

Table of Contents

- 1. [Introduction](#)

NodeJS Info

Wat?

Node.js is een platform dat gebaseerd is op de V8 JavaScript engine van Google. Node.js is eigenlijk een back-end variant van de JavaScript die wij kennen (voor front-end development van websites). Dit platform zorgt voor snelle I/O events zoals web calls, netwerk communicatie, files, .

Evented architecture

Idee van concurrency = laat verschillende zaken op hetzelfde tijdstip gebeuren. Java zal een nieuwe thread starten voor elke connectie. Nadeel multi-threading: 4MB memory voor elke connectie. Gelimiteerd aantal thread. Op 4GB server betekent dit ongeveer 4000 connecties.

JavaScript = single-threaded!: Single-thread maar de mogelijkheid om verschillende code paths in parallel te draaien door EVENTS. Single thread betekent dat een synchrone blok code de web page zal locken. Daarom ASYNCHRON CODE.

Hoe werken callbacks

```
user = User.findById(1000)
```

Deze code queries de database voor id=1000. Het pauzeert de thread/process tot de database terug retourneert. In javascript gaan we een callback meegeven die opgestart wordt wanneer de taak afgelopen is:

```
User.findById(1000, function(user){  
  //do something with user  
});
```

Wanneer de database het resultaat aan het zoeken is, is de thread vrij om iets anders te doen.

Node.js prompt

```

C:\WINDOWS\system32\cmd.exe - node
Microsoft Windows [Version 10.0.10240]
(c) 2015 Microsoft Corporation. Alle rechten voorbehouden.

C:\users\tom_ptr\documents\visual studio 2015\Projects\Node_1\Node_1>node
> var foo = "bar"
undefined
> var foo = "bar";
undefined
> foo
'bar'
> var temp = 1;
undefined
> temp
temp
> temp + 23
24
>
^C again to quit
-

```

Standard libraries, vb. http

async by default: (=callback function)

```

var http = require("http");
http.createServer(function(req,res){

    res.writeHead({
        "Content-Type":"text/plain"
    });
    res.end("Hallo Tom");
}).listen(3000);

```

Opstarten node server

```
node httpLib.js
```

Developing & Debugging

Wil je graag dat je server automatisch opstart nadat je wijzigingen hebt gemaakt? Installeer nodemon:

```
npm install nodemon --global nodemon server.js
```

```

C:\Users\tom_ptr\Documents\Aptana Studio 3 Workspace\Node1\Node-PPT>nodemon httpLib.js
25 Oct 08:15:48 - [nodemon] v1.5.1
25 Oct 08:15:48 - [nodemon] to restart at any time, enter `rs`
25 Oct 08:15:48 - [nodemon] watching: *.*
25 Oct 08:15:48 - [nodemon] starting `node httpLib.js`
25 Oct 08:16:10 - [nodemon] restarting due to changes...
25 Oct 08:16:11 - [nodemon] starting `node httpLib.js`
25 Oct 08:16:16 - [nodemon] restarting due to changes...
25 Oct 08:16:16 - [nodemon] starting `node httpLib.js`

```

Debugging

NodeJS-inspector: <https://github.com/node-inspector/node-inspector>

```
c:\users\tom_ptr\documents\visual studio 2015\Projects\Node_1\Node_1>npm install
1 -g node-inspector

> v8-profiler@5.2.9 preinstall C:\Users\tom_ptr\AppData\Roaming\npm\node_modules\node-inspector\node_modules\v8-profiler
>
> v8-debug@0.5.2 preinstall C:\Users\tom_ptr\AppData\Roaming\npm\node_modules\node-inspector\node_modules\v8-debug
>

npm WARN optional dep failed, continuing default-browser-id@1.0.2

> utf-8-validate@1.1.0 install C:\Users\tom_ptr\AppData\Roaming\npm\node_modules\node-inspector\node_modules\ws\node_modules\utf-8-validate
> node-gyp rebuild
```

```
C:\ node-debug start.js

end@0.13.0, accepts@1.2.12, type-is@1.6.7)
| ws@0.7.2 (options@0.0.6, ultron@1.0.2)
| yargs@3.21.0 (decamelize@1.0.0, camelcase@1.2.1, window-size@0.1.2, y18n@3.1
| 0, cliui@2.1.0, os-locale@1.2.0)
| biased-opener@0.2.6 (minimist@1.1.0, x-default-browser@0.3.0, browser-launch
er@2@0.4.5)
| v8-debug@0.5.2 (nan@2.0.5, node-pre-gyp@0.6.9)
| v8-profiler@5.2.9 (nan@1.8.4, node-pre-gyp@0.6.9)

c:\users\tom_ptr\documents\visual studio 2015\Projects\Node_1\Node_1>node-inspe
ctor
Node Inspector v0.12.2
Visit http://127.0.0.1:8080/?ws=127.0.0.1:8080&port=5858 to start debugging.
Received request for a method not implemented: DOM.getDocument
Received request for a method not implemented: DOM.setInspectModeEnabled
^CTerminate batch job (Y/N)? y

c:\users\tom_ptr\documents\visual studio 2015\Projects\Node_1\Node_1>node-debug
start.js
Node Inspector is now available from http://127.0.0.1:8080/?ws=127.0.0.1:8080&po
rt=5858
Debugging 'start.js'
```



