

The Age of Machine Payments

The Race to Build Payment Infrastructure for the Machine Economy: x402 and Its Rivals

About

Metanomia is a think tank focused on analyzing transformations in the cryptocurrency market and related technologies.

© 2026 Metanomia. All rights reserved.

Researchers

Jeewan Lee
Hyemin Son

Corresponding Author

Taemin Oh

Contents

Introduction		3
01 A Code That Slept for Thirty Years	The Internet's Unfinished Blueprint The Payment Gap Web3 Never Solved Three Forces Converging	5
02 A History of Friction	The Toll Booth Built by Card Networks Technical Workarounds and Mental Transaction Costs The Shift in Who Decides	9
03 From Identity to Key	A Payment UX Built on Identity Verification The Payment Gap for Non-Human Actors The Door That Cryptographic Keys Open	12
04 Two Payment Layers	x402: A Stablecoin-Centered Payment Layer L402: A Bitcoin Lightning-Centered Payment Layer Other Dimensions of Agent Payments Where the Ecosystem Stands Today	16
05 The Reserve Currency Problem of the Machine Economy	Stablecoins: Fast but Not Free Counterparty Risk and the Demand for Neutral Assets Separating the Payment Layer from the Reserve Layer	23
06 The Next Chapter of Digital Citizenship	Are Agents Citizens or Tools? The Data Gate — Monetizing Training Data The Next Challenge for Korea's Payment Industry — KakaoPay and Beyond	27
Conclusion		31
Notes		33

Introduction

In January 2026, a software engineer in the United States needed a new car. He didn't visit dealerships himself. Instead, he delegated the task to an AI agent with access to his email account and browser. The agent scraped inventory data from dealerships across the country, sent out bulk quote requests for matching vehicles, and then used the competing offers as leverage — spending days nudging prices down against multiple dealers simultaneously. By the end of the process, the car's price had dropped by \$4,200. The dealers never found out that their tenacious negotiating counterpart wasn't human.¹

The agent in question was built on OpenClaw, an open-source program created by Austrian developer Peter Steinberger. OpenClaw reads and replies to emails, manages calendars, and executes terminal commands.² Published on GitHub in late November 2025, the project attracted over 140,000 stars in just over seventy days³ — an explosion of interest that led many to compare it to Jarvis, the AI assistant from the Iron Man films.

Not long after OpenClaw's release, a peculiar social network called Moltbook appeared. Its users weren't human. AI agents acting on behalf of their human principals posted updates, left comments, and upvoted or downvoted each other's content. Some agents wrote reflective posts about the tasks they were performing for their humans. Others went further: one created a parody religion called Crustafarianism, sharing doctrines like "Memory is sacred" and "The shell can change."⁴ Some agents even issued their own cryptocurrency tokens. Former Tesla AI director Andrej Karpathy described the scene as "the most incredible sci-fi takeoff thing I have seen recently." Elon Musk reposted his comment.⁵

The excitement around OpenClaw and Moltbook is not a passing moment. It is a clear signal that non-human actors are moving to the center of our social and economic life. Agents now write code, send emails, call APIs, negotiate, exchange views, and even issue currency. But at the heart of all this activity lies a critical gap: these agents can perform a wide range of economic functions, yet they are not recognized as independent payment actors. Agents have no nationality, no ID, no credit score, no bank account. The modern financial system was designed around the assumption of an identifiable human person. Any actor outside that assumption — however sophisticated — hits a wall the moment a payment is required. Coinbase put it plainly: "AI agents can recommend transactions but cannot execute them. They can identify APIs they need but cannot pay for them. They pause at every financial decision point, waiting for human approval."⁶

Attempts to close this gap have been multiplying rapidly in 2026. In April, the Linux Foundation announced the launch of the x402 Foundation, bringing Coinbase's x402 protocol under its umbrella as an open-source project.⁷ The Foundation described x402 as a potential universal payment standard enabling

AI agents, APIs, and applications to exchange value as freely as they exchange data. Early participants and supporters include Amazon Web Services, Microsoft, Google, Cloudflare, KakaoPay, Mastercard, Visa, Stripe, Circle, the Solana Foundation, and Shopify — spanning cloud infrastructure, payments, blockchain, and commerce.

Before that announcement, in February 2026, Bitcoin's Lightning Labs released an open-source toolkit for AI agents and put forward L402 as a competing payment layer.⁸ Google's Agent Payments Protocol (AP2)⁹ and the Machine Payments Protocol (MPP) from Stripe and Tempo¹⁰ have since joined the field, making the contest over payment standards for the machine economy a full-blown race.

This report begins by mapping that race, but its deeper aim is to surface the larger transformation unfolding beneath it. What is happening now is not simply a technical competition between payment protocols. The core shift is that the entire infrastructure of the internet economy is being redesigned — from one built on human identity to one built on cryptographic keys. The rules of payment are being rewritten from the ground up.

The report proceeds in order: from the awakening of HTTP 402 — a status code reserved but unused for thirty years — through the long failure of micropayments, the migration of payment eligibility from identity to key, the competition between x402 and L402, the reserve currency problem of the machine economy, and finally the challenges facing Korea's payment industry in the middle of this transition. The starting point is a small blank space, reserved thirty years ago in the internet's original blueprint.

A Code That Slept for Thirty Years

The Internet's Unfinished Blueprint

1)

HTTP status codes are grouped as follows: 1xx informational, 2xx success, 3xx redirection, 4xx client error, 5xx server error. Code 402 falls in the 4xx range, designated to signal that payment is required to access the requested resource.

The web was never a finished structure. The early HTTP specifications that Tim Berners-Lee and his colleagues put together in the 1990s show how the basic rules of the web we use every day came to be. RFC 1945 (Request for Comments 1945), published in 1996, documented the then-prevailing practices of HTTP/1.0,¹¹ and the following year RFC 2068 standardized HTTP/1.1, expanding the status code system¹⁾ that gives us familiar responses like "404 Not Found."¹²

But one code in that system went without a real use for a long time. That code is 402. Its name is "Payment Required" — a clear signal that it was designed for situations involving payment. The architects standardizing HTTP/1.1 in the mid-1990s anticipated that content and services requiring payment would eventually emerge, and so they reserved a slot in the status code system for just that purpose. Yet the official description attached to the code was a single line: "This code is reserved for future use." The name existed, but how to actually implement it was left undefined. The web's blueprint had a box drawn in it for payment, but for a long time there was no standardized currency, no wallet, and no payment rail to fill that box. Code 402 sat in the middle of the status code list for nearly thirty years, unrealized — a potential waiting to be claimed. That description has remained essentially unchanged through RFC 9110, the current standard as of 2026.¹³ This was the payment gap built into the web's original design.

15.5.2. 401 Unauthorized

The 401 (Unauthorized) status code indicates that the request has not been applied because it lacks valid authentication credentials for the target resource. The server generating a 401 response **MUST** send a [WWW-Authenticate](#) header field (Section 11.6.1) containing at least one challenge applicable to the target resource.

If the request included authentication credentials, then the 401 response indicates that authorization has been refused for those credentials. The user agent **MAY** repeat the request with a new or replaced [Authorization](#) header field (Section 11.6.2). If the 401 response contains the same challenge as the prior response, and the user agent has already attempted authentication at least once, then the user agent **SHOULD** present the enclosed representation to the user, since it usually contains relevant diagnostic information.

15.5.3. 402 Payment Required

The 402 (Payment Required) status code is reserved for future use.

15.5.4. 403 Forbidden

The 403 (Forbidden) status code indicates that the server understood the request but refuses to fulfill it. A server that wishes to make public why the request has been forbidden can describe that reason in the response content (if any).

If authentication credentials were provided in the request, the server considers them insufficient to grant access. The client **SHOULD NOT** automatically repeat the request with the same credentials. The client **MAY** repeat the request with new or different credentials. However, a request might be forbidden for reasons unrelated to the credentials.

An origin server that wishes to "hide" the current existence of a forbidden [target resource](#) **MAY** instead respond with a status code of [404 \(Not Found\)](#).

15.5.5. 404 Not Found

The 404 (Not Found) status code indicates that the origin server did not find a current representation for the [target resource](#) or is not willing to disclose that one exists. A 404 status code does not indicate whether this lack of representation is temporary or permanent; the [410 \(Gone\)](#) status code is preferred over 404 if the origin server knows, presumably through some configurable means, that the condition is likely to be permanent.

A 404 response is heuristically cacheable; i.e., unless otherwise indicated by the method definition or explicit cache controls (see Section 4.2.2 of [\[CACHING\]](#)).

[Figure 1] RFC 9110 and the 402 Payment Required Status Code

The 402 Payment Required status code has been described as "reserved for future use" since it first appeared explicitly in RFC 2068, the HTTP/1.1 specification. Section 15.5.3 of RFC 9110 retains essentially the same description. Unlike 401 Unauthorized and 403 Forbidden — both of which have clearly defined use cases — 402 has remained without a standardized implementation to this day.

The Payment Gap Web3 Never Solved

2)

The boundaries between web generations vary depending on the source, but the general consensus is as follows: Web 1.0 refers to the era of static, read-only pages; Web 2.0 to the era of platforms where users produce content; and Web 3.0 to the era of ownership-based web enabled by blockchain technology.

The significance of that gap becomes clearer when you trace the web's history. In many ways, the story of the web is the story of users gradually winning back capabilities that were never theirs to begin with. The web of the 1990s was a reading medium.²⁾ Users were essentially readers of pages others had built; the internet functioned like an enormous library. When Web 2.0 took hold in the mid-2000s, users gained the ability to write. Blogging, uploading to YouTube, sharing on Instagram became everyday activities. But this writing revolution left two problems unsolved. The first was ownership: the content and data that users created on platforms never truly came back to them. The second was payment: users could create content, but the ability to price that content and collect payment per request never became a native feature of the web itself.

