



EDGE COMPUTING

no FRONTEND



Cícero Feijó



CÍCERO FEIJÓ



professorfeijo.com.br

Engenheiro Specialista

+ 25 anos de experiência em desenvolvimento web.

Instrutor

+ de 15 anos ministrando cursos e treinamentos.

Canal no Youtube

+ de 2k inscritos com conteúdo gratuito.

Palestras

TDC, GUFERS, FrontInPOA

Algumas perguntas para iniciar:

Você sabia que um clique “lento”
pode custar milhões?



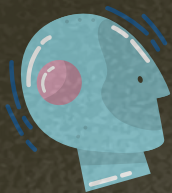
"Você já parou pra pensar onde o seu
código realmente roda?"



Alguns números

País	Serverless	Edge
United States	111 ms	93 ms
Japan	96 ms	86 ms
Singapore	95 ms	90 ms
Australia	101 ms	89 ms
South Africa	119 ms	62 ms

Conceitos iniciais



Hospedagens Tradicionais:

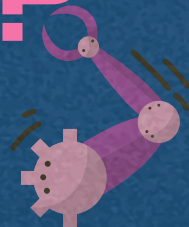
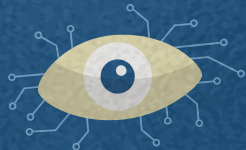
Você aluga um pedaço de servidor e sobe sua aplicação (PHP, HTML, banco, etc). O servidor está sempre ligado, mesmo que ninguém esteja acessando seu site.
Ex. Locaweb, KingHost, GoDaddy, etc.

Serverless:

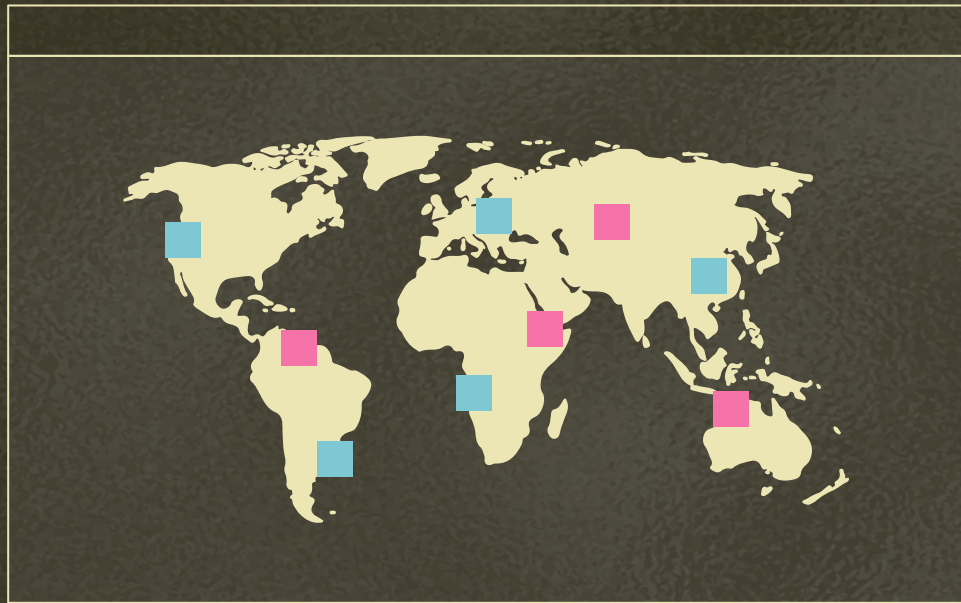
O provedor executa seu código sob demanda, aloca recursos conforme necessário, e encerra quando a execução termina. Você paga apenas pelas invocações e uso de memória/CPU.

Mas e a EDGE?

o QUE É A EDGE?



o que é Edge Computing?



Edge Computing é um modelo de computação distribuída em que o processamento e a execução do código ocorrem em servidores localizados geograficamente próximos ao usuário final, chamados de nós de borda (edge nodes).

Quais as vantagens?

Baixa latência

Ao entregar conteúdo mais próximo do usuário o tempo de resposta é reduzido significativamente.



Melhor escalabilidade global

Por rodar em múltiplos pontos distribuídos, a aplicação lida melhor com picos de tráfego e distribuição geográfica sem depender de um único data center.

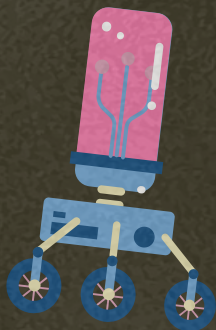
Custom Experience + rápida

Permite executar lógica de personalização (como geolocalização, A/B testing, idioma ou redirecionamento regional) antes mesmo do conteúdo ser entregue, sem atrasos perceptíveis.

