



We'll start in a few minutes

Pay the Web Forward with Web Monetization

Peter-Paul Koch

Beyond Tellerrand, 13-05-2024



Pay the web forward

@interledger #webmonetization

	iOS		iOS WebView		And. 4	AWK Web View	Chromium								Chromium WebView					UC 10	Black Berry 10	Dol phin	show page contents		IE 11	Edge 13	Firefox
	8	9	8	9			LG	Cy	HTC	S34	Ub	MM	S44	Xia	Op	Go	30	33	39				45	51			
<p>background-attachment: fixed</p> <p>The background image does not scroll; the element serves as a “window” on the background image.</p>	static		no-scroll		no	no	static	no								no-scroll					no	static	buggy	static	buggy	yes	
<p>Static Background image is correctly positioned on page load, but does not change on scrolling.</p> <p>No-scroll Works, but you can't scroll the element.</p> <ul style="list-style-type: none"> The Samsung Android 4 CRASHES on my test page. No data. Edge uses a kind-of parallax scrolling that's off. 																											
<p>background-attachment: local</p> <p>The background image scrolls with the element (and not the document).</p> <p>External test that disagrees with mine</p>	incomplete		no		no	buggy								no					buggy	no	incomplete	no	untestable	yes			
<p>Untestable Browser does not support scrollable elements.</p> <p>Incomplete Not supported when you also use <code>-webkit-overflow-scrolling: touch</code>.</p> <ul style="list-style-type: none"> The Samsung Android 4 CRASHES on my test page. No data. The Chromia need a <code>border-radius</code> on the element. This is obviously a bug. 																											
<p>background-clip</p> <p>States whether the background cov-</p>	yes		yes		yes								yes					yes									

- Introduce QuirksMode as combination of MDN and caniuse, and why I stopped working on it around 2013: I didn't make any money



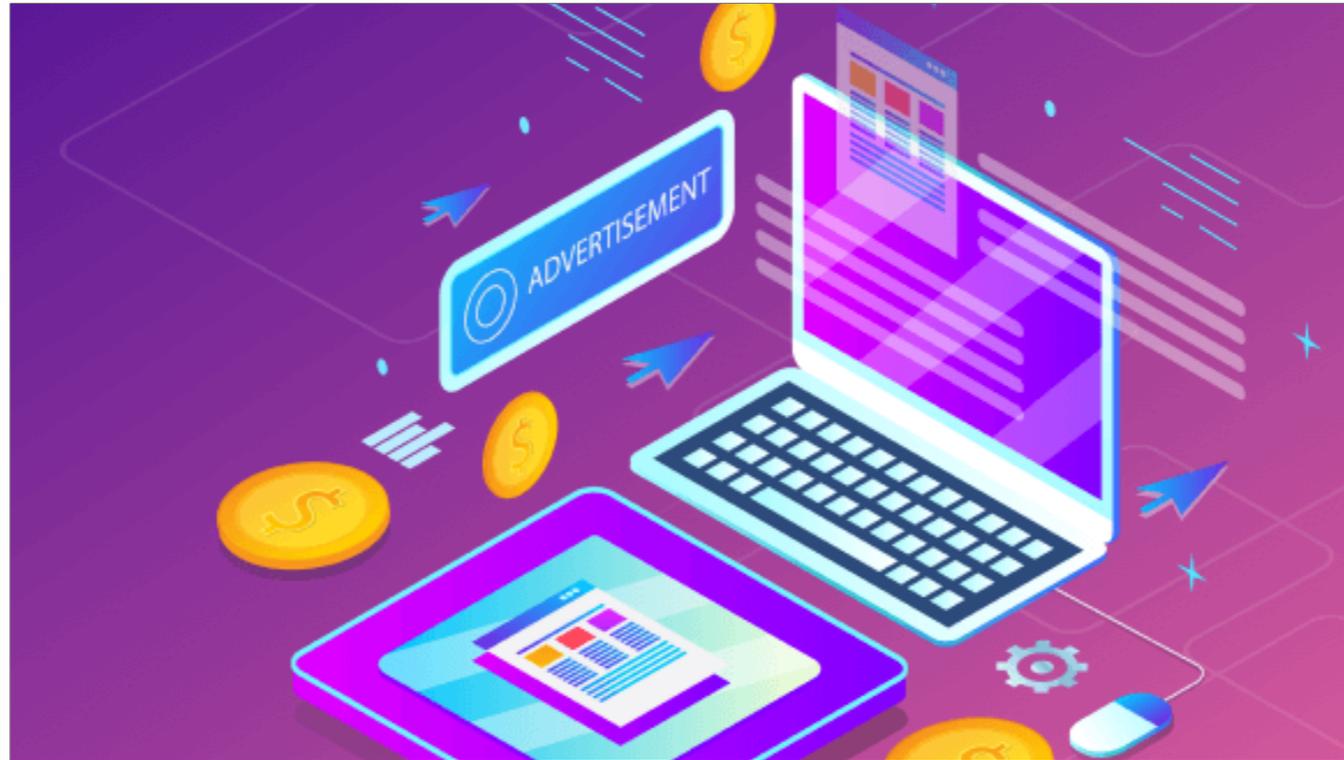
1

WHY?



Why should websites be free?

