Prototyping Cooperative Scheduling

haraken@chromium.org

2017 Jun 5

Status: DRAFT, PUBLIC

The cooperative scheduling in Blink is going to be a big project. However, I noticed that it wouldn't be that hard to prototype the cooperative scheduling with the current Blink. This document describes how to implement the prototype.

Step 1: Introduce safe points

A safe point is a place where Blink may run a cross-origin task. This condition should already be guaranteed when V8 calls back Blink (because V8 should be implemented with an assumption that Blink may run a nested message loop).

When we run V8 in third-party iframes, we register a callback function to V8 so that V8 calls back the function periodically (e.g., every 16 ms). The function works as the safe point.

However, jochen@ pointed out that it would not be trivial to implement the safe point in V8. In short term we can probably insert the safe point to normal DOM attribute getters / setters / operations (hoping that they will be called back often).

Step 2: Introduce ScheduleMainFrameTasks()

We introduce BlinkScheduler::ScheduleMainFrameTasks(), which runs one or more tasks of the main frame for a certain amount of period. Then we call BlinkScheduler::ScheduleMainFrameTasks() in the safe point.

That's it! These two steps will enable Blink to yield a long-running task of a third-party iframe and cooperatively schedule tasks of the main frame. In theory it will work :) We can prototype it and see how much it improves the responsiveness of the main frame on real-world websites.