区块链之P2P&分布式

目录

一.P2P
P2P的概念
P2P的特点
四种网络模型
集中式（中心化）
纯分布式（全分布）
全分布式非结构化拓扑
全分布式结构化拓扑
混合式（半分布）
结构化模型
比特币区块链网络
介绍区块链网络
比特币的P2P网络及节点发现
二.分布式

一.P2P

P2P的概念

点对点技术（peer-to-peer， 简称 P2P ）又称对等互联网络技术，是一种网络技术，依赖网络中参与者的计算能力和带宽，而不是把依赖都聚集在
服务器上。

P2P的特点

1. 非中心化：网络中的资源和服务分布在所有的节点上，每一个节点保存着所有的数据，信息的传输可以直接在节点之间，不需要中间环节的介
2. 可扩展性：用户可以随时加入和退出该网络，系统的资源和服务能力也同步扩充。理论上其可扩展性几乎可以是无限的。P2P 网络中的每个节
是客户端也是服务端，因此也不适合使用 HTTP 协议进行节点之间的通信，一般都是直接使用 Socket 进行网络编程。
3. 健壮性：因为服务是分散在各个节点之间的，部分节点或网络遭到破坏对其他部分的影响很小，故 P2P 具有耐攻击、高容错的特点。P2P 网
分结点失效时能够自动调整整体拓扑，保持其它结点的连通性。
4. 高性价比：P2P 架构可以有效地利用互联网中散布的大量普通结点，将计算任务或存储资料分布到所有结点上。利用其中闲置的计算能力或存
到高性能计算和海量存储的目的。
5. 隐私保护：在 P2P 网络中，由于信息的传输分散在各节点之间进行而无需经过某个集中环节，用户的隐私信息被窃听和泄漏的可能性大大缩小
6. 负载平衡：由于每个节点既是服务器又是客户端，减少了传统 C/S 模型中对服务器计算能力、存储的要求，同时因为资源分布在多个节点，更
整个网络的负载均衡。

四种网络模型

表 1　点对点网络功能及特点对比

比较项	中心化点对点网络	全分布式非结构化点对点网络	全分布式结构化点对点网络	半分布式点对点网络
P2P 应用	Napster	Gnutella	Chord、Pastry	Kazza
区块链应用	无(类似银行)	比特币	以太坊	超级账本
是否去中心化	否	是	是	否
是否可精确查找	否	否	是	否

图1

集中式（中心化）

即存在一个中心节点保存了其他所有节点的索引信息，索引信息一般包括节点 IP 地址、端口、节点资源等。集中式路由的优点就是结构简单、实现
点也很明显，由于中心节点需要存储所有节点的路由信息，当节点规模扩展时，就很容易出现性能瓶颈；而且也存在单点故障问题。

