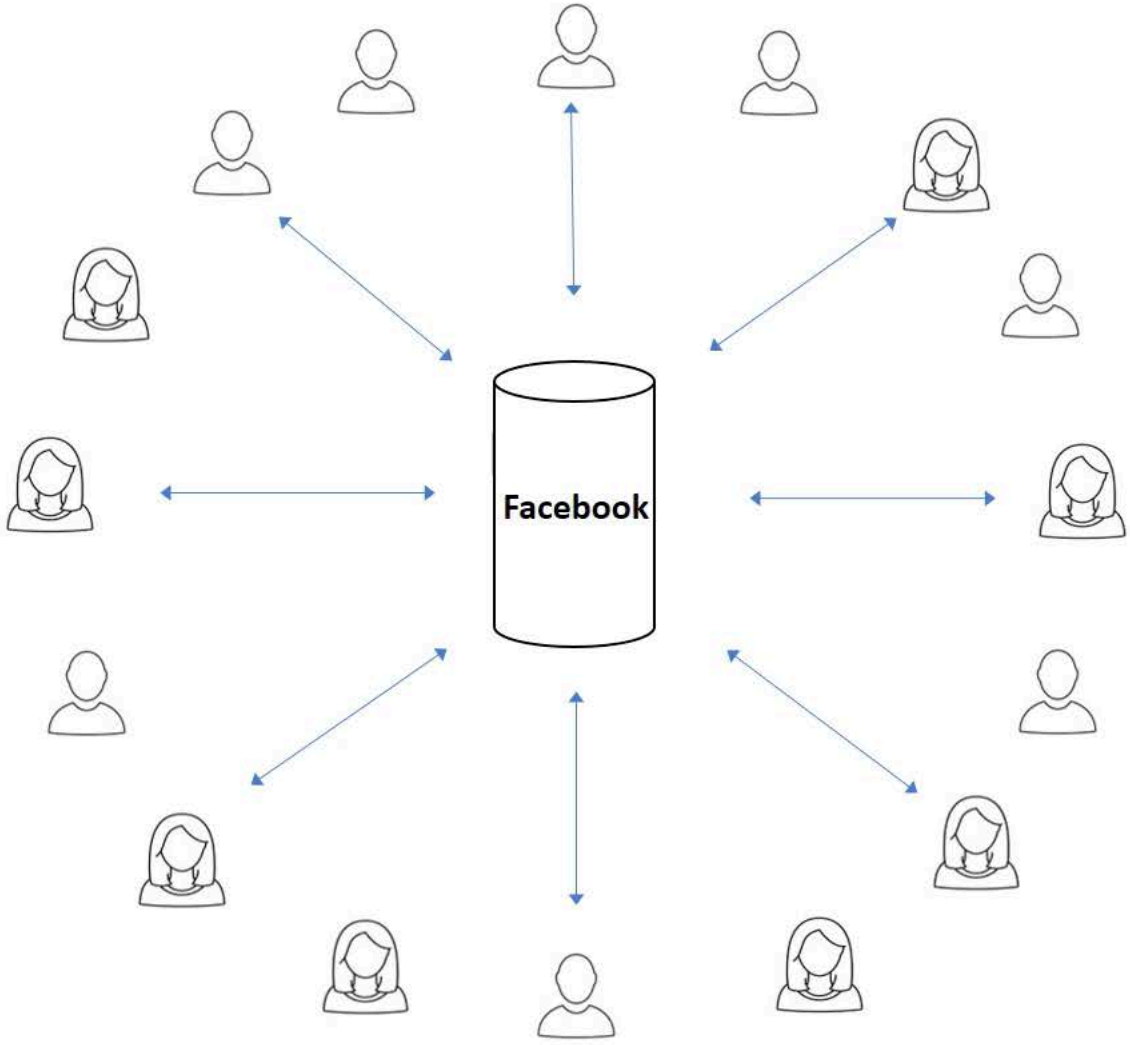
Centralization
Example: Facebook

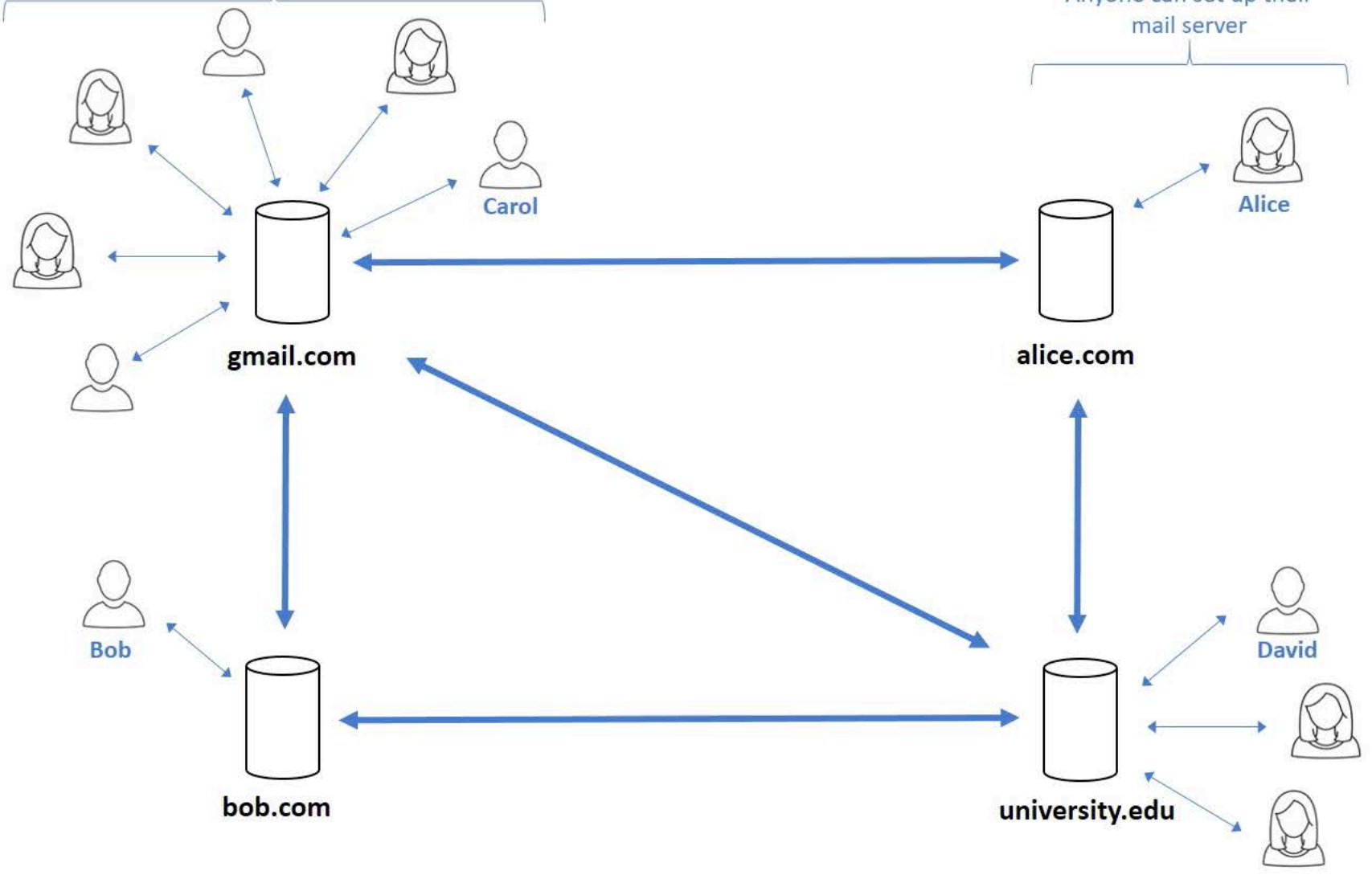


Facebook has full control over data, access, and communication.

Decentralization
Example: SMTP Email

Centralization:
