What's a Sparse Merkle Tree?

Kelvin Fichter · Follow
4 min read · Sep 14, 2018

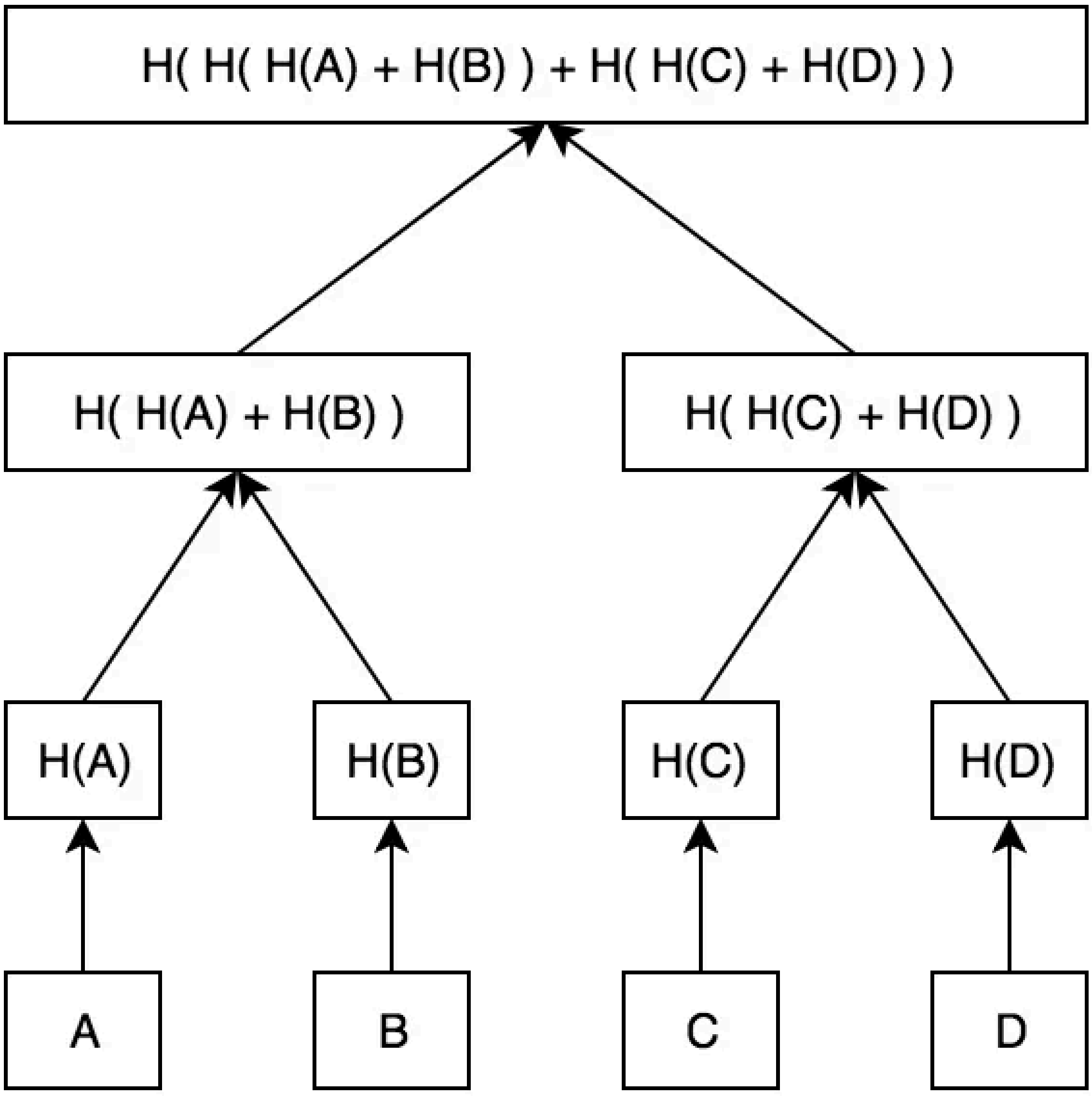
Listen Share

If you've been around the Ethereum research community lately, you might've heard of something called a sparse Merkle tree. They might sound scary, but they're really quite simple. This post will explain exactly what a sparse Merkle tree is, why they're cool, and what they're currently being used for.