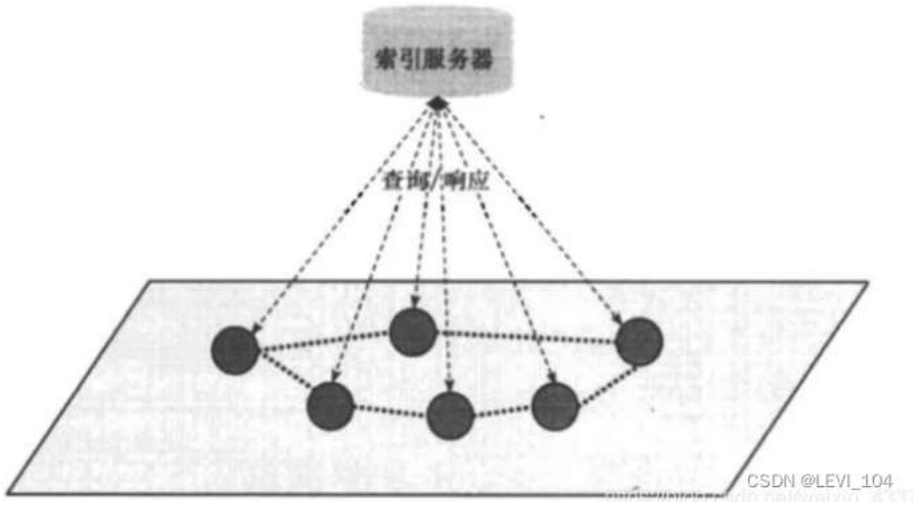

图2

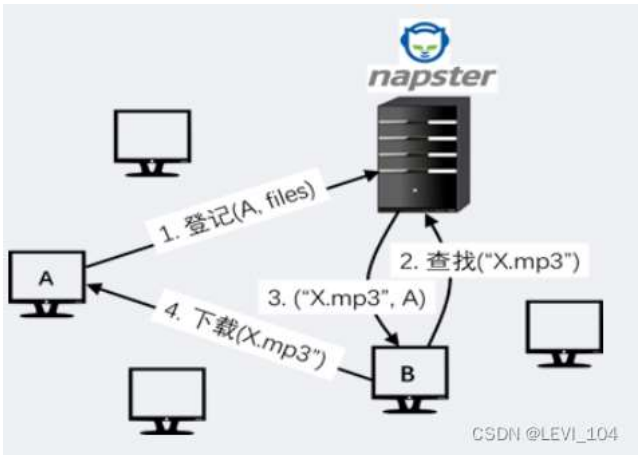

图3

当一个用户需要查找某个音乐文件时，首先需要通过中心索引服务器对音乐文件进行检索，得到拥有该音乐文件的其他用户的信息，接着可以依据
接连接到资源拥有者，实现文件传输和共享。

纯分布式（全分布）

移除了中心节点，在 P2P 节点之间建立随机网络，就是在一个新加入节点和 P2P 网络中的某个节点间随机建立连接通道，从而形成一个随机拓扑结
加入该网络的实现方法也有很多种，最简单的就是随机选择一个已经存在的节点并建立邻居关系。

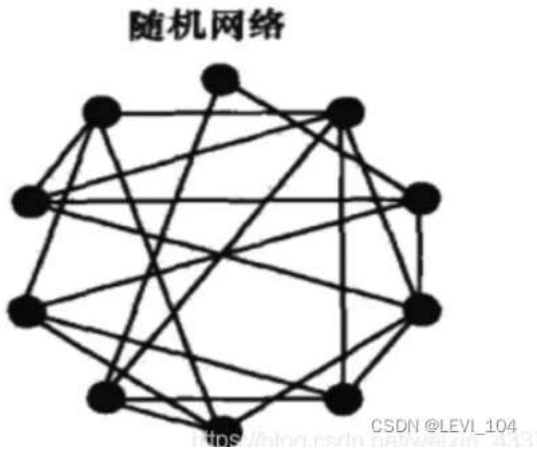

图4

像比特币的话，是纯分布式的P2P网络，是使用 DNS 的方式来查询其他节点，DNS 一般是硬编码到代码里的，这些 DNS 服务器就会提供比特币节
址列表，从而新节点就可以找到其他节点建立连

节点第一次启动的时候，程序并不知道全网任何活动节点的 IP 地址。为了发现这些 IP 地址，程序会请求一个或者多个 DNS 地址(也叫做 DNS 种子点)，这些 DNS 地址都是硬编码到 Bitcoin Core当中的，而且由比特币的社区维护者维护这些域名。比特币程序通过发送 version 消息到远程节点表到一个节点，这个消息会包含本节点的版本信息、区块和当前的时间。远程节点会返回它自身的version 信息，然后两个节点都会发送 verack 信息给表示连接已被建立。连接一旦被建立，本节点会发送 getaddr 和 addr 消息到远程节点来收集比特币网络上更多的节点信息，并且与获取的这些节点P2P连接，默认情况下，一个节点会连接到8个其他节点（链出），并允许多达125个链入节点连接进来，并且对于成功连接的节点，会将这些节点信等)保存到本端DB。

全网广播的方式就是，该节点首先向邻居节点广播，邻居节点收到广播消息后，再继续向自己的邻居节点广播，以此类推，从而广播到整个网络。这也称为泛洪机制。纯分布式结构不存在集中式结构的单点性能瓶颈问题和单点故障问题，具有较好的可扩展性，但泛洪机制引入了新的问题，主要是问题，包括两个较大的问题，一是容易形成泛洪循环，比如节点 A 发出的消息经过节点 B 到 节点 C，节点 C 再广播到节点 A，这就形成了一个循环手问题则是响应消息风暴问题，如果节点 A 想请求的资源被很多节点所拥有，那么在很短时间内，会出现大量节点同时向节点 A 发送响应消息，这节点 A 瞬间瘫痪。