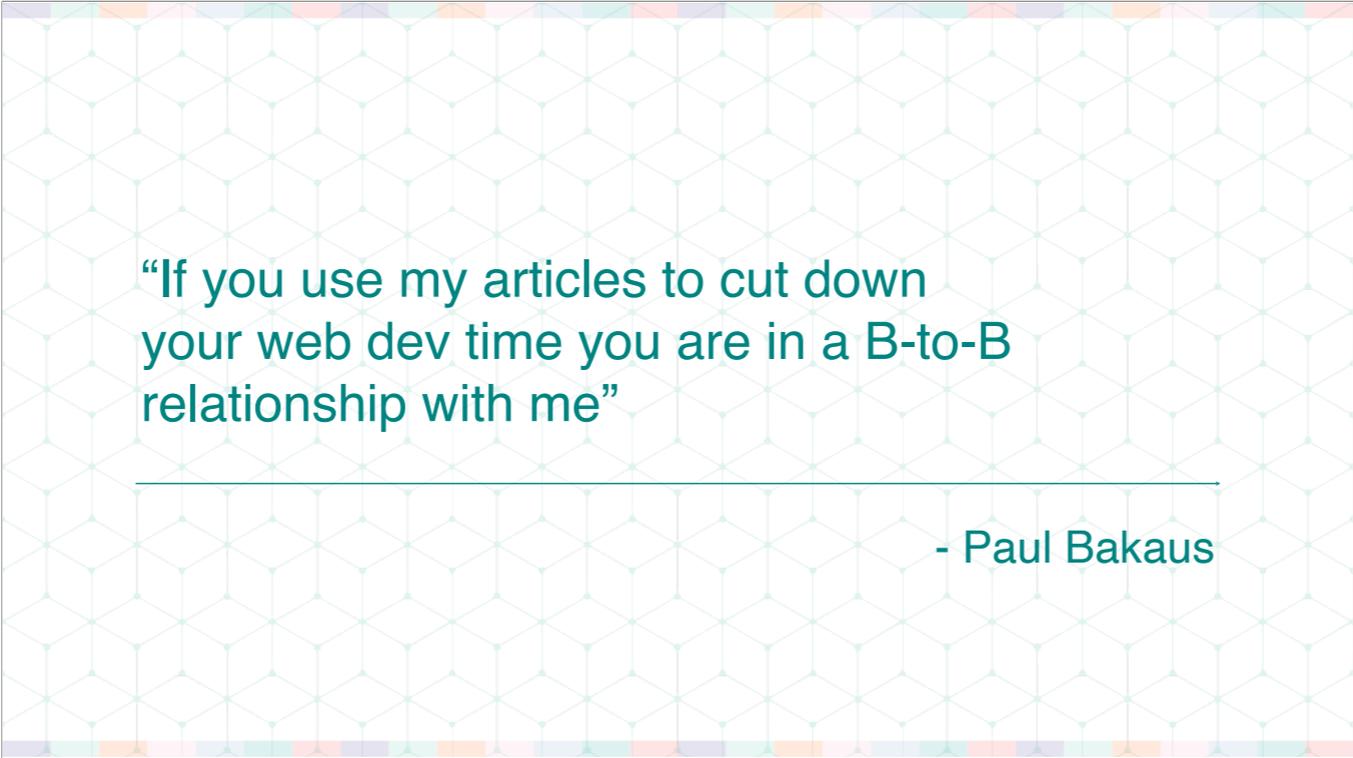
- People don't want to pay for online content. Why not? Because back in the late nineties we set the wrong example. Everything had to be free, for reasons that I don't understand in hindsight.



- If websites are free, ads are the best solution. And they're not a very good solution.

Still, easy for site owners

and convenient for visitors who can carefully side-step the question of payment



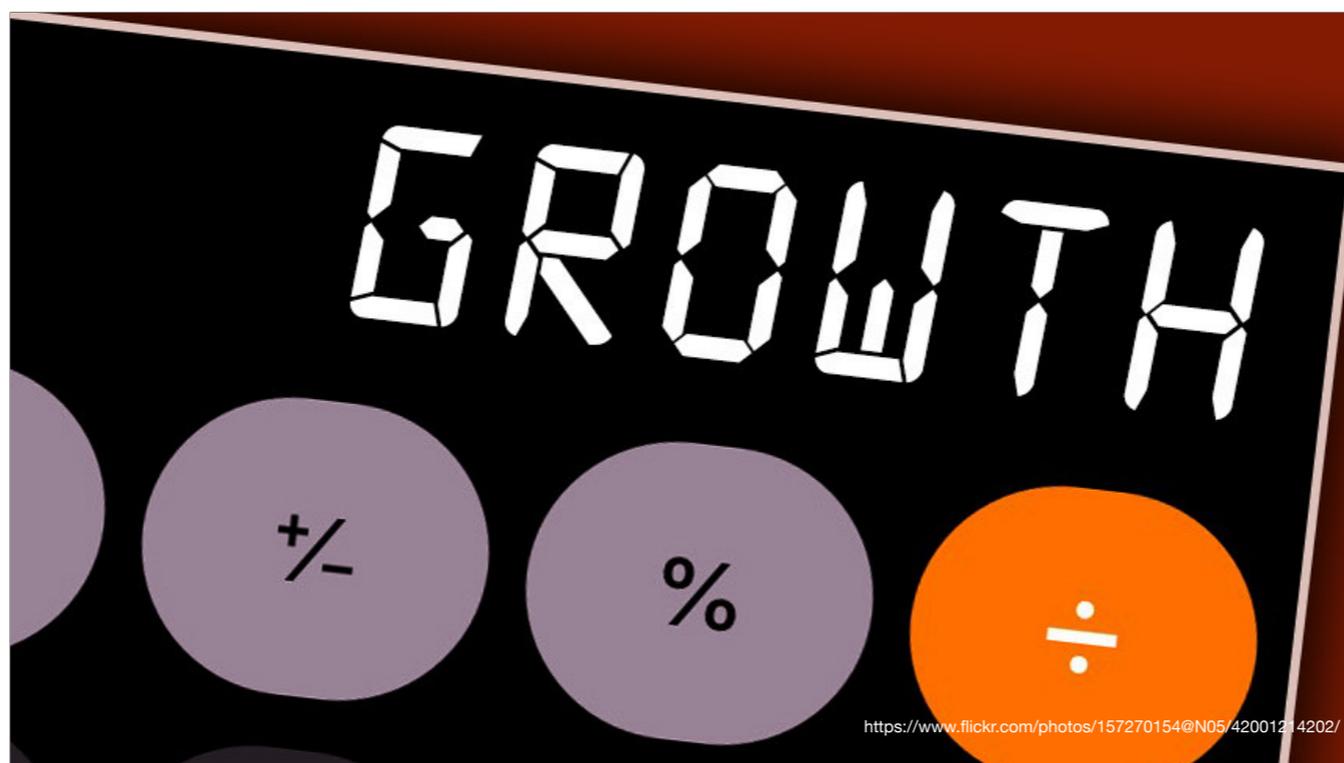
“If you use my articles to cut down
your web dev time you are in a B-to-B
relationship with me”