Merkle Trees

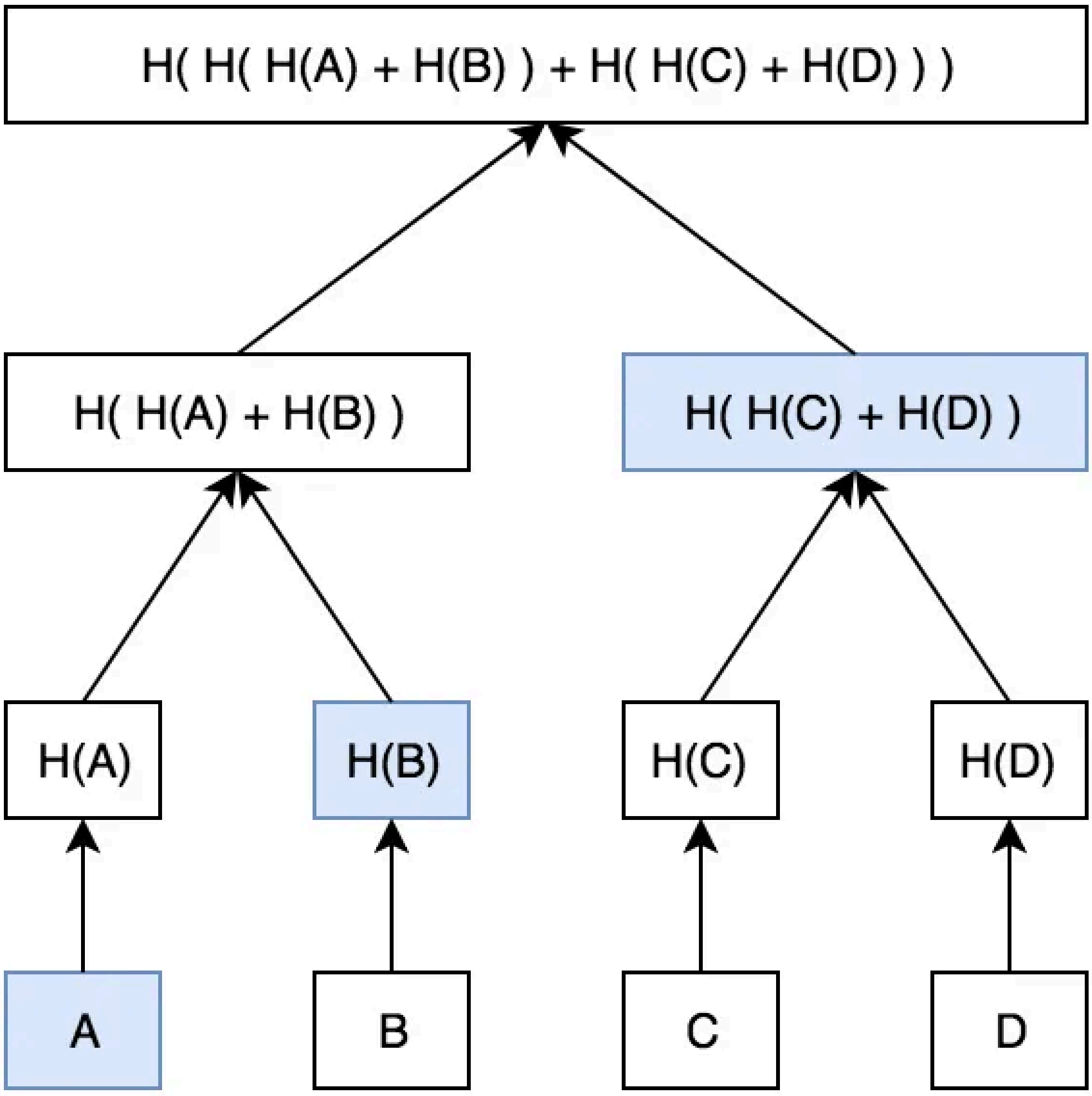
First, let's start with a quick refresher on Merkle trees. Merkle trees give us a way to cryptographically commit to a set of data. We start by hashing each piece of data in the set, and then keep hashing our way up the tree until we get to the root node.