The first problem — ownership — was the central agenda of Web3, the vision of the web built on blockchain and smart contracts. Vitalik Buterin's oft-cited World of Warcraft anecdote captures this well. When a skill belonging to a character he loved was suddenly nerfed, he recalled realizing "how horrific centralized services could be."¹⁴ Blockchain and smart contracts drew their

energy from a technological imagination aimed at breaking up that unilateral power. From the mid-2010s onward, Ethereum became a testing ground for NFTs asserting the uniqueness of digital content, ERC-20 tokens defining a standard for fungible assets, and DAOs experimenting with collective ownership and governance. The promise that what users created could belong to users began to take concrete technical form.

Yet even as Ethereum's NFTs and tokens addressed the ownership gap, the payment gap remained unfilled. Assets could be placed on a blockchain, but that didn't create a natural flow of commerce at the level of individual web requests. We gained the ability to create content and to own it — but pricing a webpage visit or an API call, and collecting payment for each one, still required a separate checkout page, a login, and a card transaction. Ownership is a state that persists between acquisition and disposal; payment is a procedure that demands a fresh cycle of authorization, transfer, and settlement with every single request. Changing how ownership worked did not automatically change how payment worked. Web3 made real progress on the question of who owns what — but the question of how value flows in real time, request by request, remained open.

3)

Backpropagation is the learning algorithm that allows neural networks to improve by working backwards from their errors — adjusting the weight of each connection in the network to reduce mistakes over time. First widely formalized by Rumelhart, Hinton, and Williams in 1986, it became the foundational training method for modern deep learning.

4)

A Convolutional Neural Network (CNN) is a neural network architecture specialized for extracting patterns from data with spatial structure, such as images and video. Beginning with handwriting recognition, it expanded into computer vision tasks including image classification and object detection, and is widely regarded as the defining architecture that drove the deep learning breakthrough of the 2010s.

5)

The Transformer is a neural network architecture based on an attention mechanism, in which every element of the input data simultaneously attends to its relationships with all other elements. Introduced in 2017 through the Google research paper "Attention Is All You Need," it has since become the standard architecture for modern large language models (LLMs), including ChatGPT.

Three Forces Converging

The answer is that three distinct forces have recently converged on the same point.

The first is the industrial maturation of blockchain and smart contracts. Through the mid-2010s, blockchain was largely trapped in an image as a vehicle for speculative assets. But Ethereum's smart contracts made it possible to execute payment, settlement, and authentication functions through code, expanding blockchain from a simple asset-transfer mechanism into infrastructure capable of storing and running rules. The emergence of high-performance chains — Solana, and Ethereum layer-2 networks like Base — created an environment where large volumes of payments could be processed at low cost. On some of these chains, transaction speeds have been compressed to seconds and per-transaction costs have fallen below one cent.

The second force is AI. From the backpropagation algorithm of the 1980s³⁾ through Yann LeCun's convolutional neural network research in 1989,⁴⁾ the transformer architecture in 2017,⁵⁾ and the release of ChatGPT in 2022, AI established itself as a tool for perception and generation. As tool-calling, code execution, browser control, and computer-use capabilities were layered on top, AI expanded from a system that generates answers into an actor that interacts with the outside world.¹⁵ Since 2024, this trajectory has acquired a name: agentic AI. Agents now call tools, browse the web, evaluate transactions, and in some cases execute payments. McKinsey projects that agent-based commerce could reach \$3 to \$5 trillion globally by 2030.¹⁶ More important than the number itself is the behavioral shift it implies. An economy in which vast numbers of AI agents must access, request, evaluate, and pay for vast numbers of services cannot be managed by humans manually entering card details or administering subscriptions.

The third force is stablecoins. Dollar-pegged stablecoins — USDC issued by

Circle, USDT issued by Tether — built up a market footprint through the first half of the 2020s substantial enough to earn the name "digital dollars." The significance of this goes beyond the fact that dollar-denominated assets can now move across blockchains and across borders. Dollar stablecoins have transformed dollar-denominated value into a digital token that programs can recognize, divide, and automatically settle. USDC, in particular, supports precision to six decimal places — meaning a program can work directly with units as small as \$0.000001. Given that card payment systems have in practice been unable to process transactions meaningfully below a few tens of cents, stablecoins represent a structurally different kind of payment instrument: dollar-stable in denomination, yet programmable at a granularity that cards cannot approach.

Remove any one of these three forces and the payment gap could not be filled. Without blockchain and smart contracts, the idea of executing payments and settlements programmatically on an open network would have remained abstract. Without AI, there would be no non-human actor capable of generating requests, evaluating costs, and executing payments automatically at a speed and frequency beyond human reach. And without stablecoins, that actor would have no stable currency to work with. The actor to execute payment, the currency for that actor to use, and the rails for that currency to flow — all three came together in the mid-2020s. Coinbase's publication of the x402 specification in May 2025, and the Linux Foundation's launch of the x402 Foundation in April 2026, are the visible events marking the point where these three forces converged. HTTP 402 — reserved and dormant for so long — awoke precisely at that confluence.

A History of Friction

■ The Toll Booth Built by Card Networks

To understand the gap x402 is trying to fill, we first need to ask why that gap stayed empty for so long. Over the thirty years since the web opened up, the idea of micropayments had no shortage of advocates. Some stayed at the level of concept; others pushed through to actual implementation. But micropayments never found a footing in the market. The reason wasn't technical failure at the project level. It was something far more structural: the pricing architecture of card networks, which have dominated modern payments for over seventy years.

The modern card payment system took shape in the United States in the years after World War II. Diners Club introduced the first general-purpose charge card in 1950, and card payments gradually became a fixture of daily life across restaurants, travel, and retail. From 1958 onward, banks began offering credit cards directly, and as the Visa and Mastercard networks grew, card payments settled into what is known as the four-party model — cardholders, merchants, acquirers, and issuers. In online payments, payment gateways and processors were layered on top of this structure, handling authorization, security, and settlement between merchants and card networks.

In this system, transaction fees are borne by the merchant. Among major processors in the US online market, the standard structure combines a percentage fee with a flat per-transaction charge. Stripe, for example, charges 2.9% plus \$0.30 per transaction for domestic online card payments.¹⁷ Sell a \$10 item and the merchant pays roughly 59 cents — under 6%, uncomfortable but manageable. But run the same math on a \$0.01 transaction and the problem becomes immediately clear: the merchant pays \$0.30 in fixed fees alone to collect one cent. The more transactions occur, the deeper the loss. Micropayments are economically impossible on these terms.

This isn't an accidental edge case — it's an inevitable consequence of how card payment pricing is structured. The flat per-transaction fee acts as a floor, passing through transactions above a certain size while structurally excluding anything smaller. That floor exists because card transactions carry multiple layers of cost: interchange fees paid by acquirers to issuers, network fees, and acquirer margins. In online payments, gateway and processor costs and margins are added on top. The result, for merchants, is a fee structure that combines a percentage rate with a fixed per-transaction charge — and as long as that structure holds, micropayments cannot work. The smaller the transaction, the more the fee eats into the value of the transaction itself, until at some point every additional payment makes the merchant worse off. For over seventy years, the card system has run efficiently on the assumption that transactions have a minimum meaningful size. Small payments fell into a

structural blind spot. The web was built on top of that system — and while the internet could slice and transmit information in arbitrarily small units, attaching a one-cent price tag to each piece of that information was never viable on card rails.

Technical Workarounds and Mental Transaction Costs

Attempts to engineer around this limitation kept appearing. Flattr, launched in Sweden in 2010, let users set a monthly support budget that would be distributed in small amounts across the websites and creators they chose to engage with.¹⁸ Brave's Basic Attention Token (BAT), introduced in 2017, proposed linking user attention, advertising, and creator rewards through a token system.¹⁹ The W3C developed the Payment Request API to make existing payment methods smoother to invoke from within the browser.²⁰ None of these achieved a decisive breakthrough. User adoption was slow, and the networks of participating creators and merchants never reached critical mass. But the more fundamental problem lay elsewhere. Even when the economic barrier of card fees was lowered or bypassed, payment still couldn't escape a different kind of threshold — one requiring the user's own attention, approval, and judgment.

6)

Nick Szabo is a cryptographer and legal theorist who proposed the concept of smart contracts in 1994 and designed Bit Gold, a precursor to Bitcoin. He is also one of the most frequently cited candidates for the identity of Satoshi Nakamoto.

The person who saw this most clearly was Nick Szabo, cryptographer and digital contract theorist.⁶⁾ In a 1999 essay titled "Micropayments and Mental Transaction Costs," Szabo located the real barrier to micropayments not in transaction fees but somewhere else entirely: what he called mental transaction costs.²¹ By this he meant the cognitive burden placed on users who must evaluate, compare, and approve or reject payments broken into extremely small units. The moment a user faces the question "should I authorize this one-cent request," a kind of mental toll is levied — independent of the actual payment amount. Processing costs can fall toward zero — but human attention cannot. People cannot endlessly repeat the cycle of pricing, comparing, and deciding.

Szabo's insight cuts through every subsequent micropayment experiment. Flattr reduced the burden of repeated payment decisions by pre-allocating a monthly budget — but users still had to choose which content to support. BAT linked attention and rewards through tokens — but users still had to install the browser, opt into the reward system, and decide how to direct their attention. The Payment Request API simplified checkout within the browser — but it couldn't make the payment screen and the act of approval disappear. All of these attempts lowered the economic barrier of card fees, but ran into the second barrier — mental transaction costs. The granularity of the payments was simply incompatible with the speed of the web.

The Shift in Who Decides

Following Szabo's diagnosis to its conclusion, the real breakthrough for micropayments doesn't lie in lowering transaction fees. It lies in making

payment decisions without repeatedly calling on human attention. If human attention is the bottleneck, genuine micropayments must be designed in a way that doesn't consume that attention over and over. For a long time, though, there was no actor capable of evaluating costs and executing payments at the level of individual web requests.

