I want to explore bundling all of our component JavaScript together into a single file, and then caching that file client side. I imagine this will be a bit of trial and error, so feel free to adjust these requirements as needed. The goal here is to minimize the number of files being downloaded with every page load.

## Resource Building

### New Client Lib

Description: This is just an empty clientlib that does nothing but embed all the other client libs used by the FDP.

#### Properties

|  |  |  |
| --- | --- | --- |
| Name | Type | Value |
| allowProxy | Boolean | true |
| categories | String[] | hf.fdp.components |
| dependencies | String[] | May need to add things like jquery here, but not sure. You’ll need to see what works. |
| embed | String[] | thf.addtocart,  thf.advancemodal,  thf.disclosurescroll,  thf.glossary,  thf.growthof10k,  thf.morningstar,  thf.favorites,  thf.myfundwidget,  thf.myteamwidget,  thf.prereg,  thf.printview,  thf.quickfacts,  thf.responsivepublish,  apps.main.thf.richTextEditor,  thf.munchkin,  thf.sectorchart,  thf.subscription,  thf.taggedList,  thf.yui |

### Fund Detail Page Template

* Add a clientlib include for category “hf.fdp.component”
  + CSS in the Head
  + JS in the Footer

### Components (for each component mentioned in embed property)

* Comment out the individual clientlib includes in each of the individual components.

## Client Side Caching

Let’s leverage the Versioned ClientLibs library from ACS AEM Commons at the below URL. The goal here is that our single hf.fdp.component client lib file will be cached on the local users machine and never need to be downloaded again unless it changes.

<https://adobe-consulting-services.github.io/acs-aem-commons/features/versioned-clientlibs/index.html>

## Question to Answer

Please find answers inline:

1. Is a single JS file (hf.fdp.component) downloaded on the Fund Detail page?

**Wipro -** Yes, there will be a single JS file downloaded to the FDP

1. If you refresh the page, is it downloaded again or pulled from the local cache?

**Wipro -** File will be pulled from local disk cache.

1. If a small change occurs in one of the individual component JS files, does the user receive an updated copy of the hf.fdp.component JS file?

**Wipro -** Yes, user will receive updated copy with latest hash value appended to individual component js file, this is happening only after removal of dispatcher cache.

1. Is there a noticeable performance improvement to page load time?

**Wipro -** There is only minimal page load time observed, after implementing these changes.

**Observations/Results for embedding Client libraries:**

1. The embed property integrates all the CSS and JS files together.
2. All the embedded clientlibs files are concatenated and load as one file.
3. The problem arises only when the component is under SDI. For such cases, even embedded clientlibs go for a separate network request. Solution to SDI issue would be to remove individual client libs inclusions in the component, embed them in the template level, create different aggregated clientlibs for different templates, and include them in the template level.
4. Following our structure in clientlibs, the embed property must be added to both the parent clientlibs folder and the default folder under themes.
5. We have code references in components and CSS for publish client library (deprecated). Need to update these references with responsivepublish client library. This will reduce number of network calls.



1. Dialog Selection should be there for FDP components like Performance, Morning Star etc. When used on other templates other than Fund Detail template for the inclusion of component client library.
2. We can add dependency of responsivepublish clientlibs into all other template libs so the number of network calls can be limited to one.
3. Upon embedding client libs for fdp, there is a difference in the page load time along with the reduced network calls.
   1. Pre Embedding

