* DNS Server

1. It contains the list of nodes in the network.
2. It contains list of current DHT (Distributed Hash Table) server.

* DHT Server(Task - 1)

1. It stores a Hash Table that maps AccountNumber to its public key hash .
2. Public key hash is stored instead of public key to reduce space consumption by hash table.
3. Each DHT Server maintains list other DHT Servers in the networks.
4. Keys to be stored are distributed among the DHT Servers.
5. Inserting a key.
6. Key’s hash is calculated and the key is stored on the DHT server whose hash is nearest to the key’s hash.
7. Backup of the key is stored to the dht node whose hash is adjacent to the node on which the key is stored.
8. Adding a new DHT Server.
9. If a new node is added to the DHT network then key’s stored are also redistributed accordingly.
10. DHT node get crashed
11. The key’s stored in that node has backup in adjacent 2 nodes so backup key is now stored to the new nearest node and it backup gets stored accordingly.

* Currency Nodes

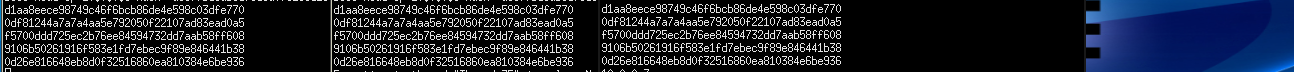
1. When the node gets online it generates its public/private key and a random Account\_Number.
2. It fetches DHT server IP address from DNS Server and sends that node a request to store its public key hash.
3. It also fetches IP of nodes in network from DNS Server and stores in local Array List.
4. 2-Phase Commit Protocol.(Task - 2)
5. Request for a transaction is send to the intermediate node.
6. Intermediate node will send a request to both sender and receiver to acknowledge the transaction.
7. If both sender and receiver acknowledge transaction then it send a commit message to both send and receiver.
8. On getting commit message from intermediate node sender broadcast the transaction to all other nodes in the network.
9. Transaction Format (JSON Format) (Task - 3)



scriptSig property above here is our digital signature.

scriptSig = publickeyhash is encrypted by RSA algorithm using private key of the sender.

1. Broadcast and verification (Task-4)
2. When a node receives a transaction it verifies its digital signature by decrypting scriptSig using sender’s public key.
3. If found correct it adds transaction to its transaction queue.
4. Hash of transaction list (Task - 5)
5. All the nodes will print their transaction list.



1. Block Chain [To prevent double spending] (Task – 6 )
2. Each block contains 2 transactions.
3. Once a node has more than 2 transactions in queue it will start mining the block for first 2 transactions.
4. Mining Block: - Is just a trial and error hash puzzle.
5. The block who find the nonce which satisfy the hash puzzle broadcast the block to others and other nodes verify the nonce if everything found correct add that to its block chain.
6. Double spend transaction might create temporary fork in the block chain but the next block mined breaks the tie and longest block chain is always selected as confirmed transaction and such a path cannot contain two transaction having using same output.