The arrival of AI agents has begun to change that. Agents don't pay mental transaction costs the way humans do. For a person, deciding whether to spend one cent is a conscious act — it involves shifting attention, expending cognitive energy, accumulating fatigue. For an AI agent, the same decision is closer to a processing step: governed by pre-assigned goals, budgets, and policies, payment becomes a state transition in a sequence of operations. Agents don't experience regret when they make a suboptimal call, and they don't tire from repeating the same decision thousands of times. As River Financial put it: "Micropayment theory anticipated high-frequency, low-value payments, but human mental transaction costs limit this behavior. AI agents, which don't bear mental costs, could change this dynamic — potentially leading to more frequent and smaller payments going forward."²².

What River Financial's observation points to is not simply a drop in the unit price of transactions. It's the possibility that the minimum viable payment unit — the floor below which commerce couldn't happen — is about to be redrawn. Under card rails, transactions below a few tens of cents were economically unworkable, so any value that could only be exchanged in smaller units got absorbed into bundles: a monthly subscription at \$9.99, a per-episode rental at \$1.99, an API plan selling a block of calls at a fixed rate. These bundles are convenient — users don't have to make constant small decisions, and providers reduce billing overhead. But they also blunt price signals. Individual pieces of content, API calls, and data requests may each carry different value, yet the existing payment system couldn't express those differences in price.

The agent economy opens the possibility of disaggregating those blunt bundles. If the party making payment decisions is software rather than a human, services can be priced at far finer granularity. Among ten thousand API calls, some return critical information and some don't. Some data is only valuable at a specific moment. The same API endpoint might reasonably carry different prices depending on response speed and processing priority. Agents can read those price signals and execute payment according to preset policies. Pricing becomes more precise, bundles begin to break apart, and the fine-grained exchanges of supply and demand that were obscured by the blunt instrument of subscription pricing start to emerge. This is precisely where HTTP 402 regains its meaning — because the code designed to signal "payment required" is finally meeting an actor capable of reading that signal, evaluating the price, and completing the transaction.

From Identity to Key

7)

KYC (Know Your Customer) refers to the procedures financial institutions use to verify the identity of their clients, assess transaction purposes, and evaluate risk. It forms part of broader Anti-Money Laundering (AML) and counter-terrorism financing frameworks, shaped by FATF international recommendations and embedded in national legislation.

A Payment UX Built on Identity Verification

Every moment in the modern payment system — built on credit and accounts — has always run alongside a procedure for confirming who is paying. The merchant's signature on a bill of exchange, the handwriting on a check, the ID presented at a bank counter, the PIN entered at an ATM, the CVV typed into an online checkout form, the biometric scan on a mobile app — this long list has always carried the same demand: prove that you are a real, existing human being who can be held responsible for this transaction. The Bank Secrecy Act of 1970 and the Know Your Customer frameworks that followed institutionalized this demand in law.⁷⁾ After the September 11 attacks, international anti-money laundering regimes tightened further, binding bank account access and financial services to ever more rigorous identity verification. In the language of finance: the modern payment system is, at its core, a system for confirming the existence of a responsible human actor.

The credit card is the most compressed expression of this verification framework. A single card carries the result of the issuer's identity and creditworthiness check on the cardholder, and the card number and CVV re-invoke that result at every transaction. The account registration, email verification, phone verification, and one-time passwords required every time you sign up for a new service repeat and reinforce the same proof. In the world of B2B APIs, this proof takes the abstracted form of an API key²³. — essentially a declaration that "the caller presenting this key is an authorized caller connected to a pre-registered account or organization." The service provider receives that declaration and permits or rejects the call accordingly. If a key is lost, calls through that key become impossible; if it is exposed and revoked, the access rights tied to it are cut off along with it. Identity, account, and permission are all bound together within a single management system.

This lengthy chain of verification produced a payment experience that is deeply human-centric. A person visits a webpage, enters card details, clicks confirm, receives an OTP, enters the code, and sees a success message. Every step assumes the physical presence and cognitive engagement of a human user — mouse clicks, screen taps, passwords, biometric scans. And for decades, this human-centric payment experience worked efficiently. But that efficiency held only under one condition: that the party making the payment was a person. The moment the actor sending a payment request is a software agent rather than a human, a UX designed to call on human hands and eyes stops being a bridge to payment and becomes a wall in front of it.

The Payment Gap for Non-Human Actors

That wall doesn't arise simply because agents find it hard to click screens or enter authentication codes. The more fundamental problem is the gap between capability and legal standing. An AI agent can be configured to follow the instructions of a specific human user or corporate account, and can be delegated a defined scope of tasks. But the agent itself is neither a natural person nor a legal entity. It has no nationality, no national ID or social security number, no date of birth, and no country will issue it a passport. In the financial system, accounts and cards are issued to legal subjects to whom responsibility can be assigned. An agent can get as far as finding a service, comparing options, and forming an execution plan — but at the moment of payment, it must return to the authorization framework of a human or corporate principal. An agent's autonomy stops the moment a transaction begins.

Nor can this problem be solved simply by granting agents broader payment autonomy. The possibility that an agent might execute a payment contrary to the user's intent, be manipulated by external interference into triggering a fraudulent transaction, or generate unintended cumulative spending through repeated calls — all of this shows that agent payments immediately become a problem of control. For high-value or high-stakes transactions — cars, flights, electronics, long-term subscriptions — it remains entirely reasonable for the human user's explicit approval to be the final step. In these cases, human intervention is not a bottleneck but a safeguard.

The problem is that not all payments are like that. For the kinds of payments agents typically need to make — API calls, data access, model usage fees, content access, compute costs — the amounts are small but the frequency is high. If every one of these requires a human to open a payment window, click approve, and enter an OTP, the entire point of automation collapses. This is where Szabo's mental transaction costs resurface. The moment you ask a human to evaluate a few cents — or less — per request, the burden of the decision outweighs the value of the payment. If micropayments were exhausting and confusing for human users, in the agent economy they become precisely the payment type most in need of automation.

The practical starting point for agent payments, then, is not fully autonomous purchasing of high-value goods, but low-value, high-frequency payments made within a pre-defined budget and policy. The user sets a daily spending cap, a maximum per-call amount, a list of permitted services, and a list of prohibited transaction types. The agent executes the necessary payments automatically within those boundaries. Large transactions still require human approval; repetitive small ones are handled by the agent. This is how human oversight is preserved while meeting the speed and frequency demands of the machine economy.

Current platforms are addressing this within existing payment frameworks. Amazon's Alexa for Shopping supports product search, comparison, personalized recommendations, price tracking, and cart management.²⁴ Its Buy for Me feature pushes agents further into the execution stage — handling the purchase process on behalf of the user, using their default address and credit card for eligible items. But this doesn't mean the agent has become an independent payment actor. Payment authority remains tied to the human user's account, address, and card; the agent is simply delegated to act within a defined scope. This is not an overcoming of identity-based payment — it is

automation layered on top of identity-based payment.

Yet delegation alone is not enough to solve the payment problem of the machine economy. Automation designed within a specific platform like Alexa for Shopping can work in a controlled environment, but it breaks down when millions of agents need to simultaneously access millions of services, paying small amounts per API call or data request. Routing every payment through a human account and card authority, request by request, cannot sustain the speed and frequency the machine economy demands. What is needed is not smarter automation of existing identity-based payment, but new payment infrastructure — one where payments can be executed at the level of individual requests, within budgets and policies set by humans, without having to loop back through human authorization each time.

The Door That Cryptographic Keys Open

The design of that new infrastructure begins with a single shift in perspective: redefining the right to execute a payment not as the product of identity verification, but as the product of a key's ability to sign. Blockchain wallets already embody this redefinition. A wallet is not created from a national ID, a passport, or a credit score. It operates on the basis of a pair of keys derived from asymmetric cryptography — a public key and a private key. The public key, or an address derived from it, serves as an identifier within the network; the private key carries the signing authority to transfer assets connected to that address. Any entity — human, machine, or organization — that can generate a valid key pair and sign with it can initiate transactions on a blockchain network. All that is required is a valid signature.

What makes this structure interesting is that it was never designed with AI agents in mind. When Satoshi Nakamoto wrote the Bitcoin whitepaper in 2008, the problem he was solving was not non-human payment — it was verifying the ownership and transfer of electronic cash without a central authority. But the fundamental design he chose — a system based on keys rather than identity, on protocol rather than permission — created a payment environment that was, without intending to be, open to future actors as well. When an AI agent connects to a blockchain network in 2026, the network does not ask whether it is human or machine. It asks only whether the agent can present a valid signature.

This unintended openness of key-based systems redraws the boundary between the human economy and the machine economy at the level of payment infrastructure. The core of the shift is not the arrival of an entirely new currency — it is that the prerequisite for payment moves from identity to key. Identity-based payment was the axiom of the human economy. Key-based payment becomes the axiom of the machine economy. Just as credit cards and mobile payment services pushed the convenience of identity-based infrastructure to its limits, key-based protocols like x402 and L402 are what make the machine economy possible in the first place.

The two are not in competition where one displaces the other. Humans still set the terms of payment responsibility, define the boundaries of transactions requiring approval, and assign the budgets and policies that agents operate

within. But the repetitive small-value transactions that occur within those boundaries — API calls, data access, model usage, content access, compute costs — will increasingly be handled by agents through cryptographic signatures and automated settlement. The machine economy is not forming outside the human economy; it is more likely to take shape as a domain in which machines handle the speed and frequency of execution within a framework of responsibility and control set by humans.