JS Calls - 60

CSS Calls - 23

**First Load - 12.23s**

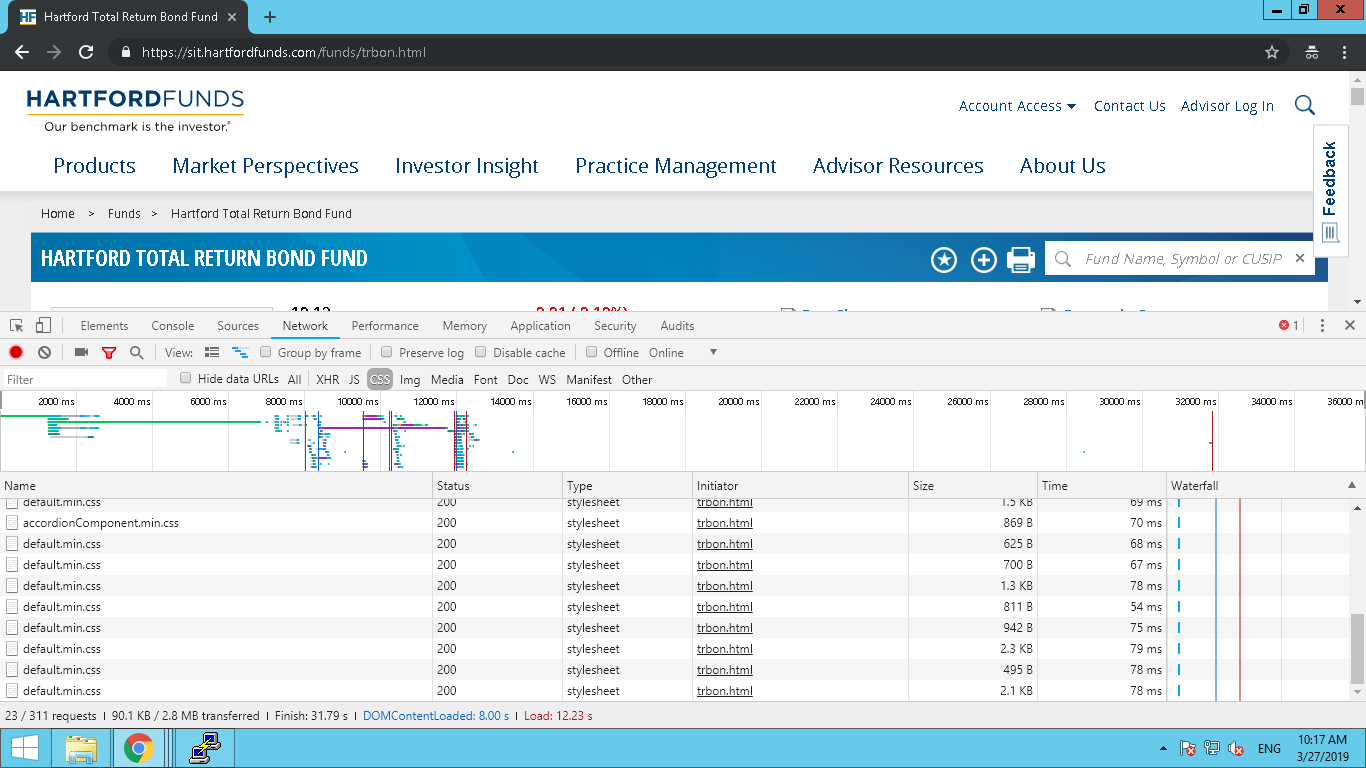


Figure 1 CSS

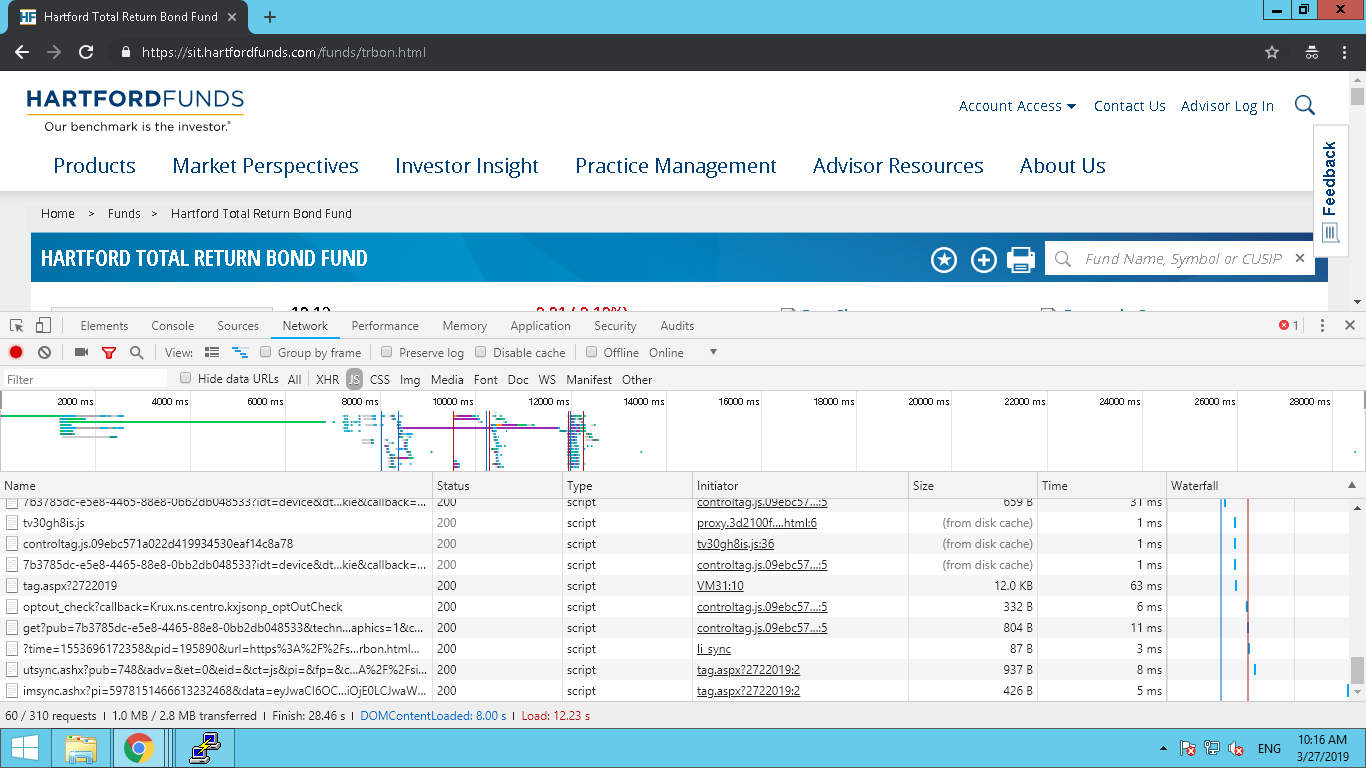


Figure 2 JS

