x402: An open standard for internet-native payments

An HTTP based protocol for agents, context retrieval, APIs, and more

By:
Erik Reppel **Ronnie Caspers** **Kevin Leffew** **Danny Organ**

Coinbase Developer Platform / x402

May 6, 2025

Contents

Abstract

x402 is an open payment standard that enables AI agents and web services to autonomously pay for API access, data, and digital services. By leveraging the long-reserved HTTP 402 "Payment Required" status code, x402 eliminates the need for API keys, subscriptions, and manual payment processing, allowing real-time, machine-native transactions using stablecoins like USDC.

With one line of code, developers can integrate pay-per-use monetization, unlocking frictionless payments for context retrieval and third-party APIs for AI-driven applications. x402 offers instant settlement, near-zero fees, and chain-agnostic flexibility, making it the ideal solution for AI-first commerce and machine-to-machine payments.

Motivation

The rapid growth of AI and autonomous systems is reshaping the internet economy, but one of the major roadblocks to achieving fully autonomous AI systems is the lack of a payment system that empowers AI Agents to function without human intervention.

Legacy payment systems are designed primarily for human interactions. As such, web services are: not able to be used autonomously by AI agents, stuck using inefficient business models like subscriptions, and hindered by operational complexities such as delayed settlement times, high transaction fees, manual invoicing, and susceptibility to fraud and chargebacks.

These challenges create significant friction for AI-driven applications and machine-to-machine (M2M) transactions, preventing the full realization of autonomous digital economies. AI agents require instant, frictionless access to real-time contextual data, API services, and distributed computing resources to function independently. They need the ability to execute microtransactions dynamically and autonomously, without the human-in-the-loop intervention or delays associated with legacy payment setups.

Browser Usage APIs have attempted to address certain requirements of agentic payments, they continue to leverage systems originally designed for human users rather than machines. Consequently, they remain burdened by manual user experience (UX) navigation, reliance on credit cards, account verification processes, and the overall human-oriented friction that impedes true automation for agentic interactions.

Ultimately, the motivation behind x402 is to unlock the full potential of autonomous AI systems and agentic commerce, fostering a more efficient, frictionless, and scalable digital economy for tool use.

By enabling machine-native payments, x402 enables AI agents to autonomously discover and procure third-party cloud resources, contextual data, and API tools—making it easier for them to achieve their targeted optimization goals without human-in-the-loop intervention. This empowers developers, businesses, and consumers to innovate freely without payment friction, accelerating the adoption and evolution of AI-driven commerce.

Onchain Payments: The Foundation of Autonomous Digital Economies

Where Traditional Payment Rails Fail

Legacy payment systems incur high costs, slow settlement times, chargeback risks, and require layers of manual setup and authorization making them impractical for AI-driven commerce.

For example, ACH bank transfers take 1–3 days to settle, while even credit card payments—despite instant authorization—can take days to finalize and remain subject to reversal for months. Additionally, traditional

systems present significant barriers to financial inclusion, leaving approximately 1.4 billion people unbanked and without access (source).

x402 bridges the gap, allowing AI agents and API providers to interact seamlessly with real-time, trustless payments—eliminating friction from legacy billing systems and unlocking new pay-per-use revenue models.

Scaling Payments with Blockchain and Digital Assets

Compared to legacy payment rails, onchain transactions via x402 settle in ∼200ms, providing instant payment finality for API providers. No rolling chargeback windows, no settlement delays—just real-time access and revenue collection. Built on permissionless blockchain infrastructure, x402 is available worldwide without requiring costly conversions or access to traditional financial rails. For a broader discussion on how onchain payments are transforming digital commerce (Base.org, 2025).

x402 eliminates these inefficiencies by leveraging stablecoins and Layer-2 scaling for low-cost, instant, and automated transactions. The table below compares x402 with traditional payment methods, demonstrating why it is optimal for AI-first, pay-per-use models.

