

Self Time		Total Time		Function
51452.7 ms		51452.7 ms		(idle)
1023.0 ms	10.43 %	1436.7 ms	14.65 %	▸ consoleCall
699.4 ms	7.13 %	699.4 ms	7.13 %	(garbage collector)
331.0 ms	3.38 %	340.6 ms	3.47 %	▸ writeBuffer
319.4 ms	3.26 %	319.4 ms	3.26 %	(program)
292.8 ms	2.99 %	292.8 ms	2.99 %	▸ writeUtf8String
219.0 ms	2.23 %	219.0 ms	2.23 %	▸ writev
184.0 ms	1.88 %	190.3 ms	1.94 %	▸ next
179.6 ms	1.83 %	179.6 ms	1.83 %	▸ stat
112.5 ms	1.15 %	575.3 ms	5.87 %	▸ SourceNode_walk
104.2 ms	1.06 %	139.1 ms	1.42 %	▸ SourceNode_add
97.5 ms	0.99 %	347.6 ms	3.55 %	▸ parse
91.7 ms	0.93 %	91.7 ms	0.93 %	▸ registerDestroyHook
78.5 ms	0.80 %	322.7 ms	3.29 %	▸ deserializeObject
77.3 ms	0.79 %	98.8 ms	1.01 %	▸ resolve
74.7 ms	0.76 %	243.8 ms	2.49 %	▸ createFunctionContext
71.2 ms	0.73 %	4119.5 ms	42.02 %	step

Self Time		Total Time		Function
27591.4 ms		27591.4 ms		(idle)
2742.5 ms	18.07 %	5335.8 ms	35.15 %	▸ consoleCall
2370.9 ms	15.62 %	2370.9 ms	15.62 %	▸ writeUtf8String
822.5 ms	5.42 %	822.5 ms	5.42 %	(garbage collector)
317.9 ms	2.09 %	317.9 ms	2.09 %	(program)
299.9 ms	1.98 %	309.1 ms	2.04 %	▸ writeBuffer
238.1 ms	1.57 %	238.1 ms	1.57 %	▸ writev
211.7 ms	1.39 %	237.0 ms	1.56 %	▸ set
208.4 ms	1.37 %	208.4 ms	1.37 %	▸ stat
204.6 ms	1.35 %	205.5 ms	1.35 %	▸ cache
129.1 ms	0.85 %	658.7 ms	4.34 %	▸ SourceNode_walk
126.9 ms	0.84 %	191.1 ms	1.26 %	▸ next
113.6 ms	0.75 %	155.6 ms	1.03 %	▸ SourceNode_add
109.1 ms	0.72 %	471.3 ms	3.11 %	▸ deserializeObject
98.4 ms	0.65 %	367.1 ms	2.42 %	▸ parse
82.0 ms	0.54 %	4834.4 ms	31.85 %	step
79.5 ms	0.52 %	79.5 ms	0.52 %	▸ getColorDepth

running tests.....
Running 20s test @ http://localhost:8080/info
100 connections

Stat	2.5%	50%	97.5%	99%	Avg	Stdev	Max
Latency	256 ms	1487 ms	1510 ms	1559 ms	1131.71 ms	538.41 ms	1581 ms

Stat	1%	2.5%	50%	97.5%	Avg	Stdev	Min
Req/Sec	0	0	0	69	5	15.83	4
Bytes/Sec	0 B	0 B	0 B	130 kB	9.39 kB	29.7 kB	7.51 kB

Req/Bytes counts sampled once per second.
of samples: 20

308 requests in 20.43s, 188 kB read
108 errors (108 timeouts)

Running 20s test @ http://localhost:8080/info?console=1
100 connections

Stat	2.5%	50%	97.5%	99%	Avg	Stdev	Max
Latency	1515 ms	1559 ms	1687 ms	1738 ms	1577.32 ms	53.9 ms	1740 ms

Stat	1%	2.5%	50%	97.5%	Avg	Stdev	Min
Req/Sec	0	0	0	100	5	21.8	100
Bytes/Sec	0 B	0 B	0 B	188 kB	9.39 kB	40.9 kB	188 kB

Req/Bytes counts sampled once per second.
of samples: 20

300 requests in 20.49s, 188 kB read
100 errors (100 timeouts)

Summary report @ 10:37:02(-0300)

```
http.codes.200: ..... 1000
http.request_rate: ..... 26/sec
http.requests: ..... 1000
http.response_time:
  min: ..... 49
  max: ..... 1089
  median: ..... 432.7
  p95: ..... 889.1
  p99: ..... 1043.3
http.responses: ..... 1000
vusers.completed: ..... 50
vusers.created: ..... 50
vusers.created_by_name.0: ..... 50
vusers.failed: ..... 0
vusers.session_length:
  min: ..... 17210.2
  max: ..... 19267.3
  median: ..... 18588.1
  p95: ..... 18963.6
  p99: ..... 18963.6
```

Summary report @ 10:29:53(-0300)

```
http.codes.200: ..... 1000
http.request_rate: ..... 19/sec
http.requests: ..... 1000
http.response_time:
  min: ..... 241
  max: ..... 2253
  median: ..... 713.5
  p95: ..... 1652.8
  p99: ..... 2018.7
http.responses: ..... 1000
vusers.completed: ..... 50
vusers.created: ..... 50
vusers.created_by_name.0: ..... 50
vusers.failed: ..... 0
vusers.session_length:
  min: ..... 26492.7
  max: ..... 27883.4
  median: ..... 27181.5
  p95: ..... 27730.6
  p99: ..... 27730.6
```



La conclusión que puedo sacar es que la ruta no bloqueante tuvo una mejor performance que la bloqueante