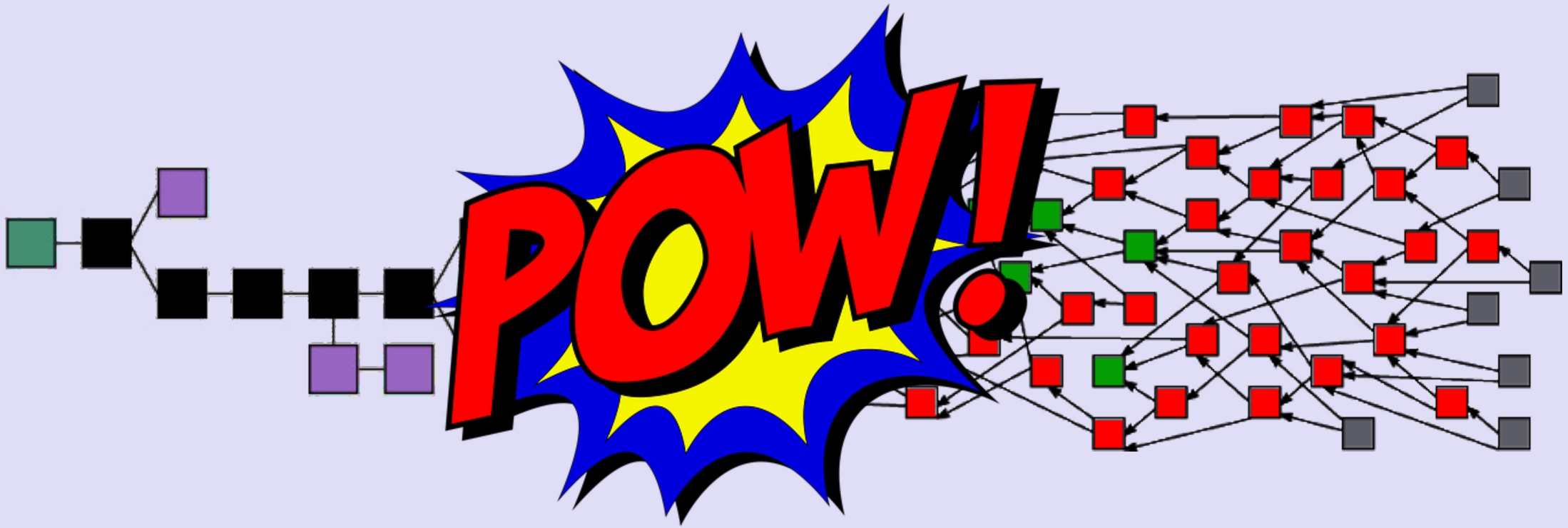
IOTA TUTORIAL 7

Proof of Work, Curl & Nonce

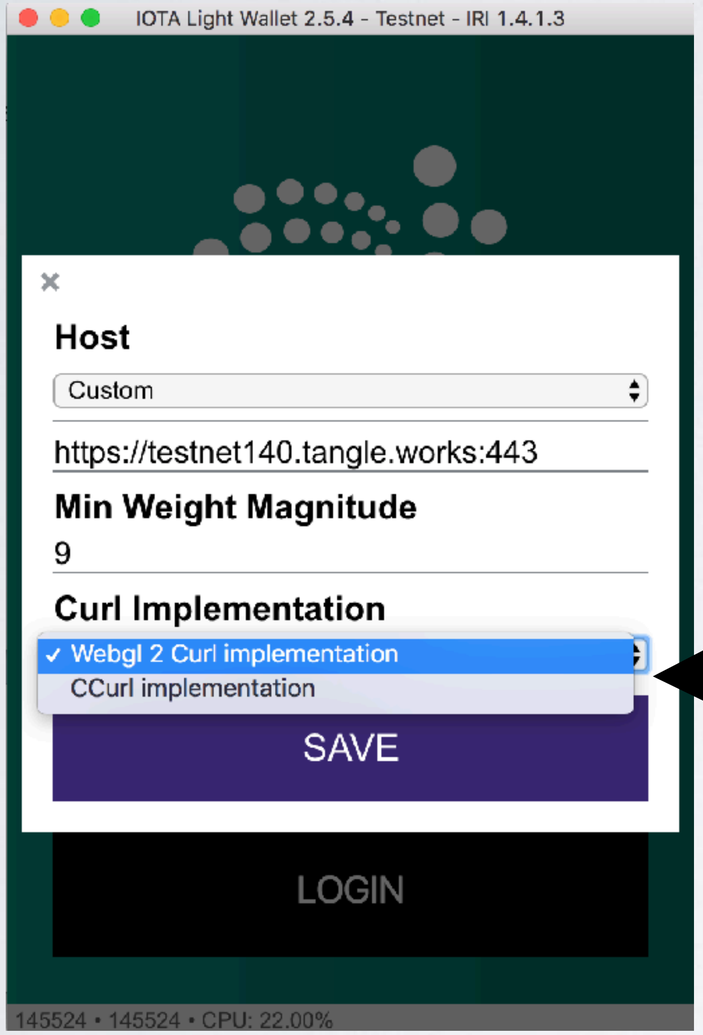


INTRO

- In this video I will explain what the purpose is of the IOTA light wallet Curl implementation and what the difference is between Webgl 2 Curl implementation and CCurl implementation.

- Furthermore I will explain what Proof of Work is and what a nonce is.

CURL IMPLEMENTATION



IOTA Light Wallet

Webgl 2 Curl implementation

CCurl implementation

MAKING A TRANSACTION

- Making a transaction is basically a 3-step process:

- Step 1. Constructing the bundle and signing the transaction inputs with your private keys.

  - IOTA uses a bundle which consists of multiple transactions containing credits to the receiving addresses (outputs) and debits from the spending addresses (inputs).

  - In IOTA there are two types of transactions: one where you transfer value and thus, have to sign inputs, and ones where you simply send a transaction to an address with no value transfer (e.g. a message). A bundle represent a transfer of value.

  - A transaction is an object containing several fields such as an address, signature, value and tag.

MAKING A TRANSACTION

- Step 2. Tip selection
  - The tip selection is a process whereby you traverse the tangle in a random walk to randomly chose two transactions which will be validated by your transaction. Your transaction checks for example if the descendants of that transaction is valid. If these transactions are valid they will be added to your bundle construct and are called branchTransaction and trunkTransaction.

MAKING A TRANSACTION

- Step 3. Proof of Work (PoW)
