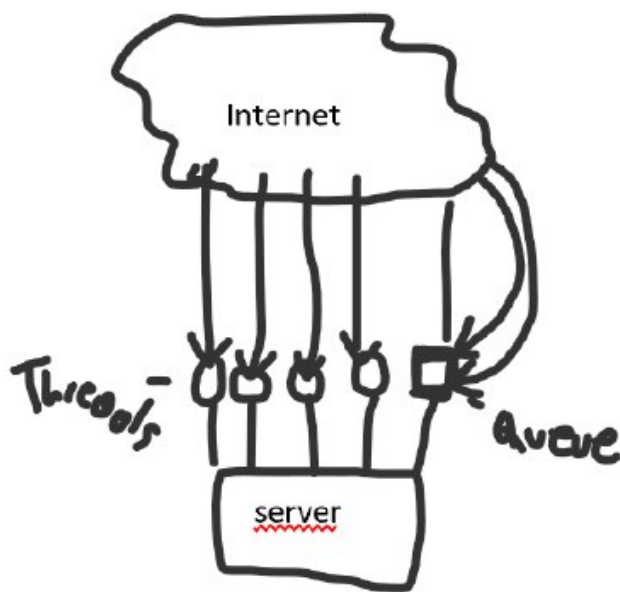
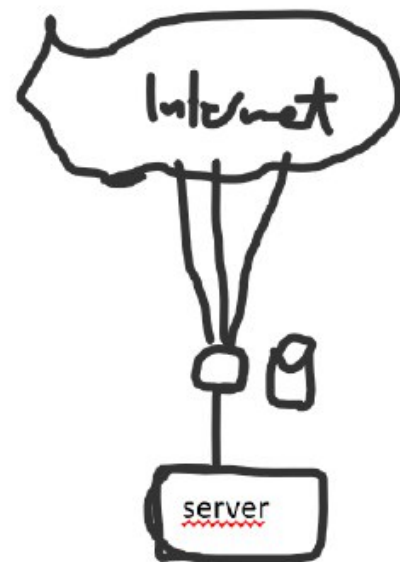


NODE.js Mitschrift

- Traditionell:
wird der Server über Threads kontaktiert. Das ist für REST Webseiten, bei denen durchgehend irgendwelche kleine Abfragen gemacht werden tödlich, da für jede Abfrage immer der Thread blockiert wird.
- Node.js:
Es gibt nur einen Thread. In diesem Thread ist eine kleine Queue drinnen, die nacheinander abgearbeitet wird. Alle Anfragen gelangen zu dem Thread.
Dieser Thread arbeitet **parallel** die Tasks ab.
Vorteile: wenig RAM nötig, nur ein Thread



Traditionell



Node.js

Aktueller JavaScript Standard: es6

Variablen Deklaration

Variablen müssen mit "var" deklariert werden. Ansonsten sind sie global. Wenn man dann in zwei verschiedenen Files die selbe Variable hat überschreiben sie sich. -> Man kann viel Blödsinn damit anstellen.

Verschiedene Funktionsdefinitionen:

```
function add5(x){  
  return x+5  
}
```

```
var add10 = function(x){
  return x+10
}

var add15 = (x) => {
  return x+15
}

function execute(fun, wert){
  console.log("Ergebnis:"+fun(wert))
}
```

Verschiedene Commands

Datei aus Filesystem einlesen:

Mit require lädt man eine Library und weist sie einer Variable zu.

```
const fs = require('fs')
```

Mit **Callbacks** können solche Aufrufe dann gemacht werden, damit sie den Thread nicht blockieren.

```
fs.readFile('Hello.txt', (err,data) => {
  if(err) throw exception
  console.log(data.toString())
})
```

Funktionen in Funktionen zurückliefern

```
function addFun(x,y){
  return () => {return "Ergebnis von der addFun: "+(x+y)}
}

var abc = addFun(1,2)
console.log(abc())
```

Node Package Manager (npm)

```
npm init
npm install --save express
```