Conditional Spending

Conditional spending is a proposed feature for cryptocurrencies that allow wallets to put automatic limitations on when or if their coins get spent. These limitations would have to be based on information that is readily accessible to every miner in the blockchain. Examples might be 'if wallet Y falls below N coins, send M coins to wallet Y.' Every node in the network has the ability to check the balance of wallet Y.

Functionally, what will happen is that when a wallet makes a conditional spend, the coins will be removed from the wallet (to prevent double spending / overspending). The conditional spend will have an expiration date (say, X blocks from being announced). If the condition is not filled by the expiration date, then the coins are returned to the sending wallet.

There is an alternative, where the coins are not removed from the triggering wallet until the condition is filled, but then the condition will be invalid if the wallet is empty (or has insufficient funds) at the time of completion of the condition. For important conditional transactions, this method is not recommended, though there is nothing inherently wrong with supporting this type of conditional spend in the protocol.

There are two ways to manage checking conditional spends. The first is to have the network check for the fulfillment of every open conditional spend every block. The second is to have the conditions around, but unchecked unless somebody posts a claim in the condition. (something like "check conditional spend of id 'xxxxxxxxxx...'"). In the first method, a large volume of conditional spends could prove computationally prohibiting to the blockchain. In the second method, any volume of conditional spends is computationally acceptable, and it is assumed that some external party is watching the conditional spends that are interesting to them, and will post a claim on a conditional spend if the conditional is ever fulfilled. There may be additional methods to handle conditional spends, but they have not been explored here.

Conditional spending can be as simple as being able to check the balances of other nodes, or as complex as having an entire turing complete scripting system with an api to all the information that is available to every node. Blockchains could even theoretically support checking external sources for information, such as fetching the real-time exchange rate between the US dollar and the Euro from some trusted source. As the parameters for conditional spending become increasingly complex, greater attention must be paid to security. For example, if you creating a scripting system that is turing complete, you must have some solution to the halting problem. Otherwise, someone could create a conditional spend that will permanently stall every wallet on the network. An example of a solution would be to run all conditional spends in a virtual machine, and to set a limit on how many instruction cycles can be used before a conditional spend is automatically rejected as unfulfilled. If the blockchain relies on an external source, there must be some way to address potential MITM attacks, failures of the external source, and conflicting responses from the external source to the various nodes (as well as a corruption of the external source).