全分布式非结构化拓扑

没有使用中心索引服务器，其节点拥有真正的对等关系
洪泛（Flooding）数据广播，即节点会将接收到的消息向邻居节点转发，直到所有节点都接收到了这个消息或消息传播的深度到达一定的限制。

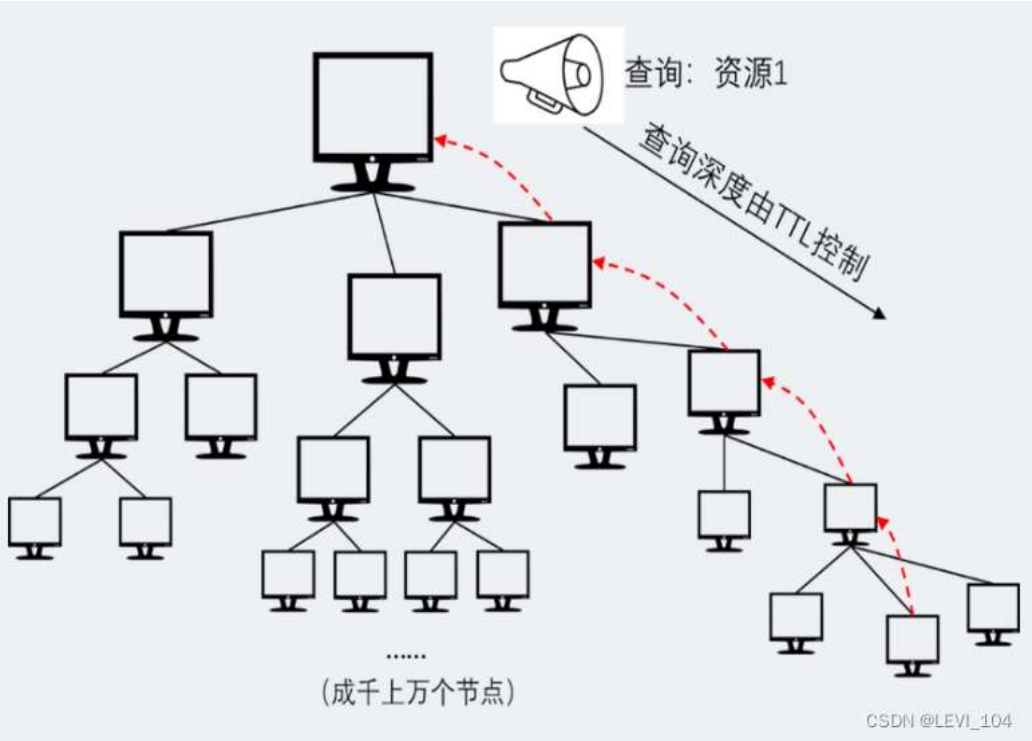

图5

特点：可能会出现广播风暴、实现快速的消息传播和资源查找

寻找过程：首先，节点会根据资源关键字向邻居发送查询请求，如果它的邻居拥有这种资源，则会与发起查询请求的节点建立连接，进行资源的传输个邻居会继续向自己的邻居扩散这个查询请求，直到找到这种资源。

全分布式结构化拓扑

采用分布式散列表（Distributed Hash Tables，简称DHT）来实现整个网络的寻址和存储，从而结构化地址管理。
分布式散列表将存储着网络中所有资源信息的散列表划分成很多不连续的小块，分散地存储在多个节点上。