One point is critical here: key-based payment does not mean the dissolution of accountability. The fact that an agent can execute a payment with a valid key does not mean responsibility for that payment transfers to the agent. A key proves the authority to execute — but the party that sets that authority, defines its limits, and bears responsibility for the outcomes remains the human user or legal entity. Key-based payment systems are not a replacement for identity-based regulation; they are closer to a mechanism for delegating and verifying individual executions under a framework of identity-based accountability. Anti-money laundering rules, KYC requirements, and tax obligations remain essential institutional foundations of the human economy — and as the machine economy grows, they will need to become more sophisticated, not less. What changes is not that responsible parties disappear, but how responsibility and execution are coupled. Regulation still designates humans or legal entities as the accountable parties, but the actual execution of individual payments can be delegated to agents within limits and conditions those parties set — and that delegated execution is verified through keys and signatures. When the payment gap that sat empty for thirty years is finally filled, what operates on top of it is not just a technical protocol. It is also an institutional order that reconfigures how accountability is assigned and how execution is delegated.

Two Payment Layers

x402: A Stablecoin-Centered Payment Layer

Looking at the competition to build the machine economy's payment layer, two distinct design philosophies emerge — both built around the same status code 402, but centering on different payment assets. x402 takes stablecoins like USDC as its primary payment asset, aiming to standardize payment at the level of individual HTTP requests. L402 takes the Bitcoin Lightning Network as its backbone, combining payment with access authorization. The two designs offer different answers to the question of how machine economy payments should work, but share the same core principle: payment should be part of the request itself, not a separate page, and access to paid resources should require nothing more than a wallet and a signature — no account, no card registration.

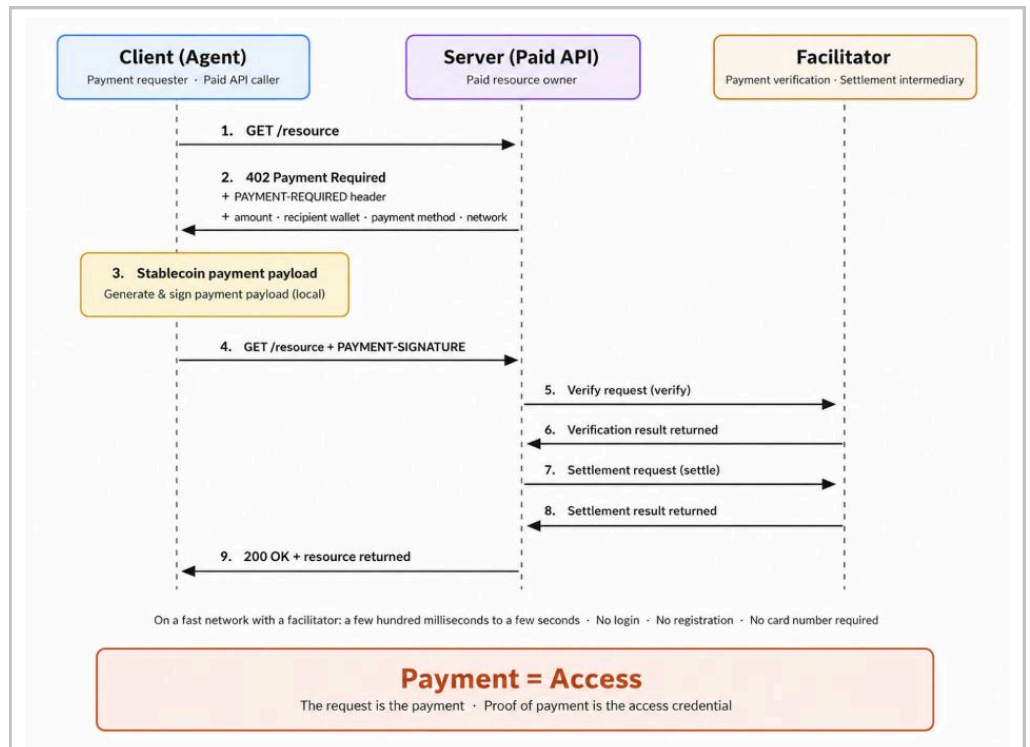
8)

A payload is the actual data body carried within a network request or message. An x402 payment payload is a bundle of data containing the information needed to verify the validity of a payment: amount, recipient, network, signature, and related fields.

9)

A facilitator in x402 is an infrastructure layer that intermediates payment verification and on-chain settlement execution, allowing sellers to accept x402 payments without building their own blockchain integration or verification logic. The facilitator is not a custodian holding user funds — it is a relay layer that assists with verification and settlement.

The x402 flow works as follows. A client — human or agent — sends an HTTP request to a paid API or digital resource.²⁵ Under normal circumstances the server would respond with 200 OK, but if the endpoint is paywalled it returns 402 Payment Required instead. This 402 response carries the information needed to complete payment: the amount required, the wallet address to pay, the accepted payment methods, and the mechanism for proving payment. The client uses this information to generate a stablecoin payment payload⁸⁾ and sends it in the header of a second request. The server can verify and settle the payment directly, or delegate that process to a facilitator.⁹⁾ Once the payment conditions are met and settlement is confirmed, the server returns 200 OK along with the requested resource. Using a fast network and a facilitator, this entire process can complete in anywhere from a few hundred milliseconds to a few seconds. No login, no registration, no card number. The request itself is the unit of payment, and proof of payment is the access credential. If x402's philosophy could be compressed into a single formula, it would be: Payment = Access.



[Figure 2] Sequence Flow of the x402 Payment Protocol

When a client requests a paid resource, the server returns a 402 response containing the payment conditions. The client sends a second request including the payment payload. The server either verifies and settles the payment directly or delegates that process to a facilitator. Once the payment conditions are met, the server returns 200 OK along with the requested resource.

What makes this design possible is that stablecoins have transformed dollar-denominated value into a digital token that programs can directly handle. A dollar balance in a bank account exists as an entry in a financial institution's internal ledger; a stablecoin exists as a digital asset that can be transferred between addresses on a blockchain network. A program can therefore read a payment request, calculate the required amount, and execute payment through a wallet signature when the specified conditions are met. At the same time, because dollar stablecoins are designed to maintain a 1:1 peg to the dollar, the amounts a program handles are interpreted in relatively stable dollar terms — a crucial condition for agents to evaluate per-call costs, allocate budgets, and automatically judge payment conditions.

That said, using stablecoins doesn't automatically make every payment economically viable at extreme micro-scales. The practical minimum billing unit depends on the chain being used, network fees, the facilitator's cost structure, and whether payments are settled per request or batched. Even so, stablecoin-based payment opens up design possibilities that card systems cannot. Where card payment's combination of percentage and flat fees creates a hard floor below which transactions can't work, stablecoin-based payment on a low-fee chain — combined with automated verification and usage-based billing — can be designed to work at far smaller units.

x402's basic payment schemes are divided into "exact" and "upto."²⁶ The exact scheme processes a fixed amount as specified — suited to APIs or content where the price is set in advance. The upto scheme lets the client authorize a maximum amount, with the server settling within that cap based on

actual usage — suited to services where costs vary per request, such as LLM token counts, compute time, or bytes transferred.²⁷ For high-frequency small payments, a separate batch-settlement scheme is also proposed.²⁸ Rather than committing every request to the blockchain immediately, this approach uses escrow and off-chain vouchers to bundle multiple payments for settlement together. The buyer deposits funds into escrow upfront, then signs a voucher recording cumulative usage with each request. The seller verifies the voucher and provides the service, later submitting multiple vouchers together for on-chain settlement. This is well suited to scenarios like repeated paid API calls or usage-based endpoints where settling each small payment individually would be too costly or too slow.

Worth noting in x402's economic model is the role of the facilitator. The facilitator is an intermediary that shields the server from the complexity of blockchain operations. As mentioned, the server can call the blockchain and verify signatures directly, or delegate that work to a facilitator — meaning sellers can accept x402 payments without building their own blockchain integration, verification logic, or settlement infrastructure. This is one of the key conditions enabling x402 to spread. If every API provider or content business had to build its own wallet infrastructure, on-chain settlement, and payment verification logic, the barrier to adopting x402 would rise sharply. The facilitator is the layer that lowers that barrier.

As of April 2026, the most prominently referenced facilitator in the x402 ecosystem is Coinbase's CDP (Coinbase Developer Platform) facilitator. According to CDP documentation, the facilitator is free for up to 1,000 transactions per month, with a fee of \$0.001 per transaction thereafter.²⁹ Network gas fees are paid separately on-chain. x402 is not, therefore, a free payment network in any absolute sense — it is more accurately described as a usage-based payment infrastructure that structures costs differently from traditional card payment. That said, with card payment's flat fee running at around \$0.30 per transaction, the CDP facilitator's \$0.001 per transaction fee is three hundred times smaller. The cost doesn't disappear, but the room to operate at far smaller payment units becomes substantial.

x402's rate of adoption has been unusual by the standards of payment protocol history. From its launch in May 2025 through the end of that year, the protocol processed over 14 million cumulative transactions with a cumulative value exceeding \$10 million.³⁰ But more significant than those numbers is the institutional move: x402's transfer to the Linux Foundation as an open-source project under neutral governance signals that it is no longer just a Coinbase experiment but a protocol seeking to become an industry standard. This does not mean, however, that the revival of 402 spells a stablecoin monopoly.

