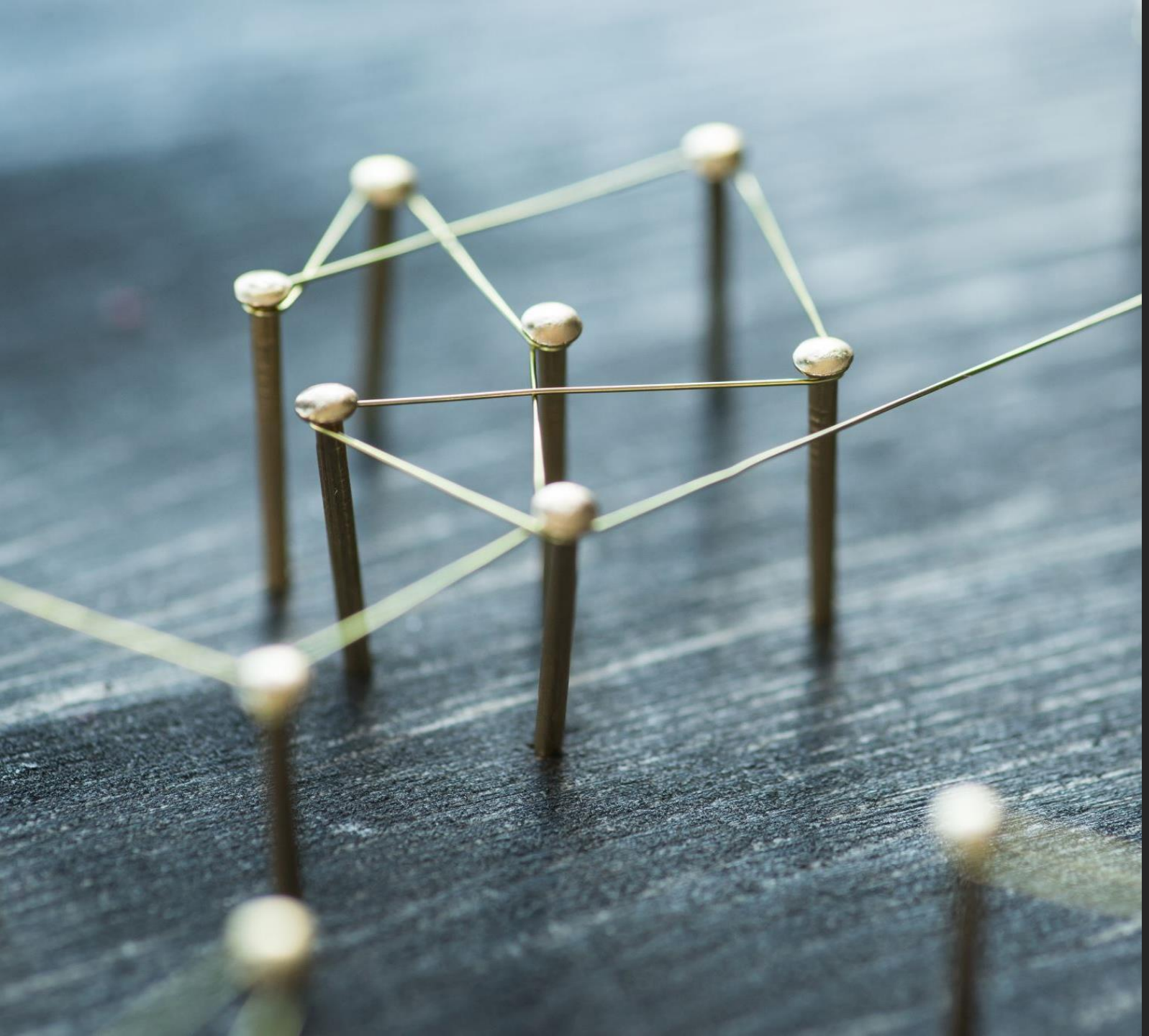
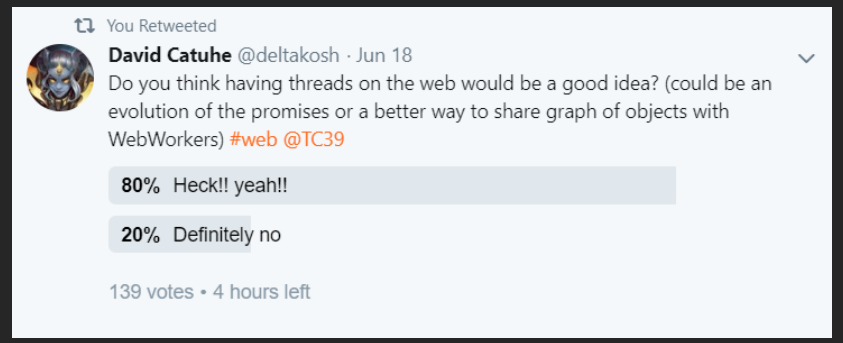


Web Threading

DAVID CATUHE - @DELTAKOSH

BABYLON.JS / MICROSOFT

What is the problem?



Today “multi-threading” is more a “multi-process” approach

We MUST be able to better leverage multi core CPUs (even on mobile) to improve web experiences and be on par with native applications

Examples for Babylon.js:

- Frustum culling
- Animation
- Physics / collisions
- Particles
- IA

Would be beneficial for the entire web ecosystem

Why not Web Workers?

1. They cannot run a specific function from your current context
2. They cannot share objects (only ArrayBuffers).
3. Transferable objects do not help either as we could have thousands of objects to pass back and forth per frame and only using ArrayBuffer is not enough
4. They require a separated js file to run
   - Impossible to run a specific function without dealing with a lot of plumbing
   - No context capture
5. They are a bit like Processes where we need Threads

How ?

Leverage promises

- Create PromiseTask that could be handled by a scheduler and ran on a different native thread (User will not have control on thread count)
- Code change is minimal for developers
- Can capture context directly (no need for transferable objects)

```
var particleSystem = new BABYLON.ParticleSystem("particles", 200, scene);

// Task can capture particle system from the context
var task = new PromiseTask((resolve, reject) => {
    // This will run on a different thread
    particleSystem.animate();
    resolve()
}).then(() => {
    // Continuation here
});
```

But…

- A lot of friction from TC39 influencers
- JavaScript enginesV8 is architected under the assumption one thread is in an isolate at one time
- Would require huge development effort from browser vendors

Some experiments:

https://webkit.org/blog/7846/concurrent-javascript-it-can-work/

How ?

Improve web workers

- Extend transferables to user object graph
  - Maybe restrain that feature to workers only
  - Allow us to efficiently work with OffscreenCanvas
  - Could be limited to POCO objects?
- Ignore DOM/WebGL to make it simple
- Allow worker creation from a function (not only from a script file)
- **Good intermediate solution**

How?

ANY OTHER IDEAS?