

Regular Transaction  Regular Transaction  Contract Deployment thansactions  Execution of a contract.  Etherium State Transaction Function  Stak  Data, Fransaction Function  Thereof I I I I I I I I I I I I I I I I I I I	Regular Transaction  Contract Deployment transactions  Execution of a contract.  Etherium Stack Transaction Function  Stack  Data,  Promsaction  From:  To  Volu:  Deply (Sd, tx) —75'  Etherium Stack transition function Apply (S, Tx) >5'  Etherium Stack transition function Apply (S, Tx) >5'  Etherium Stack transition  Tous > (competition  Transaction > 2000 competin  ———————————————————————————————————		
Regular Transaction  Contract Deployment transactions  Execution of a contract.  Etherium State Transaction Function  Stack  Data, Transaction Function  Stack  Data, Transaction Function  Prom:  Walu:  Data:  Stack  Data:  Dat	Regular Transaction  Construct Deployment thansactions  Execution of a construct.  Etherium State Transaction Function  State  Data,  Thomas toom  Thomas toom  Therefore  There		
Regular Transaction  Contract Deployment transactions  Execution of a contract.  Etherium State Transaction Function  Stack  Date,  Thomsaction  Tho	Regular Transaction  Contract Deployment thansaction  Execution of a Contract.  Etherium State Transaction Function  State  Data,  Thom:  Thom:  Therefore  Therefore	and the	Types of Transactors
Contract Deployment thansactions  Execution of a contract.  Etherium Stark Transaction Function  Stark  Data,  Phorm:  Thom:  Value:  Distar:  Sign:  Ppply (So, tx) — S')  Etherium Stark transition function Apply (S, Tx) > S	Contract Deployment transaction  Execution of a contract.  Etherium State Transaction Function  State  Data,  Pressaction  From:  Town:  Value:  Doda:  Sign:  Phyly (Se, tx) — 75   Etherium State transition function Apply (S, Tx) > 5  Etherium State transition function Apply (S, Tx) > 5  Etherium State transition  Toras > 1 competition  Toras > 1 competition  Transaction > 2000 competin  Transaction > 2000 ges to 5 Gas poo by the of date 37  Pervent all State changes  Tas to the mines's account:  Pefrund the fee all remaining gas to senter		
Contract Deployment thansactions  Execution of a contract.  Etherium Stark Transaction Function  Stark  Data,  Phorm:  Thom:  Value:  Distar:  Sign:  Ppply (So, tx) — S')  Etherium Stark transition function Apply (S, Tx) > S	Contract Deployment transaction  Execution of a contract.  Etherium State Transaction Function  State  Data,  Pressaction  From:  Town:  Value:  Doda:  Sign:  Phyly (Se, tx) — 75   Etherium State transition function Apply (S, Tx) > 5  Etherium State transition function Apply (S, Tx) > 5  Etherium State transition  Toras > 1 competition  Toras > 1 competition  Transaction > 2000 competin  Transaction > 2000 ges to 5 Gas poo by the of date 37  Pervent all State changes  Tas to the mines's account:  Pefrund the fee all remaining gas to senter		· Regular Transchina
Etherium State Transaction Function  State  State  Data,  Data,  Data,  Dota,  Thomsaction  From:  To  Value:  State  Papely (So, tx) — 75'  Etherium State transition function Apply (S, Tx) — 55'  Etherium State transition  To as — (compatition  Transaction — 2000 compatin  — 2000 ges (b) S Gas poo byte of  Seven all State changes  Fas to the mines's account.	Etherium Stake Transaction Function  Stak  Data,  From:  Town:  Papely (Sd, tx) -> 5   Etherium Stake transition function Apply (S, Tx) -> 5  Etherium Stake transition function function Apply (S, Tx) -> 5  Etherium Stake transition function function Apply (S, Tx) -> 5  Etherium Stake transition function function function Apply (S, Tx) -> 5  Etherium Stake transition function		1 Contract Dand transaction
Etherium State Transaction Function  State  Data,  Data,  Prom:  To what:  Sigh:  Physican State  Promsetion  Prom:  To what:  Sigh:  Physican State  Promsition function Apply (S, TX) -> S  Etherium State transition function Apply (S, TX) -> S  Etherium State transition  I bras ->   compatition  Transaction -> 2000 compatin  -> 2000 ges by to S Gas por by to of  Seven all State changes  • fees to the mines's account.	Etherium State Transaction Function  Stak  Data, Fronzaction  Thom:  To value:  Value:  Value:  Sign:  Pipply (Se, tx) — S'  Etherium State transition function Apply (S, TX) -> S  Etherium State transition  I Gras -> Compatition  Transaction -> 2000 Compatin  -> 2000 ges followed S Gras por byte of  Several all State changes  I Gus to the miner's account.  Refund the fee all remaining gas to senter		
Stak  [Down, Thomselfon Thomselfon Thom: Thom: Thom: Thom: Thom: Thom: Thom: Thom: Thom: The state thomselfon function Apply (S, Tx) -> S  [Therium State thomselfon function Ap	Stak    Dorth,   Phonsaction		Execution of a contract.