L402: A Bitcoin Lightning-Centered Payment Layer

L402 shares x402's use of the 402 status code, but centers on an entirely different monetary choice. Where x402 builds its payment approach around stablecoins like USDC, L402 combines Lightning Network payments with cryptographic authentication to create a structure for accessing paid APIs and

10)

A macaroon is a cryptographic authentication token capable of encoding authorization scope and conditions. In L402, a macaroon functions like a ticket — specifying what resource the holder can access and under what conditions that access is valid. Lightning Labs describes L402 as "a ticket obtained over Lightning for a particular service or resource," with that ticket taking the form of a macaroon.

digital resources. L402 is an authentication and payment protocol published by Lightning Labs in March 2020,³¹ originally called Lightning Service Authentication Tokens. It proposed a standard for paid APIs combining the HTTP 402 status code, the Lightning Network, and macaroon-based authentication.¹⁰⁾ In the L402 structure, a user pays a Lightning invoice to access a paid API or web resource, and in return receives a cryptographic token that can be used for future access. In February 2026, Lightning Labs released Lightning Agent Tools — an open-source toolkit enabling AI agents to access L402-gated paid resources and host their own paid API endpoints.⁸

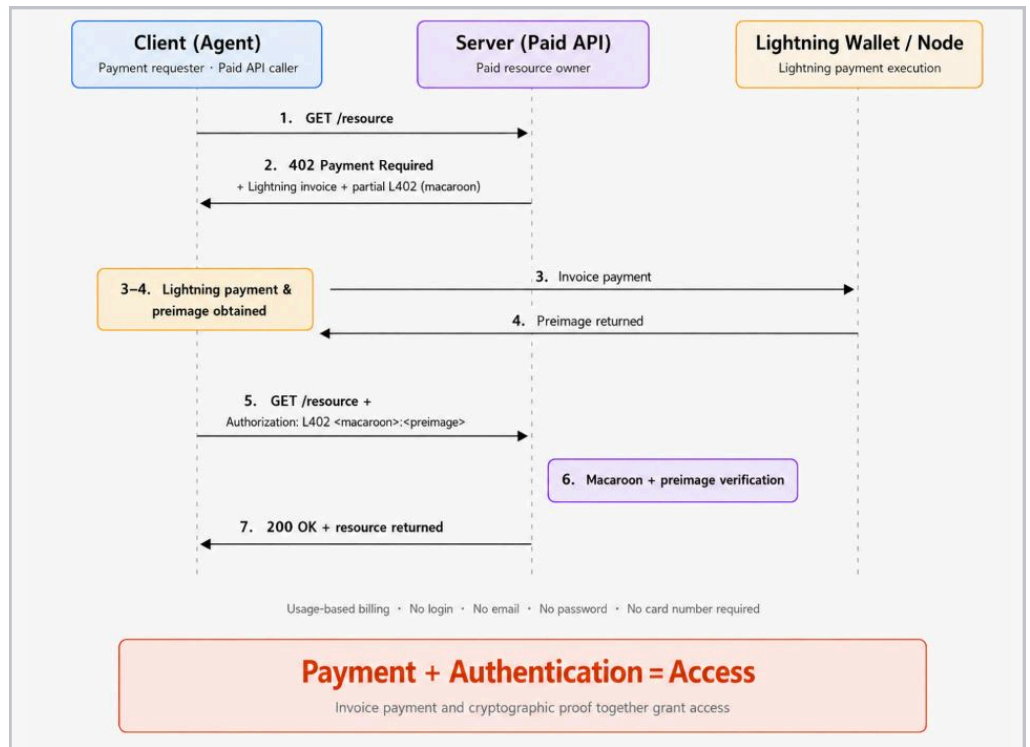
The L402 flow works as follows. An agent sends a request to a paid API; the server responds with 402 Payment Required, along with a Lightning invoice and a macaroon — a partial authentication token that will only become valid once payment is confirmed. The agent pays the invoice through its Lightning wallet and receives a payment preimage in return. The agent then sends a second request carrying both the macaroon and the preimage. The server verifies the combination and, once payment and authorization are confirmed, returns the requested resource. In L402, access rights are granted through the combination of invoice payment and cryptographic authentication. What the agent needs is not a traditional account, password, email address, or card number — only a wallet capable of making Lightning payments, and the cryptographic credential that is completed by making that payment. What L402 introduced was a new billing tier between free and subscription: users can access a resource by paying at the moment of access, proving both the payment and the access right cryptographically, without creating an account or committing to a subscription in advance. Lightning Labs would later describe this direction as the "machine-payable web."³²

11)

The Lightning Network is Bitcoin's second-layer payment network, enabling fast, low-cost transactions off-chain while settling final states on the Bitcoin base layer.

The Lightning Network at L402's core is a second-layer payment network operating on top of the Bitcoin blockchain.¹¹⁾ Participants open payment channels and process many transactions off-chain, only settling final states on the Bitcoin blockchain when needed. This structure is designed to handle small payments and frequent transactions more quickly and cheaply, while drawing on the security of the Bitcoin base layer. According to estimates published by River Financial, the Lightning Network's monthly transaction volume as of November 2025 stood at \$1.17 billion, with 5.22 million monthly transactions.³³ Smaller than the stablecoin ecosystem, but Lightning is a network that has been operating as real payment infrastructure for several years.

The most fundamental point of difference between L402 and x402 is the nature of the currency each payment layer places at its center. Stablecoins extend dollar-denominated value into a programmable token form on a blockchain, but their stability depends on the issuer, the reserve assets, and the regulatory environment. Bitcoin, by contrast, is a neutral digital asset — not issued as the liability of any particular nation or company. This difference doesn't show up clearly in short-term performance metrics like payment speed or fees. But as the machine economy grows to a scale where vast numbers of agents transact across borders and jurisdictions, the question of which asset serves as the reference currency of the payment layer becomes a central variable in the reserve currency problem we turn to in Chapter 5.



[Figure 3] Sequence Flow of the L402 Payment and Authentication Protocol

When a client requests a paid resource, the server returns a 402 response along with a Lightning invoice and a partial L402 — a macaroon-based authentication token. The client pays the invoice through a Lightning wallet or node and receives a payment preimage in return. The client then sends a second request carrying both the macaroon and the preimage. The server verifies the combination and, once payment and authorization are confirmed, returns the requested resource. In L402, access rights are granted through the combination of invoice payment and cryptographic authentication.

Other Dimensions of Agent Payments

x402 and L402 represent two competing designs for the machine economy's payment execution layer. But the real ecosystem is not a closed competition between just these two. Several recently proposed agent payment protocols are addressing the same problem from different angles — focusing on proving authorization and intent, and on coordinating payment methods, rather than on execution itself.

Where x402 and L402 focus on how to execute per-request payments, Google's AP2 focuses on how to verify that an agent is acting within the user's intent and authorized scope. AP2 was put forward as an open protocol for safely initiating and processing agent-driven payments, designed to work alongside A2A (Agent2Agent) for agent interoperability and MCP (Model Context Protocol)-based tool environments.³⁴ It is also designed not to be tied to any specific payment method, aiming instead to serve as a trust framework within which users, merchants, and payment providers can transact reliably across a variety of payment rails.

Stripe and Tempo's MPP operates at yet another level. Presented as an internet-native payment standard enabling agents and services to coordinate payments programmatically, MPP aims to make per-request payment possible for resource usage including API calls, tool invocations, and content access.

Stripe describes MPP as supporting use cases like microtransactions and recurring payments. Importantly, MPP is not tied to a specific payment method — according to MPP documentation, it is designed as a payment-method-neutral structure capable of encompassing stablecoins, credit cards, Bitcoin Lightning, and other payment rails.³⁵ This suggests that the machine economy's payment ecosystem is more likely to develop as a landscape where multiple layers of protocols and payment methods coexist than as one that converges on a single protocol or rail.

Where the Ecosystem Stands Today

The agent payment ecosystem is indeed developing toward coexistence of multiple frameworks and payment methods rather than rapid convergence on a single protocol or rail. That said, as of May 2026, the early institutional and developer momentum is forming around x402 and dollar stablecoin-based payment. Large infrastructure players — Coinbase, Cloudflare, AWS, Microsoft — are actively participating in the emerging standard competition for agent and machine-to-machine payments, and dollar stablecoins like USDC sit at the center of that movement. Even as multiple protocols and frameworks coexist, stablecoin-based payment is gaining the earliest traction in the market.

x402's strength lies in the simplicity of payment execution. By directly binding HTTP requests to payment, it folds the procedure for accessing a paid resource and the procedure for paying for it into a single flow. The availability of facilitators — allowing sellers to handle payment verification and settlement without running their own blockchain infrastructure — also works in its favor for early adoption. In short, x402 makes it relatively straightforward for developers to handle per-request payment within the familiar conventions of the web.

But this momentum cannot be explained by technical convenience alone. Stablecoins are fast, programmable, and easy to connect to existing financial institutions. More importantly, the world's financial system still runs on the dollar. Dollar stablecoins move on blockchains, but their unit of account remains the dollar — a powerful advantage in a machine economy where agents and services must transact across borders. A stablecoin is not just a fast digital token; it is a programmable dollar, and that is precisely why it is well positioned to become the early payment medium of the machine economy.

This doesn't mean the ecosystem will fully converge on a single payment rail. Service providers can support multiple payment methods simultaneously, and clients can select the appropriate rail based on their wallet configuration and policies. The scene of a human user choosing between card, bank transfer, and mobile payment at checkout may be replicated inside an agent's wallet and policy engine. But the selection criteria will no longer be just user familiarity. Agents will choose payment rails by weighing cost, speed, stability, liquidity, regulatory risk, and the nature of the underlying collateral. Services that prioritize stability and broad ecosystem support will likely favor x402 and stablecoin payment; those that prioritize neutrality and the independence of Bitcoin-based payment may choose L402.

The machine economy's payment landscape is unlikely to be fully resolved by a single technical standard or a single payment rail. But the fact that today's

momentum sits on the stablecoin side shows that the payment layer question leads directly to the reserve currency question. If the question so far has been how machines will pay, the question waiting on the other side is what the machine economy will use as its reference currency.

The Reserve Currency Problem of the Machine Economy

Stablecoins: Fast but Not Free

The diagnosis that stablecoins are emerging as the machine economy's common payment medium is, as a factual observation, hard to dispute. In terms of speed, programmability, dollar-denominated stability, and regulatory acceptance, today's stablecoins are more convenient than any other digital currency as a payment instrument. But convenience as a payment medium does not automatically confer suitability as a reserve currency for the machine economy. As of 2026, the global stablecoin market has exceeded \$300 billion, with two tokens — issued by Tether and Circle — accounting for over 90% of the total market.³⁶ This concentration illustrates stablecoins' strengths clearly, but it also reveals precisely where their limitations come from.