  - Once the bundle is constructed, signed and the tips are added to the bundle, the PoW has to be done for each transaction in the bundle. Every transaction in a bundle requires a nonce (this is the result of the PoW) in order to be accepted by the tangle network.
  - IOTA's PoW is directly comparable to Hashcash, as it serves a similar purpose to prevent spam, and in IOTA's case, also to prevent sybil-attacks.
  - When the PoW is done, the nonce of the transaction object should be updated. The transaction can now be broadcasted to the tangle network and wait for it to be approved by someone else.

CURL

- The IOTA team created their own cryptographic hash function called Curl.
- This hash function is used for a number of purpose in IOTA, but in this video I am only focussing in the usage of Curl in the context of PoW.
- In this video I am not discussing the Curl algorithm itself and how it is implemented.

CURL USED IN PROOF OF WORK

- The IOTA light wallet uses the Curl hash algorithm for PoW.
  There are two methods to execute the Curl hash algorithm:

  - Method 1: Using **Webgl 2 Curl implemention**.
    The IOTA light wallet is build using the Electron framework. Electron combines Chromium (used by Google Chrome browser) and Node.js into a single app, which can be packaged for Mac, Windows, and Linux. The Curl hash algorithm is ported to Javascript (curl.lib.js) to work in WebGL-enabled browsers such as Chromium.
    Web **G**raphics **L**ibrary (WebGL) is a JavaScript API for rendering interactive 2D and 3D graphics within any compatible web browser.
    See: http://webglsamples.org/

  - WebGL uses the system Graphics Processing Unit (**GPU**).