Analogia rápida

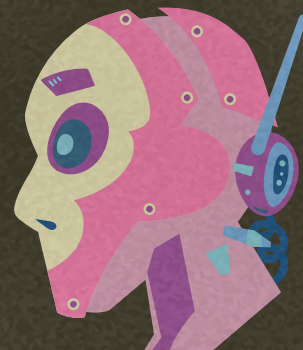
Serverless

É como ter um restaurante central em São Paulo. Todos os pedidos do Brasil são enviados para lá, preparados, e depois entregues.



Edge

É como ter pequenas cozinhas espalhadas por várias cidades. O pedido é preparado e entregue localmente — muito mais rápido.



Agora comparando os modelos

Característica	Tradicional	Serverless	Edge
Infraestrutura	Servidor fixo	Alocado sob demanda.	Nós de borda globais.
Escalabilidade	Limitada	Automática por invocação.	Automática e distribuída.
Local de execução	Data center fixo	Data center regional (ex: us-east-1)	Mais próximo do usuário (edge node)
Latência	Média a alta	Média	Baixa
Tempo de vida do processo	Longo (sempre ligado)	Curto (sob demanda)	Muito curto (na borda)

Mercado de edge



Azion

Brasil

Fundada em 2011



Cloudflare

EUA

Fundada em 2009



Fastly

EUA

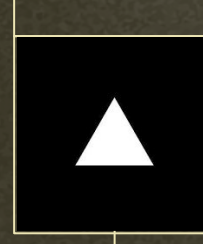
Fundada em 2011



Akamai

EUA

Fundada em 1998



Vercel

EUA

Fundada em 2015

Como a Edge sabe onde estou?

DNS Anycast

Protocolo de rede que direciona a requisição automaticamente para o servidor Edge mais próximo do ponto de origem da requisição.

- A decisão é feita pela infraestrutura da rede (BGP) e não pelo navegador.
- O provedor (ex: Cloudflare, Vercel, Akamai) opera vários nós espalhados pelo mundo, e o tráfego "flui" até o nó mais próximo com base em menor número de saltos (hops) ou menor latência.
- Faz uso de IP's Anycast para resolução de servidores, diferente do modelo IP Unicast.

Como a Edge sabe onde estou?

Sobre o BGP

BGP (Border Gateway Protocol) é o protocolo de roteamento principal da internet. Ele é usado para decidir por onde os dados devem trafegar entre redes diferentes, escolhendo o melhor caminho possível.

Uma espécie de WAZE da internet.

EXEMPLO:

O domínio "professorfeijo.com.br" resolve para um endereço de IP Anycast.

IP's Anycast está associado a diversos servidores Edge ao redor do mundo.

O BGP é o responsável por roteá-lo até o servidor mais próximo, com base em:

- Menor número de saltos (hops)
- Melhor performance de rede
- Menor latência
- Políticas específicas do provedor

Como a Edge sabe onde estou?

Endereço IP do Usuário

O nó Edge que recebe a requisição consegue inferir a localização aproximada do usuário analisando seu endereço IP.

Plataformas como Vercel, Netlify e Cloudflare expõem isso em headers como:

- x-vercel-ip-country
- cf-ipcountry
- x-vercel-ip-city

Isso permite personalizar o conteúdo na borda: idioma, moeda, redirecionamento, A/B testing, etc.

Como a Edge sabe onde estou?

Distribuição automática pelo provedor

Plataformas como Cloudflare, AWS CloudFront, Vercel e Netlify fazem a replicação automática de código e ativos estáticos para seus pontos Edge ao redor do mundo.

Quando o usuário faz uma requisição:

- A rede direciona para o nó Edge mais próximo.
- O nó executa o código local (se for uma função Edge) ou entrega o arquivo cacheado.

Pontos de presença

<https://vercel.com/docs/edge-network>

Vercel Edge Network

Our global Edge Network has 119 globally distributed points of presence.



VERCEL:

- 119 pontos
- 94 cidades
- 51 países

Possui clientes como:

- Adobe
 - Notion
 - Pinterest
 - Loom
- entre outros.

Pontos de presença

<https://www.azion.com/pt-br/produtos/nossa-rede/>



AZION:

- Mais de 100 pontos globais.
- 54 pontos no Brasil.

Possui clientes como:

- Magalu
 - Dafiti
 - Renner
 - Marisa
 - Pernambucanas
- entre outros.

E-Commerce na Black Friday



Latência

Milhares de usuários simultâneos.

