

Here is the discussion forum for this part of the course. Please either post your comments/observations/questions or share your creations.

Let us suggest some topics of discussion and optional projects:

## SUGGESTED TOPICS

- Did you hear about [the Fetch API](#) and the [Service Workers API](#)? These emergent APIs are not mature yet and not ready for mainstream use, but they are promising... We will certainly cover them in a next version of this course. However, have a look at them! The Fetch API is a candidate successor for XHR and XHR2. Compare [this example from the Week 1 course](#) and [this same example that uses Fetch](#) (and two polyfills to make it work in your browser!!!).
- Did you note that using XHR2 for monitoring progress is really simple and efficient? I used to rely on jQuery for Ajax request but a few years ago I switched for native XHR2. What is your experience? Please share ;)
- How can we monitor the speed of an upload/download in bytes per second? What would you propose? Did you find some interesting resources on the Web that explain that?

## OPTIONAL PROJECTS

- If you know how to program server-side code, please make a small app that will upload files, monitor the progress of the upload, save the files server-side, and send back a message containing the URLs of the files. Better: create a Web page that displays links to the uploaded files.
- Try to write an `assetLoader` function that will download a set of images and sound (maybe using the `BufferUtility` seen during Week 1), but this time with a progress bar. This could be useful for a game, or for a Web app that needs to load resources before starting.