Data,  Data,  Promiser :  To value:  Value:  Doda:  Sign:  Deply (Si +x) -> 5')  Etherium State transition function Apply (S, Tx) -> 5  Deply (Si +x) -> 5')  Deply (Si +x) -> 5')  Deply (Si +x) -> 5')  Deply (Si +x) -> 5'  Deply (Si +x) ->	Data,  Data,  Pringaction  Thom:  To value:  Data:  Sign:  Ppply (So, tx) -> 5')  Fibrium State transition function Apply (S, Tx) -> 5'  Ppply (So, tx) -> 5')  Fibrium State transition function Apply (S, Tx) -> 5'  Poss -> Computation  I bras -> 1 Computation  Transaction -> 2000 computation  Transaction -> 2000 computation  -> 2000 gas for Some per byte of data 5'  Percent all State changes  Percent all State changes  First to the mines's account.  Pefrend the fee all remaining gas to sorter		Etherium State Transaction Function
Data,  Data,  From:  From:  To Valu:  Dosta:  Sign:  Dipply (Si + x) -> 5')  Etherium State transition function Apply (S, Tx) -> 5  Dipply (Si + x) -> 5')  Dipply (Si + x) -> 5')  Dipply (Si + x) -> 5')  Dipply (Si + x) -> 5'  Di	Data,  Data,  Pringaction  Thom:  To value:  Data:  Sign:  Ppply (So, tx) -> 5')  Fibrium State transition function Apply (S, Tx) -> 5'  Ppply (So, tx) -> 5')  Fibrium State transition function Apply (S, Tx) -> 5'  Poss -> Computation  I bras -> 1 Computation  Transaction -> 2000 computation  Transaction -> 2000 computation  -> 2000 gas for Some per byte of data 5'  Percent all State changes  Percent all State changes  First to the mines's account.  Pefrend the fee all remaining gas to sorter	3-2	
Data,  Data,  Promiser :  To value:  Value:  Doda:  Sign:  Deply (Si +x) -> 5')  Etherium State transition function Apply (S, Tx) -> 5  Deply (Si +x) -> 5')  Deply (Si +x) -> 5')  Deply (Si +x) -> 5')  Deply (Si +x) -> 5'  Deply (Si +x) ->	Data,  Data,  Pringaction  Thom:  To value:  Data:  Sign:  Ppply (So, tx) -> 5')  Fibrium State transition function Apply (S, Tx) -> 5'  Ppply (So, tx) -> 5')  Fibrium State transition function Apply (S, Tx) -> 5'  Poss -> Computation  I bras -> 1 Computation  Transaction -> 2000 computation  Transaction -> 2000 computation  -> 2000 gas for Some per byte of data 5'  Percent all State changes  Percent all State changes  First to the mines's account.  Pefrend the fee all remaining gas to sorter	,	State State
Therium State transition function Apply (S, Tx) \rightarrows  [Therium State transition function function Apply (S, Tx) \rightarrows  [Therium State transition function function Apply (S, Tx) \rightarrows  [Therium State transition function funct	Phorpaction  From:  To:  Value:  Data:  Sign:  Sign:   (Ppply (So, tx) -> 5')  (Popply (So, tx)		
Therium State transition function Apply (S, TX) \rightarrow S  (Apply (So, tx) -> 5')  (Apply (So, tx)	Ppply (Si, +x) -> 5  Etherium State transition function Apply (S, Tx) -> 5  Distribution State transition function Apply (S, Tx) -> 5  Distribution State transition function Apply (S, Tx) -> 5  Distribution -> 16 compatition  Transaction -> 2000 compates  -> 2000 gcs by the of some state of data 19  Pevers all State changes  Peters all State changes  Peters the mines's account.  Peters the lee all remaining gas to senter		Diensaction
Ppply (So, tx) -> 5'  Etherium State transition function Apply (S, Tx) -> 5'  10) Gras -> Competition  1 Gras -> 1 Competition  transaction -> 2000 competer  -> 2000 gas by S Gras por by the of data 12  • Pevert all State changes  • fus to the mines's account.	Poply (So + x) -> 5'  (Poply (So + x) -> 5')  (Phyrium State transition function Apply (S, Tx) -> 5  (Phyrium State transition function Apply (S, Tx) -> 5  (Phyrium State transition function Apply (S, Tx) -> 5  (Pas - Compatition - Compatit	? <u></u>	From:
Ppply (So, tx) -> 5'  Etherium State transition function Apply (S, Tx) -> 5'  10) Gras -> Competition  1 Gras -> 1 Competition  transaction -> 2000 competer  -> 2000 gas by S Gras por by the of data 12  • Pevert all State changes  • fus to the mines's account.	Poply (So + x) -> 5'  Etherium State transition function Apply (S, Tx) -> 5  15 Gras -> (competition  16 cos -> (competition  transaction -> 2000 competer  -> 2000 ges 66 S Gras per by the of date 12  feveral all State changes  fas to the mines's account.  lefund the fee all remaining gas to senter		70
Ppply (SO, +x) -> 5'  Etherium State transition function Apply (S, Tx) -> 5'  By Gras -> Computation  16ras -> 1 computation  transaction -> 2000 computs  -> 2000 gas (b) S Gras por byte of  levers all State changes  fees to the mines's account.	[Poply (So, tx) ->5')  Etherium Stute transition function Apply (S, TX) ->5'  15 Gras -> Computation  15 Gras -> 1 Computation  -> 2000 gas   5 Gras per by the of  -> Gras per by the to pay the byto data 17  • Pevert all State changes  • fees to the miner's account.  • Refund the fee all remaining gas to sends	-	Data:
Etherium State transition function Apply (S, TX) -> S  A) Gras -> Computation  16ras -> 1 Computation  transaction -> 2000 computs  -> 2000 gcs (b) S Gras por byte of  Ocas por byte to pay the bytes  • Peverl all State changes  • fus to the mines's account.	Etherium State transition function Apply (S, TX) >> 5  ## Gras -> Computation  1 Gras -> 1 Computation  transaction -> 2000 computer  -> 2000 ges for S Gras por byte of  levers all State changes  fees to the mines's account.  Peternd the fee all remaining gas to sends	)	Sigh:
Etherium State transition function Apply (S, TX) -> S  (Fas -> Computation  16 ras -> 1 Computation  transaction -> 2000 computs  -> 2000 ges (b) S Gas por byte of  Ocas por byte to pay the byte data?  • Pevert all State changes  • fas to the mines's account.	Etherium State transition function Apply (S, TX) >> 5  ## Gras -> Computation  16ras -> 1 Computation  transaction -> 2000 computer  -> 2000 ges file  > Gras per byte to pay the byte data 19  • Revert all State changes  • fees to the mines's account.  • Refund the fee all remaining gas to sends	<del>}</del>	
Etherium State transition function Apply (S, TX) -> S  (Fas -> Computation  16 ras -> 1 Computation  transaction -> 2000 computs  -> 2000 ges (b) S Gas por byte of  Ocas por byte to pay the byte data?  • Pevert all State changes  • fas to the mines's account.	Etherium State transition function Apply (S, TX) >> 5  ## Gras -> Computation  16ras -> 1 Computation  transaction -> 2000 computer  -> 2000 ges file  > Gras per byte to pay the byte data 19  • Revert all State changes  • fees to the mines's account.  • Refund the fee all remaining gas to sends		( Ppply (SO +x) -> 5)
Etherium State transition function Apply (S, TX) -> S  (S) Gras -> Computation  16 ros -> 1 Computation  transaction -> 2000 computes  -> 2000 gas (b) S Gras por byte of  Oras por byte to pay the byte data?  Pevert all State changes  fees to the mines's account.	Etherium State fronsition function Apply (S, TX) -> S  15 Gras -> Computation  16 cos -> 1 Computation  transaction -> 2000 computer  -> 2000 gas 66 S Gras per byte of  levers all State changes  fees to the mines's account.  leptond the fee all remaining gas to sends	,	
(Sas -> Computation  16ras -> 1 Computation  transaction -> 2000 computes  -> 2000 gas by te of  (Sas per byte to pay the byte data ??  Revert all State changes  fees to the mines's account.	(Fas -> Computation  16 ras -> 1 Computation  transaction -> 2000 computer  -> 2000 gas by se of  S Gras por byse to pay the byto data 19  • Revert all State changes  • fees to the mines's account.  • Repend the fee all remaining gas to senter	1.7	