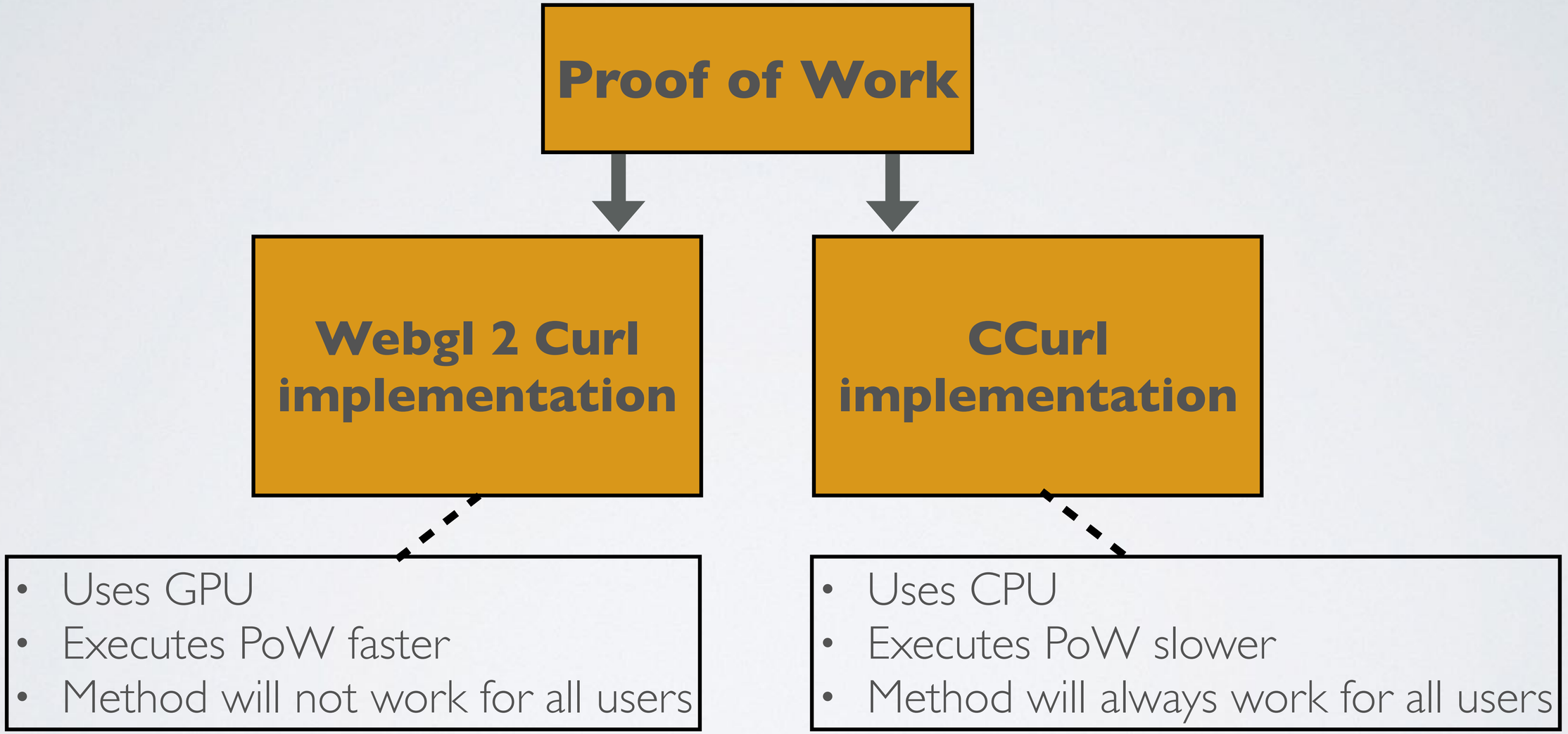
CURL USED IN PROOF OF WORK

- Method 2: Using **CCurl implementation**.
  CCurl means C port of the Curl library, which uses the system Central Processing Unit (**CPU**) (aka native route)

- PoW is executed faster using GPUs instead of CPUs.

- By default the IOTA light wallet uses the "Webgl 2 Curl" implementation thereby speeding up the PoW. However, some people will find that they get an "**Invalid Transaction Hash**" when they use this setting. In that case select the light wallet option "CCurl implementation".

CURL USED IN PROOF OF WORK



CCURL IMPLEMENTATION

- CCurl implemenation means using the C port of the Curl library for the PoW.

