

SOK: TRANSPARENT DISHONESTY FRONT-RUNNING ATTACKS ON BLOCKCHAIN.

SHAYAN ESKANDARI, SEYEDEHMAHSA MOOSAVI, JEREMY CLARK

3rd Workshop on Trusted Smart Contracts @ FC'19
February 2019



UNIVERSITÉ

Concordia

UNIVERSITY

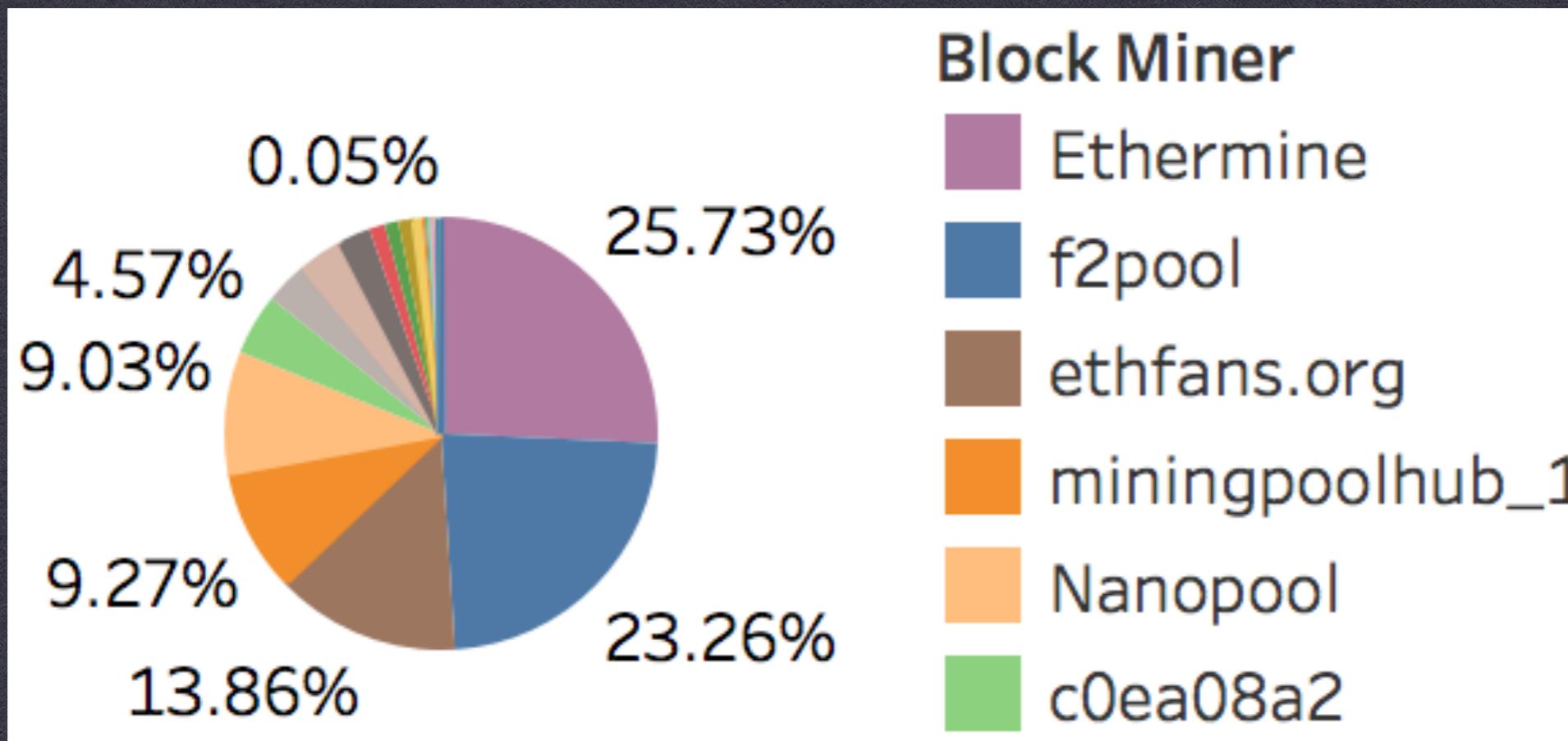
CONSENSYS

Diligence

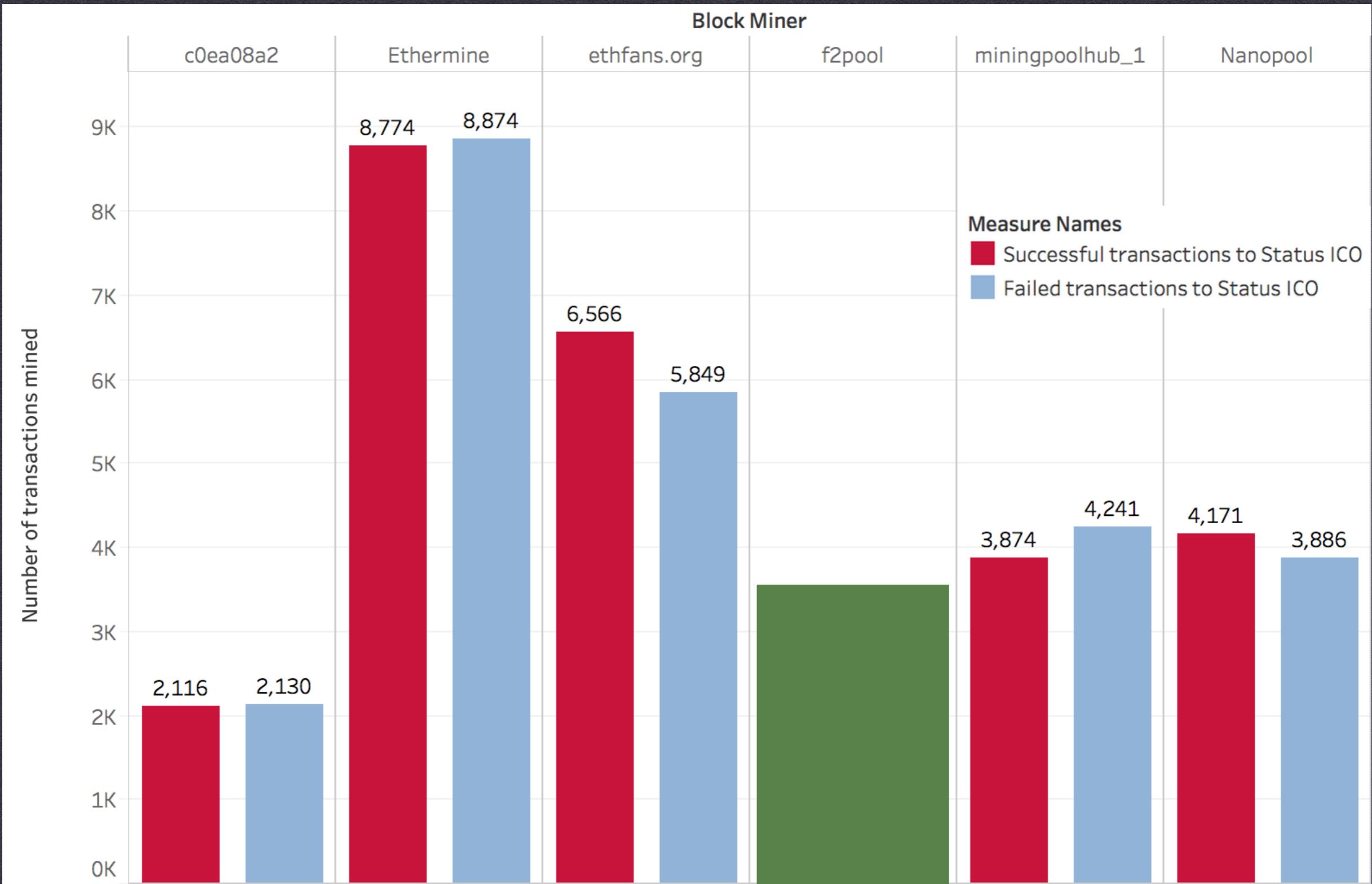
STORY 1: ICO