**Second Load - 4.78**

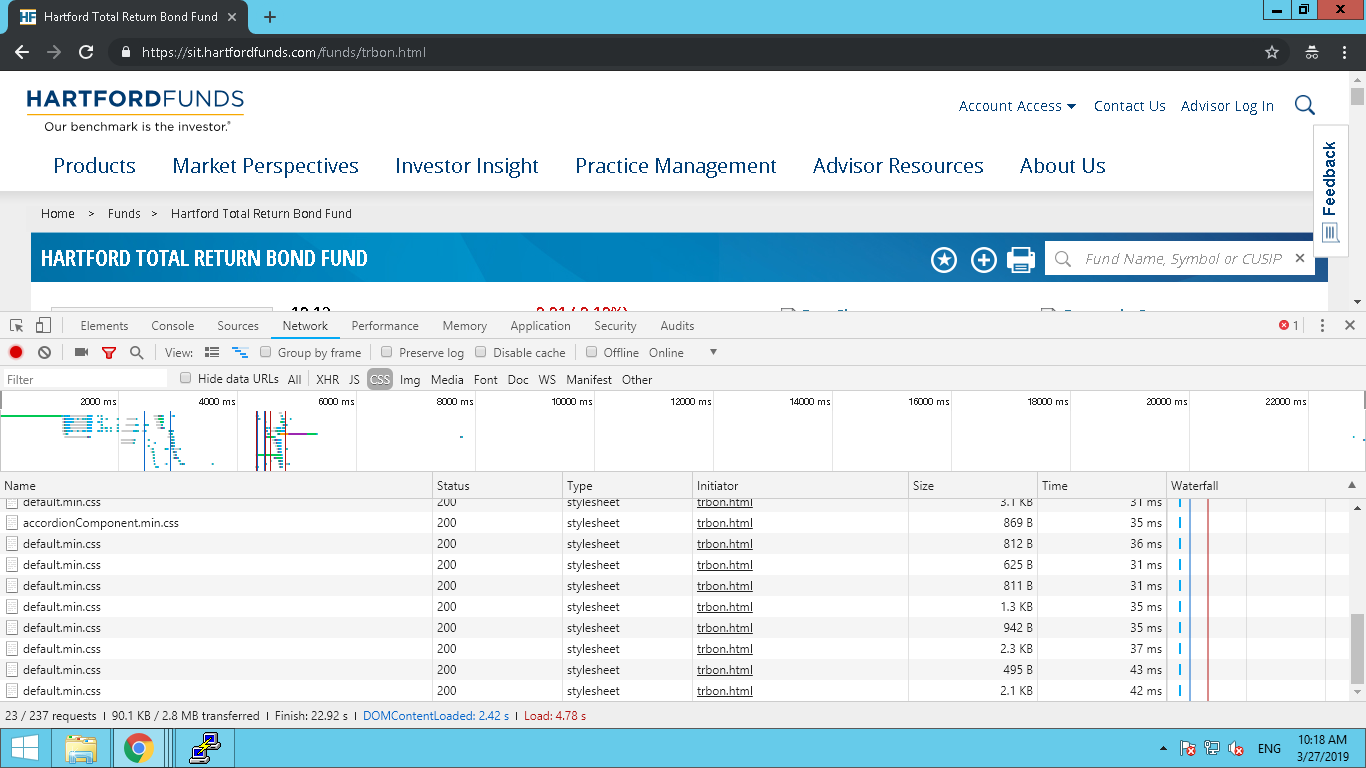


Figure 3 CSS

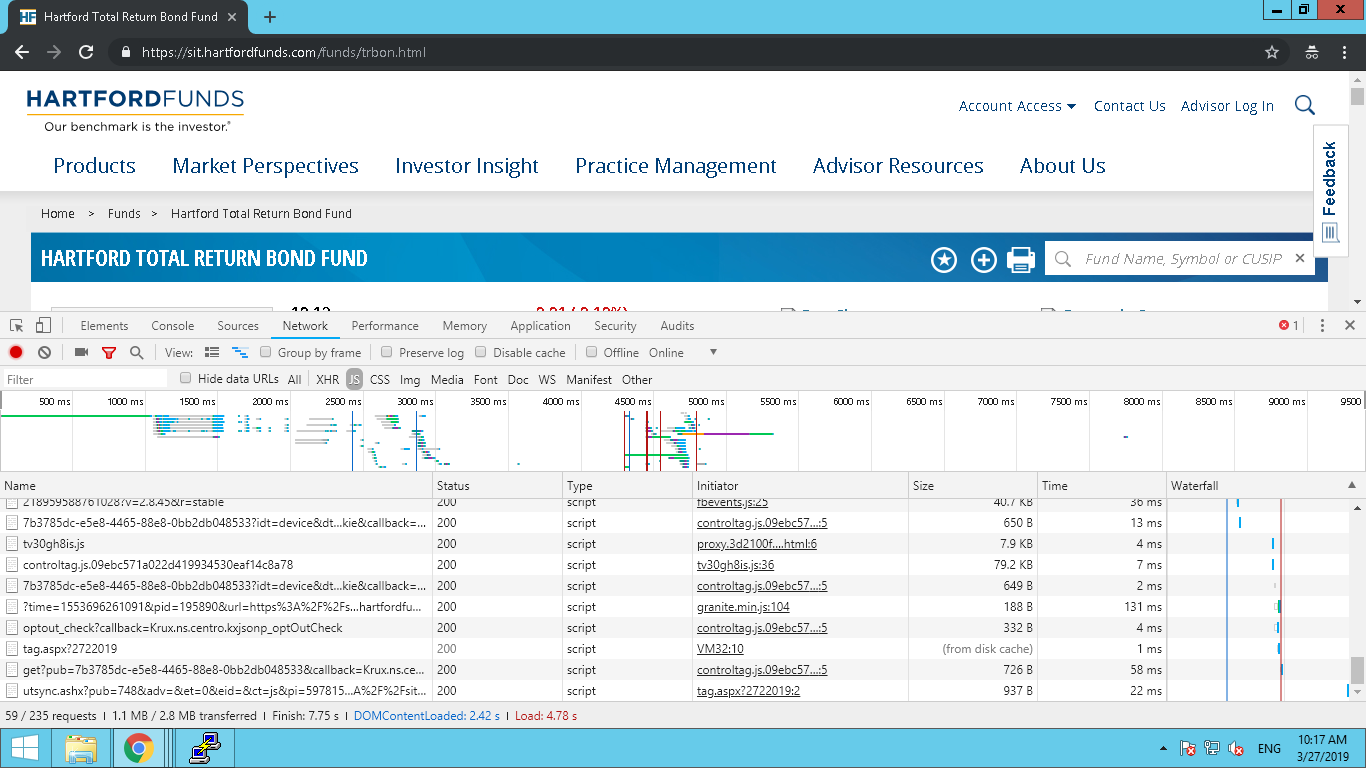


Figure 4 JS

* 1. Post Embedding