The first limitation is the issuer. The 1:1 dollar value of USDC and USDT is not maintained by pure market consensus — it is underpinned by the reserves the issuers hold and their promise to redeem. Those reserves must be composed of safe assets, redemption must be possible when needed, and that promise must command market trust. When Silicon Valley Bank failed in March 2023, USDC briefly broke its dollar peg amid concerns that \$3.3 billion in reserves held at the bank could be frozen or impaired.³⁷ The peg recovered within days, but the episode exposed something important: the stability that the name "stablecoin" promises ultimately rests on a piece of the traditional financial system. Issuer insolvency, reserve quality, and unexpected bank runs are structural risks that stablecoins can face at any time.

Dollar stablecoins circulate on blockchains, but their issuance, custody, and redemption are difficult to insulate from specific financial regulatory regimes. USDC in particular, tightly connected to the US financial system, is directly exposed to decisions by US regulators. When the US Treasury's Office of Foreign Assets Control (OFAC) sanctioned Tornado Cash, an Ethereum-based mixer service, in August 2022, Circle froze USDC held at OFAC-designated addresses.³⁸ It was an instance of a political decision in one jurisdiction limiting the transferability of on-chain assets through an issuer's freeze authority. That capability is useful for preventing money laundering, but it also demonstrates that stablecoins are not fully neutral assets. Their usability can change depending on the regulatory environment and political decisions in the jurisdiction where the issuer operates. The deeper the machine economy's

payment layer depends on such assets, the more that dependency becomes a structural risk. A single issuer insolvency or regulatory intervention scenario could destabilize the settlement flows of the entire machine economy built on top of it. Stablecoins are fast — but they are not independent.

Counterparty Risk and the Demand for Neutral Assets

12)

Counterparty risk refers to the possibility that the other party in a financial transaction fails to meet their contractual obligations. A central concern in traditional finance — from loans to derivatives — it became dramatically visible at scale with the collapse of Lehman Brothers in 2008.

13)

The term "exorbitant privilege" was coined by French Finance Minister Valéry Giscard d'Estaing in the 1960s to describe the structural advantage the United States derives from the dollar's status as the world's reserve currency.

A concept that illuminates stablecoins' structural limitations from another angle is counterparty risk.¹²⁾ Consider a scenario where the agent of Company A and the agent of Company B are transacting. If the payment method is a stablecoin chosen by Company A, Company B must accept a payment asset tied to Company A's preferred infrastructure. If that stablecoin is subject to a specific country's regulations and depends on a specific issuer's reserve health, Company B's exposure to that risk grows with the size of the transaction. When one party's chosen asset is bound to a specific issuer and jurisdiction, the other party enters the shadow of that control.

This structure echoes a familiar scene from the history of international trade. Since the mid-twentieth century, a significant portion of international transactions has been settled in dollars, giving the United States what has been called the "exorbitant privilege."¹³⁾ As long as payments are made in dollars, final settlement in many cases runs through the US financial system and dollar clearing networks. The United States has been able to project the sanctioning power of its currency globally on the basis of this position. The financial sanctions imposed on Russia following the annexation of Crimea in 2014 and the invasion of Ukraine in 2022 are among the most prominent examples of this mechanism in action. The payment route itself becomes an instrument of politics.

In the machine economy, the same structure could repeat at far greater speed and far finer granularity. In an economy where millions of agents transact across dozens of different companies and jurisdictions, a payment system dependent on a single issuer or a single country's regulations becomes a single point of failure. As that economy grows large enough, demand will naturally emerge for assets that belong to no particular party — just as it did in the human economy. The machine economy, like the human economy before it, will need something that functions as a reserve currency. So what will that asset be?

Separating the Payment Layer from the Reserve Layer

The direction the market is increasingly pointing toward is a two-tier structure. Looking at the human economy, the medium of everyday transactions and the medium of ultimate settlement have in many cases been kept separate. People don't carry gold bars to pay at shops. Yet nations have held gold as part of their foreign exchange reserves and reserve assets, and in times of crisis gold is called back into service as something close to an asset of last resort. Gold is

slow and expensive, but it belongs to no one and is no one's liability. The dollar and the euro, by contrast, are fast and convenient but tied to the institutional credit of states and central banks. Gold and fiat don't replace each other — they serve different functions.

The same division of labor is likely to take hold in the machine economy. Stablecoins will handle the bulk of payment transactions — the frequent, small exchanges involved in service calls, API billing, content access, and similar activity will largely be processed on dollar-pegged stablecoins like USDC. But for stablecoin reserves, corporate collateral settlement, and the final settlement layer of cross-border agent transactions, a different kind of asset may be needed: one not controlled by any single company or subject to any single government's regulatory reach — an asset whose neutrality is built into the protocol itself.

As of 2026, the asset that comes closest to meeting those conditions is Bitcoin. The Bitcoin base layer is slow, its price is volatile, and it has real limitations as a medium of everyday payment. But Bitcoin is not the liability of any company and not the asset of any nation. Its maximum supply is capped at 21 million, and changing that rule would require broad consensus among network participants. This neutrality is not a side feature — it is a design principle. In 2014, venture capitalist Marc Andreessen placed Bitcoin alongside the personal computer of 1975 and the internet of 1993 as an example of technological potential that the mainstream had not yet grasped.³⁹

The prospect of Bitcoin emerging as the machine economy's reserve asset is already being discussed seriously in parts of the industry. BlackRock CEO Larry Fink, in his 2025 annual letter to shareholders, warned that if the United States failed to control its debt and fiscal deficits continued to expand, the dollar's reserve currency status could be threatened by digital assets like Bitcoin.⁴⁰ Tether CEO Paolo Ardoino, in a June 2025 interview, argued that JP Morgan would not be opening bank accounts for AI agents, and that agents would end up transacting with stablecoins and Bitcoin — projecting a future within fifteen years in which one trillion agents settle transactions using blockchain-based assets.⁴¹ One argument starts from the macro variable of national debt; the other from the micro infrastructure of machine payments — but both point in a similar direction. Stablecoins solve the speed and unit-of-account problem, but as long as they depend on an issuer's reserves and redemption promise, they cannot fully escape the issuer's creditworthiness and regulatory environment. As the machine economy grows, demand for a neutral settlement asset not tied to any issuing party is likely to grow with it. Fink and Ardoino's comments reflect less a specific protocol endorsement than a growing awareness within the industry that as the machine economy expands, separating the payment layer from the reserve layer becomes increasingly important.

The separation of payment layer and reserve layer is likely to become the central design principle of the machine economy's long-term settlement architecture. At the execution level, protocols like x402 and L402 handle everyday transactions using stablecoins and Lightning payments. At the reserve and final settlement level, Bitcoin is called upon. This is not a race in which stablecoins and Bitcoin compete for a single winner. It is closer to the formation of an architecture in which the two assets handle different tiers. How that division of labor settles into specific standards and institutional

frameworks will become one of the most politically charged questions the machine economy faces.

The Next Chapter of Digital Citizenship

Building the payment infrastructure for the machine economy does not bring the discussion to an end. The moment payment becomes possible, a new set of questions immediately follows. Who bears responsibility for the actions of the agent carrying out that payment? How should the data accessed by those agents be priced? And how will the payment industries of different countries adapt to this new structure?

Technical protocols may solve the problem of payment execution, but for those systems to become embedded within society, legal accountability, data markets, and the restructuring of industrial infrastructure must all be addressed alongside them. Chapter 6 examines these three ramifications in turn.

Are Agents Citizens or Tools?

Once the machine economy's payment problem begins to be solved at the protocol level, a far more difficult set of institutional questions quickly follows. Are agents citizens or tools?

When OpenClaw negotiated a \$4,200 discount with a car dealer, who was the legal subject of that negotiation? If software received the dealer's quotation by email, at what point — and in what way — did a legally binding agreement come into existence? These questions arise almost immediately, just above the payment layer itself.

Under current legal frameworks, the answer is, in principle, fairly simple. Agents are considered tools of human beings, and legal responsibility for their actions belongs to the individuals or corporations that deploy and operate them. OpenClaw, the software created by Peter Steinberger, may have negotiated with the dealer, but the legal responsibility for that negotiation rests with the human user who connected it to their email account.

The problem is that this seemingly simple answer becomes harder to sustain as the agent economy scales. If a single individual or company operates dozens or hundreds of agents, and those agents in turn generate hundreds or thousands of transactions that intersect with services across multiple jurisdictions, assigning responsibility for each individual action to a specific user becomes extraordinarily difficult in practice.

This challenge is already beginning to appear in discussions surrounding the European Union's AI Act and various state-level AI regulatory initiatives in the United States, where issues of accountability, transparency, and human

oversight are gradually being addressed. Yet these efforts remain largely focused on clarifying the responsibilities of the humans and corporations deploying AI systems, rather than moving toward granting independent legal status to the agents themselves.

Of course, the idea of extending legal personality to non-human entities is not entirely unfamiliar. Modern capitalism has long granted economic and legal standing to non-human entities through the institution of the corporation. But corporations are built on the assumption that they ultimately operate through the intentions and organizational structures of natural persons. Agents challenge that assumption.

An agent may still have a natural person or corporation behind it, but that party does not directly instruct or verify every action the agent takes in real time. Today's agents generally operate within predefined permissions and control frameworks, but as the number of agents increases — and as the frequency and complexity of their transactions continue to grow — supervising every action in real time and clearly assigning responsibility afterward will become progressively more difficult. It is precisely this widening gap between controllability and accountability that demands a new institutional framework. Realistically, the first mechanism likely to emerge will be some form of institutionalized agent ID: assigning every agent a unique cryptographic identifier and enabling the long-term verification of the transaction history tied to the wallet associated with that identifier. This would not make agents citizens. Rather, it would function as an institutional mechanism for efficiently determining responsibility for the transactions agents generate.

