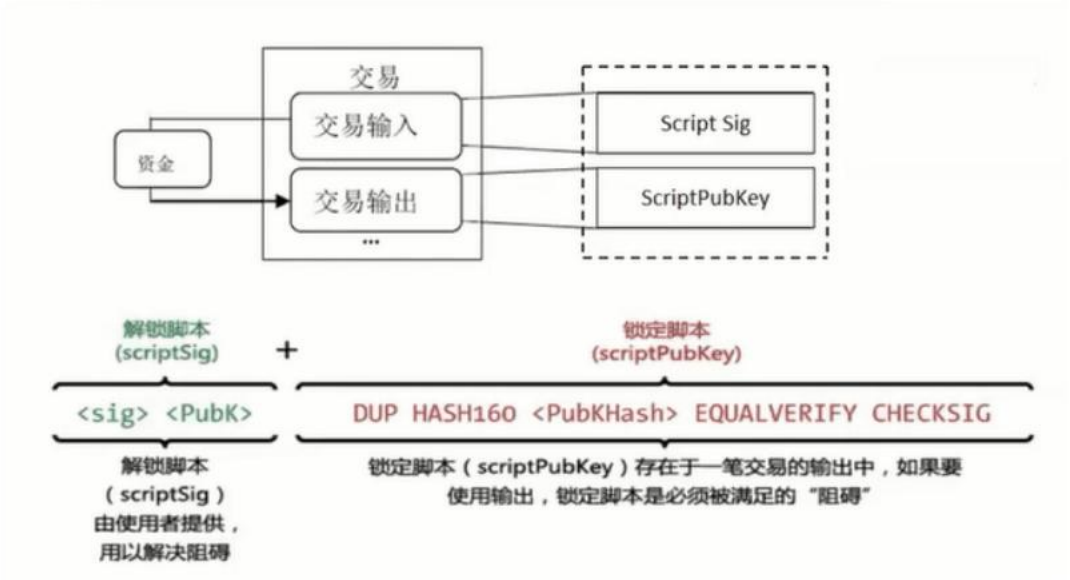
ECDSA 是比特币和以太坊信任基础的核心。其主要用于交易环节的数字签名，在交易过程中，有一个非常重要的信息，那就是需要签名。换言之，一笔交易至少涉及三个方面：付款人、收款人和支付金额。例如张三支付李四时，张三为付款人，李四为收款人，支付金额假设为 5 代币。当张三准备好交易时，需要发送到整个网络。其他人看到交易时，需要核实交易是否由张三发起。在这个环节中，采用了数字签名技术。




交易模块分为：交易输入和交易输出。一个事务由多个输入或输出组成。

交易输入代表支付信息，交易输出代表收款信息。当然，会有付款金额。“付款金额”字段显示在事务输出中。

这种类的交易与我们通常理解的交易非常相似。它的资金流动是从交易输入到交易输出，即从付款人到受款人。

图的右边有一个虚线框。框中的脚本 sig 是脚本签名，属于事务输入。类似地，scriptpubkey 属于事务输出。脚本包含一些数据和操作码，以支持脚本语言的运行。

Scriptsig 和 scriptpubkey 在中文中可以直观地解释为：解锁脚本和锁定脚本。

为什么 scriptsig 也叫 unlock script？这是因为 scriptsig 在前一个连接的事务输出中锁定了一些资金。

如果张三要花费前面的代币，也就是说，要花费一个未使用的事务输出，他需要在另一个事务中构造一个事务输入。在这个事务输入中，将设置签名字段和公钥，以证明张三有资格花费尚未花费的事务输出。

这种情况称为解锁脚本，即打开现有资金使用。

在这个过程中，unlock 脚本中有两个字段数据，sig 和 PubK（缩写为 pubkey）。这两个字段分别表示：签名和公钥。两者都由用户提供，以解决障碍。

Lock 脚本中还有一些字段和操作码：DUP 是复制操作码；hash160 是哈希操作码；是字段；equivalverify 是验证操作码；checksig 也是验证操作码。

它的总体含义是：执行复制——进行哈希——执行字段——验证是否相等——校验签名是否正确。

在签名验证过程中，将调用 ECDSA 签名验证算法。