STATUS.IM ICO

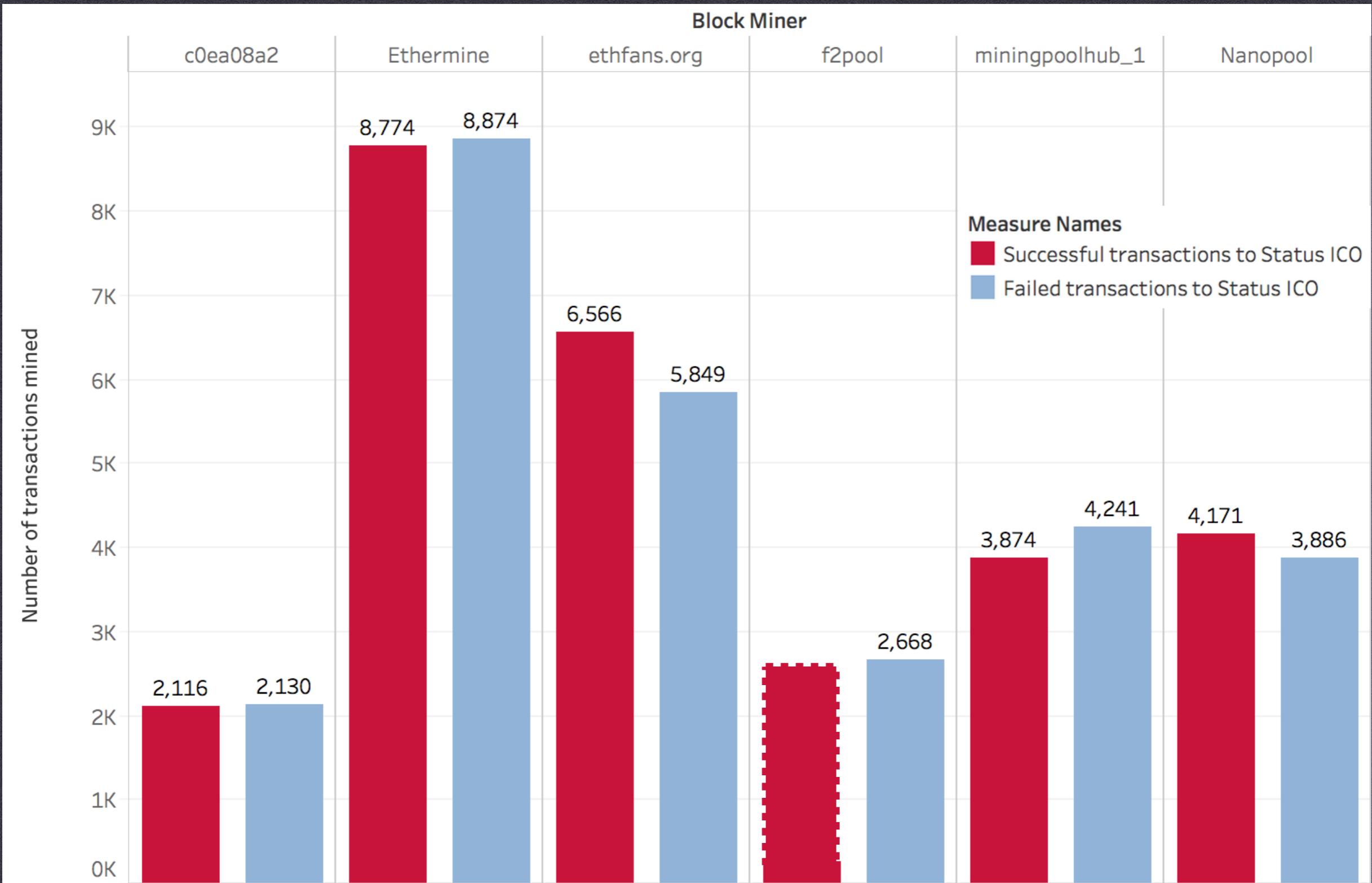
- * JUNE 2017
- * ~300,000 ETH IN 16 HOURS
- * FAIRNESS:
 - * (GASPRICE > 50 GWEI) -> INVALID
 - * DYNAMIC CAP