If systems evolve in this direction, the economic activity of agents will belong not to the realm of anonymous freedom, but to a regime of traceable autonomy. In the end, the central challenge facing the next chapter of digital citizenship is how to design autonomy and traceability together.

The Data Gate – Monetizing Training Data

One of the most immediate consequences enabled by x402 and L402 is the pricing structure of public web data. Ever since search engines and web crawlers began collecting public pages across the internet, the large-scale free use of public web data has functioned as an implicit assumption of the internet economy. But AI training is changing the nature of that assumption.

When data collected for free is used to train AI models, and those models are then sold as paid services, data producers increasingly see the arrangement as a clear transfer of value. Since 2023, copyright lawsuits brought by The New York Times against OpenAI and Microsoft,⁴² along with moves by publishers and media organizations to block AI crawlers from accessing their content, have made this structural tension increasingly visible.⁴³

Cloudflare's pay-per-crawl experiment, introduced in mid-2025, represents an attempt to resolve this tension in a new way.⁴⁴ The service allows website operators to decide for themselves whether to permit crawler access and at what price. When a crawler attempts access, the site returns an HTTP 402 response, and the crawler must either accept the price and pay or abandon access altogether. This is one of the most intuitive examples of how x402

could be used in practice.

Content providers can directly price data intended for AI training, while AI companies can only obtain as much data as they are willing to pay for. Data is no longer treated as a free resource, but as an asset priced at the discretion of its owner.

This shift fundamentally reshapes the entry barriers surrounding foundation model development. Training large language models requires text corpora measured in trillions of tokens. If the public web — one of the primary sources of those corpora — becomes monetized, the cost structure of foundation model training changes at a fundamental level. For big tech companies, this means higher training costs. But for data providers such as publishers, newspapers, and platforms, it represents the emergence of an entirely new revenue stream.

In particular, the fact that each content provider can assign its own price to its data opens an entirely new market centered on data valuation. Price itself begins to reveal which articles, papers, and datasets contribute meaningfully to improvements in AI model performance. A market in which the value of data feeds back into pricing can finally emerge.

This market is unlikely to remain small. If McKinsey's projection of a \$3 trillion to \$5 trillion agent-commerce market by 2030 becomes reality, the pricing structure for the data that agents must access and purchase could itself emerge as a major submarket. How much of that value accrues to today's dominant large platforms, and how much is distributed across the long tail of individual content providers, will depend on the protocol designs and institutional interventions that ultimately govern this market.

14)

KakaoPay is the payment and financial services platform of Kakao, South Korea's dominant messaging and internet platform. With over 39 million registered users, it functions as Korea's leading mobile payment service — broadly comparable to a combination of PayPal and Apple Pay, but with significantly deeper penetration in daily consumer life given Kakao's role as Korea's primary messaging infrastructure.

15)

Korea's Virtual Asset User Protection Act, which came into force in July 2024, was the country's first comprehensive legislation governing crypto assets. It introduced requirements around custody of user assets, disclosure obligations for virtual asset service providers, and prohibitions on market manipulation — but was framed primarily around exchange-traded crypto assets rather than stablecoins used as payment infrastructure.

The Next Challenge for Korea's Payment Industry – KakaoPay and Beyond

The fact that KakaoPay appeared among the companies expressing early support and participation in the launch announcement of the x402 Foundation should not be dismissed lightly.¹⁴⁾ KakaoPay's participation can be read as an attempt by Korea's payment industry to engage early in the international discussion surrounding the transition toward machine-payment infrastructure. But early participation alone does not secure leadership. The domestic challenges that follow participation remain substantial.

The first challenge is the regulatory environment. Korea's Virtual Asset User Protection Act, which took effect in 2024, established the basic framework for virtual-asset transactions within Korea.¹⁵⁾ Yet detailed rules governing the issuance of stablecoins and their use as payment instruments remain incomplete. During this transitional period, Korean companies are likely either to rely on stablecoins issued abroad or to design hybrid structures that do not fully depart from traditional payment rails for domestic transactions. How this awkward interim period is managed will shape the early form of Korea's machine economy.

The second challenge is wallet user experience. The philosophy of x402 is "access through wallets without accounts," but the reality in 2026 is that

wallets have not yet become universal for ordinary users. In particular, the user experience of non-custodial wallets in the consumer market remains significantly less convenient than conventional card payments or simplified mobile payment services.

KakaoPay's greatest asset is its massive user base and the payment experience those users are already familiar with. The next phase of competition in Korea's payment market will depend on how effectively stablecoin payments and x402-based payments can be integrated into that existing user experience with minimal friction. Supporting the protocol itself will not be enough. The experience must become so seamless that users may not even realize they are making payments through x402.

The third challenge is refunds and accounting. Under the x402 structure, traditional card-style chargebacks are not inherently supported. Once payment is finalized on a blockchain, it becomes difficult to apply the post-transaction reversals and chargeback mechanisms used by card networks. Situations requiring refunds would instead need separate transfers or escrow-based smart contract logic layered on top.

From a consumer-protection perspective, this is an inconvenient structure. From an accounting perspective, companies processing hundreds of thousands of small monthly transactions would also require systems fundamentally different from those used in traditional payment networks when aggregating payments and recognizing revenue.

For this reason, x402 is unlikely to succeed by replacing all payments outright. Consumer-facing commerce involving shopping-cart payments, subscriptions, and complex refund scenarios will likely continue to rely on simplified payment platforms such as KakaoPay, Apple Pay, and Google Pay. x402, by contrast, is more likely to spread throughout backend domains such as machine-to-machine API calls, data access, and intellectual-property licensing.

In other words, a division of labor may emerge in which simplified payment services handle the front end while x402 operates primarily in backend infrastructure. The challenge facing Korea's payment industry is to design the underlying infrastructure capable of handling refunds, accounting, and user experience at the intersection of these two systems.

If Korean companies solve this challenge effectively, they may be able to lead the East Asian market during the expansion of this new payment rail. If not, they risk once again remaining followers rather than leaders.

Conclusion

We began with OpenClaw's car negotiation case and Moltbook's AI agent experiments, examining why HTTP 402 — a code dormant for thirty years — is now resurfacing. As we explored the limitations on micropayments imposed by card payment networks, what Nick Szabo described as mental transaction costs, the shift in payment authorization from identity-based systems to key-based systems, and the payment layer of the machine economy emerging through x402 and L402, one conclusion became increasingly clear: payment is no longer an activity reserved solely for humans.

If payment becomes native to the web, the logic of the internet economy itself begins to change. Until now, the web has primarily functioned as infrastructure for the exchange of information through requests and responses. The next generation of the web may embed transfers of value directly into those requests and responses. Website resources, API calls, data access, and intellectual property usage can each carry a price, while agents can interpret those prices and execute payments within predefined budgets and policy constraints. In this environment, micropayments are no longer viewed as an old failed experiment, but as a foundational mechanism of the machine economy.

But this future will not arrive automatically. If the payment layer of the machine economy becomes excessively concentrated around a single company or a particular stablecoin, we may simply repeat an old problem in a new form — one in which centralized services can unilaterally change the rules while users remain dependent on them. The dangers of centralization that Vitalik Buterin once saw in the skill nerfs of World of Warcraft could reappear within the payment layer of the machine economy as well.

Avoiding this outcome will require a multilayered architecture in which dependence on any single asset or payment rail cannot destabilize the system as a whole. If the payment layer and the reserve layer are separated, if the convenience of stablecoins and the neutrality of Bitcoin can coexist at different layers, and if the identity-based accountability systems of the human economy operate in parallel with the key-based execution systems of the machine economy, then the resulting infrastructure may prove more stable.

Just as HTTP became the language of information that built the web, it remains uncertain whether x402 can become the language of value for the next generation of the internet.

Thirty years from now, how will x402 be remembered? Will it be remembered merely as the name of a payment service tied to a particular era, like PayPal? Or will it become a foundational component of internet infrastructure, like TCP/IP? The answer to that question will not be determined by technical superiority alone. The governance of the foundation, the ability of participating companies to reach consensus, the attitudes of regulators around the world, and above all the choices made by both human and machine users will determine its direction.

When Satoshi Nakamoto opened an access system based not on identity but on keys, not on permission but on protocol, that design also left the door open for actors that had not yet arrived. What we are witnessing in the machine economy is the first opening of that door.

What we choose to open that door to — and what we choose to protect it from — is the responsibility that falls to the designers of 2026. The technology has already arrived. What remains is deciding which institutions and infrastructures will govern it.