- Paul Bakaus

- Goes for QuirksMode, but also for MDN, caniuse, and maybe even for Stack Overflow? And there will be plenty of other sites that are worthy of your money.



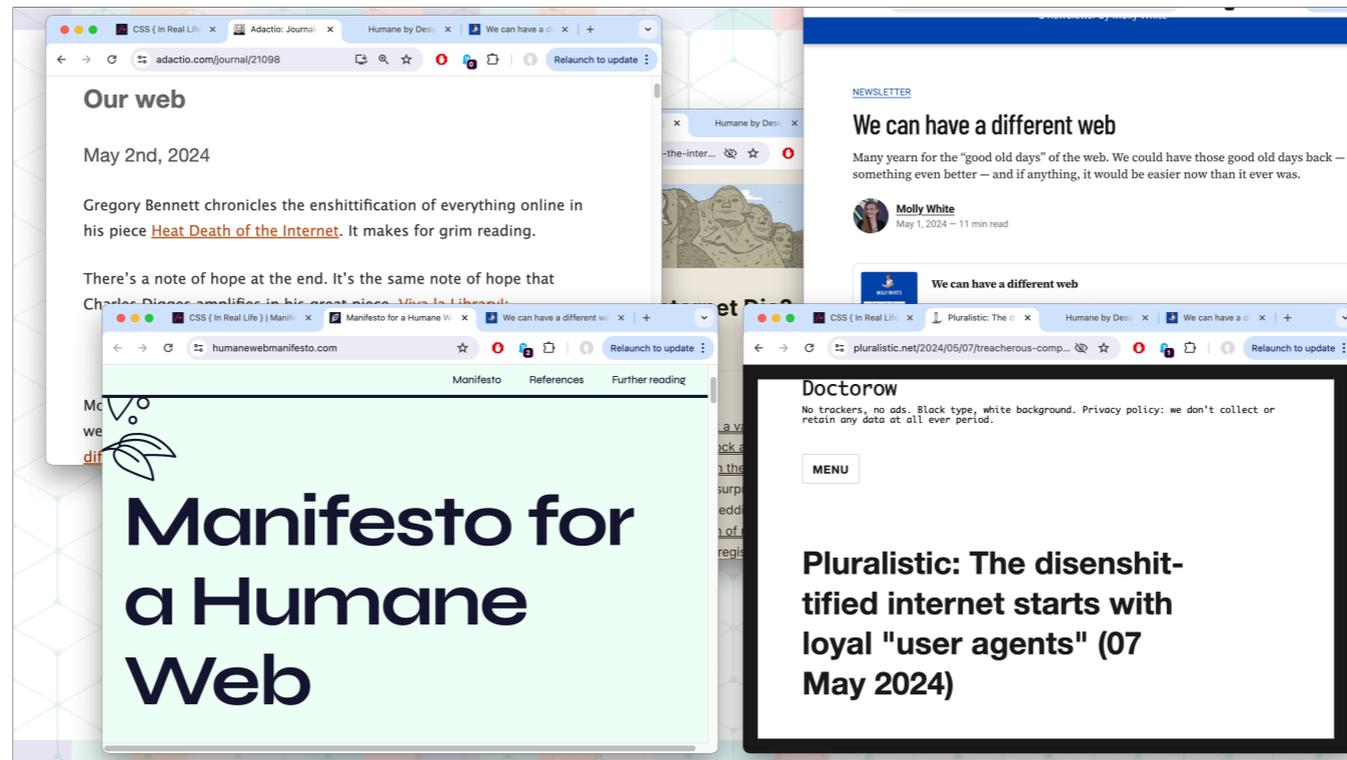
Because if we don't pay for the content we consume



we only see the content that VC capital decides is profitable



which means AI-generated content.



We're seeing the start of a Take Back The Web movement. Great!
But you know what none of these articles are talking about?



<https://www.flickr.com/photos/mpastwa/2181343296/>

- MONEY

All this will fail if no one is able to make some money off of creating good websites.

I should know. I've been there.