JS Calls - 46

CSS Calls - 12

**First Load - 4.16s**

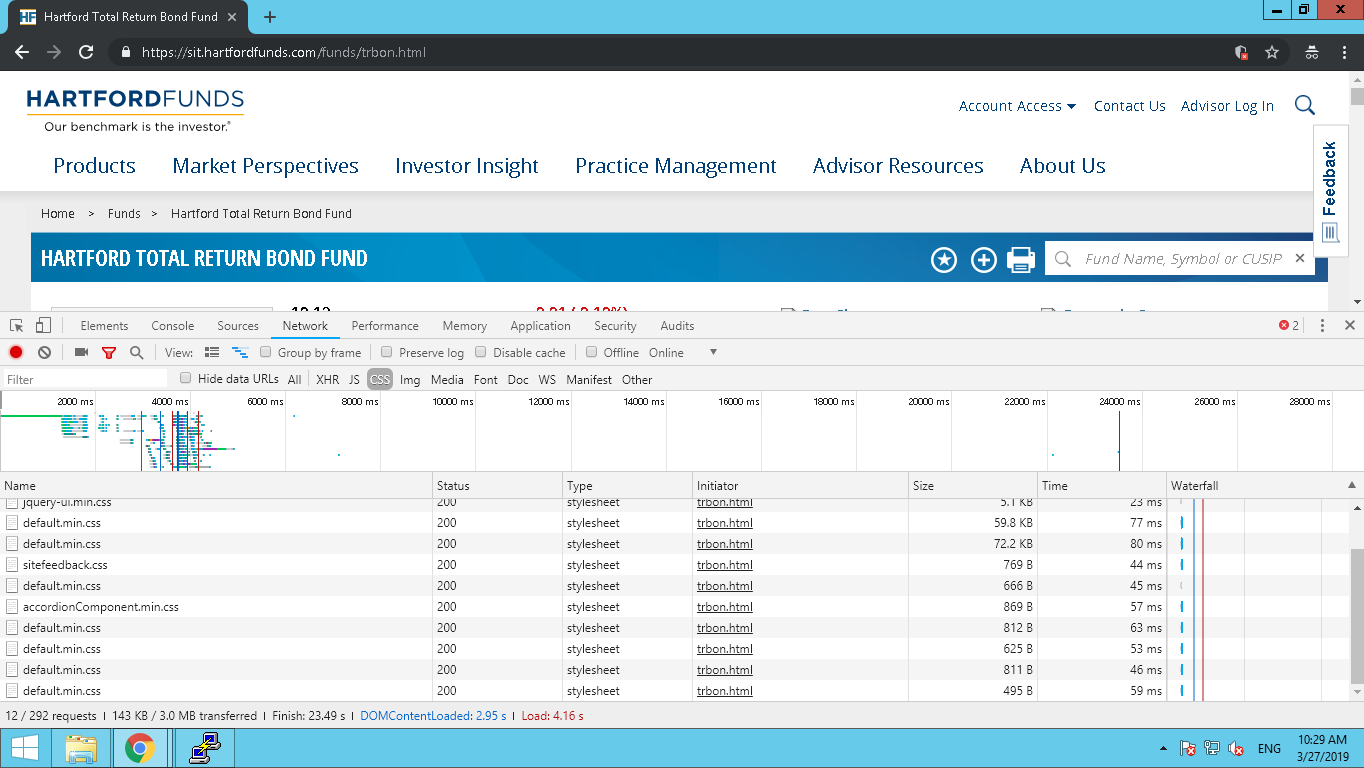


Figure 5 CSS

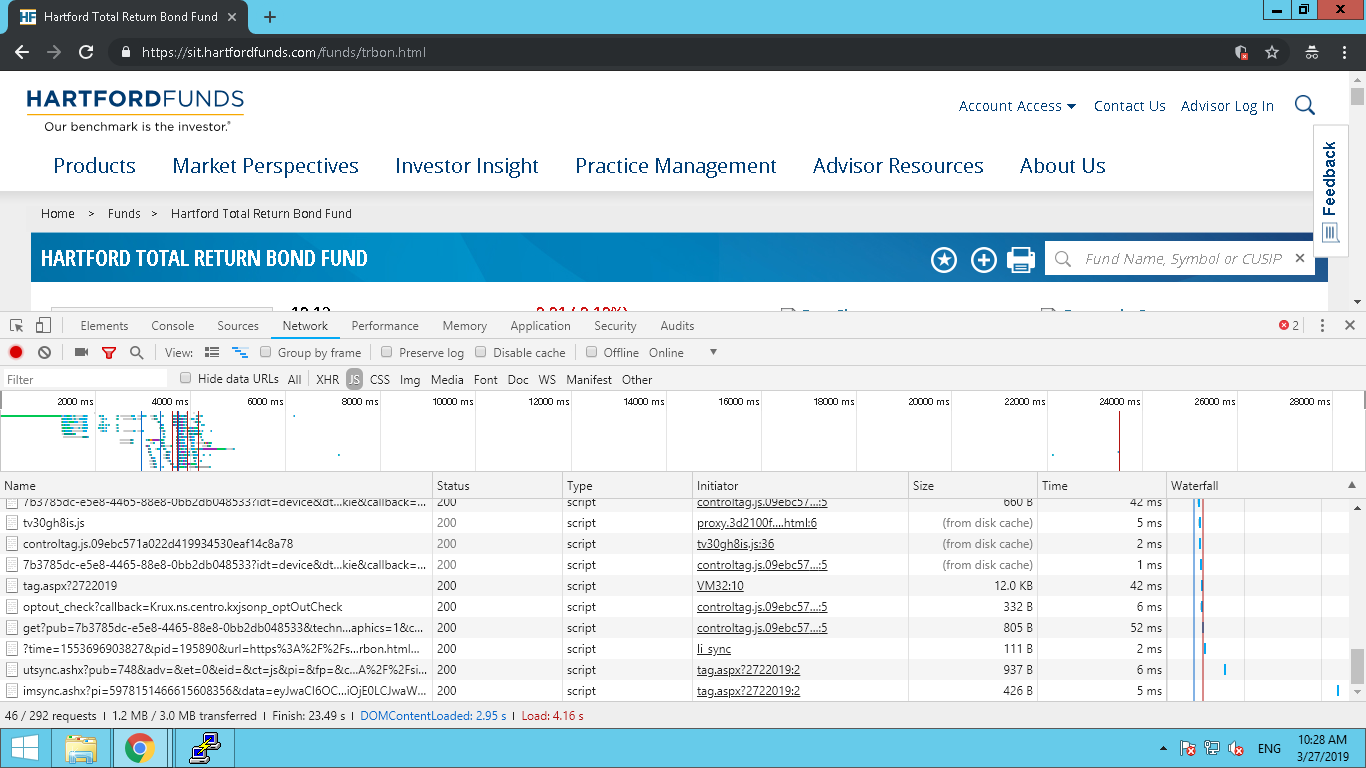


Figure 6 JS