Escalabilidade

Evita sobrecarga de um unico data center.

Customização

Exibição de banners e promoções por região.

Segurança

Firewalls, rate limits e bloqueios de bots.

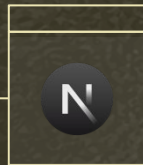
A/B Testing

Testes de UI, preços ou frete por ex.

Experimentando diretamente no código



Aplicação em NextJS rodando na Vercel.



Duas rotas configuradas:

- /api: usando modelo serverless.
- /api/edge: configurada para recursos de edge.

src/api/routes.ts

MODELO SERVERLESS

```
import { NextResponse } from "next/server";

export async function GET() {
  return NextResponse.json({ source: "API tradicional", now: Date.now() });
}
```

src/api/edge/routes.ts

MODELO EDGE

```
export const runtime = "edge"; // 📌 importante!
```

```
export async function GET() {  
  return new Response(  
    JSON.stringify({  
      source: "API no edge",  
      now: Date.now(),  
    }),  
    {  
      status: 200,  
      headers: {  
        "Content-Type": "application/json",  
      },  
    }  
  );  
}
```

Teste de Latência

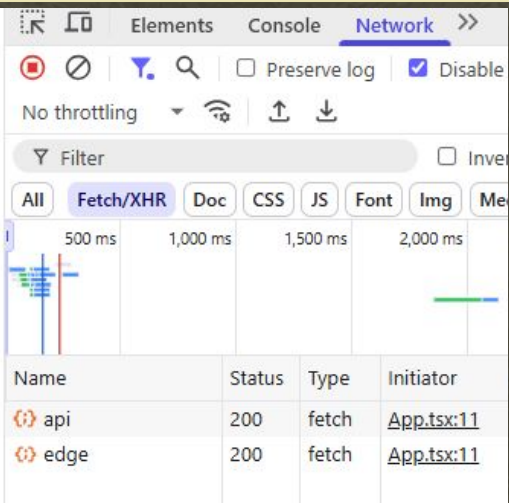
Chamar API Tradicional

216.8 ms

Chamar Função no Edge

78.6 ms

BENCHMARK



<https://back-end-na-edge-gufers.vercel.app/api>

<https://back-end-na-edge-gufers.vercel.app/api/edge>



176%

De diferença entre os
tempos das respostas.

Interpretando a entrega

MODELO SERVERLESS

X-Matched-Path	/api
X-Vercel-Cache	MISS
X-Vercel-Id	gru1:iad1:4hl2g-1752372806124-0f0440caf859

API Serverless:

Nó de borda que recebeu a requisição: gru1 (São Paulo, Brasil)

API Serverless:

Servidor central onde o código foi executado: iad1 (Virgínia, EUA).

Interpretando a entrega

MODELO EDGE

X-Matched-Path	/api/edge
X-Vercel-Cache	MISS
X-Vercel-Id	gru1:gru1:pbknr-1752372806936-165e2756af83

API Serverless:

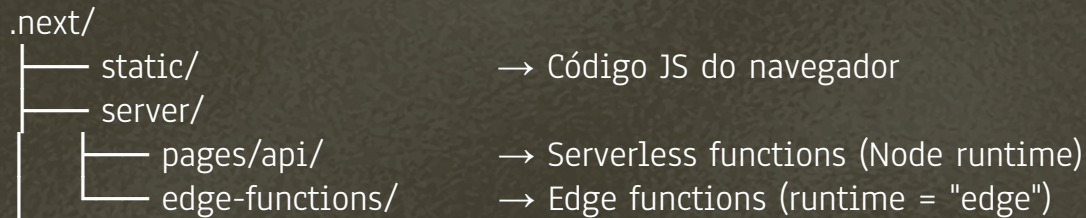
Nó de borda que recebeu a requisição: gru1 (São Paulo, Brasil)

API Serverless:

Servidor central onde o código foi executado: gru1 (São Paulo, Brasil).

Empacotamento na Edge

MODELO EDGE



Código Serverless:

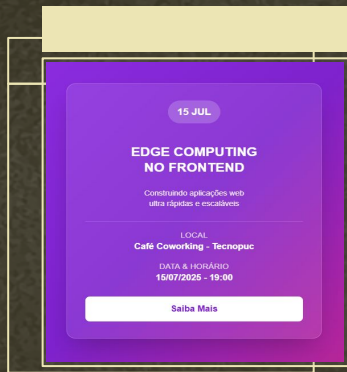
- Contém as rotas de API padrão (Serverless), em arquivos .js.
- Esse código roda em ambientes como AWS Lambda ou Vercel Serverless Functions.