Do you, as a web developer, have an obligation to pay?

I'm not sure, but I **am** sure of what happens if you don't pay. And so are you.



- If payments had existed in 2010 I might have continued working on QuirksMode
- I want the next QuirksMode to have that option, that's why I got involved in Web Monetization.
- Please think about who YOU should be paying. Who recently gave you valuable advice that cut your web dev time?



2

HOW?



Paying online has a lot of hassle and friction.

Why?

You are allowing companies to do something with YOUR money



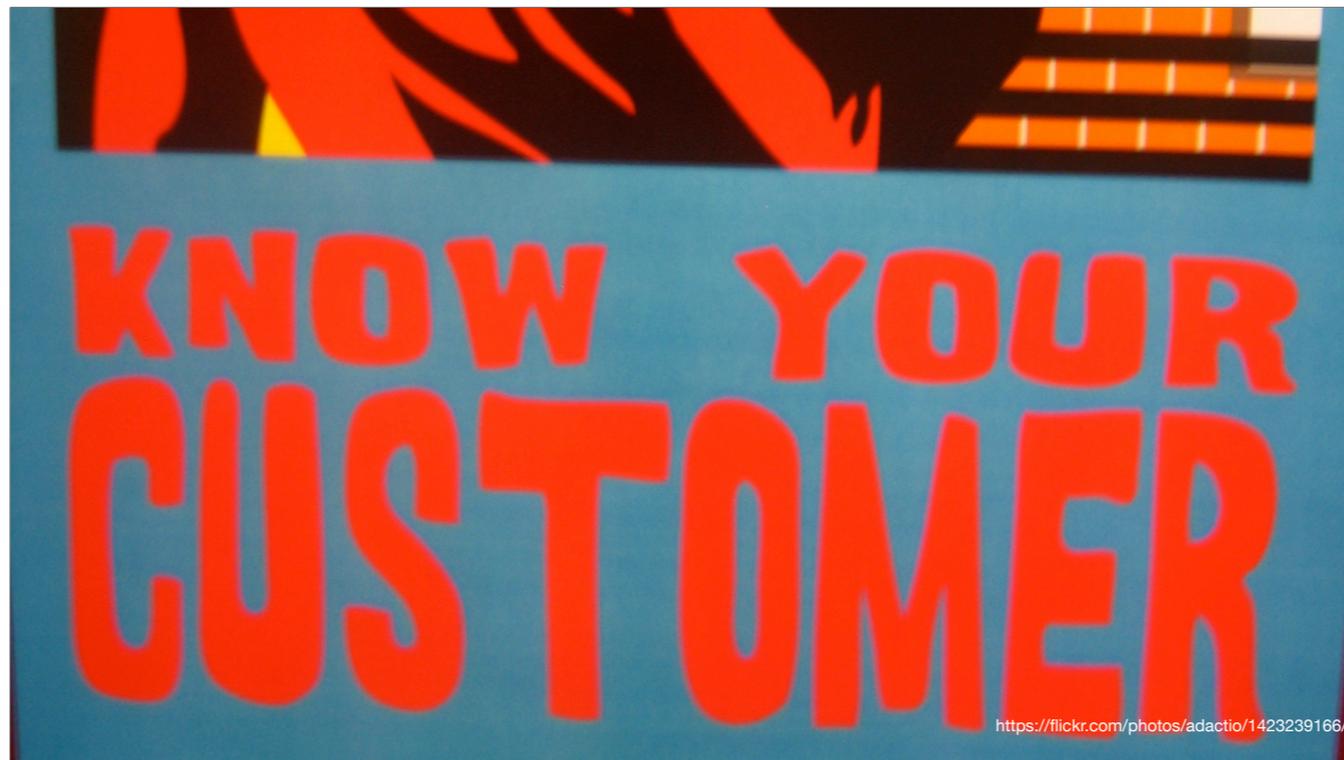
- And that's why ...
- The rules and regulations of your jurisdiction apply. They are meant to protect you and your money, so they are usually fairly thorough and fairly hard to comply with.
- Onus on the company, not on the consumer
- Purpose: They are meant to facilitate ...



- TRUST
- Between you and companies that handle your money,
- but also between all the other companies in the transaction chain.
- In the end, the friction is about building trust.



- The friction you notice is known as Know Your Customer.
- Companies that handle your money, and that promise your money to yet other companies, must know that you're good for your money.
- So some companies must know you. And they'll ask a lot of questions.

