
Teoria współbieżności

Przetwarzanie asynchroniczne (wstęp do Node.js)

Podstawy

1. Strona domowa: <http://nodejs.org>
2. Rejestr modułów - NPM: <http://npmjs.org>

Instalacja potrzebnego modułu:

```
npm install <nazwa_modułu>
```

3. Zarządzanie zależnościami: plik [package.json](#)

```
{
  "name": "my-project",
  "description": "...",
  "author": "John Doe <jdoe@mail.com>",
  "dependencies": {
    "q": "1.5.x",
    "async": "3.2.x"
  },
  "engine": "node >= 14.15.x"
}
```

Instalacja wszystkich zależności z pliku package.json:

```
npm install
```

4. [Zbiór artykułów](#)

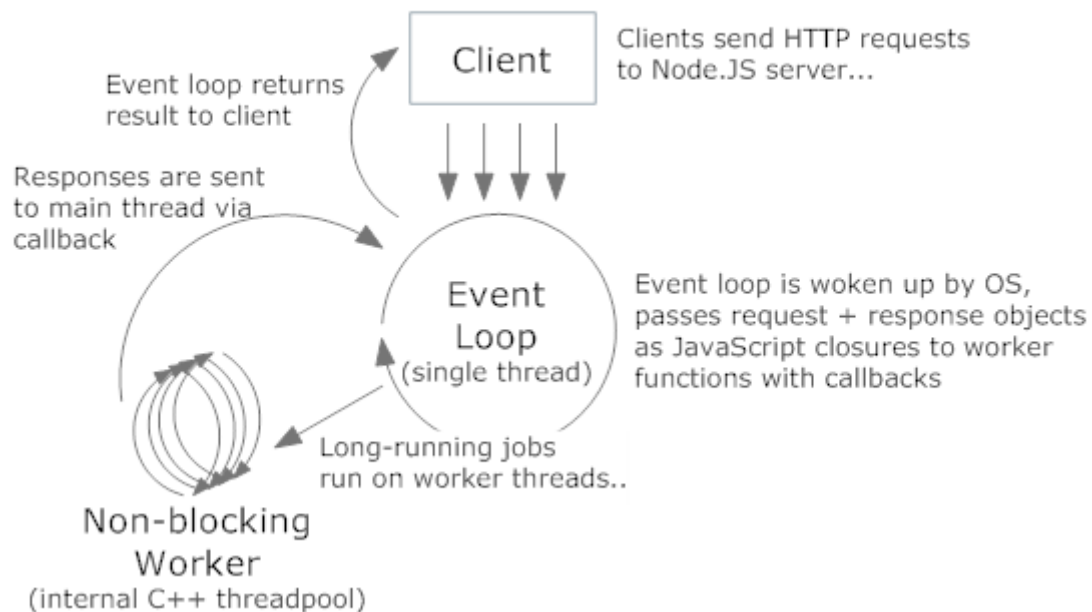
Motywacja powstania: wysoki koszt operacji I/O

The cost of I/O	
L1-cache	3 cycles
L2-cache	14 cycles
RAM	250 cycles
Disk	41 000 000 cycles
Network	240 000 000 cycles

Model przetwarzania Node.js

1. *Everything runs in parallel except your code.*
2. [Przetwarzanie asynchroniczne](#)
3. Jak działa pętla zdarzeń w JavaScript? [artykuł 1](#) [artykuł 2](#)
4. Obiekt [process](#)
5. Działanie [process.nextTick\(\)](#)

Node.JS Processing Model



Przetwarzanie asynchroniczne

- Uruchom kilkakrotnie następujący [program](#).
- Wymuszanie sekwencyjnego wykonania:
 - [Rozwiązanie 1](#)
 - [Rozwiązanie 2](#)

Mechanizm [obietnic](#)

- [Rozwiązanie 3](#)

Składnia [async/await](#)

- [Rozwiązanie 4](#)

Zadanie 1

1. **Zadanie 1a:** Zaimplementuj funkcję `loop`, wg instrukcji w pliku z Rozwiązaniem 3.
2. **Zadanie 1b:** wykorzystaj funkcję `waterfall` biblioteki [async](#).

Zadanie 2

Proszę napisać program obliczający liczbę linii we wszystkich plikach tekstowych z danego drzewa katalogów. Do testów proszę wykorzystać zbiór danych [Traceroute Data](#). Program powinien wypisywać liczbę linii w każdym pliku, a na końcu ich globalną sumę. Proszę zmierzyć czas wykonania dwóch wersji programu:

- z synchronicznym (jeden po drugim) przetwarzaniem plików,
- z asynchronicznym (jednoczesnym) przetwarzaniem plików.

Przydatne moduły:

- [walkdir](#) -- trawersacja drzewa katalogów
- [fs](#) -- operacje na systemie plików (moduł wbudowany)

Do obliczania liczby linii w pliku tekstowym proszę wykorzystać następujący fragment

kodu:

```
fs.createReadStream(file).on('data', function(chunk) {
    count += chunk.toString('utf8')
    .split(/\r\n|[\n\r\u0085\u2028\u2029]/g)
    .length-1;
}).on('end', function() {
    console.log(file, count);
}).on('error', function(err) {
    console.error(err);
});
```

Fragment ten tworzy strumień i rejestruje trzy funkcje obsługi zdarzeń (wczytanie fragmentu danych, koniec strumienia i wystąpienie błędu). (Zobacz [obsługa zdarzeń](#) w Node.js).

W implementacji proszę wykorzystać wzorzec asynchronicznego przetwarzania równoległego opisany [tutaj](#).