That ends up looking like this:



Proving Inclusion

The root of this tree is just a hash — it tells us nothing about the contents of the tree. We can use something called a "Merkle proof" to show that some content is actually part of this tree. For example, let's prove that `A` is part of the above tree. All we need to do is provide each of `A`'s siblings on the way up, recompute the tree, and make sure everything matches.



I went ahead and highlighted the siblings in blue. With just `A` , `H(B)` , and `H(H(C)+H(D))` , we can recompute the original root hash. This is an efficient way to show that `A` is part of this tree without having to provide the entire tree!

Proving Non-inclusion

So we can easily prove that something **is** part of the Merkle tree, but what if we want to prove that something **isn't** part of the tree? Unfortunately, standard Merkle trees don't give us any good way to do this. We could reveal the entire contents, but that's sort of defeating the point of using a Merkle tree in the first place.

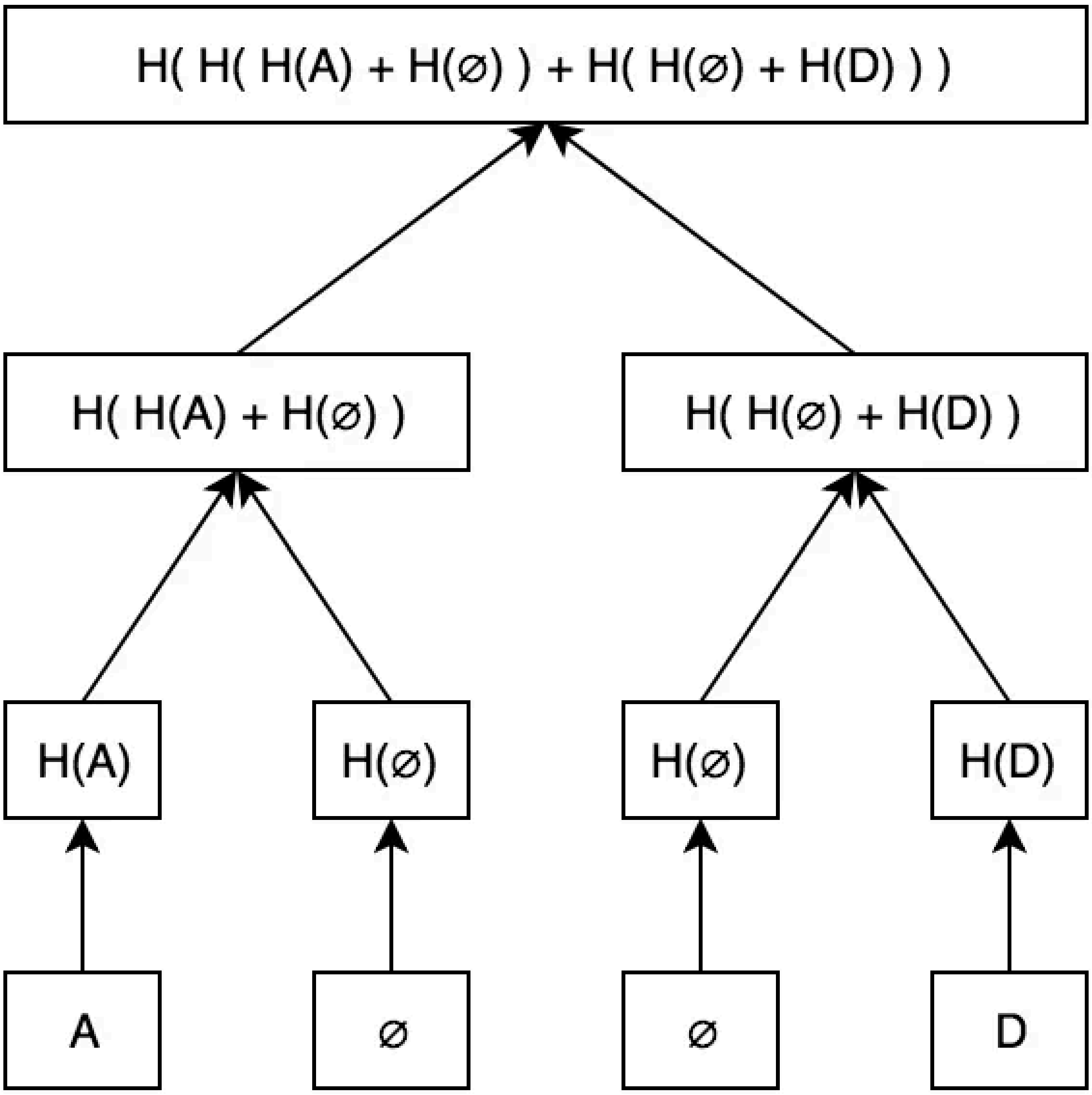
Sparse Merkle Trees

Here's where sparse Merkle trees come into play. A sparse Merkle tree is like a standard Merkle tree, except the contained data is indexed, and each datapoint is placed at the leaf that corresponds to that datapoint's index.

Let's say we have a Merkle tree with four leaves. We'll populate this tree with some letters (`A` , `D`) to demonstrate. The letter `A` is the first letter of the alphabet, so we should put it at the first leaf. Similarly, we can put `D` at the fourth leaf.

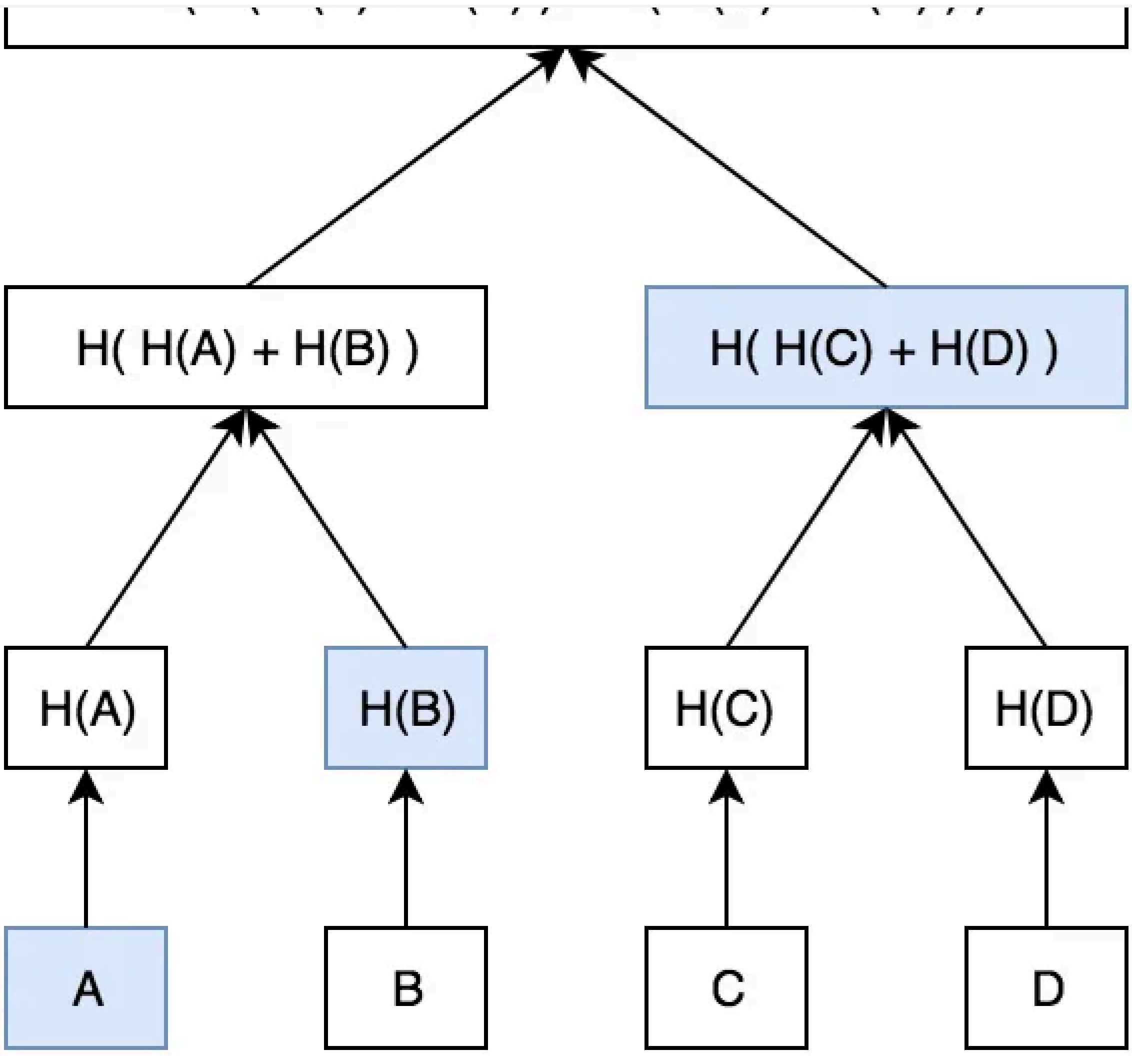
So what happens in the second and third leaves? **We just leave them empty**. More precisely, we put a special value (like `null`) instead of placing a letter.