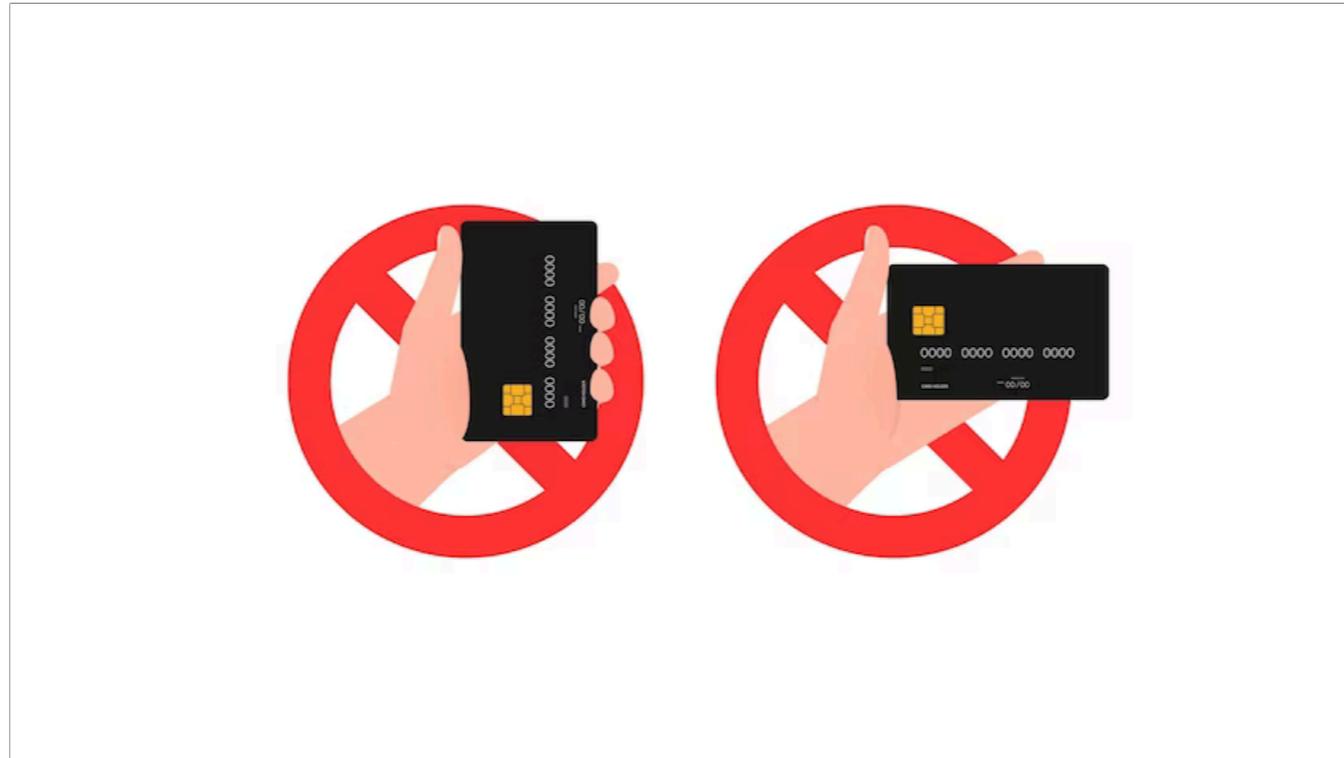


- The friction you notice is known as Know Your Customer.
- Companies that handle your money, and that promise your money to yet other companies, must know that you're good for your money.
- So some companies must know you. And they'll ask a lot of questions.



Sometimes the trust friction and KYC is hidden.

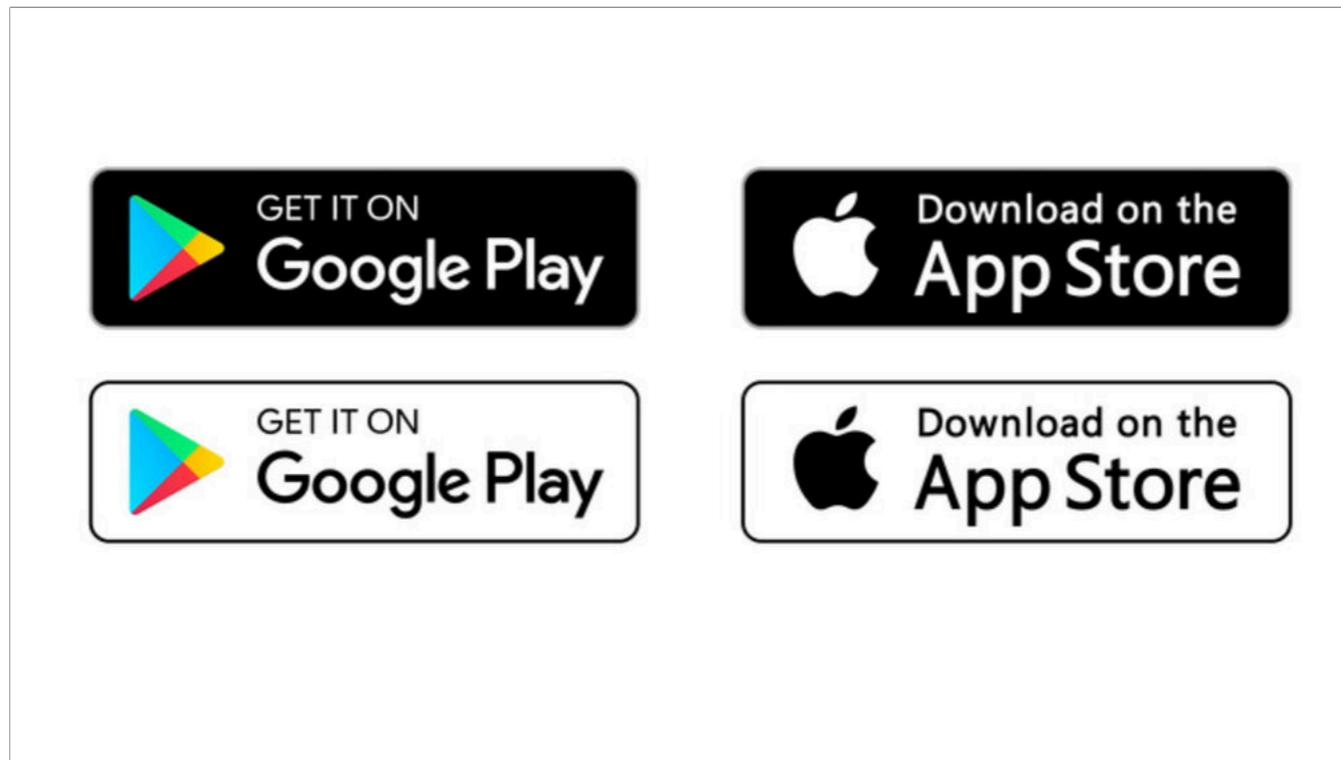
Many online payment methods are based on the credit card - and you already went through the trust friction and KYC when you applied for it.