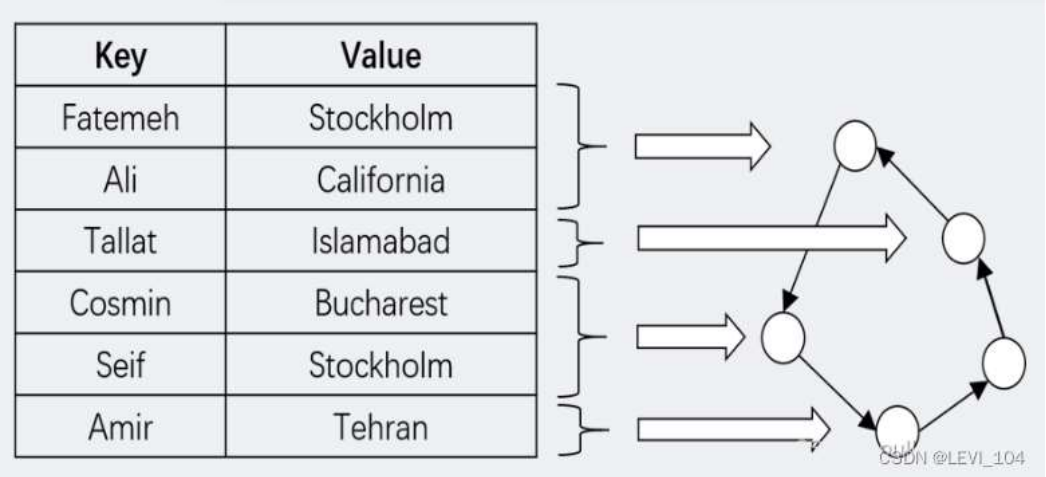

图6

特点:维护机制较为复杂、良好的健壮性、可扩展性和动态适应性

当一个节点需要请求某种资源时，首先找到包含对应资源关键词的散列表所处的节点，从该节点中获取资源对应的地址信息，最后依据地址信息连接实现资源的请求与传输。

混合式（半分布）

混合式其实就是混合了集中式和分布式结构，网络中存在多个超级节点组成分布式网络，而每个超级节点则有多个普通节点与它组成局部的集中式网的普通节点加入，则先选择一个超级节点进行通信，该超级节点再推送其他超级节点列表给新加入节点，加入节点再根据列表中的超级节点状态决定体的超级节点作为父节点。这种结构的泛洪广播就只是发生在超级节点之间，就可以避免大规模泛洪存在的问题。在实际应用中，混合式结构是相对较有效的组网架构，实现难度也相对较小，因此目前较多系统基于混合式结构进行开发实现。

特点：消除了网络拥塞的隐患，并在性能和可扩展性上具有一定的优势 对超级节点的依赖性较大

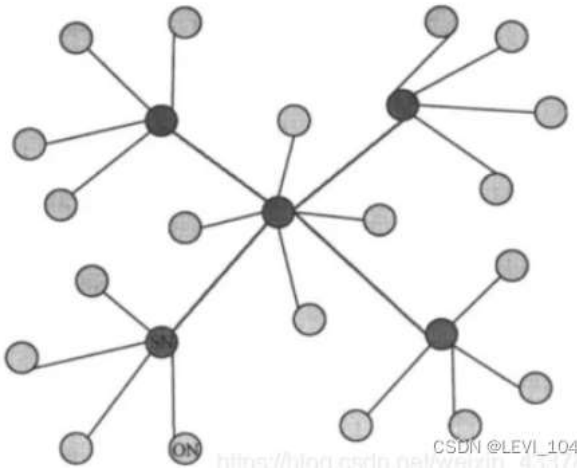

图7

对于半分布式拓扑上的资源查找，会先在普通节点所在的簇内进行，如果簇内的超级节点查询到该资源在与超级节点相邻的叶子节点上，超级节点求转发给对应的节点；否则，超级节点间会进行有限的洪泛，经由其他超级节点继续对这个文件进行查询。

结构化模型

它也是一种分布式网络结构，但与纯分布式结构不同。纯分布式网络就是一个随机网络，而结构化网络则将所有节点按照某种结构进行有序组织，比环状网络或树状网络。而结构化网络的具体实现上，普遍都是基于 DHT(Distributed Hash Table，分布式哈希表)算法思想。DHT 只是提出一种网络涉及具体实现，主要想解决如何在分布式环境下快速而又准确地路由、定位数据的问题。具体的实现方案有 Chord、Pastry、CAN、Kademlia 等算Kademlia也是以太坊网络的实现算法，很多常用的 P2P 应用如 BitTorrent、电驴等也是使用 Kademlia。因为篇幅有限，就不展开讲这些算法的具体前，我们主要理解 DHT 的核心思想即可。

P2P四种模型总结

在 P2P 网络中，可以抽象出两种空间：资源空间和节点空间。资源空间就是所有节点保存的资源集合，节点空间就是所有节点的集合。对所有资源进行编号，如把资源名称或内容用 Hash 函数变成一个数值(这也是 DHT 常用的一种方法)，这样，每个资源就有对应的一个 ID，每个节点也有一个和节点 ID 之间建立起一种映射关系，比如，将资源 n 的所有索引信息存放到节点 n 上，那要搜索资源 n 时，只要找到节点 n 即可，从而就可以避免播，能更快速而又准确地路由和定位数据。当然