The tree ends up looking like this:



Proving Inclusion

Just like in a standard Merkle tree, we can use a Merkle proof to prove that `A` is part of this tree. This proof looks just like your standard Merkle proof:

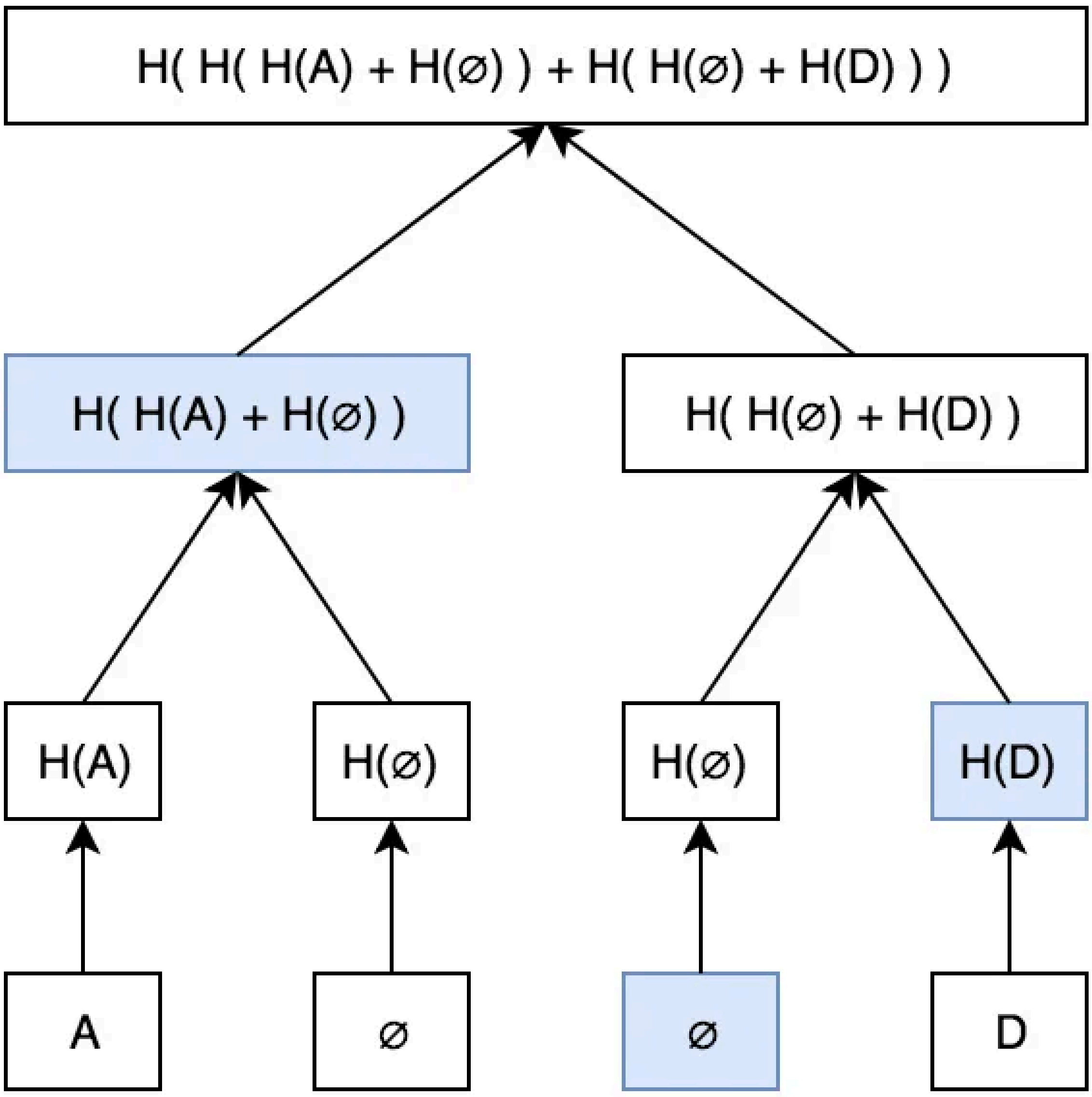


Again, we're just providing the siblings, `H(null)` and `H(H(null)+H(D))`, and checking that it matches the root.

Proving Non-inclusion

Here's where the magic happens. What happens if we want to prove that `C` is **not** part of this Merkle tree? It's easy! We know that if `C` were part of the tree, it would be at the third leaf. If `C` isn't part of the tree, then the third leaf must be `null`.

All we need is a standard Merkle proof showing the third leaf is `null`!



This looks just like a standard Merkle proof, except that we're proving the leaf contains `null` instead of `C`.

The best part of a sparse Merkle tree is that they're really representing key-value stores, inside of a Merkle tree!

Drawbacks

Sparse Merkle trees are really cool because they give us **efficient proofs of non-inclusion.** However, this can also mean they get really, really big. 26 letters isn't much, but most of the time we're talking about 2^{256} hashes! That's just way too many indices to generate a tree manually.

Luckily, there are some techniques to efficiently generate Merkle trees. The key to these techniques is that these giant sparse Merkle trees are mostly... sparse. `H(null)` is a constant value, and so is `H(H(null))`, etc. etc. Huge chunks of the tree can be cached!