Small number of centralized webmail providers are dominant: Gmail has 1 billion users.

Decentralization:
Anyone can set up their mail server



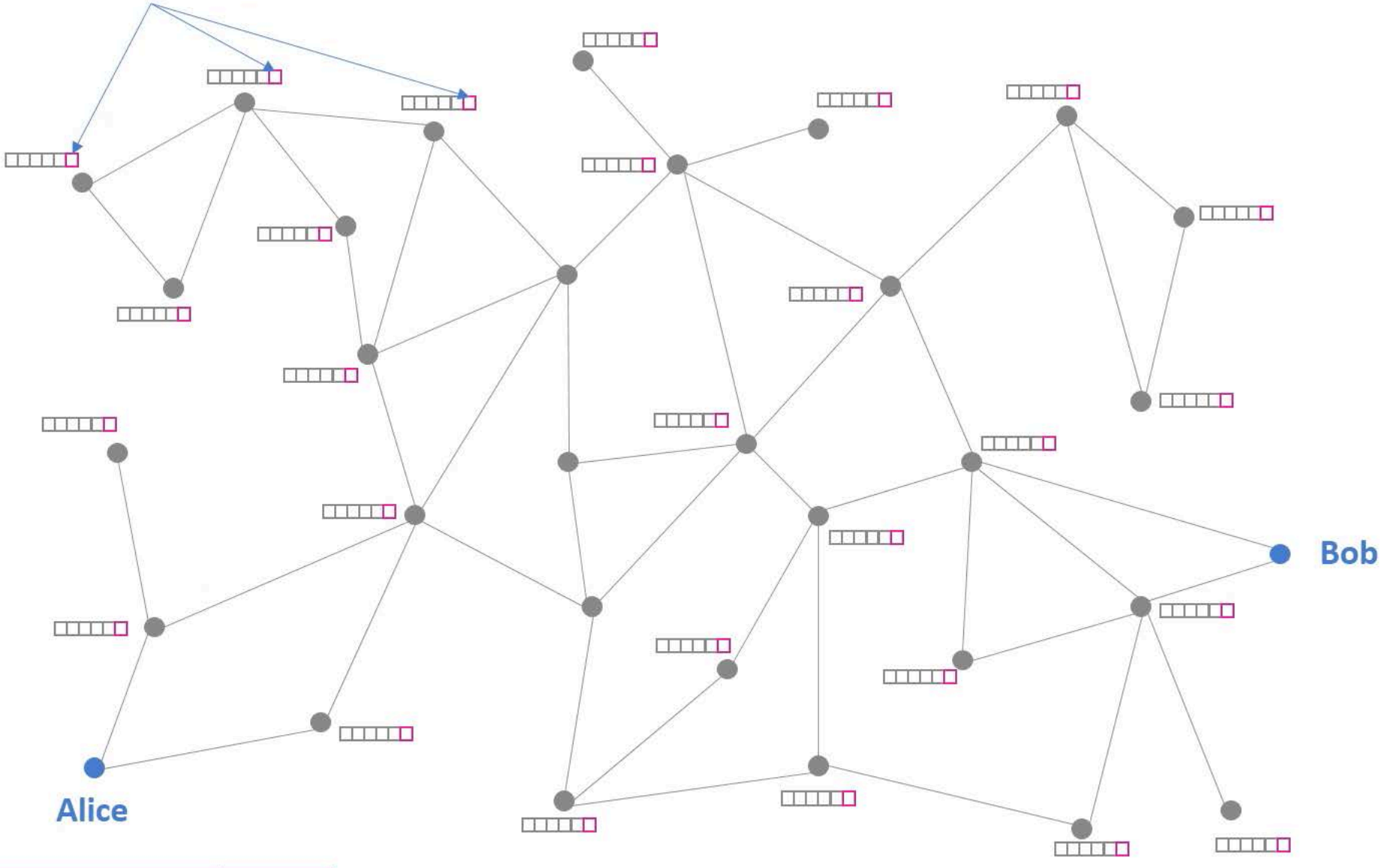
Systems fall on different points on the centralization/decentralization spectrum

