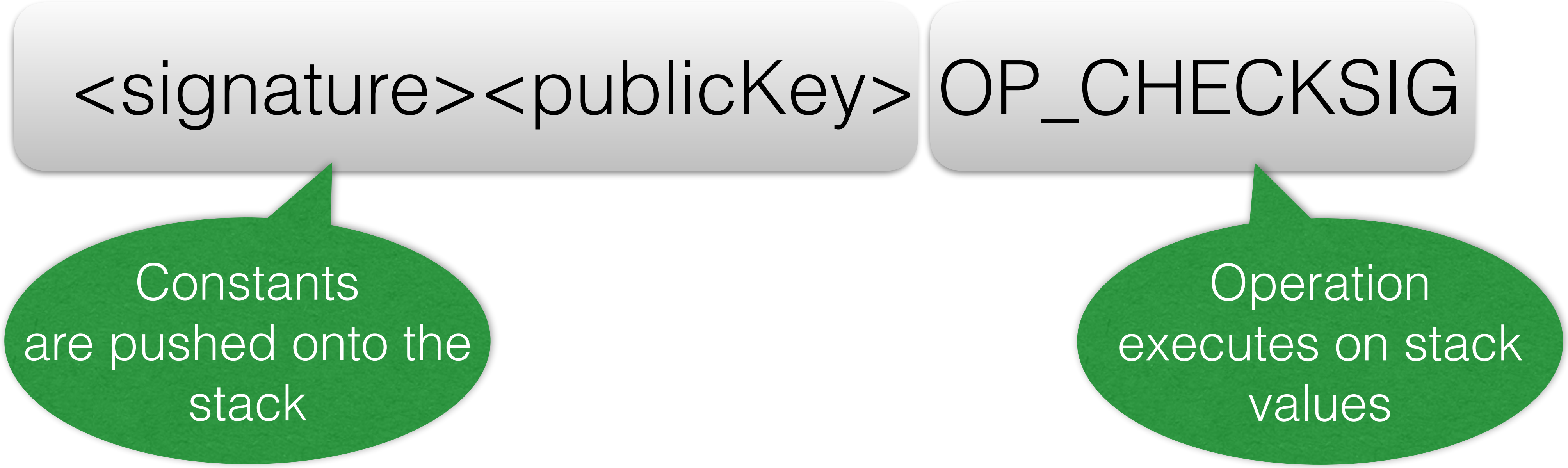
Bitcoin
Script

Script

- Stack based programming language
- If evals to *true* —> Bitcoin transaction is valid
- Many opcodes
- Execution time is critical to prevent DoS attacks

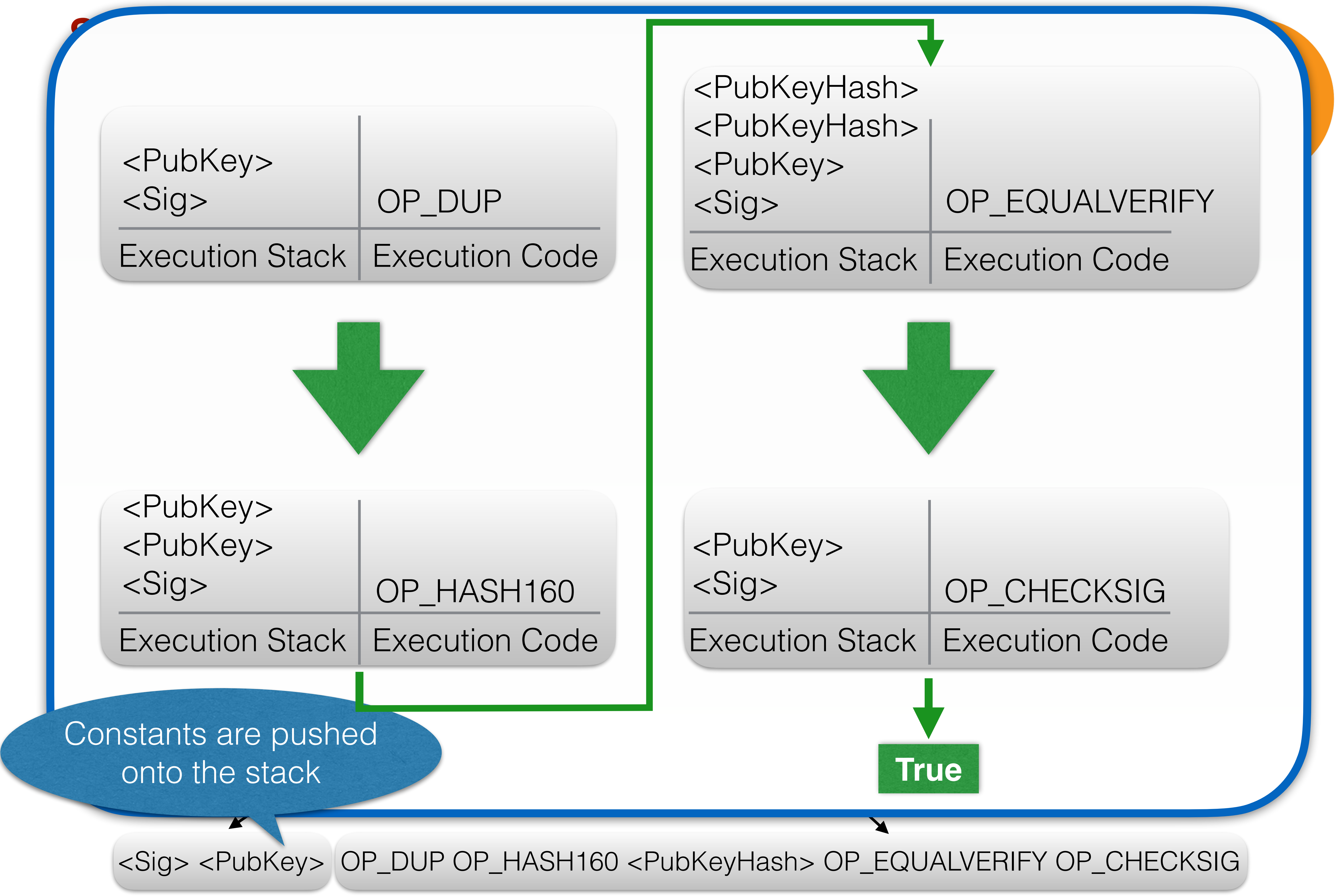
Example Script



<signature><publicKey> OP_CHECKSIG

Constants are pushed onto the stack

Operation executes on stack values



<PubKey> <Sig>	OP_DUP
Execution Stack	Execution Code

<PubKey> <PubKey> <Sig>	OP_HASH160
Execution Stack	Execution Code

<PubKeyHash> <PubKeyHash> <PubKey> <Sig>	OP_EQUALVERIFY
Execution Stack	Execution Code

<PubKey> <Sig>	OP_CHECKSIG
Execution Stack	Execution Code

Constants are pushed onto the stack

<Sig> <PubKey> OP_DUP OP_HASH160 <PubKeyHash> OP_EQUALVERIFY OP_CHECKSIG

Transaction Types

- P2PKH - Pay to Public Key Hash
  - Redeemer needs a public key and signature
- P2SH - Pay to Script Hash
  - Redeemer needs a script that matches a pre-defined hash
- Multisignature (m-n)
  - Requires multiples signatures to be redeemable
  - **m** out of **n** signatures required