Use Cases

Sparse Merkle trees are already being used in the wild to do some cool blockchain stuff.

Plasma Cash makes use of sparse Merkle trees to store information about deposited assets. Every Plasma Cash asset has a unique ID. Whenever an asset is transferred to a new user, a transaction is included in the sparse Merkle tree at the asset's index! Proofs of inclusion (or non-inclusion!) are used to prove that a given transaction history is valid.

Sparse Merkle trees might even make their way into Ethereum! Ethereum researchers are looking into sparse Merkle trees as a replacement for the Merkle Patricia tries currently used to store Ethereum state.

As always, feedback/comments/questions are more than welcome. If you spot any typos, or you think anything needs clarification, let me know!

Blockchain Merkle Tree Ethereum Plasma

Follow

Written by Kelvin Fichter

266 Followers · 1 Following

Strange chains

Responses (6)

What are your thoughts?

Respond

Chjango Unchained ⛓
about 6 years ago

Take heed, Hanshake has done the research and implemented prototype SMTs and concluded that they are *not* practical data stores and are inappropriate for blockchain applications. This is due to the heavy caching required for database lookups, which……

Read More

4

Reply

Jannik Bach
5 months ago

> H(null) is a constant value, and so is H(H(null)), etc. etc. Huge chunks of the tree can be cached!

