

Node --prof

Profiling realizado con `--prof` de NodeJs y procesado con `--prof-process`, sobre un test realizado con Artillery emulando 50 conexiones concurrentes con 20 request cada una sobre la ruta `"/info"` agregando o extrayendo un `console.log` de la información colectada antes de devolverla al cliente.

Resultados:

. con console.log()

```
[Summary]:
  ticks  total  nonlib   name
    15    0.4%  100.0%  JavaScript
     0    0.0%   0.0%    C++
    11    0.3%   73.3%    GC
  3391   99.6%                Shared libraries
```

. sin console.log()

```
[Summary]:
  ticks  total  nonlib   name
    10    0.3%  100.0%  JavaScript
     0    0.0%   0.0%    C++
    10    0.3%  100.0%    GC
  2852   99.7%                Shared libraries
```

Conclusión:

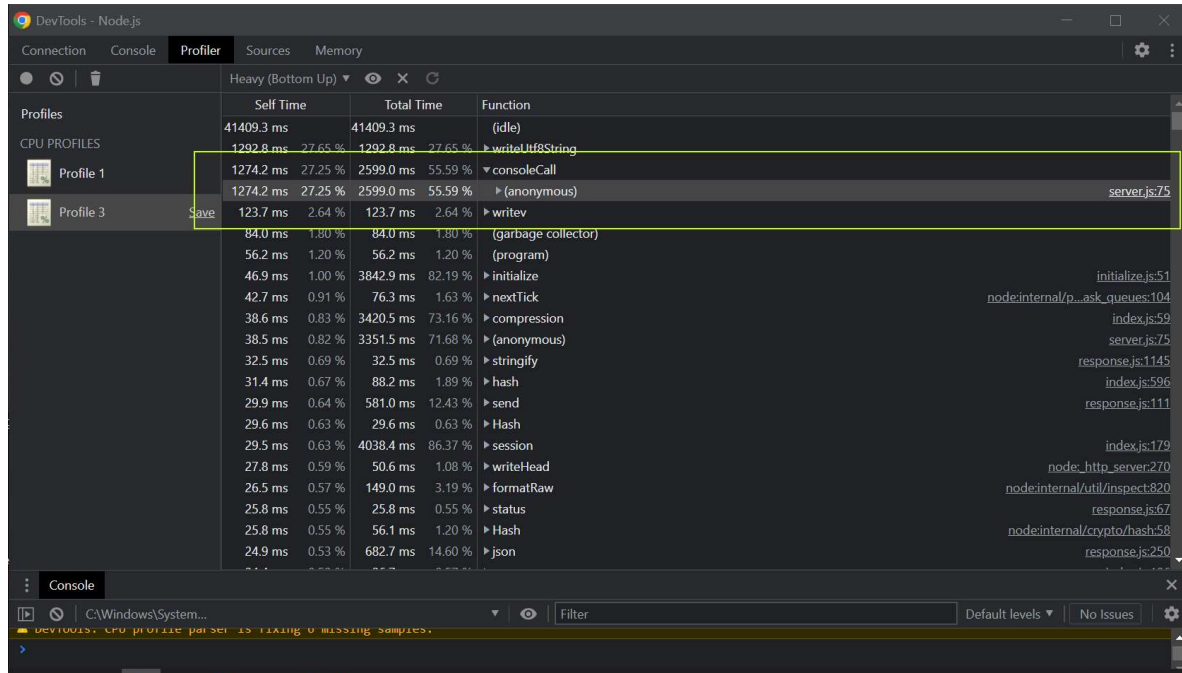
Los ticks de JavaScript aumentaron un 50% con el `console.log()`, y los ticks de librerías compartidas un 20%