- The CCurl library can be found at:
  https://github.com/iotaledger/ccurl

WEBGL 2 CURL IMPLEMENTATION

- Webgl 2 Curl implementation means using the curl.lib.js ported Javascript library to work in WebGL enabled browsers for the PoW.

- The curl.lib.js library can be found at: https://github.com/iotaledger/curl.lib.js

PROOF OF WORK

- As mentioned earlier the purpose of the PoW is to prevent spam and sybil-attacks.
- **PoW means calculating the nonce for every transaction in a bundle.**
- When making a value or non value transaction you pay no fee.
- **However there is a small cost you are paying. You pay for the electrical energy that you spend for the PoW.**

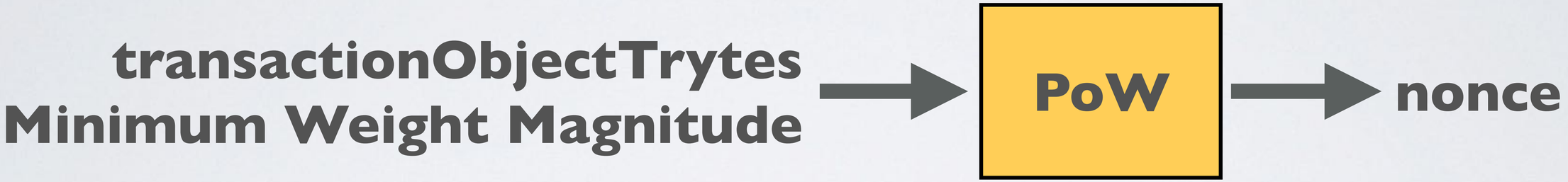
PROOF OF WORK

- Get the Minimum Weight Magnitude (MWM).
  The Minimum Weight Magnitude is the difficulty of PoW.
  More information about the Minimum Weight Magnitude, watch IOTA Tutorial #4.

- An IOTA transaction data is encoded and stored in a string of 2673 trytes.
  (= transactionObjectTrytes)
  The last 81 trytes of the transactionObjectTrytes is **reserved** for the nonce.
  More information about the anatomy of a transaction can be found at:
  https://iota.readme.io/

PROOF OF WORK

- Execute the PoW using the transactionObjectTrytes and Minimum Weight Magnitude as input. The PoW outputs the nonce which is 81 trytes in size.



- Insert the nonce in the transactionObjectTrytes (= transactionObjectWithNonceTrytes).

VALIDATE THE PROOF OF WORK

- Convert transactionObjectWithNonceTrytes into trits
  (= transactionObjectWithNonceTrits)

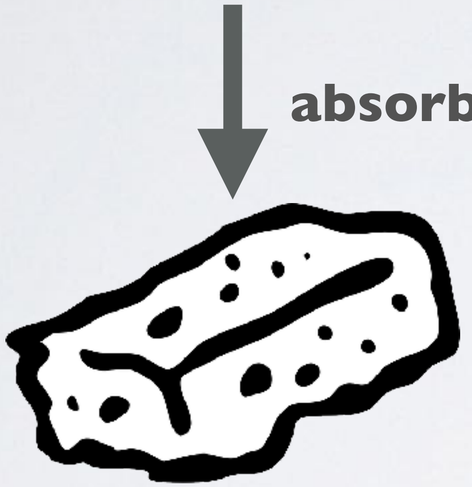
- Create and initialise a CheckHash object (type: Int32Array(243)).
  This object will hold the Curl hash algorithm result.

- Create and initialise a CurlHash object (type: Int32Array(3x243)).
  This object will:
  - receive inputs (absorb the transactionObjectWithNonceTrits)
  - execute the Curl hash algorithm
  - outputs the result (squeeze data into the CheckHash object)

VALIDATE THE PROOF OF WORK

- Apply the Curl hash algorithm:

transactionObjectWithNonceTrits



absorb

CurlHash object executes the Curl hash algorithm

squeeze

CheckHash

VALIDATE THE PROOF OF WORK

- The CheckHash object will hold the Curl hash algorithm result in trits.
- The number of 0's at the end of the CheckHash value must be at least the Minimum Weight Magnitude. If that is the case the nonce is valid.
- A valid nonce is required for the transaction to be accepted by the tangle network.