I don't get how this property is important. I think hash reuse of completely empty branches is way more important to calculate the tree efficiently. The hashes up the nodes are calculated as H(H(0) + H(0)) not H(H(0)). So how is this transitive property helpful?

Reply

Simon Jentzsch
over 5 years ago

nice and simple explanaition. But I think the Patricia Merkle Trie used in Ethereum is also capable of proving non-inclusion. It is just a bit more tricky. Because in this case you need to provide a path which may even lead to a different leaf or just until a branch is reached where the hash of the next nibble is NULL.

Reply

See all responses

More from Kelvin Fichter

Kelvin Fichter

Looking at ownership in the EVM

I've been thinking a lot about how smart contracts handle state and how they represent ownership. Ethereum pushed forward a popular model...

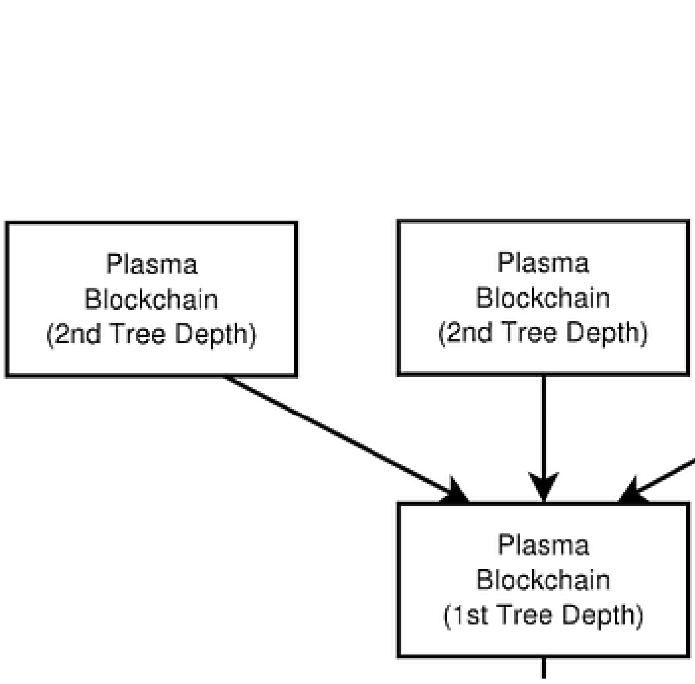
Oct 18, 2018 400 3

Kelvin Fichter

Why is EVM-on-Plasma hard?

Thanks to Dan Robinson for very useful discussions around this stuff.

Aug 1, 2018 1.2K 8


Plasma Blockchain (3rd Tree Depth)
Plasma Blockchain (2nd Tree Depth)
Plasma Blockchain (2nd Tree Depth)
Plasma Blockchain (2nd Tree Depth)
Plasma Blockchain (1st Tree Depth)

