



链滴

# IPFS 去中心化数据结构（四）

作者：[88250](#)

原文链接：<https://ld246.com/article/1556585434162>

来源网站：[链滴](#)

许可协议：[署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

本文翻译自 IPFS 社区教程 [ProtoSchool](#)。ProtoSchool 是一个可以交互式学习 IPFS 编程的网站，及代码的部分大家可到该网站上直接运行测试！

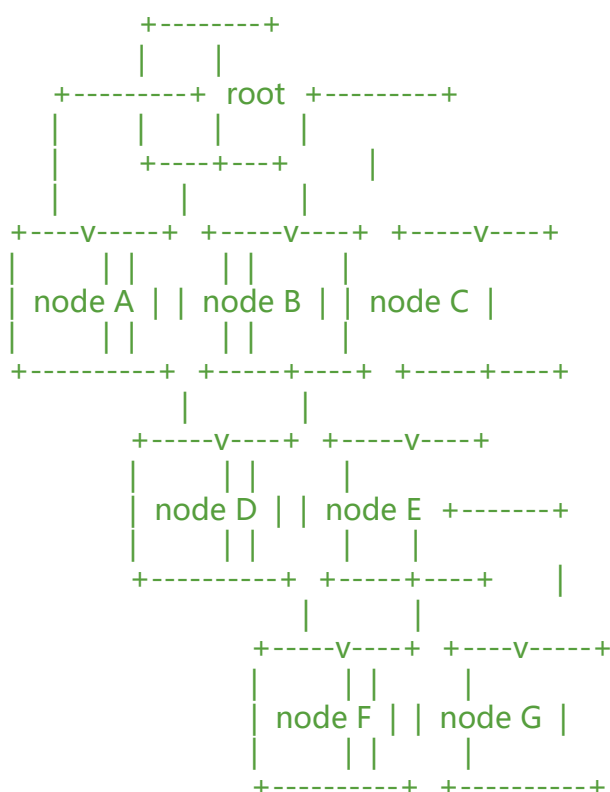
本文接上篇 [IPFS 去中心化数据结构（三）](#)。

## LESSION 5 - 默克尔树和有向无环图 (DAG)

正如我们所讨论的，去中心化的 Web 依赖于链接的数据结构。让我们来探索一下它们的样子。

### 默克尔树

默克尔树即“哈希树”，是一种树形数据结构，其中每个节点都是哈希值。



在默克尔树中，节点通过内容地址（哈希值）指向其他节点。当我们通过加密哈希运算数据后，我们得到一个“哈希值”或“内容地址”，我们可以将其视为一个链接，因此一颗默克尔树是链接节点的合。

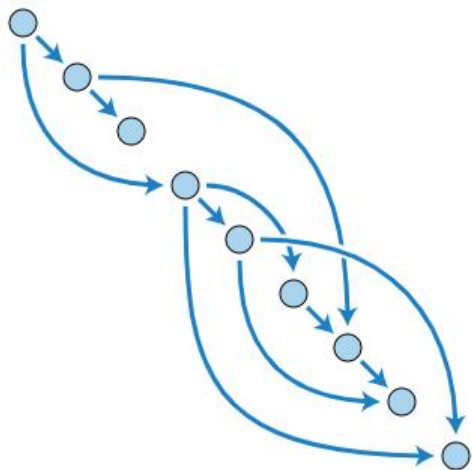
如前所述，所有内容地址都是唯一的。在上图中，节点 E 包含对节点 F 和节点 G 的哈希引用。这意味着节点 E 的内容地址（哈希值）对于包含这些地址的节点是唯一的。

有点晕么？让我们把它想象成一组文件夹。如果我们通过哈希算法运行文件夹 E，同时它包含子文件夹 F 和 G，我们获取的内容生成哈希值对其当前内容唯一的。如果我们删除文件夹 G，就像 Grace 从的小猫照片中删除那个胡须，则以前称为 E 的文件夹不再具有相同的内容，因此它会得到新的哈希值。

在构建上面的树时，根节点的最终内容地址（哈希值）对于包含该树一直向下的每个节点的树是唯一的。如果任何节点中的数据即使是单个字节发生改变，则改变的节点的哈希值也将改变，其所有父节点哈希值都会被改变。

你可能注意到了，程序员将始终需要从底部向上来构建这棵树，从叶节点到根节点。

## 有向无环图 (DAG)



DAG 是 “Directed Acyclic Graph” 的首字母缩写。这是一种描述特定类型的默克尔树（哈希树）的特方式，其中树中的不同分支可以在单个前向方向上指向树中的其他分支，如上图所示。

---

(完)