

Hi, this is my first post here. I was thinking about a concept of a “node” in general. I’m trying to write some simple code for showing any “program state” on “any screen” (= device, program etc.) on the “local network” (ethernet).

I have 42 lines of code for a welcome review (comments). And ~~will share soon~~. This is probably not directly or obviously related to ongoing other topics in this community forum (probably) but still I hope some parallel directions of thinking can lead to overall progress of a grander /decentrali`zed/ or whatever-it-should-be-called future vision.

```
const EventEmitter = require('events');

class ConnectedStore extends EventEmitter { constructor(data = {}) { super(); this.data = data; }

set(data) { this.data = data; this.emit('state_changed', { oh: 'yes', store: 'changed', state: this.data }); } }

class MultiConnectedStore extends EventEmitter { constructor(data = {}) { super(); this.store = new ConnectedStore(data);

// events recaster
this.store.on('state', obj => {
  this.emit('state', obj);
});

}

on(...args) { this.store.on(...args); }

set(...args) { this.store.set(...args); } }

const store = new MultiConnectedStore();

store.on('state_changed', obj => { console.log(✓ RECEIVED WITH ${JSON.stringify(obj)}); });

store.set({ god: 'bless', the: 'information' });
```

Main idea is to abstract away a connection between UI and a “local node” (process on the same device) OR the same UI and “network node (LAN for now because of small latency)”.

What is your opinion on various aspects of security here?

PS: the code shared here has two bigger concepts missing for better understanding but still this is the clearest possible example to sense the direction of thinking for this version of DAPPS UI LATENCY PROBLEM resolution.