STATUS ICO <> F2POOL

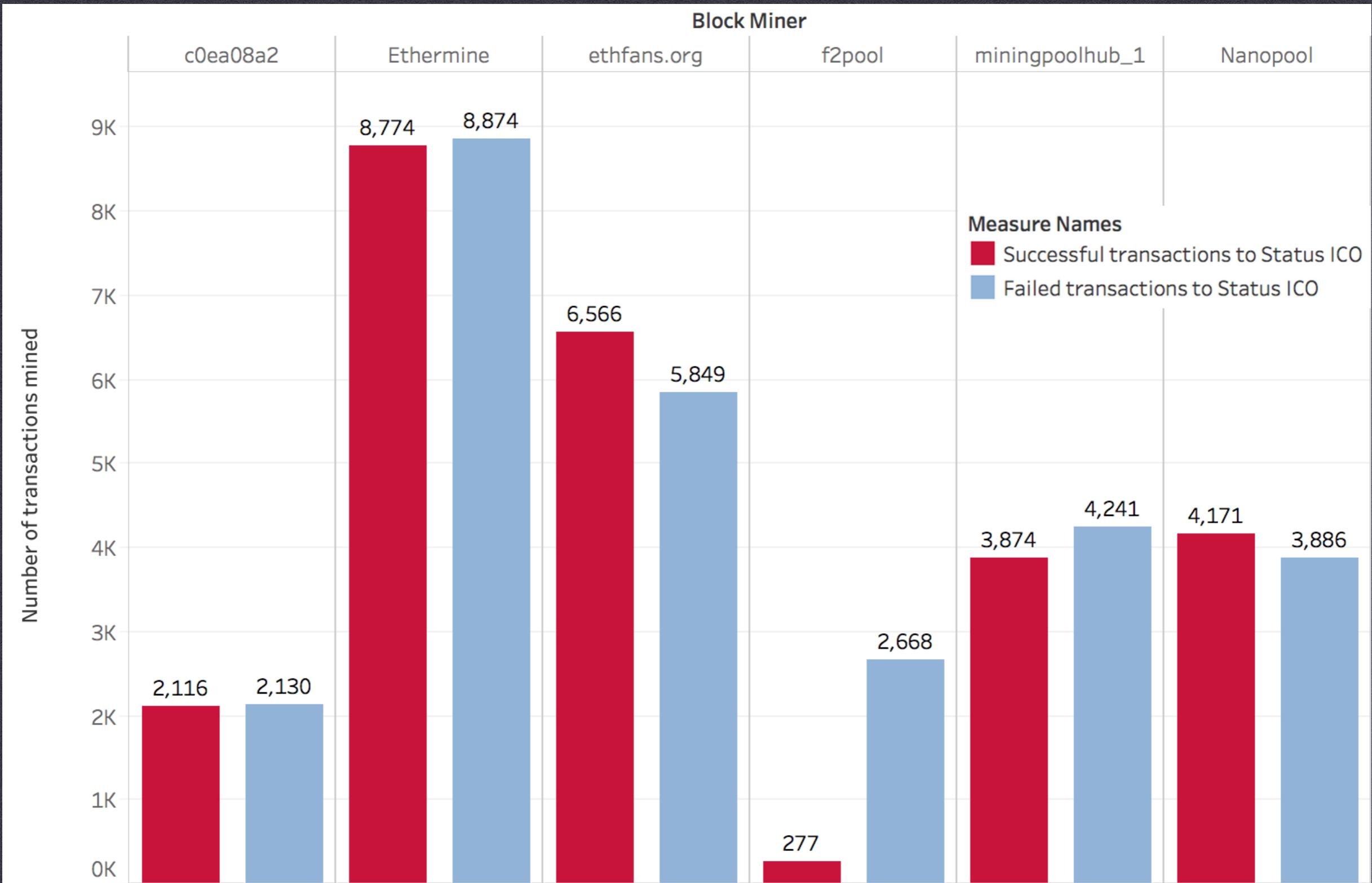


STATUS ICO <>> F2POOL