... unless, of course, you don't make enough money to get a credit card. Or you don't like them. Or it's hard to get one in your country ...



- ... and if you want to get paid you have to jump through even more hoops ...



- ... and the go-betweens take, I don't know, 30% of the money ...



- ... and if you solve all of that, you can only do explicit purchases, where the website quotes a price and you either pay it or don't.
- There's an API for that: the Payment Request API. Website initiates payment



That's where Web Monetization comes in.

And I'm standing here because I believe in it.



What do we want people to do?



<https://www.flickr.com/photos/ralf-steinberger/41940784412/>

What we want is people to say
“Hey, I like your site, here’s some money.”
Automatically.
How do we know people like your site?



Because they spend time there.

The longer the user stays, the more money is paid.

Avoids clickbait farms



- Core concept: We stream money; that is, every few seconds you spend on a website, you pay a little bit of money. You set your own hourly rate. You will also be able to block websites.

Tips come above and beyond the stream

Web Monetization

- A wallet
- An extension ✓ Easy!
- A rate ✓ Easy!

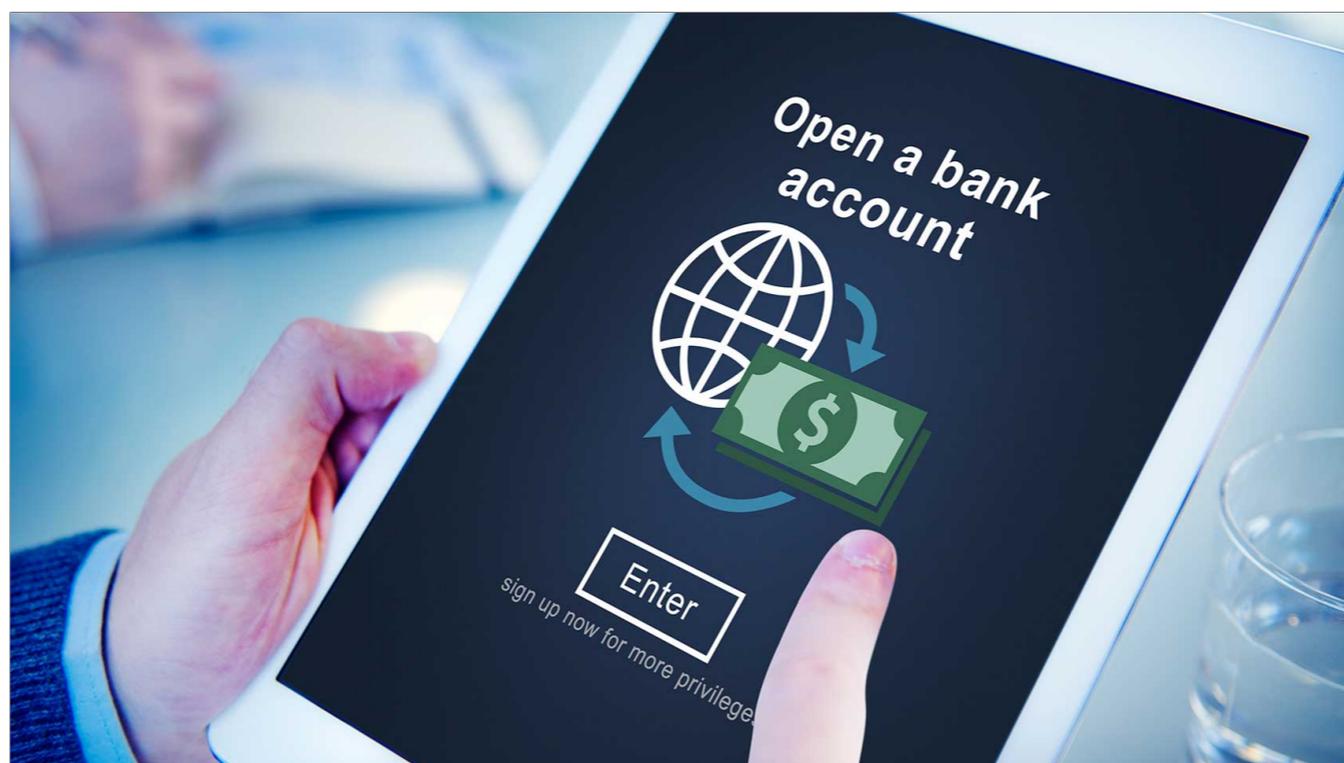
So ... how does it work?

Name and explain requirements

[CLICK] And hey, steps 2 and 3 are really easy!



- First, both you and the website you visit must have a wallet - sort of a bank account. When you visit the site, your wallet pays the owner's wallet.



- Friction alert. We discussed the whys and wherefores, but we want to handle your money, so we should gain your trust and you our trust. KYC, and so on.

Web Monetization

- A wallet
- An extension
- A rate

So that's the wallet. You could get one right now.
Gatehub. Come to booth for more info.