Decentralization
Bitcoin

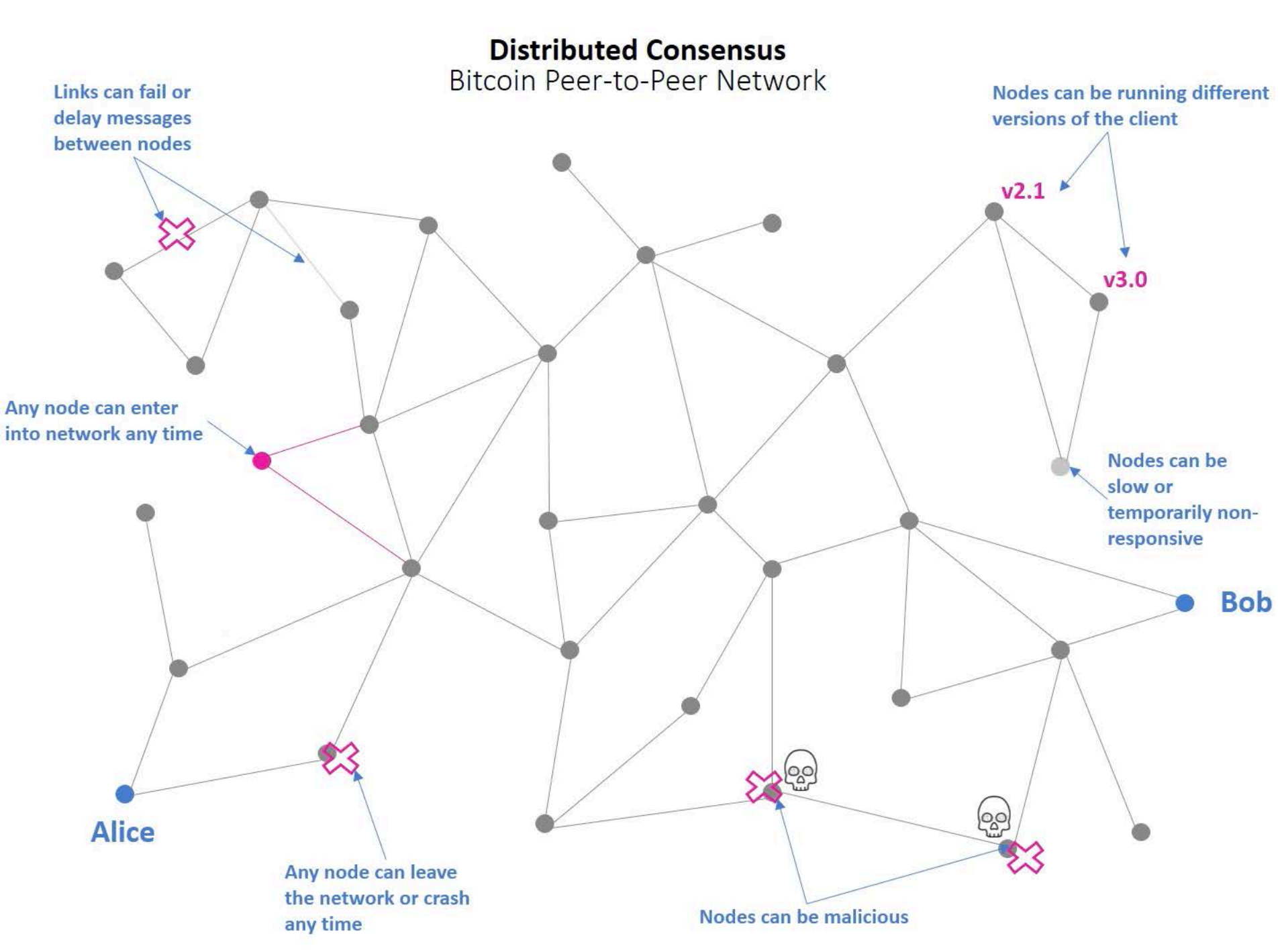
1. Who maintains the ledger of transactions?
2. Who has authority over which transactions are valid?
3. Who creates new bitcoins?
4. Who determines how the rules of the system change?
5. How do bitcoins acquire exchange value?