Payment Rail	**Typical Fees**	**Settlement Finality**	**Chargeback Risk**	**Scalability**
Credit Card	$0.30 + 2.9%	Days (batch)	Yes, up to 120d	65k TPS* *theoretical max
PayPal	∼3% + markup	Instant authorization, settlement in days	Yes	Unknown
Stripe (Pay with Crypto)	1.5%+	Depends on blockchain	No - not reversible	Depends on blockchain
Ethereum L1	$1–$5 + gas	1–2 min for confirmations	No - not reversible	15–20 TPS
x402 (on Base)	Free* *nominal gas ¡$0.0001	200 ms	No - not reversible	Hundreds to thousands TPS

How x402 Works

x402 is an open payments protocol developed by Coinbase that enables AI agents to complete transactions autonomously. It is powered by onchain technology and digital currencies (primarily stablecoins like USDC) and provides a lightweight, secure, and instantaneous payment system that we hope can help accelerate the adoption of M2M payments and agentic commerce.

The x402 protocol utilizes the long-reserved HTTP 402 "Payment Required" status code to require a payment to complete an API request or load a webpage. If an API request lacks payment, x402 responds with an HTTP 402 Payment Required status, prompting the client to pay and retry.

With this simple protocol, x402 removes the need for API keys, accounts, and subscriptions. x402 enables any API or content provider to accept pay-per-use payments through a lightweight middleware that integrates seamlessly into existing infrastructures.

Example Integration

With one line of code, companies can monetize APIs, content, and services:

```
paymentMiddleware(amount: "0.10", address: "0x...")
```

Core Payment Flow

1. **Client Request** – AI agent or app requests access to an API or digital resource.
2. **Payment Required (402)** – If no valid payment is attached, the server responds with HTTP 402, providing pricing and payment details.
3. **Agent Retries Request with Signed Payment** – The agent submits a signed payment authorization as part of the retried request.
4. **Web Service Verifies & Broadcasts Payment** – The server validates the payment, broadcasts it, and returns a response to the API request.

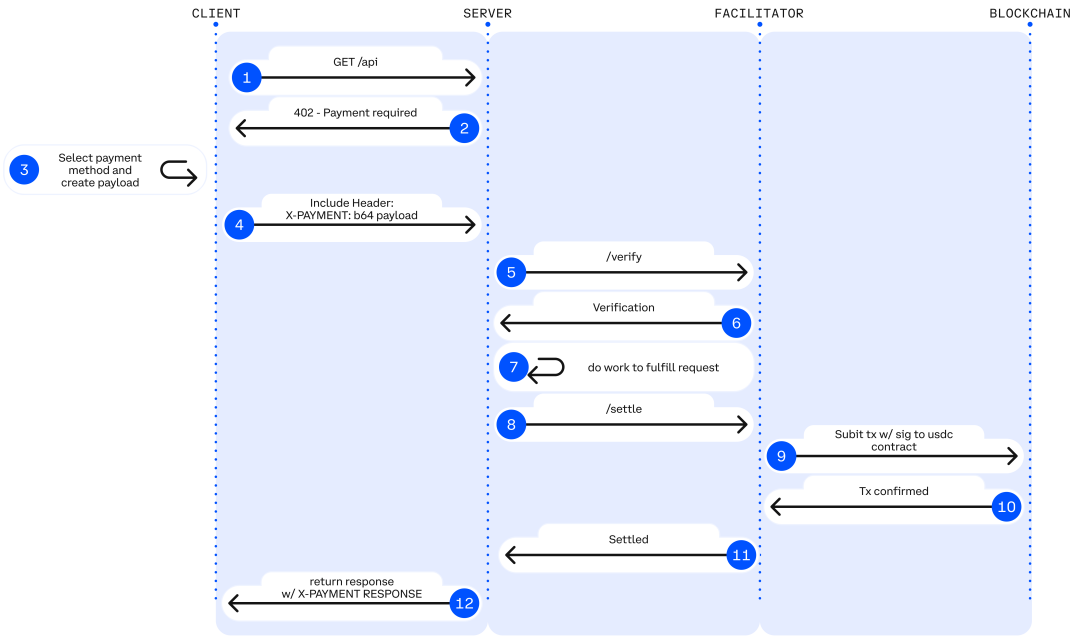

Figure 1: x402 Payment Flow: AI Agent $\leftrightarrow$ *APIServer* $\leftrightarrow$ *Blockchain*

x402 Enables Frictionless Payments

x402 removes account and billing friction from payments, enabling true pay-per-use access without subscriptions, prepaid credits, or manual invoicing. Here's how both AI agents and humans can use x402 to access digital services instantly:

