Merkle DAG和Merkle Tree（默克有向无环图）——文件切片求hash，并构建文件与这些hash的有向无环图，可以实现内容寻址，防篡改，去重

DHT（分布式哈希表）——全网维护一张表（可以理解成字典），CID是信息索引的key，查到哪些节点拥有所需的资源，为p2p下载提供资源检索

SFS（文件自验证系统）——它将所有的文件保存在同一个目录下，所有的文件都可以在相对路径中找到，CID即SFS路径名是其原路径与公钥的哈希，并确保ipfs搭建的私有网络的安全性

BitTorrent：块交换协议，通过私有网络内的各个节点进行下载加速

Git：版本控制，使文件在增量更新后，可以访问历史版本

使用的Hyperledger Fabric相关技术：

1帐本（Ledger）

Fabric帐本（Ledger）是一系列有序和防篡改的状态转换的记录，结构由一个区块链构成，并不可变的、有序的记录存放在区块中。

2 状态数据库（State Database）：

状态数据库是存储所有在交易中出现的键值对的最新值，调用链码执行交易可以改变状态数据， LevelDB是默认的内置的数据库。

3 智能合约（Smart contract）

智能合约又称为链码，是在区块链上运行的一段代码，是应用系统与区块链底层交互的中间件，通过智能合约可以实现各种复杂的应用。目前使用Go语言来开发智能合约以外，还可以使用java、node.js开发，支持最好的还是Java语言。

4 通道（Channel）

通道是两个节点（Peer）或多个节点之间信息通信的私有空间，在通道内的交易的数据与通道外隔绝，保证通道内数据的安全。在网络上的交易都要在某个通道（Channel）上执行，参互交易的每个成员都需要进行身份验证和授权，才能在通道（Channel）上进行处理。

5  节点（Peer）

节点（Peer）是区块链的交易处理和账本维护的主体，主要负责参与共识过程和通过执行链码（chaincode）实现对账本的读写操作。节点（Peer）根据功能不同分为背书节点（Endorser peer）和提交节点（Committer peer）；根据通讯不同分为锚节点（Anchor peer）和主节点（Leading peer）。

6 Hyperledger Fabric的身份管理Identity management

Hyperledger Fabric 提供用户识别服务来管理用户的ID和参与者在网络内的授权。访问控制列表可以作为额外的许可层提供特定网络操作授权。比如，一个特定的用户ID可以执行chaincode应用，但是无法部署新的chaincode。

7 Security & Membership Services 安全和成员服务

Hyperledger Fabric 加强了交易网络，所有参与者都有已知的身份。公钥基础设施用来生成与组织，网络组件以及最终用户或客户端应用程序相关联的加密证书。因此，数据访问控制可以在网络和channel层面进行操作和管理。Fabric的“许可”概念，加上channel的存在和功能，有助于解决隐私和机密性至关重要的场景。

同步相关技术：

同步策略是整个文件发送、发布和接收以及保证系统内所有用户数据库一致的关键步骤，本系统中的同步策略基于观察者模式进行设计。观察者队列即为已登录用户列表，同时为便于实现同步中的各项操作，每一台客户机在启动系统后都需要始终运行专门用于处理同步操作的守护进程，直至退出系统。守护进程将持续监听指定端口，完成发送广播消息和接收广播消息的操作。根据客户机的不同在线情况，分别采取以下对应的同步策略：

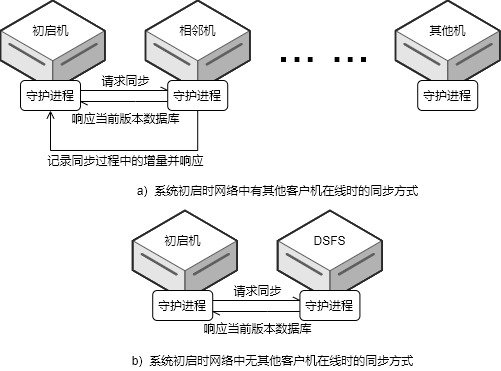


图1 系统初启的数据同步方式

1. 首次使用客户机登录系统或关闭系统后再次启用的，需初始化数据：

（1）检查当前网络中的用户在线情况，若网络中已有其他在线用户，如图1 a)所示，则从网络中与本机逻辑相邻的客户机处获取观察者队列，将本机加入队后广播更新后的队列，转到第（2）步；若当前网络中只有本机一个在线用户，如图1 b)所示，从DSFS上获取所需的数据库文件，并生成观察者队列，将本机入队，转第（3）步。

（2）如图1 a)所示，从与本机逻辑相邻的客户机处获取截止当前时间的数据库文件，相邻主机在传输本机所需的数据文件的同时，将记录截止时间之后发生的数据增量变动，若产生了数据增量，将再次给本机发送造成这些增量的操作详情，本机收到这些增量详情后将在本地执行对应的增量操作，以此防止传输过程中产生数据增量造成数据丢失。

（3）数据初始化结束后，根据已获取的数据展示相关信息和通知。应保持守护进程运行直至退出系统。

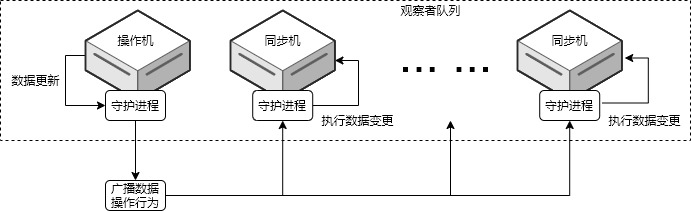


图2 已登录系统的客户机上执行数据同步

2. 已登录系统的客户机上执行数据同步：

（1）如图2所示，本机发生数据增量更新时，将本次更新操作通过守护进程在观察者队列内广播。此处的本地增量更新操作与广播操作应为一个事务，具有原子性，即：若由于网络或其他种种原因导致广播失败，本机的增量更新也应中止或回滚。

（2）其他客户机发生数据增量更新时，本机将通过守护进程收到广播消息，根据此广播消息执行对应的增量更新操作即可。

3. 客户机即将退出系统时的数据同步：

（1）检查观察者队列，若此时队列中还有其他客户机，则广播本机即将下线的消息，其他客户机收到该广播消息后将本机从观察者队列中移除。基于在线时的所有数据增量操作和广播具有原子性，本机无需再进行其他操作，直接完成下线即可。

（2）若观察者队列中除本机外没有其他成员，则需要将本机的数据库文件上传至DSFS保存，以保证此后登录的客户机获取的是最新的数据库文件。数据库文件上传成功后本机方可完成下线。

4. 文件收发时的消息推送：

（1）执行文件发送操作：本机将文件保存到DSFS后更新本地文件收发表，并将文件发送的操作在观察者队列广播。

（2）接收消息、接收文件和审批文件：当前在线的客户机在收到其他客户机的广播消息后，应在本地数据库执行对应操作以更新文件收发表，并检查新记录中本机是否属于审批者或接收者，若是则给出对应提示，否则系统静默；离线的客户机下次登录时直接在初始化数据时同步文件收发表，然后完成后续操作。