Node inspect

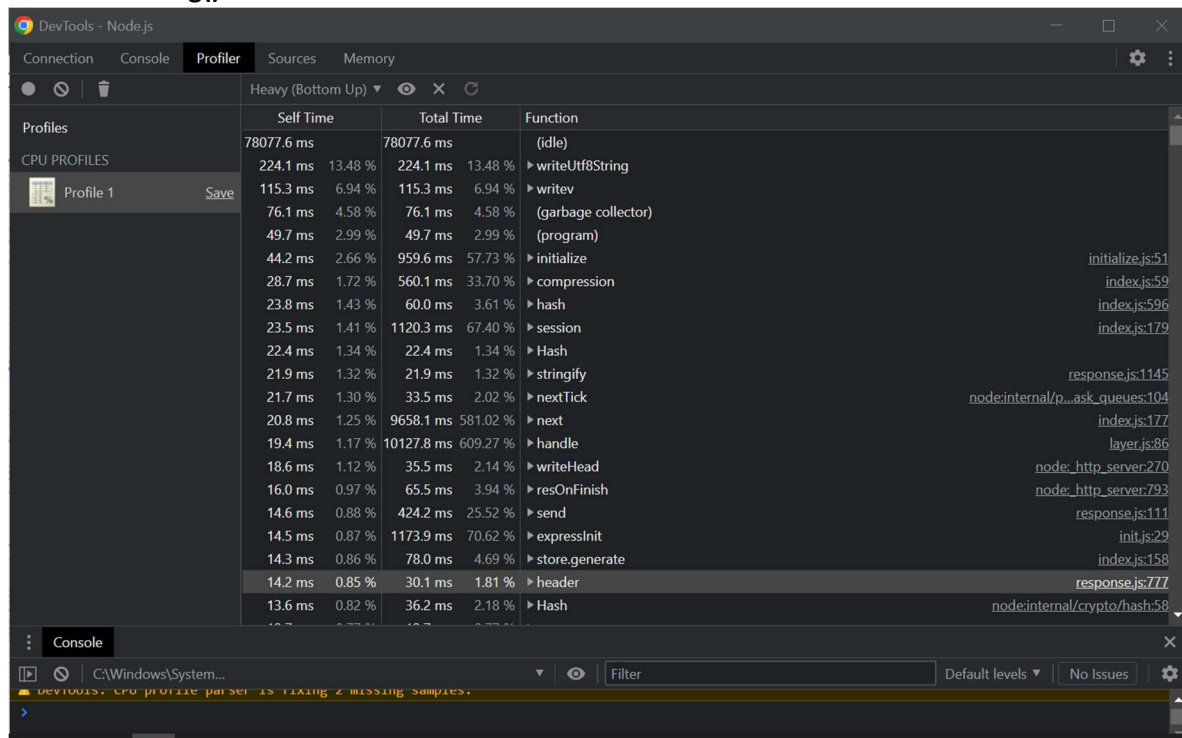
Profiling realizado con `–inspect` de NodeJs, sobre el mismo test que el caso anterior.

Resultados:

. con `console.log()`



. sin `console.log()`



Conclusión:

Aparecen las llamadas a Consola como uno de los procesos menos performantes.

Autocannon

Test realizado con Autocannon emulando 100 conexiones concurrentes realizadas durante un tiempo de 20 segundos sobre la ruta “/info” agregando o extrayendo un console.log de la información colectada antes de devolverla al cliente.

Resultados

. con console.log

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
PS C:\Mico\Programacion\CoderHouse\backend\CoderHouse\clase32\entregableClase32> npx autocannon -c 100 -d 20 "http://localhost:8080/info"
npm WARN config global '--global', '--local' are deprecated. Use '--location=global' instead.
Running 20s test @ http://localhost:8080/info
100 connections
```

| Stat | 2.5% | 50% | 97.5% | 99% | Avg | Stdev | Max |
|---------|--------|--------|--------|--------|-----------|---------|--------|
| Latency | 162 ms | 231 ms | 290 ms | 316 ms | 232.79 ms | 30.6 ms | 404 ms |

| Stat | 1% | 2.5% | 50% | 97.5% | Avg | Stdev | Min |
|-----------|--------|--------|--------|--------|--------|---------|--------|
| Req/Sec | 357 | 357 | 430 | 452 | 426.7 | 24.54 | 357 |
| Bytes/Sec | 209 kB | 209 kB | 252 kB | 264 kB | 250 kB | 14.3 kB | 209 kB |

Req/Bytes counts sampled once per second.
of samples: 20
9k requests in 20.05s, 4.99 MB read
PS C:\Mico\Programacion\CoderHouse\backend\CoderHouse\clase32\entregableClase32>

. sin console.log()

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
PS C:\Mico\Programacion\CoderHouse\backend\CoderHouse\clase32\entregableClase32> npx autocannon -c 100 -d 20 "http://localhost:8080/info"
npm WARN config global '--global', '--local' are deprecated. Use '--location=global' instead.
Running 20s test @ http://localhost:8080/info
100 connections
```

| Stat | 2.5% | 50% | 97.5% | 99% | Avg | Stdev | Max |
|---------|-------|-------|--------|--------|----------|----------|--------|
| Latency | 60 ms | 68 ms | 107 ms | 114 ms | 73.06 ms | 13.89 ms | 156 ms |

| Stat | 1% | 2.5% | 50% | 97.5% | Avg | Stdev | Min |
|-----------|--------|--------|--------|--------|---------|---------|--------|
| Req/Sec | 994 | 994 | 1394 | 1467 | 1356.15 | 122.84 | 994 |
| Bytes/Sec | 582 kB | 582 kB | 816 kB | 859 kB | 793 kB | 71.9 kB | 581 kB |

Req/Bytes counts sampled once per second.
of samples: 20
27k requests in 20.04s, 15.9 MB read
PS C:\Mico\Programacion\CoderHouse\backend\CoderHouse\clase32\entregableClase32>

Resultado:

Con console.log() el tiempo de latencia aumenta un 300% aproximadamente, y la cantidad de request por segundo disminuye también un 300%.

FlameGraph de Ox

Se realizao el gráfico de flamas con la librería Ox sobre el test anterior.

Resultados

.con console.log()



.sin console.log()



Conclusión

No llego a apreciar grandes cambios.