Manage UTXO

UTXOs are core to the logic of the OMG Network.

Merging UTXOs

Merging UTXOs is desirable in the following scenarios:

1. Exiting your funds as a single UTXO

Standard Exits are initiated on a single UTXO and not a specified amount. However, a user may want to exit an amount greater than the value of any UTXO one owns. For example:

- Alice owns 4 UTXOs worth 1 ETH each.
- Alice would like to withdraw all her funds from the network.
- To exit her funds, Alice would need to initiate a `Standard Exit` (with an `exit bond`) on each UTXO.

In the above scenario, it would be more economical for Alice to merge these UTXOs and exit a single one.

2. Fitting Inputs into a Transaction

A transaction can have a maximum of 4 inputs but a user may not own four UTXOs that can cover the amount needed for a given transaction. For example:

- Alice owns 5 UTXOs worth 1 ETH each.
- Alice would like to send 5 ETH to Bob.
- Alice cannot send 5 ETH to Bob in a single transaction as a transaction can take four inputs only.

In the above scenario, Alice can merge her UTXOs to send the desired amount to Bob in a single transaction.

Implementation

1. Install `omg-js`

To access network features from your application, use our official libraries:

Node Browser React Native

Requires Node >= 8.11.3 < 13.0.0

Copy

```
npm install @omisego/omg-js
```

JavaScript (ESNext)

2. Import dependencies, define constants

Merging UTXOs involves using 2 `omg-js` objects. Here's an example of how to instantiate them:

Copy

```
import { ChildChain, OmgUtil } from "@omisego/omg-js";
const childChain = new ChildChain({ watcherUrl });

// define constants
const plasmaContractAddress = plasmaContractAddress;

const utxoMerge = {
  currency: OmgUtil.transaction.ETH_CURRENCY,
  address: "0x8CB0DE6206f459812525F2BA043b14155C2230C0",
  privateKey: "CD55F2A7C476306B27315C7986BC50BD81DB4130D4B5CFD49E3EAF9ED1EDE4F7"
}
```

- `watcherUrl` - the Watcher Info URL for defined environment (personal or from OMG Network).
- `plasmaContractAddress` - `CONTRACT_ADDRESS_PLASMA_FRAMEWORK` for defined environment.

3. Merge UTXOs

Note, the minimum number of UTXOs to merge is 2, the maximum — 4. You also can't mix ETH and ERC20 UTXOs while performing merging.

> The merging UTXOs process is the same for both ETH and ERC20. This method demonstrates merging for ETH UTXOs. If you want to merge ERC20 tokens, change the `currency` value to a corresponding smart contract address.

```
async function mergeUtxo() {
  // retrieve all utxos
  const addressUtxosAll = await childChain.getUtxos(utxoMerge.address);

  // filter utxos based on the token
  const addressUtxos = addressUtxosAll.filter(
    (u) => u.currency === utxoMerge.currency
  );

  // slice the array to only 4 utxos
  const utxosToMerge = addressUtxos.slice(0, 4);

  // merge utxos
  const mergedUtxo = await childChain.mergeUtxos({
    utxos: utxosToMerge,
    privateKey: utxoMerge.privateKey,
    verifyingContract: plasmaContractAddress,
  });

  return utxo;
}
```

No fee is charged for merge transactions on the OMG Network since these are transactions that benefit the network.

Lifecycle

1. A user calls the `getUtxos` function to retrieve the list of all available UTXOs.
2. A user filters an array of UTXOs and returns an UTXOs for the desired currency (`ETH_CURRENCY` for ETH or `ERC20_CONTRACT_ADDRESS` for ERC20 tokens). This step is important because you can't merge ETH and ERC20 tokens together.
3. A user calls the `mergeUtxos` function and returns an array of merged UTXOs.

Splitting UTXOs

A user may also want to *split* UTXOs if one would like to exit an amount smaller than the value of any UTXO one owns. For example:

- Alice owns 1 UTXO worth 5 ETH.
- Alice wants to withdraw 2 ETH back onto the root chain and keep 3 ETH on the child chain.
- Alice cannot exit 2 ETH unless she splits her UTXO.

In the above scenario, Alice can split her UTXO to withdraw 2 ETH.

Currently, UTXO splitting is not available via omg-js lib but you can use the OMG Network Web Wallet to achieve the same result.

Network Considerations

Users are highly encouraged to merge UTXOs continuously as a way of mitigating the vulnerability of OMG Network funds in a mass exit event. Read more in the FAQ Section.

Demo Project

This section provides a demo project that contains a detailed implementation of the tutorial. If you consider integrating with the OMG Network, you can use this sample to significantly reduce the time of development. It also provides step-by-step instructions and sufficient code guidance that is not covered on this page.

JavaScript

For running a full `omg-js` code sample for the tutorial, please use the following steps:

1. Clone omg-js-samples repository:

```
git clone https://github.com/omgnetwork/omg-js-samples.git
```

2. Create `.env` file and provide the required configuration values.

3. Run these commands:

```
npm install
npm run start
```

6. Open your browser at http://localhost:3000.

7. Select `Show UTXOs`, `Merge UTXOs` or `Split UTXOs` on the left side, observe the logs on the right.

> Code samples for all tutorials use the same repository — `omg-js-samples`, thus you have to set up the project and install dependencies only one time.