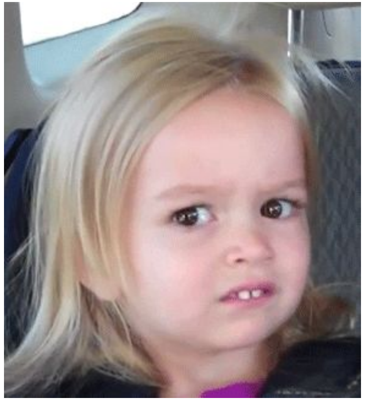
What do we want?

The sidebar to remain open or closed between page navigation!

When do we want it?

In a reasonable time-frame!

How will we build it?



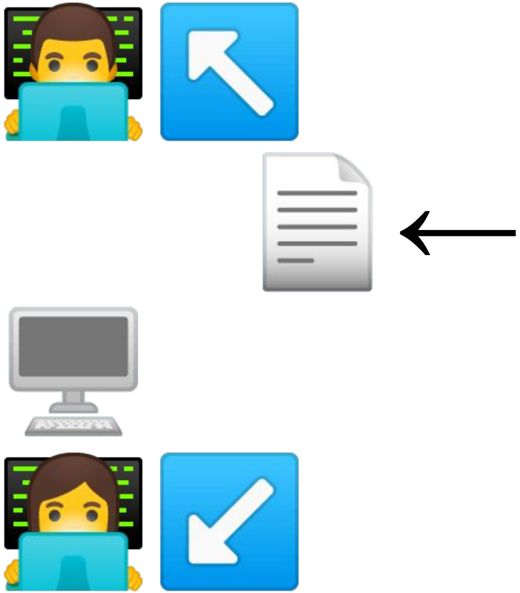
How Wikipedia works: Caching

Something to keep in mind while exploring technical options.

WIKIMEDIA
FOUNDATION

Anonymous users

- Anons **only** get a cached copy of article pages.
- Cached pages are the **same** for all anons.
- There is only **one** version of that page in the cache.

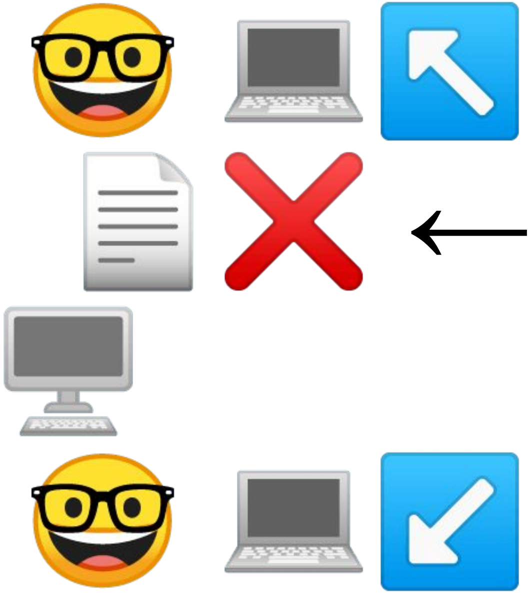


This is the only reason Wikipedia scales.

Logged-in users

- **Bypass** the cache, and get content straight from the application server.
- This content can be **customized** (with user-styles or preferences etc).
- Hitting the application servers is computationally expensive, which is why the logged-in user experience is **slower**.

This is why, if everyone had an account, our servers would melt.



The question for anons is:

If all anons get the same cached HTML, how do we modify the page *in the browser* so that the sidebar maintains its state between page views?

The question for anons is:

If all anons get the same cached HTML, how do we modify the page *in the browser* so that the sidebar maintains its state between page views?

1. **Never refresh the page** - This is how the prototype works. Desktop refresh becomes a single page application and Marvin 2.0 is born!

The question for anons is:

If all anons get the same cached HTML, how do we modify the page *in the browser* so that the sidebar maintains its state between page views?

👩‍💻
👨‍💻

1. **Never refresh the page** - This is how the prototype works. Desktop refresh becomes a single page application and Marvin 2.0 is born!

2. **Move heaven and earth to fragment the cache** - This has been an ongoing conversation since 2005.

The question for anons is:

If all anons get the same cached HTML, how do we modify the page *in the browser* so that the sidebar maintains its state between page views?

1. **Never refresh the page** - This is how the prototype works. Desktop refresh becomes a single page application and Marvin 2.0 is born!

2. **Move heaven and earth to fragment the cache** - This has been an ongoing conversation since 2005.

3. **Inline javascript** - We execute a small piece of code at the top of the page that adds a CSS class based on a value stored locally in the browser.

The pitfalls of adding an inline script:

1. **Bad for performance** because it is **render blocking** (i.e it must read from storage and add class before continuing the page-load).

2. If executed **async**, (non-render blocking) will cause a flash of unstyled content (**FOUC**) and layout reflow (this currently happens with the TOC, which *does* persist for anons).

3. The script will execute on every single page-load, for every single browser, including **grade C**.

👩‍💻
👨‍💻

Lest we forget, performance = SEO.

The pitfalls of adding an inline script:

1. **Bad for performance** because it is **render blocking** (i.e it must read from storage and add class before continuing the page-load).