Distributed Consensus

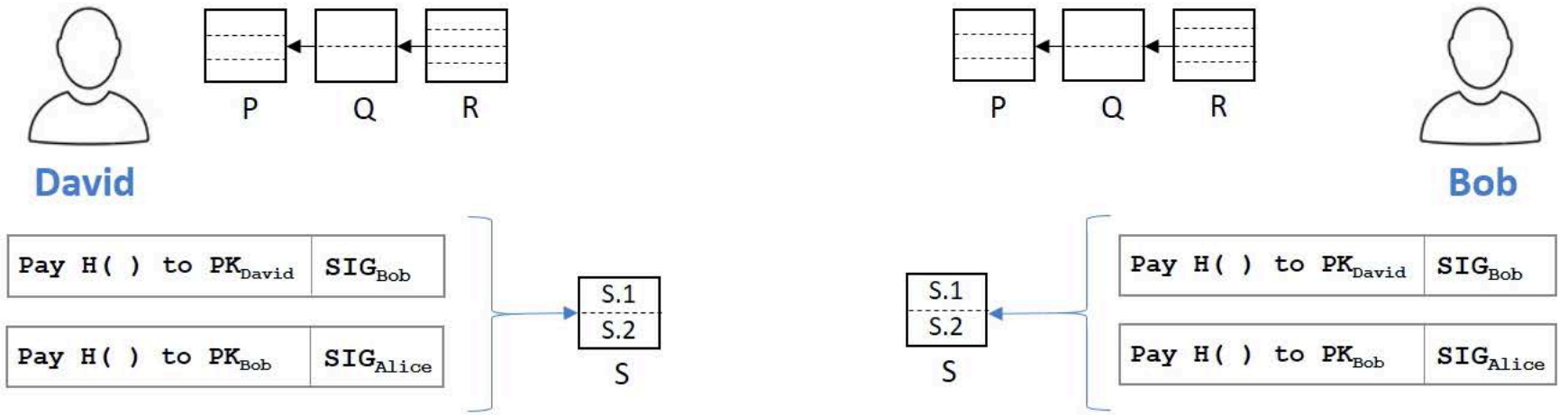
Bitcoin Peer-to-Peer Network



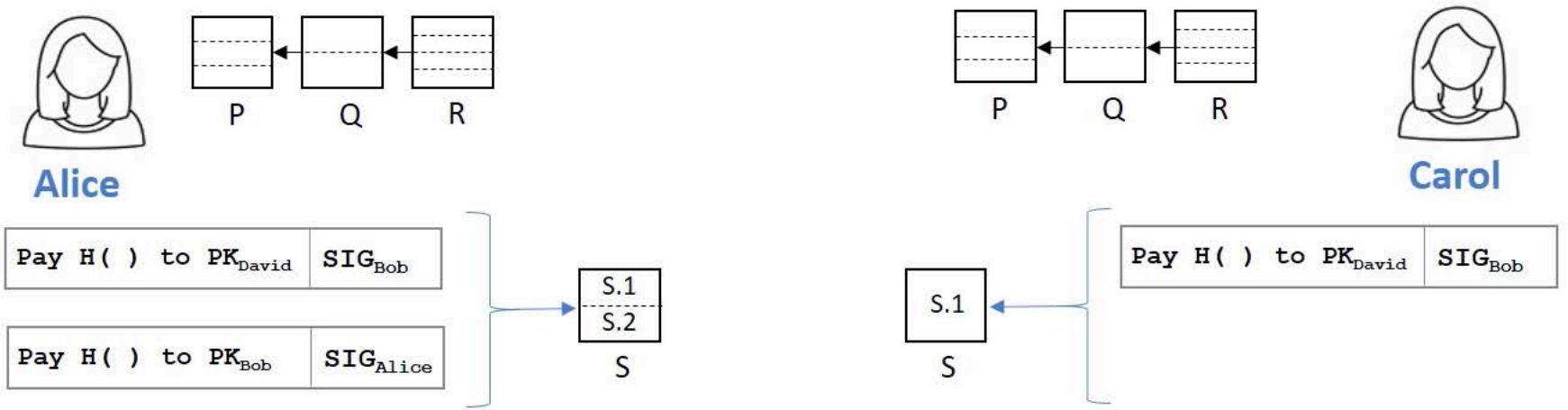
Pay H( ) to PK_{Bob}	SIG_{Alice}

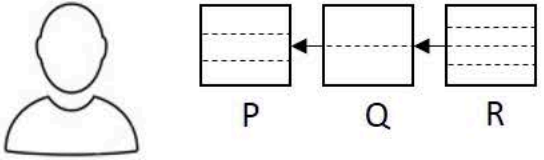

Distributed Consensus
Bitcoin Peer-to-Peer Network
Links can fail or delay messages between nodes
Nodes can be running different versions of the client
v2.1
v3.0
Any node can enter into network any time
Nodes can be slow or temporarily non-responsive
Bob
Alice
Any node can leave the network or crash any time
Nodes can be malicious

Bitcoin Consensus Protocol



Step 2: Each node creates its own block with its received transactions.