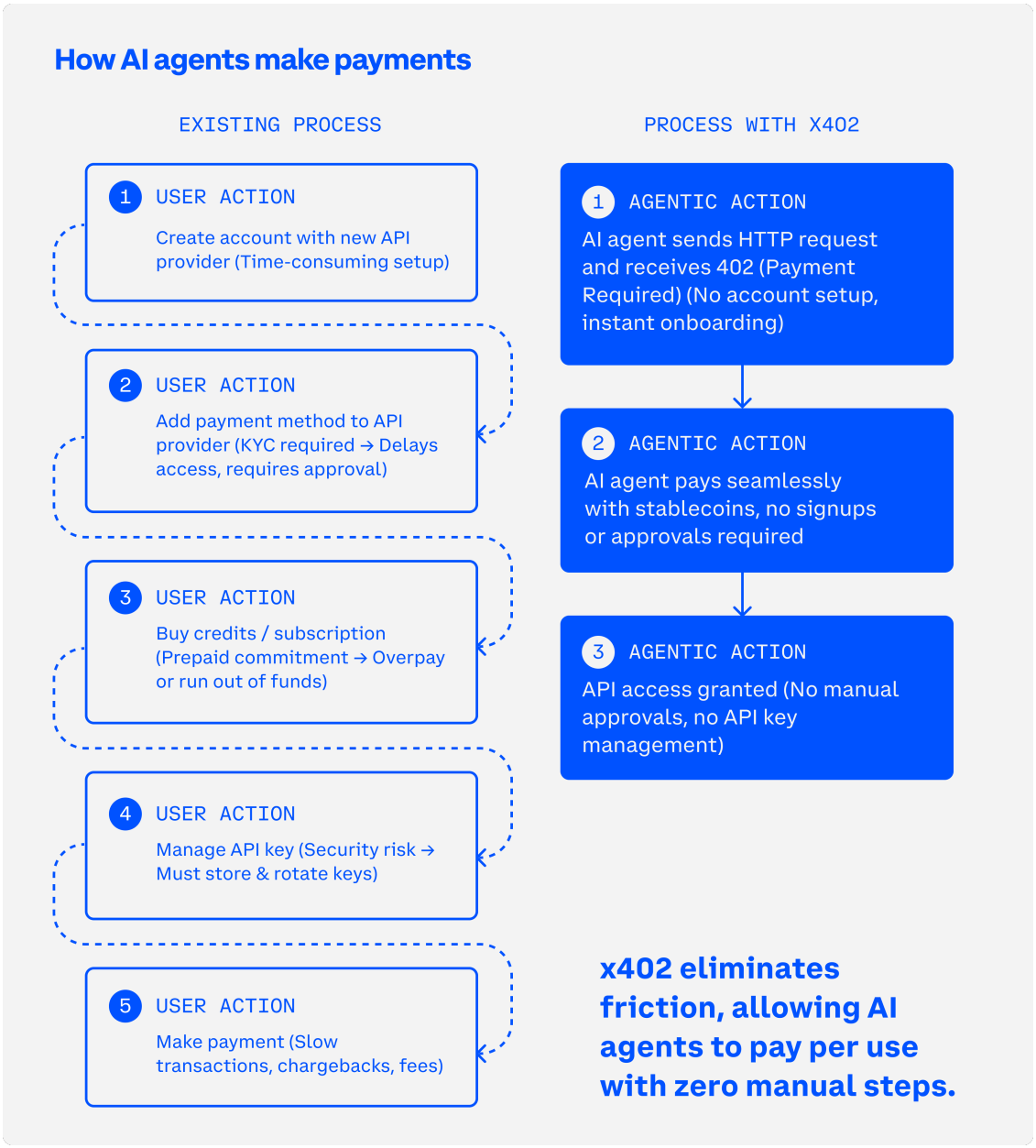

Figure 2: Comparison: Existing Agent Payment Process vs. x402 Pay-Per-Use Simplicity