**Second Load – 3.85s**

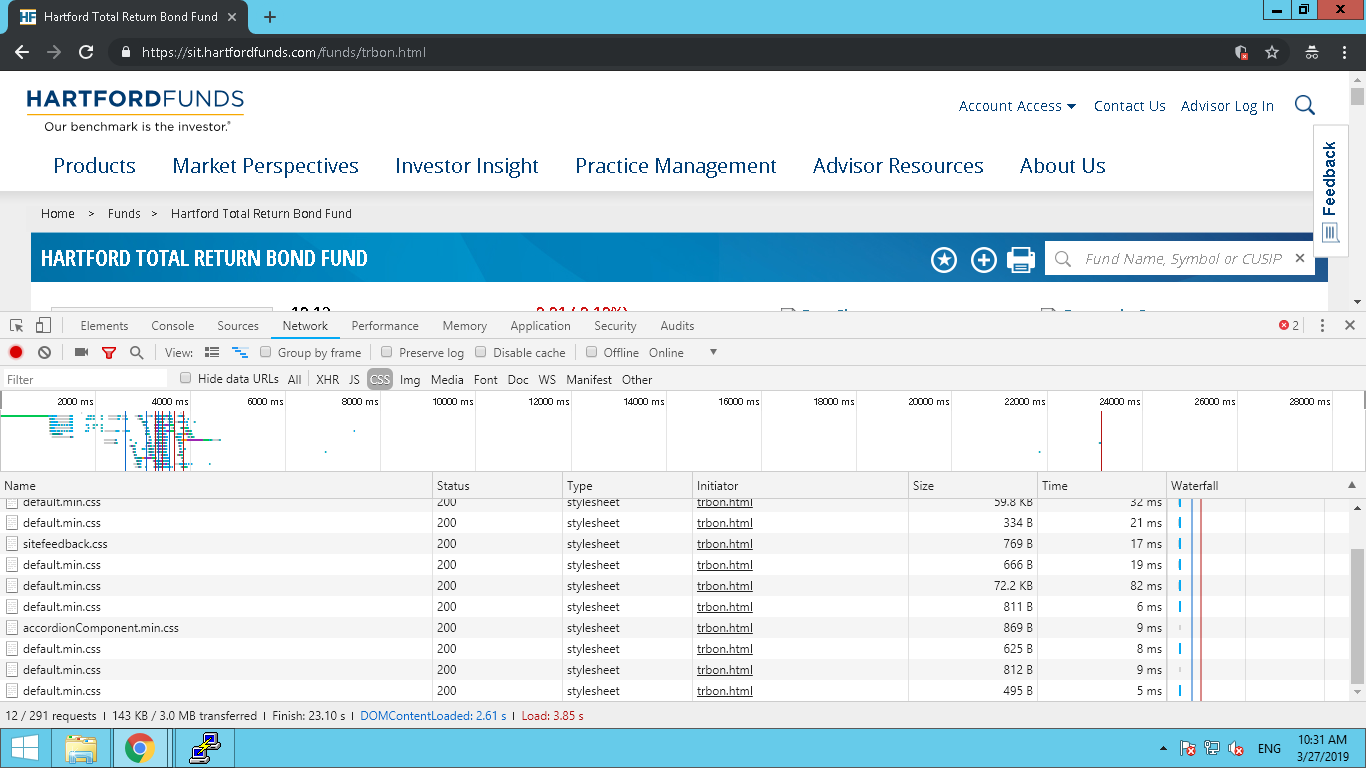


Figure 7 CSS

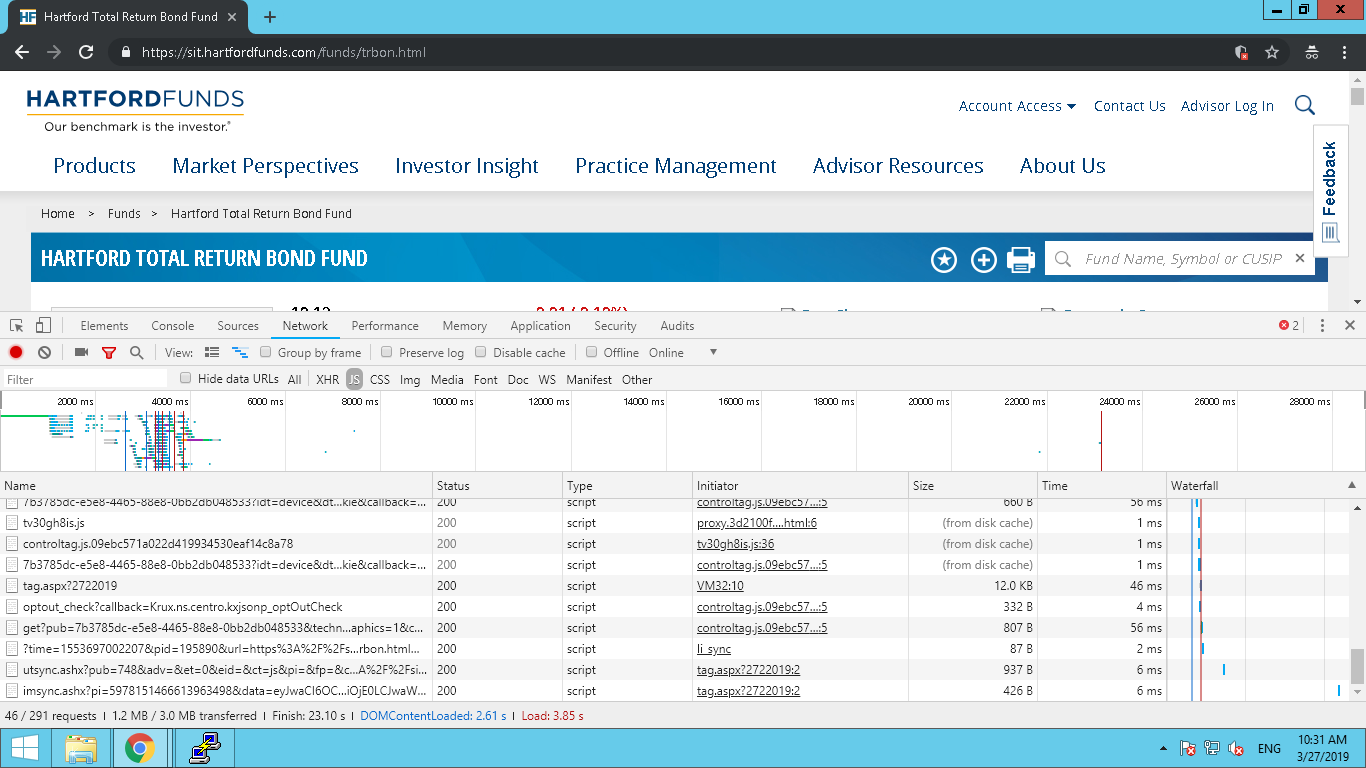


Figure 8 JS