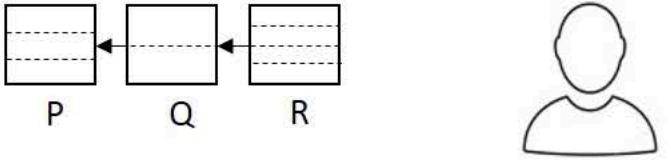




David

Pay H( ) to PK_{David}	SIG_{Bob}

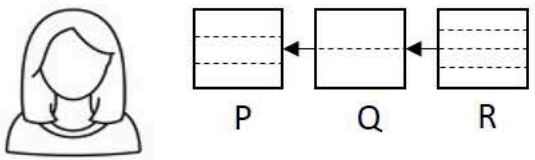
Pay H( ) to PK_{Bob}	SIG_{Alice}



Bob

Pay H( ) to PK_{David}	SIG_{Bob}

Pay H( ) to PK_{Bob}	SIG_{Alice}



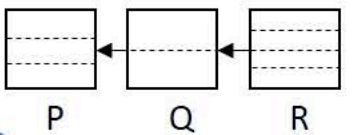
Alice

Pay H( ) to PK_{David}	SIG_{Bob}

Pay H( ) to PK_{Bob}	SIG_{Alice}

Carol did not receive Alice's transaction due to network latency

Round 4

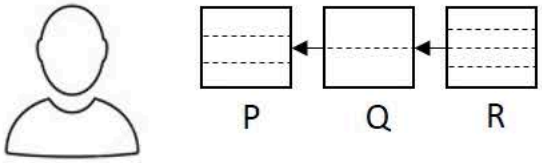


Carol

Pay H( ) to PK_{David}	SIG_{Bob}

?

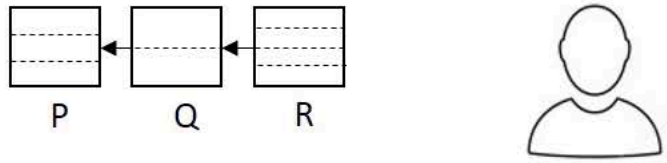
Bitcoin Consensus Protocol



David

Pay H( ) to PK_{David}	SIG_{Bob}

Pay H( ) to PK_{Bob}	SIG_{Alice}

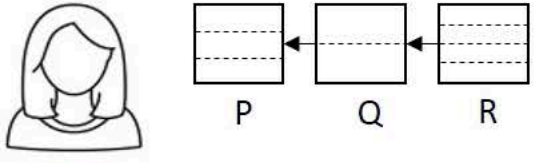


Bob

Pay H( ) to PK_{David}	SIG_{Bob}

Pay H( ) to PK_{Bob}	SIG_{Alice}

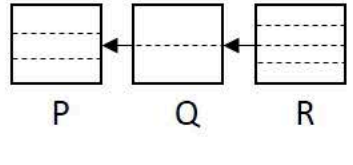
Step 3: A random node gets to dictate the next block by broadcasting it to the network