Scenario	Traditional Process	With x402
AI Agents: Autonomous Research Assistant	• Requires multiple account subscriptions for inference and data access. • Designed for humans to manually create an account & set up API keys. • API whitelisting or approval may be required before use.	1. AI agent requests market data API 2. API responds with HTTP 402 Payment Required and cost details 3. AI agent attaches USDC payment and retries 4. Instant API access granted, immediate context retrieval
Human Users: Pay-Per-Article News Access	• Requires account signup & payment details. • Forces subscription model, even if the user only wants one article. • User must cancel manually to avoid recurring charges.	1. User clicks paywalled article 2. HTTP 402 Payment Required shows USDC cost 3. User confirms payment in crypto wallet 4. Article instantly unlocked, no stored credit cards

Access Control

To further enhance content security and access control after payment, x402 can be combined with decentralized authorization solutions, enabling content providers to enforce granular access rules (e.g., single-use URLs, IP-restricted content streams) without centralized authorization servers.

Creating a Payment Layer for Agentic Commerce

Today's legacy payment systems were built for humans—requiring credit cards, subscriptions, and manual invoicing—making them incompatible with autonomous AI agents, APIs, and machine-to-machine transactions.

As AI systems take on more automated, decision-making roles, they need a payment infrastructure that is seamless and frictionless, enabling autonomous agents to rapidly access context, retrieve real-time data, and execute actions on third-party systems without delay or human intervention.

Empowering Agents to Transact Autonomously

AI models can now operate autonomously—but payments remain a bottleneck. Without a machine-native solution, language models still largely depend on human intervention to access context and actions external to the agent model.

x402 eliminates this need for human intervention, providing a crypto-native payment standard that allows AI systems to:

- Pay per API request, data query, or AI model inference without pre-registering an account.

- Fetch real-time information dynamically (e.g., financial market data, cloud resources).
- Seamlessly interact with onchain and offchain services, triggering payments autonomously.

This enables fully autonomous, AI-driven commerce—allowing goal-oriented agents to operate independently in an on-demand, permissionless economy.

Enabling New Business Models

Pragmatic Micropayments

Legacy payment rails operate on an account-based model and generally require some degree of trust/credit risk among counterparties.

These rails were designed for humans, and don't work for small, high-frequency transactional services like API requests. With fees as high as $0.30 per transaction, microtransactions become impractical—forcing businesses to rely on subscriptions and bundled pricing, causing many potential consumers to abstain from use.

As a payment protocol, x402 makes it possible to charge per request, per service, or per second of usage with:

- Near-zero transaction costs that enable payments as low as $0.001 cents per request.
- True pay-per-use pricing for APIs, AI inference, and on-demand content.
- Machine-to-machine transactions that allow IoT devices and AI agents to autonomously pay for resources.

For the first time, businesses can profitably support micropayments at scale, creating new monetization opportunities for AI-driven platforms. This is made possible because of low cost and high speed of transactions on rollups like Base, and trustworthy stablecoins like USDC that are pegged to the USD.

Seamless Pay-Per-Use Monetization

Accepting payments online has traditionally meant forcing users into accounts, managing API keys, and handling manual billing cycles. x402 removes these barriers by offering:

- Seamless per-request payments—No subscriptions, no prepayments, no lock-in.
- Instant, finalized transactions—No chargebacks, no fraud risks, no intermediaries.
- AI-native monetization—Let AI agents and human users pay dynamically without pre-approvals or API keys.

For developers and businesses, this means higher revenue, lower costs, and a seamless payment experience.

Simplifying Payments Operations

Mitigating Fraud, Chargebacks, and Compliance Overhead

Beyond transaction fees, legacy payment systems expose businesses to risks of chargebacks, fraud, operational losses, and compliance overhead. With x402:

- Payments settle instantly onchain, eliminating chargebacks and disputes.

- No PCI compliance required for developers, unless a facilitator chooses to accept card payments directly
- No reliance on banks or third-party approvals, ensuring global, permissionless access.

By removing these roadblocks, x402 allows businesses to focus on growth, not payment complexities.