关系等，基于这些关系就能建立两者的映射关系。这就是 DHT 的核心思想。DHT 算法在资源编号和节点编号上就是使用了分布式哈希表，使得资源空间的编号有唯一性、均匀分布式等较好的性质，能够适合结构化分布式网络的要求。

比特币区块链网络

介绍区块链网络

首先，比特币网络中的节点主要有四大功能：钱包、挖矿、区块链数据库、网络路由。每个节点都会具备路由功能，但其他功能不一定都具备，不同可能只包含部分功能，一般只有比特币核心(bitcoin core)节点才会包含所有四大功能。

所有节点都会参与校验和广播交易及区块信息，且会发现和维持与其他节点的连接。有些节点会包含完整的区块链数据库，包括所有交易数据，这种**全节点**(Full Node)。另外一些节点只存储了区块链数据库的一部分，一般只存储区块头而不存储交易数据，它们会通过“简化交易验证(SPV)”的方式验，这样的节点也称为 SPV节点或**轻节点**(Lightweight Node)。钱包一般是 PC 或手机客户端的功能，用户通过钱包查看自己的账户金额、管理钱包钥、发起交易等。除了比特币核心钱包是全节点之外，大部分钱包都是轻节点。挖矿节点则通过解决工作量证明(PoW)算法问题，与其他挖矿节点相新区块。有些挖矿节点同时也是全节点，即也存储了完整的区块链数据库，这种节点一般都是**独立矿工**(Solo Miner)。还有一些挖矿节点不是独立挖和其他节点一起连接到矿池，参与集体挖矿，这种节点一般也称为**矿池矿工**(Pool Miner)。这会形成一个局部的集中式矿池网络，中心节点是一个矿其他挖矿节点全部连接到矿池服务器。矿池矿工和矿池服务器之间的通信也不是采用标准的比特币协议，而是使用矿池挖矿协议，而矿池服务器作为再与其他比特币节点使用主网络的比特币协议进行通信。

在整个比特币网络中，除了不同节点间使用比特币协议作为通信协议的主网络，也存在很多扩展网络，包括上面提到的矿池网络。不同的矿池网络可不同的矿池挖矿协议，目前主流的具体矿池协议应该是 Stratum协议，该协议除了支持挖矿节点，也支持瘦客户端钱包。

另外，挖矿这块还有特殊需求。我们知道，矿工创建新区块后，是需要广播给全网所有节点的，当全网都接受了该区块，给矿工的挖矿奖励才算是有后才好开始下一个区块 Hash 的计算。所以矿工必须最大限度缩短新区块的广播和下一个区块 Hash 计算之间的时间。如果矿工之间传播区块比特币那无疑会有很高的网络延迟，所以，需要一个专门的传播网络用来加快新区块在矿工之间的同步传播，这个专门网络也叫比特币传播网络或比特币中(Bitcoin Relay Network)。

比特币的P2P网络及节点发现

比特币开启了区块链时代，任何节点开启客户端后即可实现去中心化可信任的比特币交易。然而当一个全新的节点加入比特币网络时，首先要做的是由于比特币完全去中心化，节点自由加入、退出导致新加入的节点无从获取网络中节点地址从而接入网络。为此，比特币设计三种节点发现方式：

a）种子节点。Napster采用的是中央服务器进行索引，由于中央服务器的存在，新加入节点可以稳定地接入网络。比特币虽然没有中心化服务器，了 Napster的思路，设立“种子”节点。比特币将一部分长期稳定的节点硬编码至代码中，这些节点在初始启动时提供最初接入网络的入口节点。新节稳定节点作为中介连接其他节点，并且可以持续获取区块链网络节点地址列表，所以这些节点也称之为种子节点。下图为比特币源码，方框中这些地币初始化时加载的种子地址。

```
// Note that of those which support the service bits prefix, most only support a subset of
// possible options.
// This is fine at runtime as we'll fall back to using them as a oneshot if they don't support the
// service bits we want, but we should get them updated to support all service bits wanted by any
// release ASAP to avoid it where possible.
vSeeds.emplace_back("seed.bitcoin.sipa.be"); // Pieter Wuille, only supports x1, x5, x9, and xd
vSeeds.emplace_back("dnsseed.bluematt.me"); // Matt Corallo, only supports x9
vSeeds.emplace_back("dnsseed.bitcoin.dashjr.org"); // Luke Dashjr
vSeeds.emplace_back("seed.bitcoinstats.com"); // Christian Decker, supports x1 - xf
vSeeds.emplace_back("seed.bitcoin.jonasschnelli.ch"); // Jonas Schnelli, only supports x1, x5, x9, and x
vSeeds.emplace_back("seed.btc.petertodd.org"); // Peter Todd, only supports x1, x5, x9, and xd
```