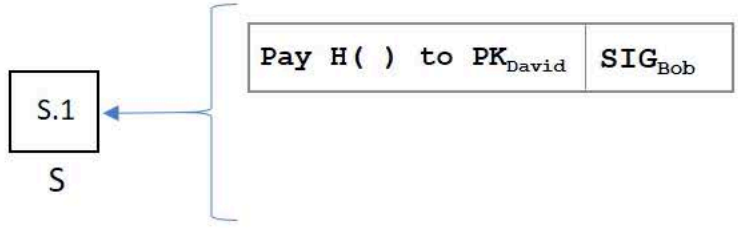
Alice

Pay H( ) to PK_{David}	SIG_{Bob}

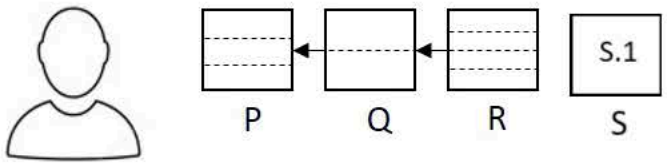
Pay H( ) to PK_{Bob}	SIG_{Alice}



Carol



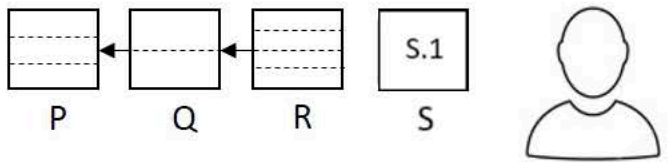
Bitcoin Consensus Protocol



David

Pay H( ) to PK_{David}	SIG_{Bob}

Pay H( ) to PK_{Bob}	SIG_{Alice}

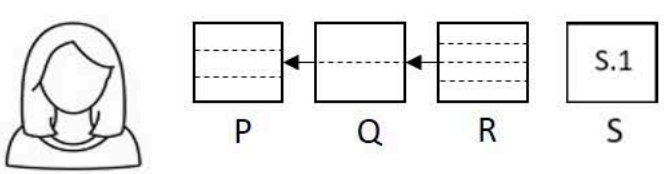


Bob

Pay H( ) to PK_{David}	SIG_{Bob}

Pay H( ) to PK_{Bob}	SIG_{Alice}

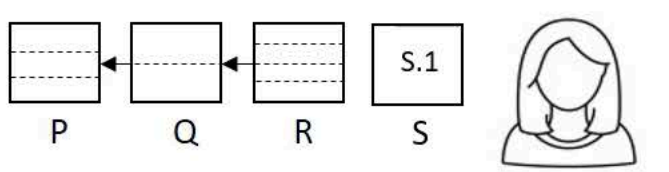
Step 4: Each nodes that receives the broadcasted block add the block its ledger, if the block is valid.



Alice

Pay H( ) to PK_{David}	SIG_{Bob}

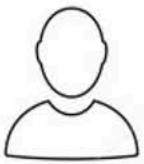
Pay H( ) to PK_{Bob}	SIG_{Alice}

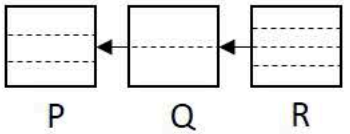


Carol

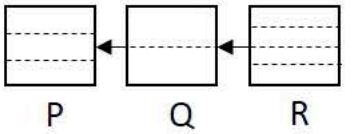
Pay H( ) to PK_{David}	SIG_{Bob}

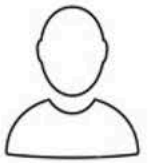
Bitcoin Consensus Protocol
Double-spending Attack





