

EIP 2718: Typed Transaction Envelope

[Return to Ethereum ERCs](#)

: **Note:** The following is an excerpt from the official Ethereum site. It is provided here as a convenience and is not authoritative. Refer to the original document as the authoritative reference.

Table 1: Data sheet for Typed Transaction Envelope

Title	Typed Transaction Envelope
EIP	2718
Author	Micah Zoltu (@MicahZoltu)
Status	Final
Created	2020-06-13
Description	https://github.com/ethereum/EIPs/blob/master/EIPS/eip-2718.md
Specification	https://github.com/ethereum/EIPs/blob/master/EIPS/eip-2718.md#Specification
Category	Core

Abstract

TransactionType || **TransactionPayload** is a valid transaction and **TransactionType** || **ReceiptPayload** is a valid transaction receipt where **TransactionType** identifies the format of the transaction and ***Payload** is the transaction/receipt contents, which are defined in future EIPs. Motivation

In the past, when we have wanted to add new transaction types we have had to ensure they were backward compatible with all other transactions, meaning that you could differentiate them based only on the encoded payload, and it was not possible to have a transaction that matched both types. This was seen in [EIP 155: Simple replay attack protection](#) where the new value was bit-packed into one of the encoded fields. There are multiple proposals in the discussion that define new transaction types such as one that allows EOA accounts to execute code directly within their context, one that enables someone besides **msg.sender** to pay for gas, and proposals related to layer 1 multi-sig transactions. These all need to be defined in a way that is mutually compatible, which quickly becomes burdensome to EIP authors and to clients who now have to follow complex rules for differentiating transaction types.

By introducing an envelope transaction type, we only need to ensure backward compatibility with existing transactions and from then on we just need to solve the much simpler problem of ensuring there is no numbering conflict between **TransactionTypes**.

From:
<https://www.omgwiki.org/dido/> - **DIDO Wiki**

Permanent link:
https://www.omgwiki.org/dido/doku.php?id=dido:public:ra:xapend:xapend.b_stds:defact:ethereum:eip:2718

Last update: **2022/05/21 13:30**