图8

另一种方式获得种子节点： 叫做 DNS-seed，又称 DNS 种子节点，DNS 就是中心化域名查询服务，比特币的社区维护者会维护一些域名。比如 seed.bitcoin.sipa.be 这个域名就是由比特币的核心开发者 Sipa 维护的，如果我们通过 nslookup 会发现大约二十多个 A 纪录的 IPv4 主机地址。我们令尝试连接域名下的某个主机的 8333 端口会发现连接成功，运行结果如下。

```
admin@admin  ~  nc -nvv 149.202.179.35 8333
found 0 associations
found 1 connections:
     1: flags=82<CONNECTED,PREFERRED>
 outif en0
 src 192.168.1.104 port 62125
 dst 149.202.179.35 port 8333
 rank info not available
 TCP aux info available
Connection to 149.202.179.35 port 8333 [tcp/*] succeeded!
```

b）地址广播。某个节点接入后，就能以这个节点为中介获取网络其他节点地址，这种方式被称为地址广播。广播包含如下两种方式：

（a）主动广播（推）节点 A地址。比特币通过 Net.AdvertiseLocal()方法，将自身节点信息推送给其他节点。如图9所示，这是一个主动单向过程，收后只保存在本地，并不作回应；同样，节点 B与其他节点间也可以通过推送自身地址传递给其他连接的节点，通过推的方式使得连接节点获得自址信息。

（b）主动获取（拉）其他节点地址。推送自身地址，只局限于自身连接的节点，并不足以将地址广播出去，比特币还有另外一种地址广播机制，通Net.Addresses()方法主动拉取地址。如图9所示，节点 A请求地址后获取节点 B，节点 B返回节点 B收录的地址信息。一次请求一次响应，以避免的浪费。节点 B也可以通过向其他节点请求地址来扩充自身地址库。其他节点主动推送的地址也在拉取动作时返回请求节点，达到地址广播的目的。两种地址获取方式是独立的，这样可以防止攻击者通过伪造大量假的地址并广泛散播，导致正常节点无法接入比特币网络或接入虚假网络。

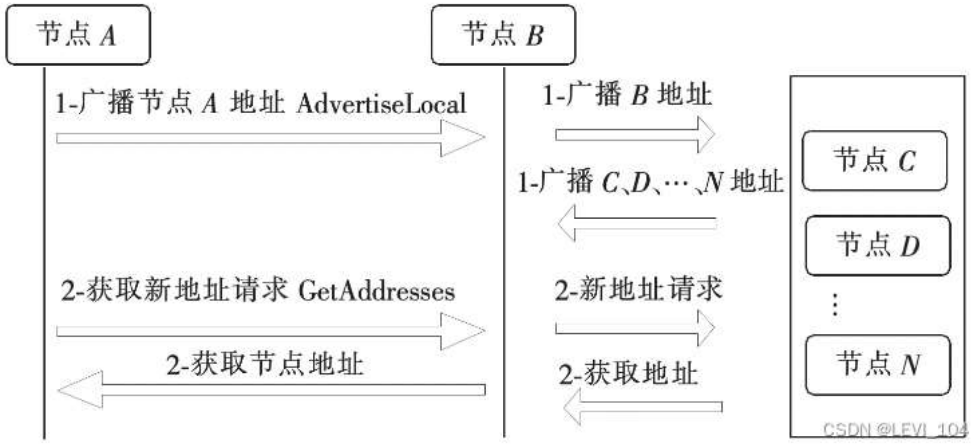

图9

c）地址数据库。为避免种子节点连接数及带宽限制，比特币节点接入网络后通过广播获取其他节点信息，并将结果保存，以便下次使用时接入网络客户端将获取的节点信息保存在 peers.dat本地文件中，使用的是 levelDB格式储存数据。如果节点 A与已建立的连接节点 B没有数据通信，节点 A送 ping消息（一个 8 bit的随机数）到节点 B；节点 B会回复 pong消息，它是一个包含 ping随机数的响应报文，用以维持连接。如果节点持续某个20min没有任何响应，它会被认为已经从网络中断开。比特币客户端启动后会发起定时循环任务，检查连接节点是否在线，并将失败的节点保存在b地文件中。

