字节币是第一个基于CryptoNoto技术，致力于匿名反机枪池的超前的一种货币，2012年就已经发布。市面上有出现过同样名字bytecoin的币种，或许是因为此种缘故，导致其一直不被大众所熟知。字节币的出块奖励主要根据当前出币总量动态改变。出块奖励BaseReward = (MSupply - A)/2的18次方。MSupply = (2的64次方 - 1)， A 等于已经出来的币的数量。特币(Bytecoin)，又名字节币，缩写为BYC，算法采用SHA256，就像一个字节等于八个比特一样，总数量也是比特币的八倍，共1.68亿个。百特币的货币符号是β。百特币每个块产生的货币数量为100个，平均5分钟出一个矿，每840000个块后产生数量减半，即约8年后将产生总量的一半，之后每过8年新产生数量又减半。难度每2016个块(平均一个星期)调整一次。百特币的P2P端口是8888，此端口用于传递交易数据及维持整个货币网络，RPC端口是8889，RPC端口用于挖矿等远程调用。另外百特币也运行了一个测试网络，用于开发者验证程序的功能，测试网络P2P端口采用18888，RPC端口是18889。Byteball(字节球)是一个去中心化的系统，允许任意数据的防篡改存储，包括可转移价值的数据，例如货币，产权，债务，股份等。这些存储单元彼此链接，每个存储单元包括一个或多个早期存储单元的散列值，既用于证实早期的单元又用于确立它们的偏序关系。链接单元之间形成 DAG(定向非循环图)。没有管理或协调新单元进入数据库的单一中心实体，允许每个人添加新的单元，只要他签署并支付的费用等于添加数据字节的大小。其他后来用户通过自己单元内的散列值来确认早期单元，并收取费用。随着新单元的添加，每个早期单元，包括其中的散列值，直接或间接的接收越来越多后来单元的确认。数据库结构当用户想要向数据库添加数据时，他创建一个新的存储单元并将其广播给他的对等节点。存储单元(除了别的以外)还包括：要存储的数据。一个单元可以包括多个数据包，称之为信息。
有许多不同类型的信息，且各有自己的结构。 其中一种信息类型是支付，用于向对等节点发送bytes(字节币)或其他资产。创建单元的一个或多个用户的签名。
用户由其地址标识。 个人用户可以(并且鼓励)拥有多个地址，就像比特币。最简单的情况，地址源于公钥，再次类似于比特币。引用由其哈希值标识的一个或多个先前的单元(父母单元)。引用父母单元是建立单元的次序(目前为止只有部分次序)和推广区块链结构。由于我们不局限于连续块之间的单亲 - 单子关系，所以我们不必争取近同步(性)，并且可以安全地承受大的延迟和高吞吐量：每个单元只会有更多的父母单元和更多的子单元。如果我们沿着父子链在历史上前进，当同一单元被多个后来的单元引用时，我们将观察到许多分叉，并且当同一单元引用多个较早单元时，许多单元逐渐融合(开发者已经习惯看到这个动态)。这种结构在图论中称为有向无环图(DAG)。 单元是顶点，父子链是图的边缘。

连接成一个 DAG 存储单元。箭头是从子单元到父母单元，G
是创始单元在新的单元极少到来这种特殊情况下，DAG 将看起来几乎就像一个链，偶尔分叉而又快速融合。类似于让每个新块确认先前所有块(以及其中的交易)的区块链中，DAG 中的每个新子单元确认其父母单元，父母单元的所有父母单元，父母单元的父母单元的父母单元等。如果有人尝试编辑一个单元，他也必须改变它

的哈希值。 不可避免地，这将破坏所有引用此单元哈希值的子单元，因为子节点的签名和哈希值取决于父哈希值。因此，不能在窃取其私钥或是不与其所有子单元达成合作的情况下修改单元。子单元们不能在没有与子单元合作的情况下修改他们的单元(原单元的孙子单元)，等等这些。 一旦一个单元被广播到网络中，并且其他用户开始在它上面构建它们的单元(将其称为父单元)，编辑这个单元所需的二次修改就如雪球一样增长。这就是为什么我们称之为 Byteball(我们的雪花是数据中的字节)。双花(双重支付)如果用户尝试使用两次相同的输出，有两种可能的情况：1、在尝试使用相同输出的两个单元有序，即一个单元(直接或间接)中包括另一个单元，并且在它之后。在这种情况下，我们显然可以安全地拒绝后面的单元。2、他们之间无序关系，在这种情况下，我们都接受。我们建立单位之间的总序后，当他们隐藏在足够深的新单位里(见下文我们如何怎么做)。在总序较早出现的一方视为有效，另一方视为无效。有一个简化定义总序的协议规则。我们要求，如果相同的地址发布超过一个单元，则它应当(直接或间接地)在每个后续单元中包含其所有先前单元，即来自相同地址的连续单元之间应当有序。换句话说，从同一作者的所有单元应连续。如果有人违反这一规定并发布两个单元，使得它们无序(非序列单元)，则这两个单元被视为双重支付，即使它们没有尝试使用相同的输出。这种非系列单元如上面情况 2所述处理。

图 1双花(双重支付) 它们之间无序如果用户遵循这个规则，但仍尝试两次花费相同的输出，则双重支付变得明确有序，并且我们可以安全地拒绝后一种情况，如上面情况 1 所示。因此，同时不是非序列的双支出容易被过滤掉。这个规则其实很自然。当用户组成一个新的单元时，他选择最近的其他单元作为其单元的母单元。通过把他们列在他的母名单上，他向外宣布他的图片，这意味着他已经看到了这些单元。因此，他看到了母单元里的所有母单元，母单元的母单元等等，直到创世块。这个巨大的集合应该显然包括他自己已经产生的一切，并且他自己可见。用户通过不包含一个单元(甚至是间接地通过父母单元)来否认他看到的这个单元。如果用户通过不包含他自己以前的单元来否认已经看到的这个单元，我们会说这很奇怪，这事有蹊跷。我们不鼓励这种行为。主链我们的 DAG 是一个特殊的 DAG。在正常使用中，人们通常将他们的新单元链接到略小的最近单元上，这意味着 DAG 仅在一个方向上增长。人们可以把它画成一条里面带有许多交错线的粗线。该属性表明我们可以在 DAG 中沿着“子-父链”选择单个链，然后将所有单元关联到此链。所有的单元要么将直接位于这条我们称之为主链的链上，要么沿着 DAG 的边缘从相对少量的跃点到达。它就像一条连接着侧面道路的高速公路。官方网址：https://bytecoin.org/交易平台 Poloniex ：https://poloniex.com/