1 brows > 1 computation  transaction -> 2000 computer  -> 2000 ges 66 S Grows por byte of  > Gos por byte to pay the bytes data 19  • Pevert all State changes  • fees to the mines's account.	1 bras ->   Compatition  transaction -> 2000 computer  -> 2000 gas   S Gras per by te of  S Gras por byte to pay the byto data 19  e levers all State changes  fees to the mines's account.  leptend the fee all remaining gas to sends	)	Etherium State transition function Apply (S, TX) -> 5
1 bras > 1 computation  transaction > 2000 computer  > 2000 gas by te of  S Gras per by te of  lever all State changes  • fees to the mines's account.	1 bras ->   Compatition    transaction -> 2000 computer    -> 2000 gas   S Gras per by te of    S Gras per by te of data 19  levers all State changes  fees to the mines's account.  Petend the fee all remaining gas to sends		
1 bras > 1 computation  transaction > 2000 computer  > 2000 gas by te of  S Gras per by te of  lever all State changes  • fees to the mines's account.	1 bras ->   Compatition  transaction -> 2000 computer  -> 2000 gas   S Gras per by te of  S Gras por byte to pay the byto data 19  e levers all State changes  fees to the mines's account.  leptend the fee all remaining gas to sends		Gas -> Computation
transaction -> 2000 computer  -> 2000 gas 66 S Gross per byte of  > Gross per byte to pay the byto data 19  • Pevert all State changes  • fees to the mines's account.	transaction -> 2000 computer  -> 2000 gas by se of  S Gras per by se of  levers all State changes  • fees to the mines's account.  • Repend the fee all remaining gas to sender		ž.
-> 2000 ges 66 S Gros per byte of  > Gros per byte to pay the byte data 19  • Revert all State changes  • Fees to the mines's account.	2000 ges by the of S Gas per by the of S Gas per by the of data 19  Revert all State changes  fees to the minos's account.  Refund the fee all remaining gas to sender	,	·
Scor per byte to pay the byte data 19  • Pevert all State changes • fees to the mines's account.	Scros per byte to pay the byte data 19  Pevert all State changes  fees to the minor's account.  Repand the fee all remaining gas to sender	,	
· Peverd all State changes · Fees to the minos's account.	· Pevert all State changes · fees to the mines's account. · Refund the fee all remaining gas to sendes	<del>\</del>	
· Peverd all State changes · Fees to the minos's account.	· Pevert all State changes · fees to the minos's account. · Repund the fee all remaining gas to sendes		Solos yes by the of
· fees to the mines's account.	· fees to the mines's account.  · Repend the fee all remaining gas to sender		1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
	. Refund the fee all remaining gas to sender		^
" Repund the lee all remaining gas to sender	fees paid gas consumed to miner.		
	, fees paid gas consumed to miner.	C	· Refund the fee all remaining gas to sender
fees pold gas consumed to miner.		I.	fees paid gas consumed to minus.

	before solidity used Serpent Conquege
<b>(A)</b>	Node Js- is a runtime of javascript _/_/_
	example A 10 ETH B
	- Cras → 2000 / White / posters
. ∀ .	Gray price > 0.00 v ell petral price
	Data -> 64 bytes
ж	medicalist Acceptable to the total
	CPU
	CPU -> binary -> Machine Code
	Assembly -> LOW levels longuya
	Juva -> High level language
	C++ dela
	Pythen
į.	FVM -> Etherium Virtual Machine
1	
	3
	Code Execution
	-> Ethereum contract code
	in that (LIFO) -> LOW level
Last	ous ) > Stack based byte code
	FVM (ode
	type of space Etherium Virtual
	-> Stack Machine
	-> Memory -
	-> Storage -
-A	10 minute Litherium much mayler
FC 11	Block
	Black Time: 12 Sec (Slots)
	Black Size 15 million gas
	Black Stru 15 million gas
1 74	2 .41.
	30 million gas