Future-Proof, Chain- and Token-Agnostic Payments

x402 is built to support any stablecoin, digital asset, or blockchain— unlocking flexibility beyond traditional payment providers.

- While USDC is one of the first supported assets, x402 is designed to support stablecoins, digital assets, and multiple blockchain networks.
- Businesses and developers can choose the best cryptocurrency for their use case, ensuring maximum flexibility.

This future-ready design ensures that x402 will continue evolving as AI-driven commerce scales.

The x402 Spec: A Flexible HTTP Standard for Payments

The x402 middleware is a lightweight integration that enables pay-per-request API payments. The following section details how developers can configure x402 and how AI agents handle payment-required responses in real time.

Middleware Configuration

```
paymentMiddleware(amount: "0.10", address: "0x...")
```

Parameter	Description
amount	Cost per request (e.g., "$0.10")
address	Wallet address where payments are received

Handling Requests Without Payment

If a request is submitted without payment, the server responds with HTTP 402 (Payment Required). The response provides structured feedback for AI agents and humans, including:

```
{
  "maxAmountRequired": "0.10",
  "resource": "/api/market-data",
  "description": "Access to real-time market data requires payment.",
  "payTo": "0xABCDEF1234567890ABCDEF1234567890ABCDEF12",
  "asset": "0xA0b86991C6218b36c1d19D4a2e9Eb0cE3606EB48",
  "network": "ethereum-mainnet"
}
```

Field	Description
maxAmountRequired	Payment amount required for access (e.g., $0.10)
resource	The requested API endpoint or service
description (optional)	Custom message describing payment details
payTo	Developer's wallet address (receiving payment)
asset	Contract address for the transaction
network	Blockchain network identifier

Technical Specifications

x402 implements a standardized approach to HTTP 402 responses with precise specifications for payment requests and verification.

Payment Request Format

When an API returns a 402 response, it includes a structured JSON payload with the following fields:

Field	Description
maxAmountRequired	Maximum payment amount required (e.g., "0.10")
assetType	Token standard (e.g., "ERC20")
assetAddress	Contract address of the payment token
paymentAddress	Recipient's wallet address
network	Blockchain network identifier (e.g., "base-mainnet")
expiresAt	Timestamp after which this payment request is no longer valid
nonce	Unique identifier to prevent replay attacks
paymentId	Unique identifier for this payment request

Payment Authorization

When submitting payment, clients include a cryptographically signed message containing:

- All fields from the payment request
- The actual payment amount (must be $\leq$ maxAmountRequired)
- Timestamp of the authorization
- Cryptographic signature from the paying wallet

The signature follows the EIP-712 standard, enabling clear and secure presentation in wallet interfaces when users authorize transactions.

Transaction Settlement

x402 implementations can handle settlement through various methods:

- On-chain settlement: Direct blockchain transactions
- Layer-2 settlement: Using optimistic or ZK rollups for lower fees
- Payment channels: For high-frequency micropayments between trusted parties
- Batched settlements: Combining multiple micropayments into a single transaction

This flexibility allows developers to choose the most efficient settlement mechanism for their specific use case and transaction volume.

Integration Examples

x402 is designed for straightforward integration across various programming languages and frameworks.

Server-Side Implementation

For NodeJS applications using Express:

```
// Install with: npm install @x402/express-middleware

const express = require('express');
const { x402PaymentRequired } = require('@x402/express-middleware');

const app = express();

app.get('/premium-data', x402PaymentRequired({
  amount: "0.10",
  address: "0x1234...",
  assetAddress: "0x2345...", // USDC contract
  network: "base-mainnet"
}), (req, res) => {
  // This code only runs after valid payment
  res.json({ premiumData: "Valuable information" });
});

app.listen(3000);
```

Client-Side Implementation

For web applications using the x402 client library:

```
// Install with: npm install @x402/client

import { x402Client } from '@x402/client';
import { connectWallet } from 'your-wallet-connector';

const client = new x402Client();
const wallet = await connectWallet();

// Configure the client with the user's wallet
client.setWallet(wallet);

// Make API requests with automatic payment handling
try {
  const data = await client.fetch('https://api.example.com/premium-data');
  console.log(data); // The premium data
} catch (error) {
  console.error('Payment failed:', error);
}
```

Wallet Integration

x402 is designed to work with any cryptocurrency wallet that supports standard signing methods. When a payment is required, the wallet interface displays a confirmation screen showing:

- The request domain (e.g., api.example.com)
- The requested payment amount
- The payment token (e.g., USDC)
- The specific resource being accessed

This transparency ensures users always know exactly what they're paying for, maintaining trust between service providers and users.