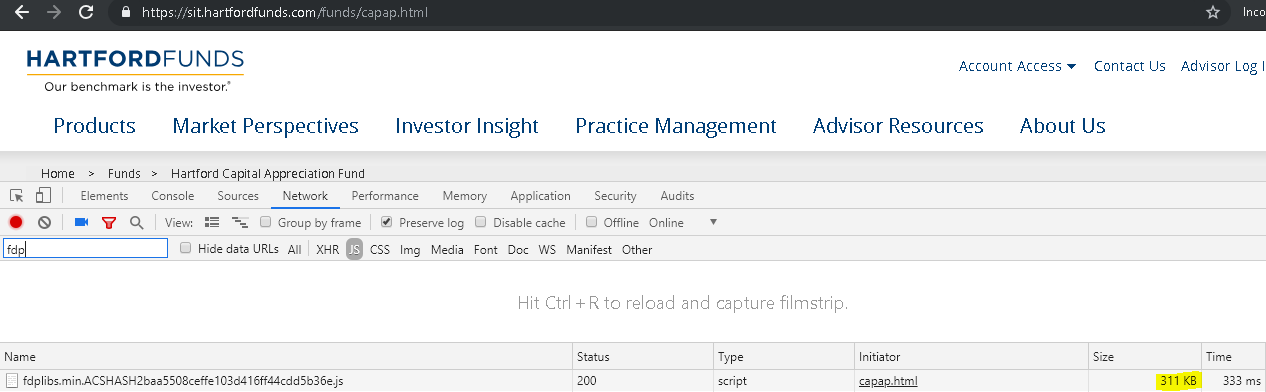
**AEM versioned Client Libraries:**

We have made necessary changes and implemented the versioned client libs on SIT environment

