

# Análisis de test de carga

## result\_prof-nobloq

```
[Summary]:
  ticks  total  nonlib   name
    9     0.2%  100.0%  JavaScript
    0     0.0%   0.0%    C++
   28     0.6%  311.1%    GC
  4571    99.8%                Shared libraries
```

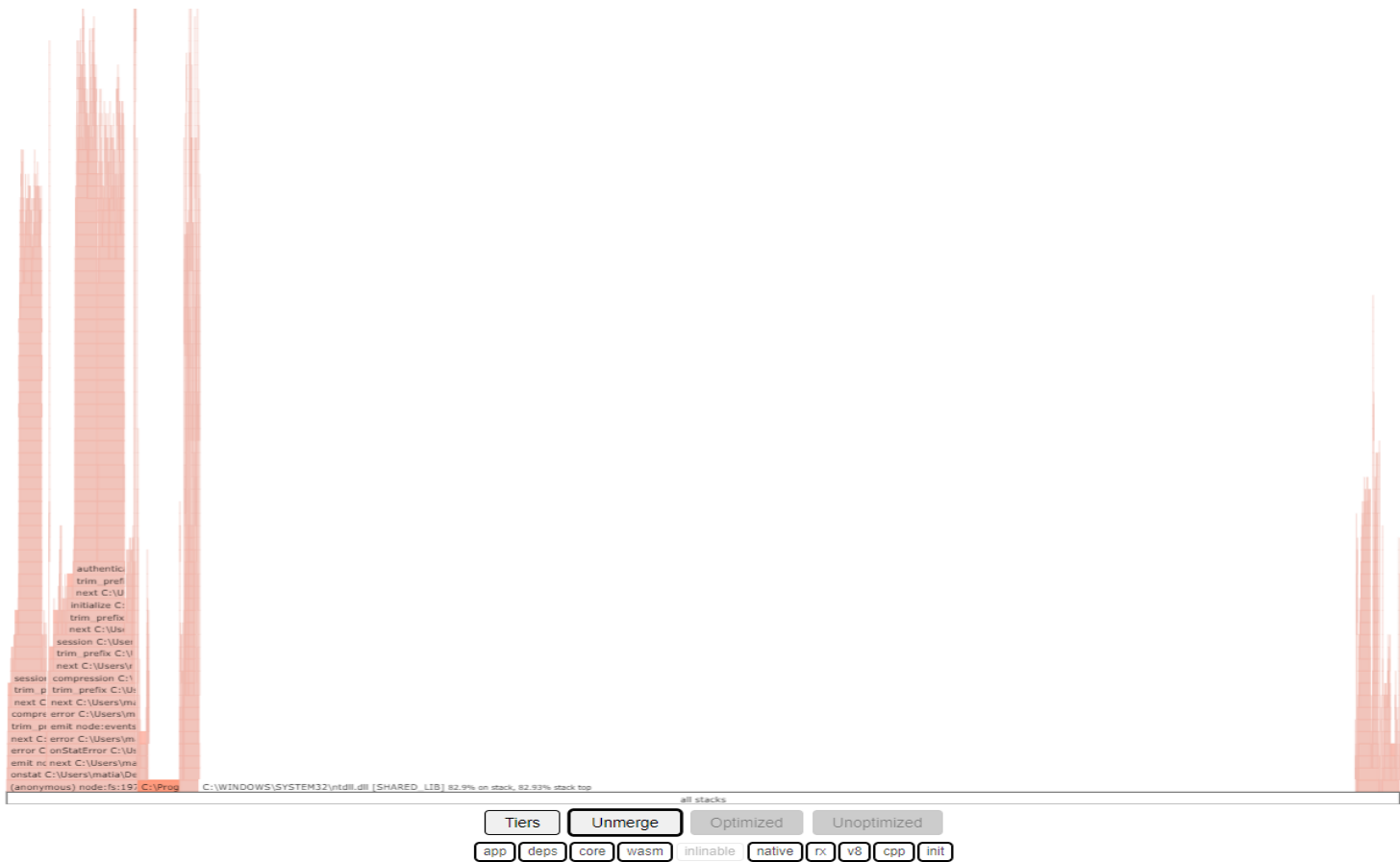
## result\_prof-bloq

```
[Summary]:
  ticks  total  nonlib   name
   15     0.2%  100.0%  JavaScript
    0     0.0%   0.0%    C++
   28     0.3%  186.7%    GC
  9788    99.8%                Shared libraries
```

## Conclusión

Podemos concluir mediante el análisis de perfilamiento mediante **node profile** que en el proceso no bloqueante tiene casi la mitad de ticks (sección "Shared libraries") que el bloqueante, esto quiere decir que los procesos tienen una aproximadamente del 50% menor.

# Análisis Gráfico Ox



## Aclaración

Lado izquierdo de la grafica →Proceso bloqueante

Lado izquierdo de la gráfica →Proceso no bloqueante

## Conclusión

Al igual que en el análisis anterior, el proceso no bloqueante tiene menos duración respecto al bloqueante. Podemos concluir que el proceso no bloqueante es mas eficiente en la carga.