Código Edge (quando runtime = "edge"):

- Contém os bundles das Edge Functions, como middlewares e APIs marcadas com runtime = 'edge'.
- Código empacotado e otimizado para execução em ambientes Edge Isolate.

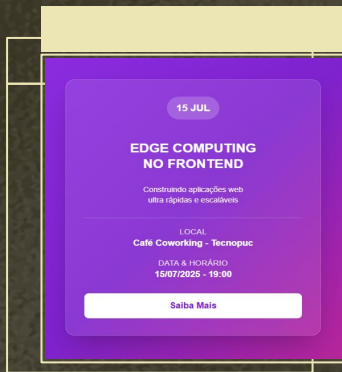
Benchmarking de um front

Tradicional



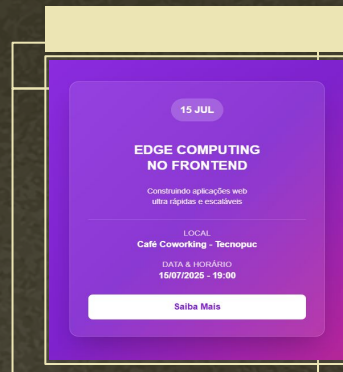
Rodando em servidor Apache na Locaweb.

Serverless



Rodando na Vercel sem uso da Edge.

Edge



Rodando na Vercel com a Edge habilitada.