F2POOL



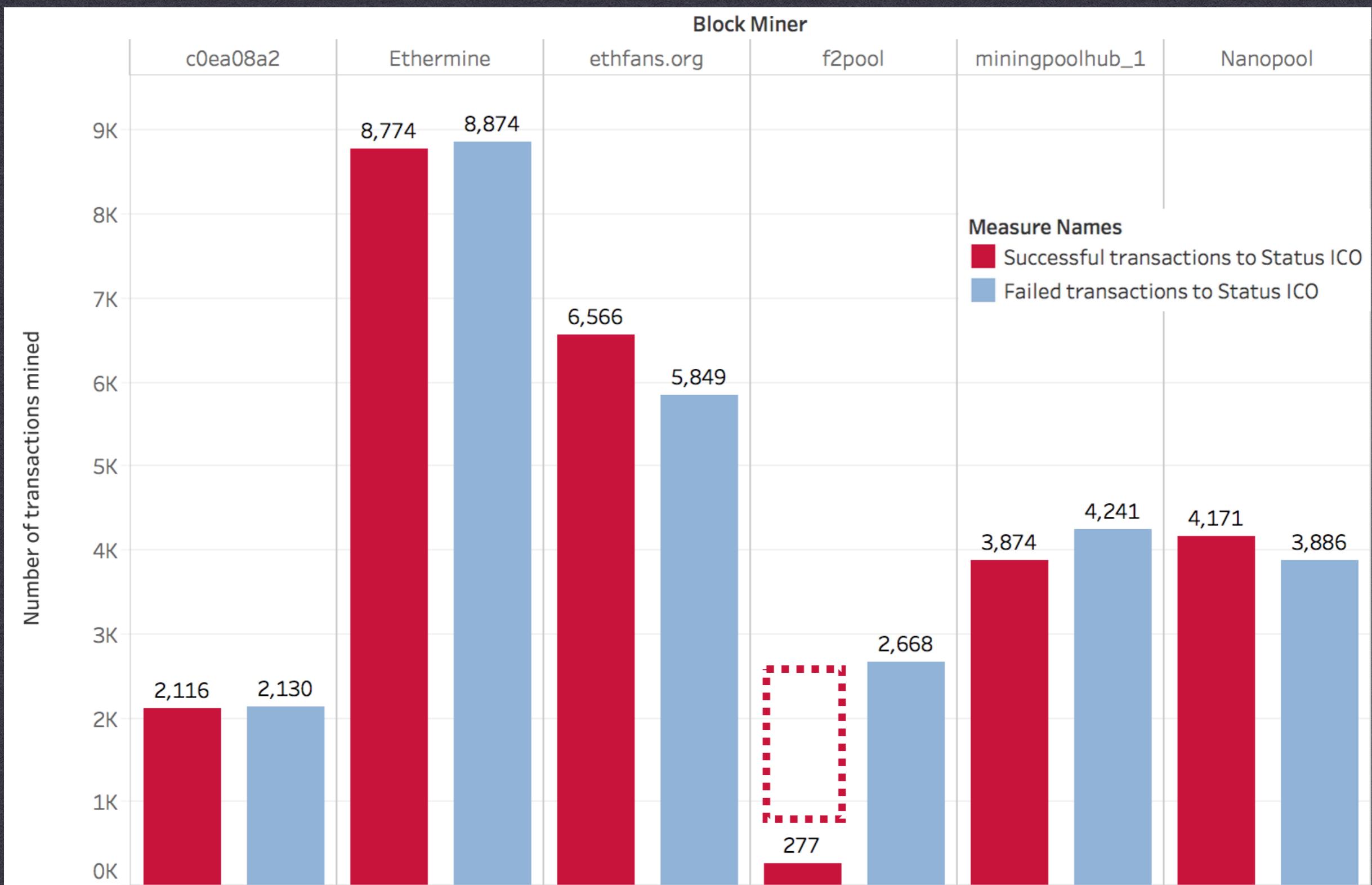
STATUS ICO <> F2POOL

