IN5420 Data Structures

Michael Eikeland

February 2018

Merkle tree

A Merkle tree is a binary tree where the leaf nodes are data blocks labelled with their hash. The parent (branch) nodes are labelled by the hash the concatenated of their children's labels. The Merkle tree provides easy verification for the underlying data strictures. It is used in Bitcoin blocks, where the data is the transactions contained in a block.

Patricia tree

A Patricia tree is the same as a Radix tree where the *radix* is 2. In essence, it means that it is a compressed binary tree. It is compressed in the way that if any node only has one child, the child is merged with the parent.

Ethereum trie

The Ethereum trie is a more complex data structure that encompasses the properties of both Merkle and Patricia trees. While Bitcoin only uses Merkle trees for transactions, Ethereum has different use cases. Particularly for the state data structure that will partially persist and partially mutate relatively frequently. Consequently, using only a Merkle tree, this would mean a lot of recalculations where there is no need for any. Ethereum implements a Merkle-Patricia-tree backed as a *(key, value)* database. The way it is implemented means that only the parts of the tree that needs recalculation will be recalculated. It is deterministic, so as long the structure contains the same data it will give the same Merkle root.

By flagging nodes as blank, branch leaf or extension, it offers some optimization in terms of insertions, look ups and storage.