> 比特币通过以上三种方式保持稳定的网络接入，实现对频 繁进出节点的地址列表维护，不需要中心机构的存在即可保持 节点可以自由加入网络，障比特币 P2P网络稳定。

二.分布式

我的理解：就是和中心化存储信息对立。中心化存储信息是由一个服务器来存储信息，而分布式就是每一个参与的用户都来存储信息数据（可能是完也可能是部分信息），然后各个用户之间相互分享信息数据

区块链中的分布式，可以简单的理解为：

1.**分布式的信息发布与传输**：每个参与系统的节点都可以发起信息，每一个参与的节点可与相邻节点进行信息交互，全网公开传递有价值的信息。

2.**分布式记账**：每个参与系统的节点只要根据共识机制，完成工作量的设定，便能取得数据库的记账权，且记录可以追溯查询，但不可篡改。

3.**分布式储存**：分布式记账后在记录信息加工中加上一个时间戳，便会产生区块数据，网络广播出去后，就会在区块链中形成，每个节点可以选择储据或者是部分数据。而且，每个点都可以拥有一份完整的本地数据储存实时的更新。

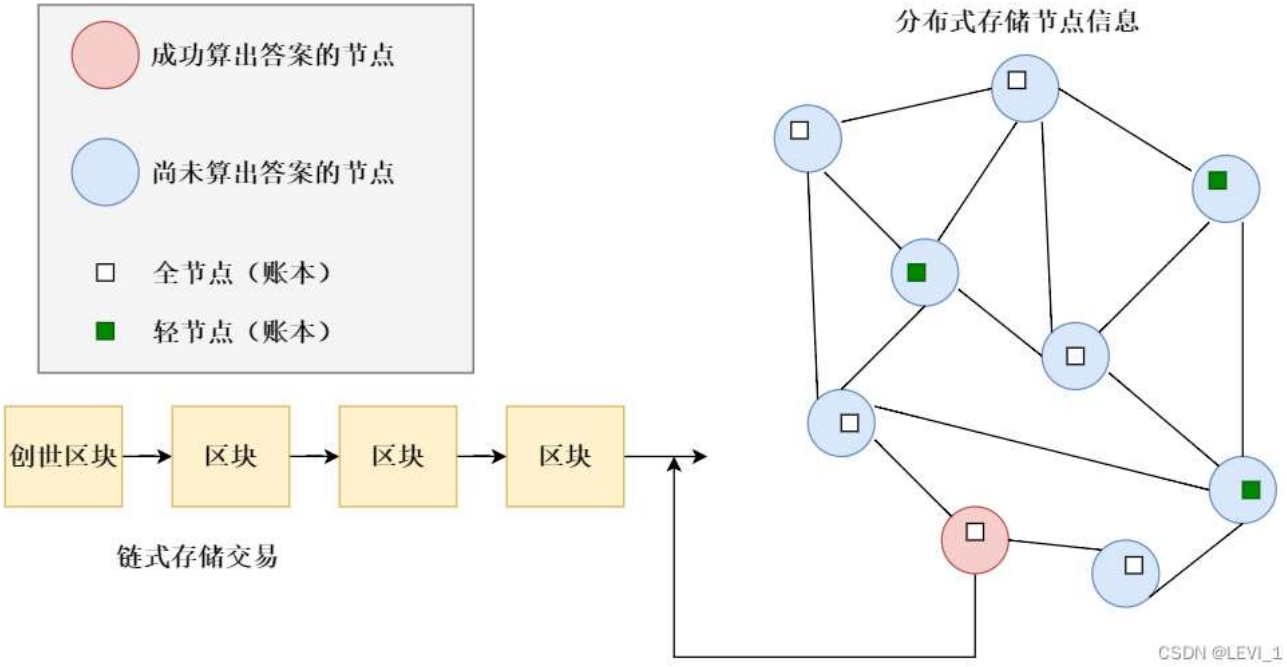

图10

这是我的理解，不知道对不对，不对请联系我