Kelvin Fichter

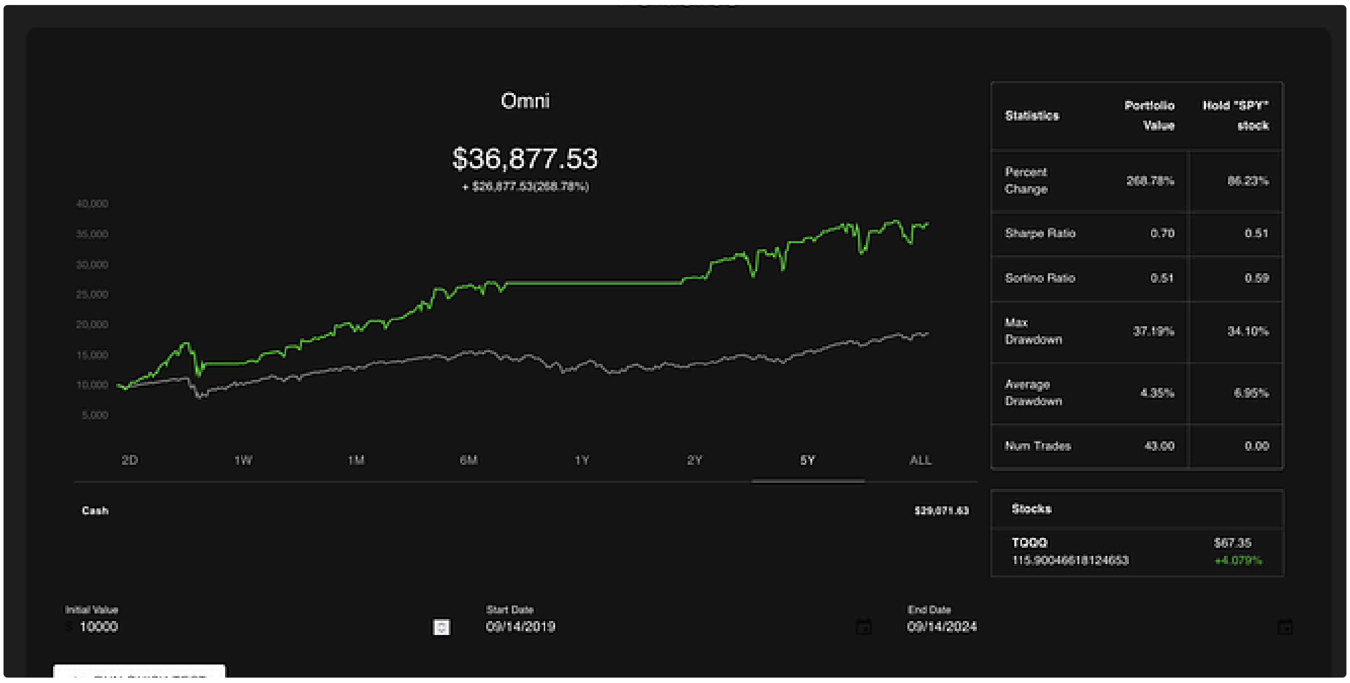
Whatever happened to nested plasma chains?

The idea of “nested plasma chains” was first mentioned in the original plasma white paper. Basically, the proposal was that plasma chains…

Nov 30, 2018 132 2

See all from Kelvin Fichter

Recommended from Medium



Statistics	Portfolio Value	Hold "SPY" stock
Percent Change	268.78%	86.23%
Sharpe Ratio	0.70	0.51
Sortino Ratio	0.51	0.59
Max Drawdown	37.19%	34.10%
Average Drawdown	4.35%	6.95%
Num Trades	43.00	0.00

Cash $29,071.63

Stocks
TQQQ 115.90046618124653 $67.35 +4.079%

Initial Value $ 10000
Start Date 09/14/2019
End Date 09/14/2024

In DataDrivenInvestor by Austin Starks

I used OpenAI’s o1 model to develop a trading strategy. It is DESTROYING the market

It literally took one try. I was shocked.

Sep 15, 2024 8.1K 205

Transaction
(Any length)
Hash Function
keccak256
0x11...
32-byte Hash
value

Swastika Yadav

Merkle proofs explained

With the increasing adoption of blockchain technology, Merkle proofs have become crucial for ensuring data integrity and efficiency. They…

Jul 18, 2024 61

Lists

data science and AI

40 stories · 311 saves

My Kind Of Medium (All-Time Faves)

106 stories · 629 saves

MODERN MARKETING

204 stories · 975 saves

Mohd Anas

Learning Go (Golang)

As a quite new programming language developed by Google, Go—or Golang—aims at providing a more liberal setting for building complex...

Dec 26, 2024 107 8

Kashish Gupta

My Interview Journey at Rippling: A Detailed Account with Code Insights

Interviewing for a role at Rippling was an experience that tested my technical skills, design thinking, and ability to handle complex...

Sep 1, 2024 14

Jessica Stillman

Jeff Bezos Says the 1-Hour Rule Makes Him Smarter. New Neuroscience Says He’s Right

Jeff Bezos's morning routine has long included the one-hour rule. New neuroscience says yours probably should too.

Oct 30, 2024 19K 504

Carlyn Beccia

Trump Fiddles While Musk Burns Down MAGA

Musk unleashes his anger on MAGA Republicans and vows "war."

Dec 28, 2024 18.1K 396

See more recommendations