CommonJS Modules

CommonJS modules heeft require en exports functionaliteit Geeft de mogelijkheid om andere modules in te laden (require) of variabele naar andere modules te exposen.

Properties worden ge-exposed door een setter te gebruiken op het exports object, of door gebruik te maken van de module.exports variabele.

```
exports.hello = "Hallo";
```

Externe modules kunnen geïmporteerd worden door de require() functie:

```
example = require("./exportVB1");  
  
console.log(example.hello);
```

Voorbeeld 2:

```
exports.info = function(msg){  
    console.log(msg);  
};  
  
exports.error = function(msg){  
    console.error(msg);  
};
```

Hoe te gebruiken:

```
var log = require("./logger");  
  
log.info("info bericht");  
  
log.error("error bericht");
```

Voor het exposen van objecten moet je gebruik maken van de module.exports variabele:

```
function Point(x,y){  
  
    this.x = x;  
    this.y = y;  
  
    this.print = function(){  
        console.log(this.x + "," + this.y);  
    };  
}  
  
module.exports = Point;
```

Klasse Point gebruiken:

```
var point = require("./Point");

var pt = new point(19,22);
pt.print();
```

NPM : Node's Package Manager

NPM wordt automatisch geïnstalleerd met de Node omgeving, en is te gebruiken via de command line. Het laat gebruikers toe om Node.js applicaties (en dependencies) te installeren (die in de npm registry beschikbaar zijn).

- <https://www.npmjs.com/>
- <https://docs.npmjs.com/getting-started/>

vb. npm install twilio

Npm installeert de dependencies automatisch. Je kan in node_modules kijken welke modules je hebt geïnstalleerd. Package.json: info over de package (Bij je eigen node app heb je ook een package.json file). Index.js : entry point voor de module

Oefening

Maak een standard http server. Bij ontvangen van een request lees je een csv file in (key values pairs) en retourneer dit in JSON formaat.

csv file: Key1,Value1 Key2,Value2 Key3,Value3

Hoe te starten:

1. Importeer de http en fs libraries (require).
2. Maak een http server
3. Lees de csv file in met de readFile functie
4. Maak een array van de ingelezen file door: data.toString().split("\n");
5. Lees de key en values uit door: for(i in array) {

```
var parts = array[i].split(',');
```

en maak een associatieve array

6. Schrijf json terug naar de webpagina: res.writeHead({ "Content-Type": "application/json" });

```
res.end(JSON.stringify(responseData));
```

Express Server

Express is een NPM package voor het bouwen van web servers (meest gebruikte om web applicaties met Node.js te bouwen)

Installeer express: `npm install express --save`

```
var express = require("express");

var app = express();

app.get("/", function(req, res){
  res.send(200, "Hallo Tom");
});

app.listen(3000);
```

Express geeft ons een veel "cleanere" en gestructureerde code.

Bouw een Node.js API

We willen een API bouwen om een lijst van personen op te vragen en een gebruiker toe te voegen. De eindpunten zijn als volgt:

- GET: `/api/users` : retourneert een JSON array van alle personen
- POST: `/api/post` : verwacht een JSON array van een persoon om uiteindelijk in een database te bewaren

Read via de API

Oplossing:

```
var express = require("express");
var bodyparser = require("body-parser");

var app = express();
app.use(bodyparser.json());

app.get("/api/users", function(req, res){

  var personen = [
    {
      "naam": "Peeters",
      "voornaam": "Tom"
    },
    {
      "naam": "Vandepierre",
      "voornaam": "Mieke"
    }
  ];

  res.json(personen);
});

app.listen(3000);
```

Controleer eerst je package.json bestand en kijk of body-parser geïnstalleerd is: body-parser wordt door express gebruikt om JSON van POST requests te lezen.

`npm install -save body-parser`

Nu hebben we /api/users "gestubbed".

Post via de API

```
app.post("/api/users", function(req, res){  
  
  console.log("Post received");  
  console.log(req.body.naam);  
  console.log(req.body.voornaam);  
  
  res.send(201);  
});
```

res.send(201): http codes te vinden op: [http codes: http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html](http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html)

Om posts te versturen kan je gebruik maken van de Postman app van chrome om deze API te testen