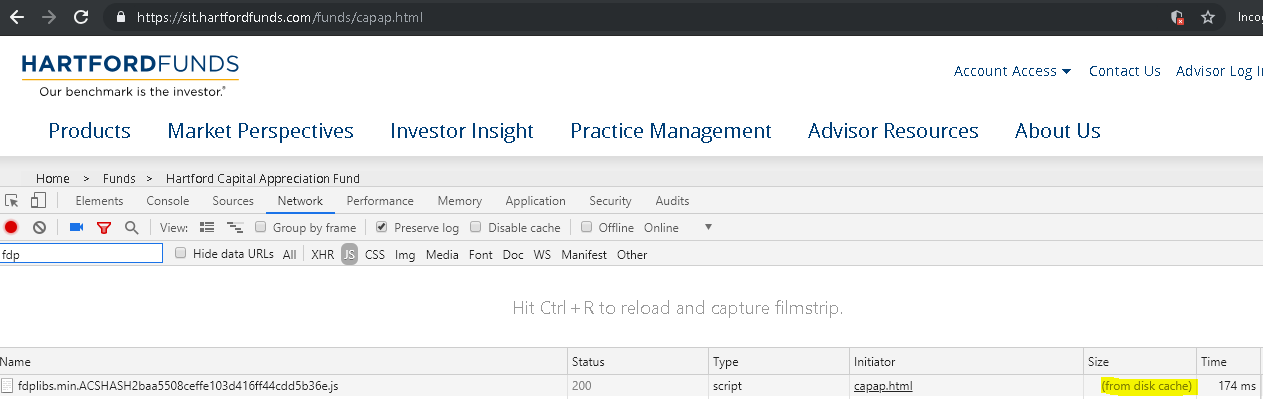
(Implemented changes as per <https://adobe-consulting-services.github.io/acs-aem-commons/features/versioned-clientlibs/index.html>)