Testing and Development

The x402 toolkit includes a local development environment with:

- Test wallets pre-loaded with test tokens
- Mock API servers implementing the x402 protocol
- Tooling for simulating various payment scenarios
- Detailed logging for debugging payment flows

These tools allow developers to build and test x402 integrations without connecting to production blockchains or spending real tokens.

Use Cases: Real World Applications of x402

Here's how AI agents and humans are using x402 in various contexts:

Agents Accessing APIs for On Demand Requests

- A research platform enables pay-per-article access, eliminating bundled paywalls and allowing AI-driven tools to pay only for relevant content.
- A video streaming service leverages x402 to charge per second of content watched, replacing traditional subscription-based monetization.
- A trading AI retrieves real-time stock market data for $0.02 per request, paying only when needed.

Pay-Per-Use AI Model Inference Monetization

- A computer vision API charges $0.005 per image classification instead of a fixed enterprise fee.
- A synthetic voice AI charges $0.10 per audio clip, enabling flexible monetization.

Agents Paying for Cloud Compute & Storage

- An autonomous agent purchases GPU resources for $0.50 per GPU-minute, paying per compute cycle.
- A goal-driven AI model expands cloud storage as needed for context and reinforcement learning, paying per GB stored.

Context Retrieval for Agents

- A financial AI assistant pays $0.25 per premium news article for research.
- A legal research agent accesses court rulings at $0.10 per document, avoiding full database subscriptions.

Micropayments for Human Access to Content

- A Substack writer charges $0.25 per article for casual readers, allowing pay-as-you-go access instead of full subscriptions.
- A premium research journal uses x402 to let readers pay per whitepaper download instead of requiring an annual membership.
- A high-quality podcast enables per-episode payments instead of forcing a monthly subscription.
- A game charges a user per-play instead of requiring a large purchase or relying on advertising revenue.

With x402, providers of contextual data can now seamlessly monetize using existing frameworks like the Model Context Payment (MCP) protocol.

Key Takeaways

The evolution of AI-driven systems demands a payment infrastructure that is as seamless and autonomous as the agents using it. x402 removes the friction of traditional payment rails, enabling real-time, pay-per-use

transactions without accounts, subscriptions, or manual billing.

- **Instant, low-cost transactions.**
- **No API keys, no subscriptions, no middlemen.**
- **AI-first, developer-friendly, and blockchain-agnostic.**

By standardizing payments at the protocol level, x402 paves the way for a more open, efficient, and scalable digital economy—one where AI agents, developers, and service providers can interact natively and autonomously.

Reference Implementation

The evolution of AI-driven systems demands a payment infrastructure that is as seamless and autonomous as the agents using it. x402 removes the friction of traditional payment rails, enabling real-time, pay-per-use transactions without accounts, subscriptions, or manual billing.

By standardizing payments at the protocol level, x402 paves the way for a more open, efficient, and scalable digital economy—one where AI agents, developers, and service providers can interact natively and autonomously. Reference Implementation

The x402 protocol has a full reference implementation available as an open-source project. It includes core protocol libraries that handle the HTTP 402 response flow, server-side middleware for frameworks like Express.js and Next.js, and client libraries for both browser and Node.js environments. It also provides cryptographic utilities for signing and verifying payments, along with services for broadcasting transactions to various blockchain networks.

Developers can use these components as building blocks or examples when integrating x402 into their own services. This helps ensure they follow the protocol correctly while reducing the amount of work needed to implement it.

For those building the next generation of AI-powered applications, x402 provides a foundation for frictionless, machine-native transactions.

Learn more at: x402.org