Benchmarking de um front

USAM05 0 V0 DA VERCEL

 /  Personal Free / Edge Computin... / Edge Computing webpage Private

Chat

Design

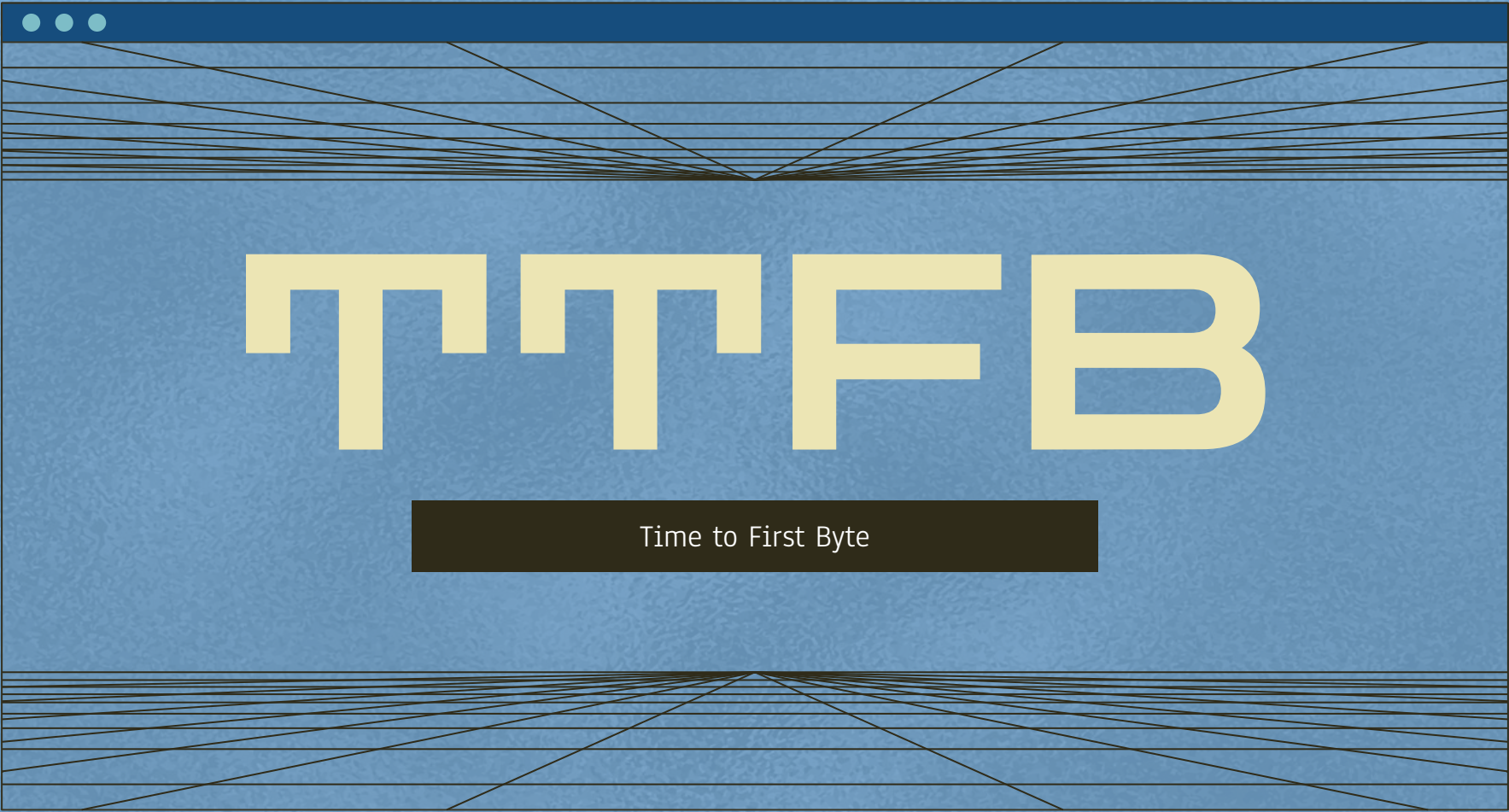


Create a single page with some basic informations about a specific content called "Edge Computing no Frontend", location: "Café Coworking - Tecnopuc" - When: 15/07/2025 - 19hr.Put a square with this information in the center of the page, use the colors of the attached image to compose the UI.



download.jpg
8.21kB

v0.dev



TTFB

Time to First Byte

TTFB Time to First Byte

Métrica de performance que mede o tempo entre o momento em que o navegador faz uma requisição para o servidor e o momento em que o primeiro byte de resposta começa a ser recebido.



Tempo de Conexão



Tempo de processamento
no servidor



Tempo até o 1º byte
chegar

TTFB Time to First Byte



Como monitorar?

PELO BROWSER:

- Web Inspector -> Lighthouse -> Performance -> Diagnostics
- Network -> Timing -> Waiting for Server Response

OBSERVABILIDADE:

- Grafana Faro
- New Relic Browser
- Datadog RUM (Real User Monitoring)
- Google Analytics V4
- webpagetest.org

Excelente
< 200ms

Aceitável
200 - 500ms

Lento
> 500ms

TTFB Time to First Byte

Configuration Choose from all browser, location, & device options

Test Location: 

Browser:

141 Runs Left | Upgrade

Start Test →

Test Settings	Advanced	Chromium	Script	Block	SPOF
Connection	Cable (5/1 Mbps 28ms RTT)				
Desktop Browser Dimensions	default (1366x768)				
Number of Tests to Run	3				
Repeat View	☒ First View and Repeat View ☐ First View Only				
Label					

webpagetest.org

- Acesso de Dulles, Virginia, EUA.
- Browser Chrome.
- Conexão de 5Mbps
- Round Trip de 28ms

TTFB Time to First Byte

Tradicional

15 JUL

**EDGE COMPUTING
NO FRONTEND**

Construindo aplicações web
ultra rápidas e escaláveis

LOCAL
Café Coworking - Tecnopuc

DATA & HORÁRIO
15/07/2025 - 19:00

Saiba Mais

TTFB
1.167s

Serverless

15 JUL

**EDGE COMPUTING
NO FRONTEND**

Construindo aplicações web
ultra rápidas e escaláveis

LOCAL
Café Coworking - Tecnopuc

DATA & HORÁRIO
15/07/2025 - 19:00

Saiba Mais

TTFB
0.308s

Edge

15 JUL

**EDGE COMPUTING
NO FRONTEND**

Construindo aplicações web
ultra rápidas e escaláveis

LOCAL
Café Coworking - Tecnopuc

DATA & HORÁRIO
15/07/2025 - 19:00

Saiba Mais

TTFB
0.177s

Links das demos

Detalhe	Repositórios	URL
UI para latência	https://github.com/csfeijo/front-end-na-edge-gufers/	-
API sem Edge - Vercel	https://github.com/csfeijo/back-end-na-edge-gufers	https://back-end-na-edge-gufers.vercel.app/api/
API com Edge - Vercel		https://back-end-na-edge-gufers.vercel.app/api/edge
Host com Apache	https://github.com/csfeijo/front-end-na-edge-gufers-sample-no-edge/tree/deploy-tradicional	http://professorfeijo.com.br/no-edge/
Host sem Edge na Vercel	https://github.com/csfeijo/front-end-na-edge-gufers-sample-no-edge	https://front-end-na-edge-gufers-sample-no.vercel.app/
Host com Edge na Vercel	https://github.com/csfeijo/front-end-na-edge-gufers-sample	https://v0-edge-computing-webpage.vercel.app/

OBRIGADE!



professorfeijo.com.br

CREDITS: This presentation template was created by **Slidesgo**, including icons by **Flaticon**, and infographics & images by **Freepik**