Below are the observations after the implementation:

1. When we load the fdp page for first time, it will load from either dispatcher cache/publisher.

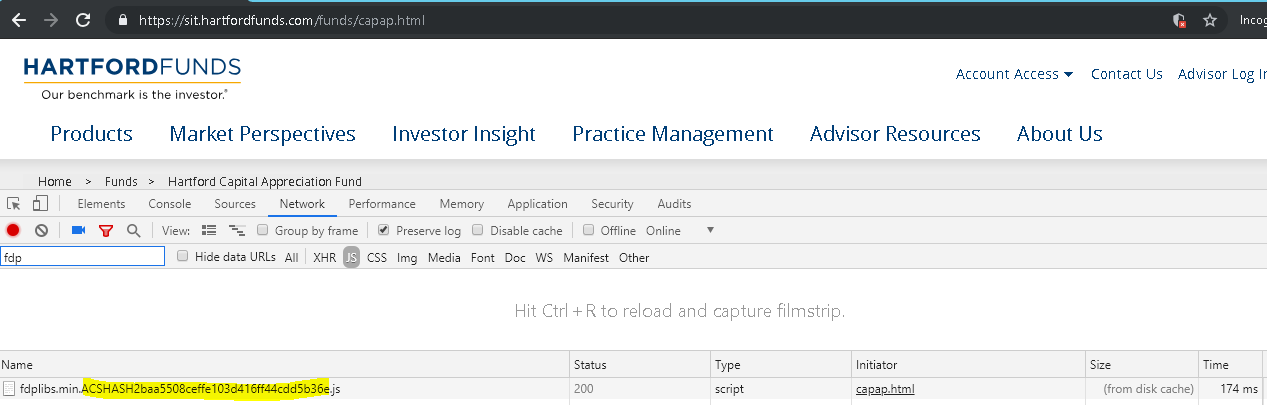


2. From next time, same fdp page will loaded from browser cache

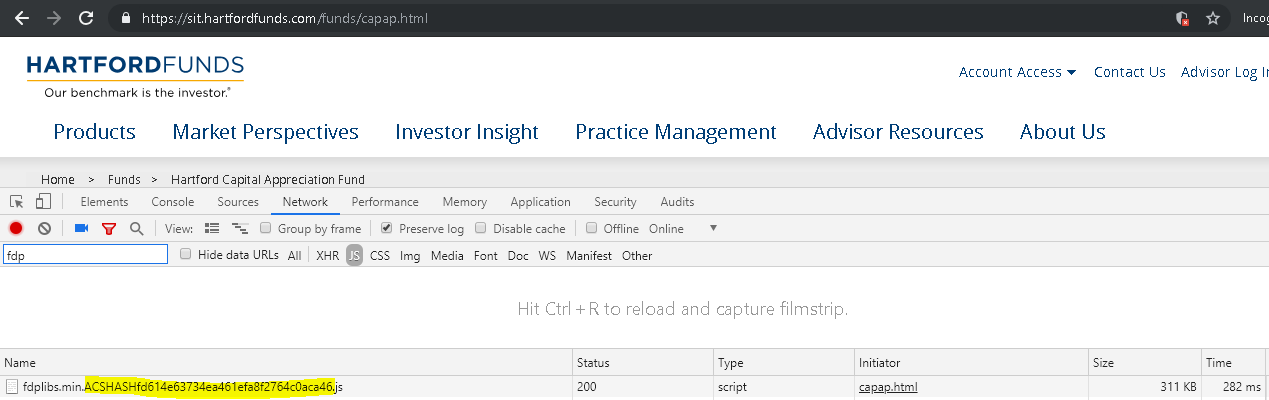


3. Clientlibs are getting versioned based on the file modifications.

4. Before modification of js/css file, the ACS hash will be as shown below for fdp page



5. After modification of js/css file, the ACS hash will be updated and the hash value will be as shown below



6. When an individual component js file is modified and to reflect updated changes we have to remove dispatcher cache manually.