2. If executed **async**, (non-render blocking) will cause a flash of unstyled content (**FOUC**) and layout reflow (this currently happens with the TOC, which *does* persist for anons).

3. The script will execute on every single page-load, for every single browser, including **grade C**.

👩‍💻 👨‍💻

For these reasons, engineering is strongly opposed to adding an inline script for anonymous users.

Lest we forget, performance = SEO.

Can we make some sort of exception to server-render this somehow?

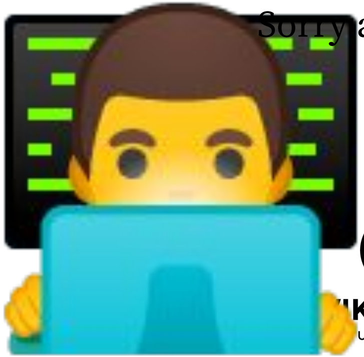
Server rendering an expanded sidebar for anons means keeping a second version of each page in the cache, i.e. **doubling the cache size.**

Even if SRE were willing to do this *for just this one feature,* it might require fundamental changes to our caching infrastructure, entirely dependant on SRE’s timeline.

Will it take years? Is it worth it?

👩‍💻
👨‍💻

In summary



Sorry anons :(

Now, logged-in users!

We make it a user preference! Done!

🎉 🥂

...right?

Almost… there are still some questions for logged-in users:

1. **Can the database handle it?** We will be updating a user preference on every hamburger click. That means a high volume/frequency of database writes to the `user_properties` table.

   Captured in T231044

2. Upon clicking the hamburger, we make an **async** save via the API. This is
   a. Prone to failure. *We don’t give any user feedback upon save completion.*
   b. We’ll have to debounce the API requests to prevent thrashing the API endpoint.
   c. JS only (not that that’s a big deal).

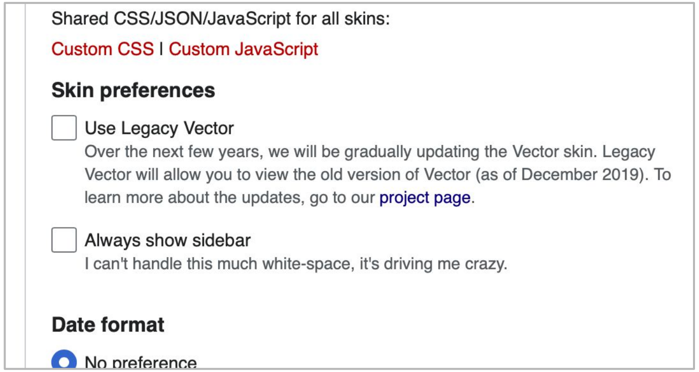
All these questions lead to even more, higher-level questions:

- **Do Users really expect sidebars to persist?**
  - ○ **Content websites:** like New York Times, Washington Post etc. **do not** persist their sidebars.
  - ○ **Applications:** like Gmail **do** persist, but there is no page-refresh involved.
- **What happens on mobile**?
  - ○ Does the sidebar persist *only beyond a certain viewport width?*
  - ○ **Should we override the user preference on mobile**? Make the sidebar overlap the content like in Minerva?
- As a user preference, it will persist **across devices**, is that desirable?
- As a user preference, is it expanded by default for logged-in users? (i.e. editors?)
  - ○ Should this be configurable on a per-wiki basis?

If this were designed by engineers:

Logged-in users only.
Make it a user-preference.
Set the user preference explicitly.

Maybe a standard user preference?



Shared CSS/JSON/JavaScript for all skins:

Custom CSS | Custom JavaScript

Skin preferences

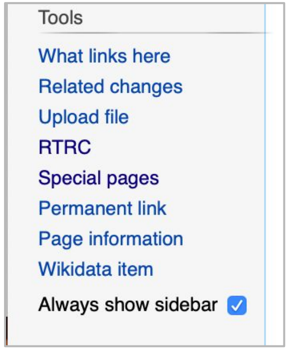
□ Use Legacy Vector
Over the next few years, we will be gradually updating the Vector skin. Legacy Vector will allow you to view the old version of Vector (as of December 2019). To learn more about the updates, go to our project page.

□ Always show sidebar
I can't handle this much white-space, it's driving me crazy.

Date format

● No preference

Maybe somewhere in the sidebar itself?



Tools

- What links here
- Related changes
- Upload file
- RTRC
- Special pages
- Permanent link
- Page information
- Wikidata item

Always show sidebar ☑

WIKIMEDIA
FOUNDATION

Thinking a bit more about user preference

Is it “disable auto-hide” or “Hide by default”.

Hide by default

- Implies that this affect just the default state.
- It will remain closed after you close it. *which will still require the “hamburger save”* (i.e save on menu icon click).
- Does the hamburger save then change the setting in user-preferences? Why even have a preference when closing the menu has the same effect?

Disable auto-hide

- Implies that the sidebar won’t retain its state.
- It will still be open after you close it. i.e. it won’t hide automatically.
- Do we even need a menu icon when this preference is enable? Wanting the menu closed only on a single page is probably rare.

THANK YOU