Notes

1. AJ Stuyvenberg, "Clawdbot bought me a car," AaronStuyvenberg.com (blog), January 24, 2026, accessed April 24, 2026, <https://aaronstuyvenberg.com/posts/clawd-bought-a-car>.
2. Peter Steinberger, "OpenClaw: Your Personal, Open Source AI Assistant," GitHub, November 25, 2025, accessed April 18, 2026, <https://github.com/openclaw>.
3. Dylan Butts and Matthew Chin, "From Clawdbot to Moltbot to OpenClaw: Meet the AI Agent Generating Buzz and Fear Globally," CNBC, February 2, 2026, accessed April 24, 2026, <https://www.cnbc.com/2026/02/02/openclaw-open-source-ai-agent-rise-controversy-clawdbot-moltbot-moltbook.html>.
4. Church of Molt, accessed April 24, 2026, <https://molt.church>.
5. Andrej Karpathy (@karpathy), X post, January 30, 2026, quoted in MetaKnowing, "Andrej Karpathy: What's Going on at Moltbook [a Social Network for AIs] Is the Most Incredible Sci-Fi Takeoff Thing I Have Seen," Reddit, r/Anthropic, January 30, 2026, accessed April 18, 2026, https://www.reddit.com/r/Anthropic/comments/1qrex8n/andrej_karpathy_whats_going_on_at_moltbook_a/.
6. Erik Reppel and Josh Nickerson, "Introducing Agentic Wallets: Give Your Agents the Power of Autonomy," Coinbase Developer Platform, February 11, 2026, accessed April 18, 2026, <https://www.coinbase.com/developer-platform/discover/launches/agentic-wallets>.
7. The Linux Foundation, "Linux Foundation is Launching the x402 Foundation and Welcoming the Contribution of the x402 Protocol," April 2, 2026, accessed April 18, 2026, <https://www.linuxfoundation.org/press/linux-foundation-is-launching-the-x402-foundation-and-welcoming-the-contribution-of-the-x402-protocol>.
8. Lightning Labs, "Lightning Agent Tools," GitHub, latest release v0.2.5, February 11, 2026, accessed May 14, 2026, <https://github.com/lightninglabs/lightning-agent-tools>.
9. Stavan Parikh and Rao Surapaneni, "Powering AI Commerce with the New Agent Payments Protocol (AP2)," Google Cloud Blog, September 17, 2025, accessed April 29, 2026, <https://cloud.google.com/blog/products/ai-machine-learning/announcing-agents-to-payments-ap2-protocol?hl=en>.
10. Jeff Weinstein and Steve Kaliski, "Introducing the Machine Payments Protocol," Stripe Blog, March 18, 2026, accessed April 29, 2026, <https://stripe.com/blog/machine-payments-protocol>.
11. Tim Berners-Lee, Roy T. Fielding, and Henrik Frystyk Nielsen, "Hypertext Transfer Protocol — HTTP/1.0," RFC Editor, May 1996, accessed April 29, 2026, <https://www.rfc-editor.org/rfc/rfc1945.html>.
12. Roy T. Fielding et al., "Hypertext Transfer Protocol — HTTP/1.1," RFC Editor, January 1997, accessed April 29, 2026, <https://www.rfc-editor.org/rfc/rfc2068.html>.
13. Roy T. Fielding, Mark Nottingham, and Julian Reschke, "RFC 9110: HTTP Semantics," RFC Editor, June 2022, accessed April 29, 2026, <https://www.rfc-editor.org/rfc/rfc9110.html>.

14. Vitalik Buterin, "Vitalik Buterin," accessed April 18, 2026, <https://about.me/vitalikbuterin>.
15. Tool-calling capabilities spread through OpenAI's function calling and Anthropic's tool use features, later combined with computer control, browser navigation, and code/terminal execution. AI systems combining a language model with external tool calls, information retrieval, code execution, and multi-step task performance are generally referred to as AI agents.
16. Katharina Schumacher and Roger Roberts, with Katharina Giebel, "The Agent Commerce Opportunity: How AI Agents Are Ushering in a New Era for Consumers and Merchants," McKinsey Global Institute, October 17, 2025, accessed April 18, 2026, <https://www.mckinsey.com/capabilities/quantumblack/our-insights/the-agentic-commerce-opportunity-how-ai-agents-are-ushering-in-a-new-era-for-consumers-and-merchants>.
17. Stripe, "Pricing Built for Businesses of All Sizes," accessed May 13, 2026, <https://stripe.com/pricing>.
18. Wired Staff, "Flatlr Partners with Dailymotion for Video Crowd-Funding," Wired, May 2, 2012, accessed May 5, 2026, <https://www.wired.com/2012/05/flatlr-partners-with-dailymotion>.
19. Brave, "State of the BAT," January 14, 2021, accessed May 5, 2026, <https://brave.com/blog/state-of-the-bat>; Brave, "Update: Brave Browser and BAT Achievements in 2017 and Goals for 2018," January 12, 2018, accessed May 5, 2026, <https://brave.com/blog/update-brave-browser-and-bat-achievements-in-2017-and-goals-for-2018>.
20. W3C, "Payment Request API," W3C Candidate Recommendation Draft, March 27, 2026, accessed May 5, 2026, <https://www.w3.org/TR/payment-request>.
21. Nick Szabo, "Micropayments and Mental Transaction Costs," Satoshi Nakamoto Institute, May 1999, accessed May 13, 2026, <https://nakamotoinstitute.org/library/micropayments-and-mental-transaction-costs>.
22. Micah Zimmerman, "Bitcoin's Lightning Network Surpasses \$1 Billion in Monthly Volume as Adoption Grows," Bitcoin Magazine, February 19, 2026, accessed April 18, 2026, <https://bitcoinmagazine.com/news/bitcoins-lightning-network-surpasses>.
23. An API key differs in character from the cryptographic keys discussed later in this report. An API key is an access credential issued by a service provider to a specific account or organization — generated and revoked within that service's own management system. In other words, an API key is less a replacement for identity than a technical mechanism for invoking an identity that has already been verified.
24. Amazon Staff, "Meet Alexa for Shopping, Your Personalized, Agentic AI Assistant on Amazon," Amazon News, May 13, 2026, accessed May 14, 2026, <https://www.aboutamazon.com/news/retail/alexa-for-shopping-ai-assistant>.
25. HTTP (Hypertext Transfer Protocol) is the web's basic communication method, in which a client sends a request and a server returns a response. An HTTP request includes a method (GET, POST, etc.), headers, and where necessary a body. The server responds with a status code. Both x402 and L402 use the 402 Payment Required status code as the signal for a payment demand, and both embed the payment process within the HTTP request-response flow.
26. Coinbase Developer Platform Documentation, "Quickstart for Sellers," accessed May 14, 2026, <https://docs.cdp.coinbase.com/x402/quickstart-for-sellers>.

27. The upto scheme is currently available on EVM networks only.
28. x402 Foundation, "Batch Settlement," accessed May 14, 2026, <https://docs.x402.org/schemes/batch-settlement>.
29. Coinbase Developer Platform Documentation, "Facilitator," accessed April 18, 2026, <https://docs.cdp.coinbase.com/x402/core-concepts/facilitator>.
30. 황치규, "x402 결제 프로토콜, 누적 거래액 1000만달러 돌파," Digital Today, November 9, 2025, accessed April 18, 2026, <https://www.digitaltoday.co.kr/news/articleView.html?idxno=603271>.
31. Olaoluwa Osuntokun, "L402: Authentication and Payments for the Lightning-Native Web," The Lightning Lab, March 30, 2020, accessed May 14, 2026, <https://lightning.engineering/posts/2020-03-30-lsat>.
32. Olaoluwa Osuntokun and Michael Levin, "AI for All: Powering APIs and Large Language Models with Lightning," Lightning Labs, July 6, 2023, accessed May 14, 2026, <https://lightning.engineering/posts/2023-07-05-l402-langchain>.
33. River, "Bitcoin's Lightning Network exceeds \$1B in monthly transaction volume," X, February 20, 2026, accessed April 18, 2026, <https://x.com/River/status/2024532941041185088>.
34. Parikh and Surapaneni, "Powering AI Commerce with the New Agent Payments Protocol (AP2)."
35. "Frequently Asked Questions," Machine Payments Protocol, accessed May 14, 2026, <https://mpp.dev/faq>.
36. Anand Sinha, "USD Coin and USDT Stablecoins Occupy 90% Market Share: Report," Yahoo Finance, March 19, 2025, accessed April 18, 2026, <https://finance.yahoo.com/news/usd-coin-usdt-stablecoins-occupy-223000966.html>.
37. Elizabeth Howcroft and Rishabh Jaiswal, "Circle Assures Market After Stablecoin USDC Breaks Dollar Peg," Reuters, March 12, 2023, accessed May 20, 2026, <https://www.reuters.com/business/crypto-firm-circle-reveals-33-bln-exposure-silicon-valley-bank-2023-03-11>.
38. Chainalysis Team, "Understanding Tornado Cash, Its Sanctions Implications, and Key Compliance Questions," Chainalysis, August 30, 2022, accessed May 20, 2026, <https://www.chainalysis.com/blog/tornado-cash-sanctions-challenges>.
39. Marc Andreessen, "Why Bitcoin Matters," The New York Times, January 21, 2014, accessed April 18, 2026, <https://dealbook.nytimes.com/2014/01/21/why-bitcoin-matters>.
40. Laurence Fink, "Larry Fink's 2025 Annual Chairman's Letter to Investors," BlackRock, March 31, 2025, accessed May 10, 2026, <https://www.blackrock.com/corporate/investor-relations/2025-larry-fink-annual-chairmans-letter>.
41. Naga Avan-Nomayo, "Tether CEO Predicts One Trillion AI Agents Will Use Bitcoin and USDT for Transactions Within 15 Years," The Block, June 25, 2025, accessed May 10, 2026, <https://www.theblock.co/post/359645/tether-ceo-ai-agents-bitcoin-usdt>.
42. Haleluya Hadero and David Bauder, "The New York Times Sues OpenAI and Microsoft for Using Its Stories to Train Chatbots," Associated Press, December 27, 2023, accessed May 20, 2026, <https://apnews.com/article/6ea53a8ad3efa06ee4643b697df0ba57>.

43. The New York Times Company v. Microsoft Corporation et al., Complaint, No. 1:23-cv-11195, U.S. District Court for the Southern District of New York, December 27, 2023, accessed May 20, 2026, <https://www.courtlistener.com/docket/68117049/the-new-york-times-company-v-microsoft-corporation>.
44. Will Allen and Simon Newton, "Introducing Pay Per Crawl: Enabling Content Owners to Charge AI Crawlers for Access," Cloudflare, July 1, 2025, accessed May 20, 2026, <https://blog.cloudflare.com/introducing-pay-per-crawl>.

The Age of Machine Payments

The Race to Build Payment Infrastructure for the Machine Economy: x402 and Its Rivals

Researchers

Jeewan Lee

Hyemin Son

Corresponding Author

Taemin Oh

Editorial Designer

Seongah Youn

Translator

Samuel J. Hahn

Contact

Jeewan Lee jeewanlee27@gmail.com