Web Monetization

- A wallet
- An extension ✓ Easy!
- A rate

- Once you have a wallet, you should install the ILF extension for your browser - and that's the part that's not yet possible.
- Your extension allows you to set your hourly rate and give tips. It is the interface.
- Note: the extension gives orders to the wallet, which is allowed to make withdrawals from your bank account.

Web Monetization

→ A wallet

→ An extension

→ A rate

✓ Easy!

- Set a rate per hour.
- As a payer you're done now.

Web Monetization

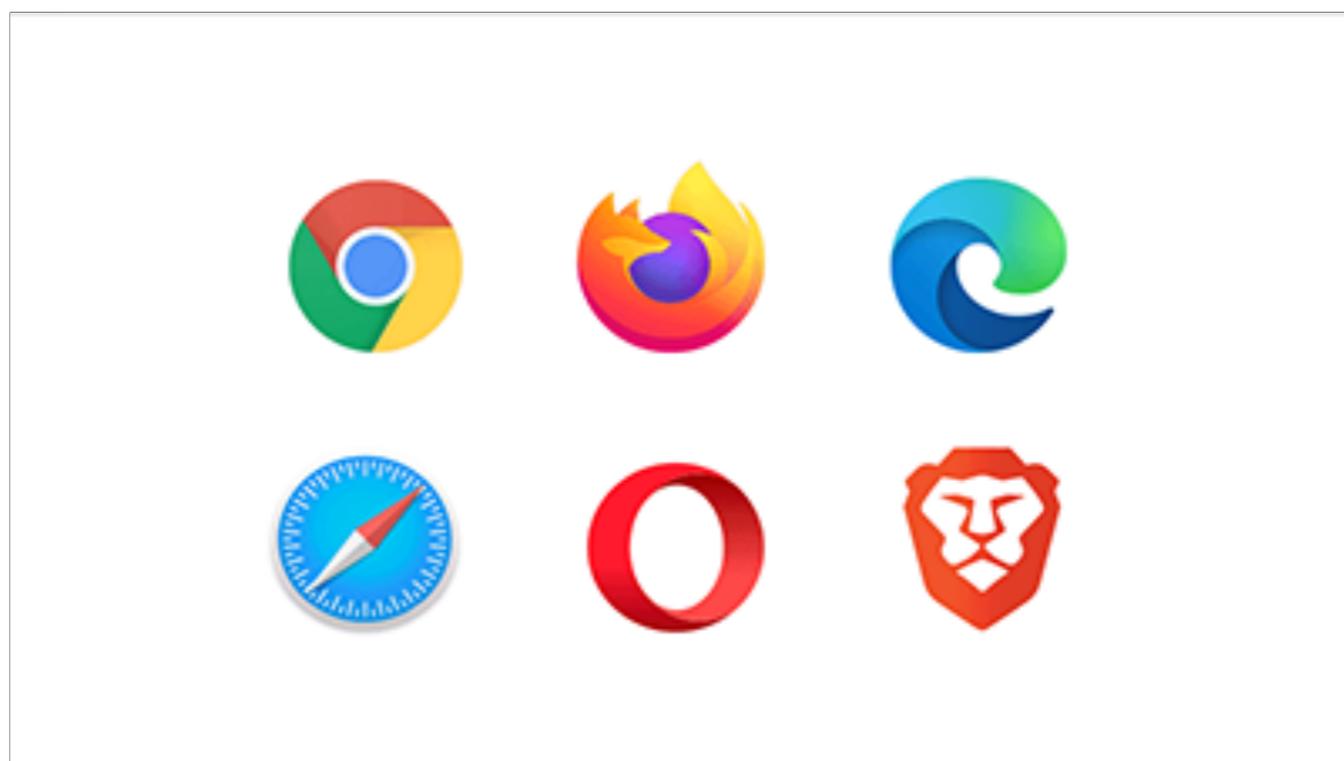
- A wallet
- An extension
- A rate
- A payment pointer in your site

As a site owner you have to put a payment pointer on your site. This is the URL of your wallet.

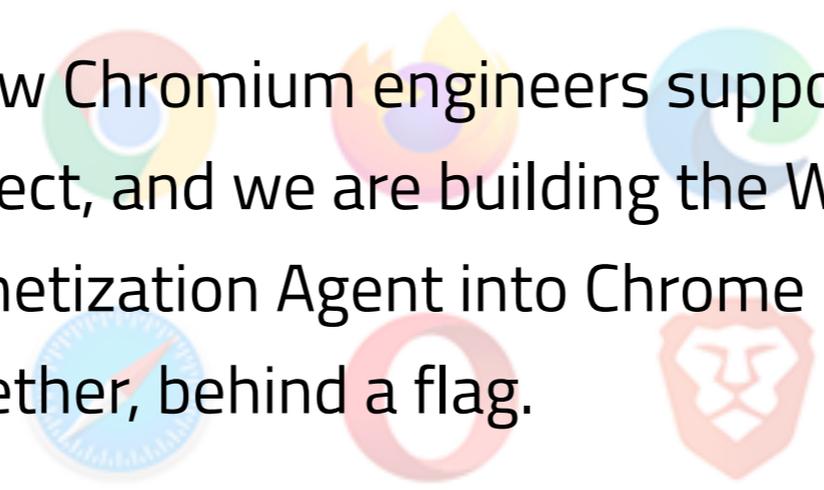
```
<link
  rel="monetization"
  href="https://wallet.example/ppk"
>
```

We call this the payment pointer - though I prefer wallet address.