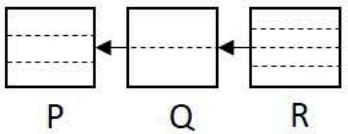
David





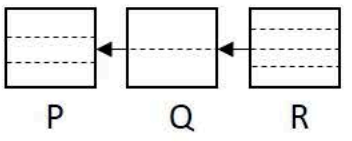
Bob

Alice wants to buy a diamond from Bob



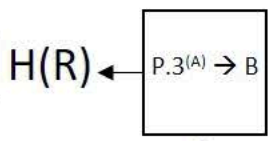
Alice

$P.3^{(A)} \rightarrow B$



Carol

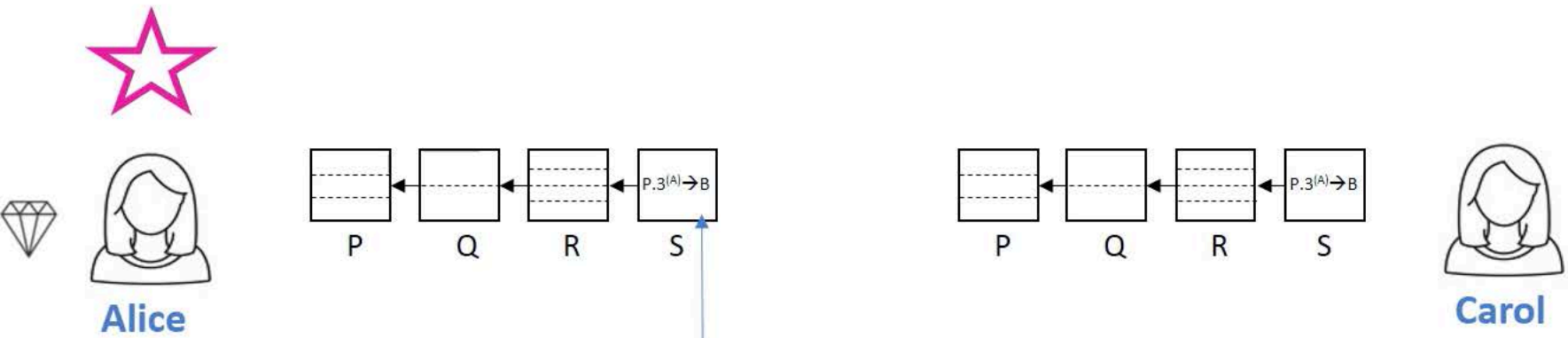
Hash pointer to previous block R



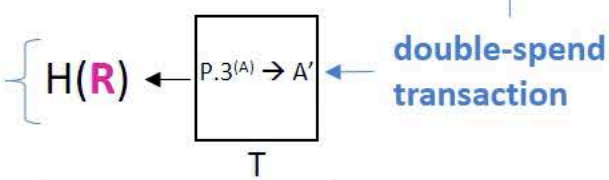
Carol's proposed block S

Round 4

Bitcoin Consensus Protocol
Double-spending Attack

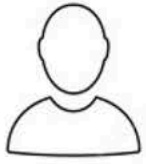


Alice sets block T's previous hash pointer to block R, instead of block S

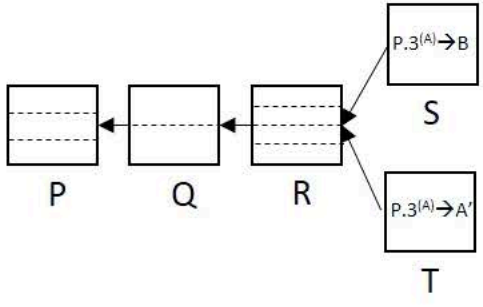


