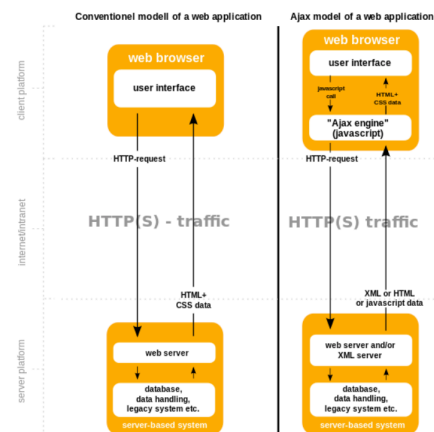


WEB3

_Intro:

- Ajax (Asynchronous JavaScript and XML)
- Focus on simple app creation -> RubyonRails / Node
- Static Pages / MVC / Applets / PHP



_Node.JS:

- Increasingly asynchronous _ JS standard VM.
- Node solves I/O bound programs problems.

EventLoop:

- Operations dispatched when event occurs
- Low-level access _ Runs parallel except your code
- EventLoop spins > event > callback listening for it
- Waiting for (Data/HTTP request/Database) runs parallel
- single threaded / IO task non-blocking

Inheritance:

- Javascript prototype-based ES6(ECMA) allows for new code.
 - Node built around idiomatic asynchronous event driven architecture. Emitter emit named events > call listeners.
- All objects emit events instance of EventEmitter with `eventemitter.on()` `eventemitter.emit()`.

Modules:

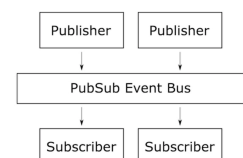
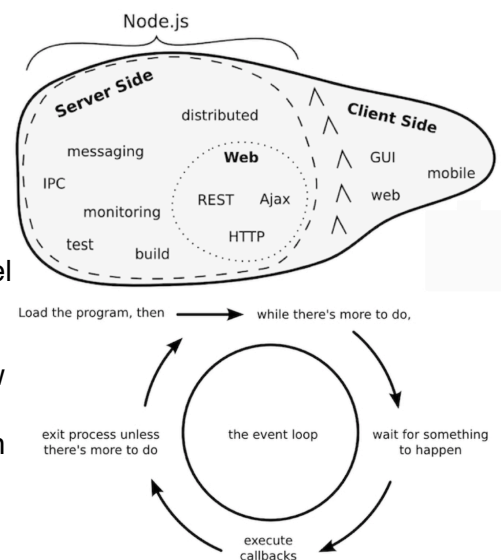
- Problem JS > naming collision 2 calendar both have `drawCalendar()`
 - Node uses Modules, when loaded new scope created so no collision.
- from inside module outside is invisible except `require` other modules > no collision possible.
- `module.exports` can be used to make module methods public.

Faye:

- publish-subscribe messaging system based on Bayeux protocol

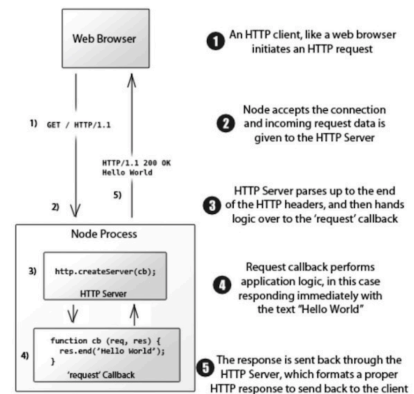
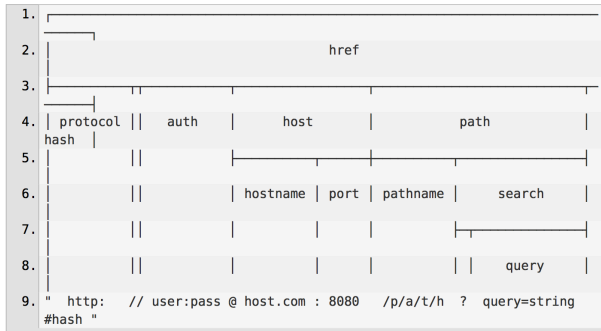
PubSubPattern: senders = publishers / receivers=subscribers

- messages not send directly but into classes subscribers express interest to get it.
 - allows to break down the app into smaller loosely coupled modules
- Faye provides backend and fronted clients / uses best mechanism for browser (Websockets > Long-Polling HTTP > CORS > JSONP)



StaticWebServer:

- Lookup the IP address some.server.com connect to PORT and ask for filename.html
- In node `http = require('http')` to create web server `createServer()` `server.listen(PORT)`
- Files have to be read > send with pathname we know which



- with url module we can extract them `url.parse(request.url).pathname`
- Problems: crashes a lot (dont ask for specific file, file does not exist)

NPM - Node Package Manager:

- we can install libraries, packages with npm manager console(`npm install package_name`) and `var http = require('http')` in node.
- package.json describes all dependencies — save saves the dependencies in the json file

Express Module:

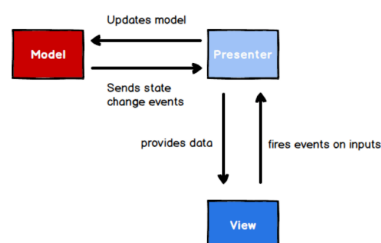
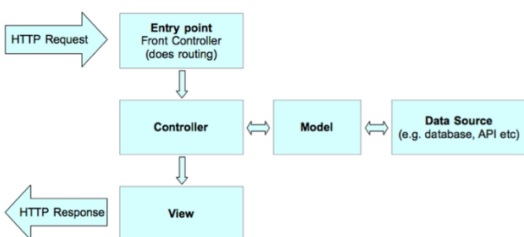
- (`npm install express —save`) makes serving files etc a lot easier.
- `app.use()` to use stuff from modules / `get/put/post/del` to register handlers for HTTP
- Middleware examples (Authentication, Logging, Zipping etc)
- (`npm install nodemon —save`) nodemon will restart node server whenever there is a change

SPA Intro: (SinglePageApplication)

- Data-Rich web applications were a problem in the past (dropbox, google drive...)
- jQuery was good in the past to make Ajax request and manipulate the DOM
- Latency is a problem with jQuery there fore SPA(1.nav to new view 2.content requested using XHR (REST) 3. Ajax makes comm. asynchronous 4. no interruption with loading)

MVC and MVP:

- Using GUI Pattern for SPA like MVC(model view controller)
- >Models: domain specific knowledge (User, Photo, Todo) can notify observers on state change
- >Views: user interface (markup) observers modules but no communication
- >Controller: handles input(clicks etc) and updates models



- **MVP** Model View Presenter: most used for building user interfaces
- >Model: interface defining data to be displayed
- >Presenter: acts upon the model and view (retrieves data from model and formats for view).
- >View: displays the data (model) and routes events to presenter to act upon.
- Frameworks(Backbone,React,Angular,Riot....) all use Components.

Components

software package for web service, resource, module that encapsulates a set of functions(data)

WEB Components: consist of (1.Custom Elements 2.Templates 3.HTML imports 4.ShadowDOM).

>CustomElements: create own fully-featured DOM elements with reacts to changes

>Templates: document fragments with are unused when loaded but maybe used at runtime

>Imports: import another HTML document into a HTML document (webception ;-)

>ShadowDOM: make code co-exist(facebock,youtube) encapsulation (youtube in another page)