If you actually follow the link you'll get a JSON file with some payment details. Browsers use that JSON for making payments



Speaking of browsers ...

A collection of five browser logos: Chrome (top left), Firefox (top middle), Edge (top right), Safari (bottom left), and Opera (bottom right).

A few Chromium engineers support the project, and we are building the Web Monetization Agent into Chrome together, behind a flag.

Mozilla, Microsoft and Apple ... unclear. Samsung at least listened to us.

Mention Igalia.

What is a Web Monetization Agent?

1. Finds `<link>`
2. Orders payments
3. Fires monetization events

Part that lives in browser

Automated

```
link.addEventListener(  
  'monetization',  
  function(e) {  
  
  }  
);
```

Also, it fires monetization events whenever a payment comes in.

```
link.addEventListener(  
  'monetization',  
  function(e) {  
    // e.amountSent.value  
    // e.amountSent.currency  
  }  
);
```

You find the amount you were just paid - and remember a new amount will come in every few seconds.

```
link.addEventListener(  
  'monetization',  
  function(e) {  
    if (e.amountSent.value > 0) {  
      // remove ads  
    }  
  }  
);
```

You may change your site if you wish ... but you're not required to.

We also want an HTTP header that notifies servers that you're monetized.



So now visitors can stream money to website owners. And that's great! If it works ...



Disadvantages of streaming

- Only works when you're online
- Will Apple block it?

Streaming vs post-paid

Streaming

- Only online
- ✓ → No need to retain payment data
- ✓ → Site owner is paid immediately
- Many requests

Post-paid

- ✓ → Offline and online
- Payment data retained
- Site owner promised to be paid “later”
- ✓ → Few requests

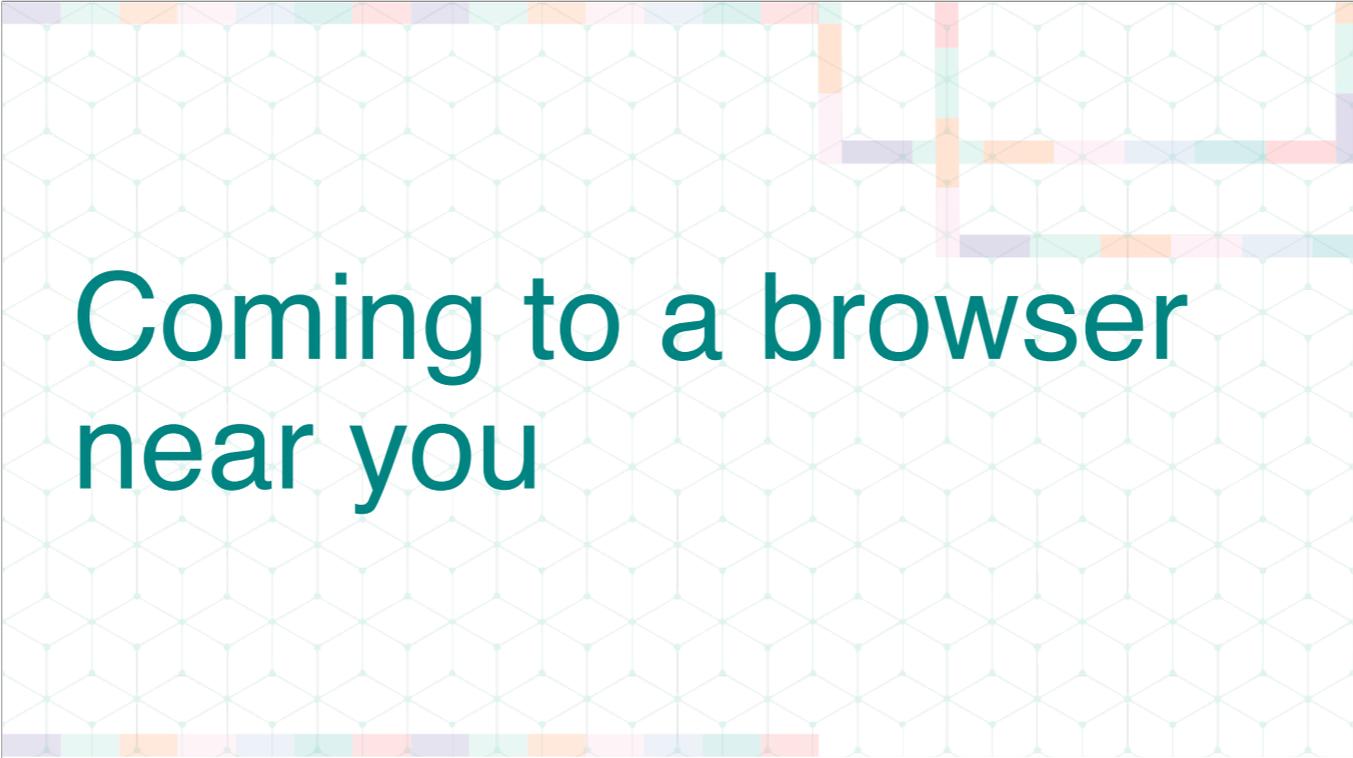
Which model is best? We don't know. So we're letting you choose.

- Streaming: a monetization stream opens immediately.
- Post-paid: the extension keeps track of the monetized sites you visit, and arranges for a single payment moment at the end of the month.

CLICK: Both have certain advantages over the other

You choose.





Coming to a browser
near you



Questions?

webmonetization.org/docs