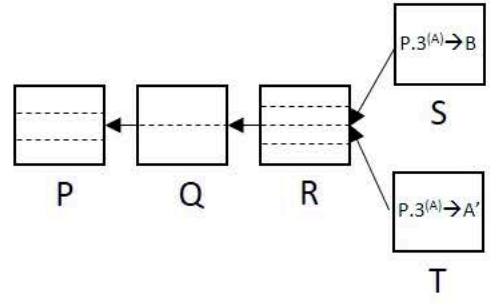
Round 5

Bitcoin Consensus Protocol
Double-spending Attack



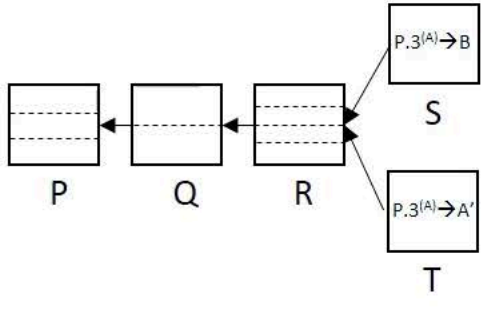
David



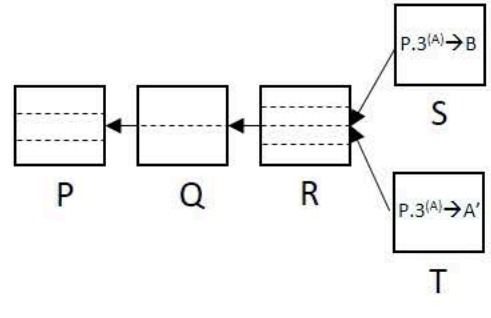


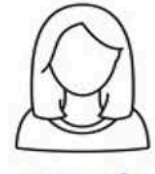
Bob

Alice

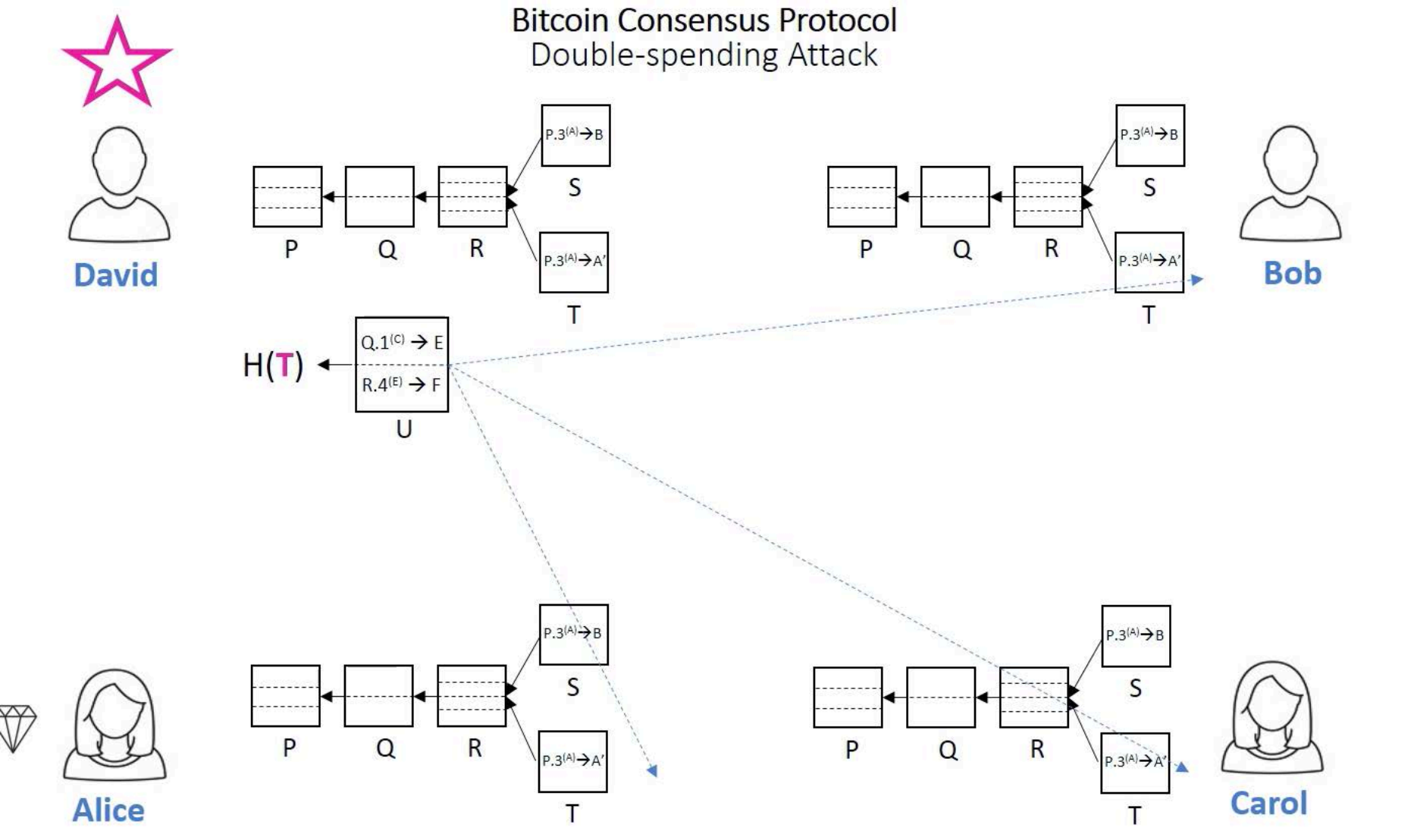


Fork



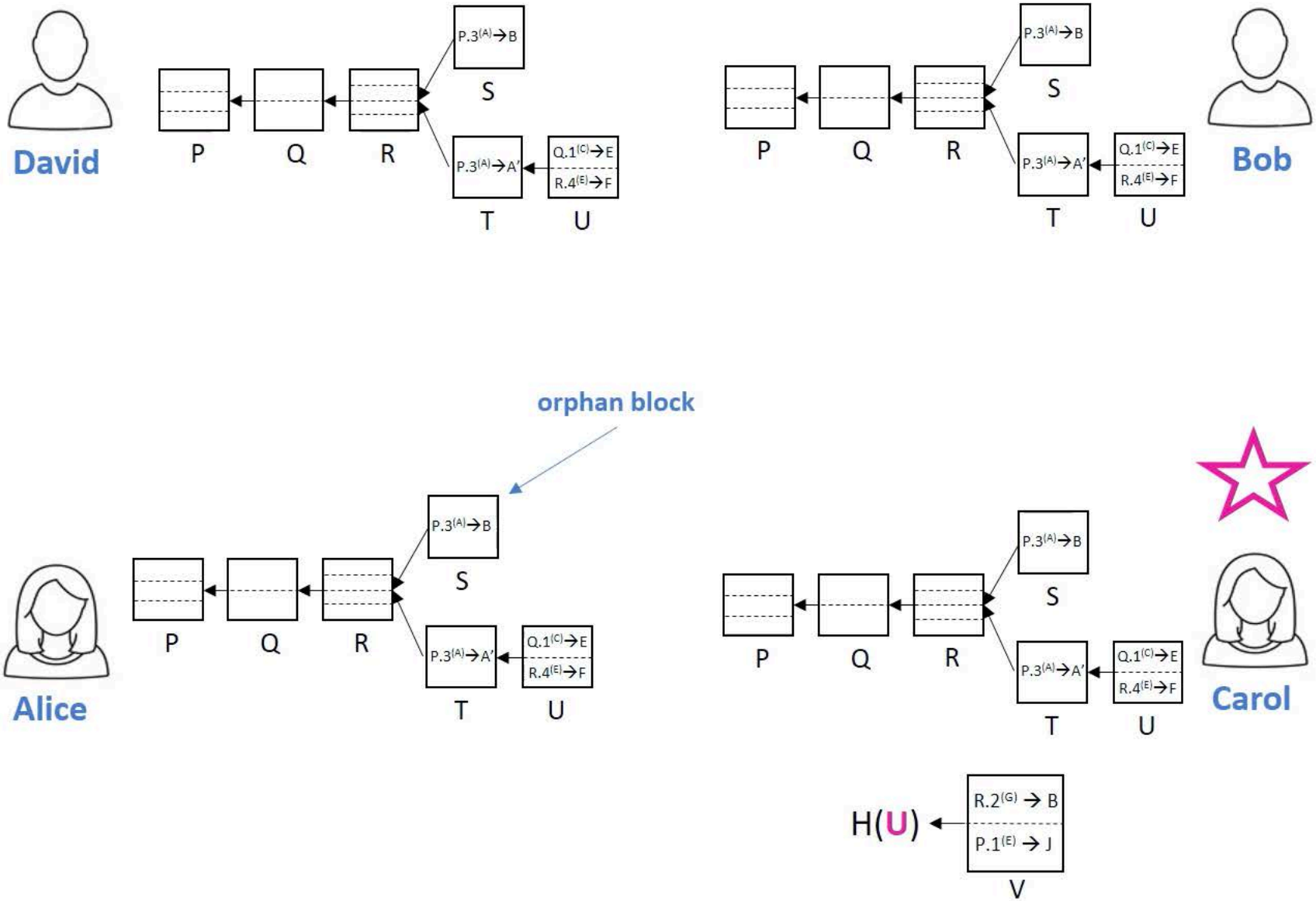


Carol


Bitcoin Consensus Protocol
Double-spending Attack
David
Bob
Alice
Carol
P
Q
R
S
T
U
P.3(A)→B
P.3(A)→A'
H(T)
Q.1(C) → E
R.4(E) → F

Round 6

Bitcoin Consensus Protocol
Double-spending Attack



Round 7