WHAT WE WERE EXPECTING TO SEE

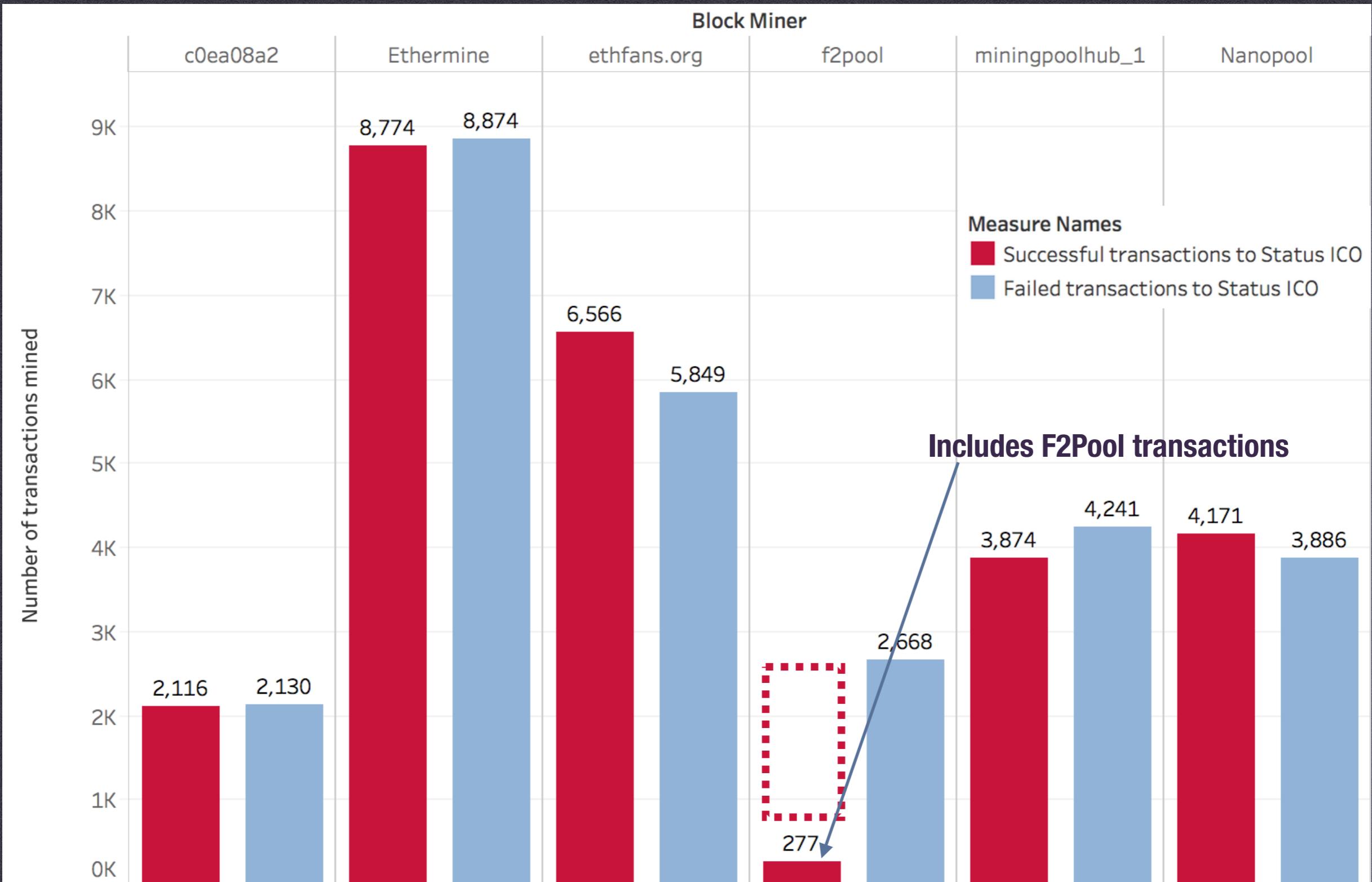


STATUS ICO <>> F2POOL

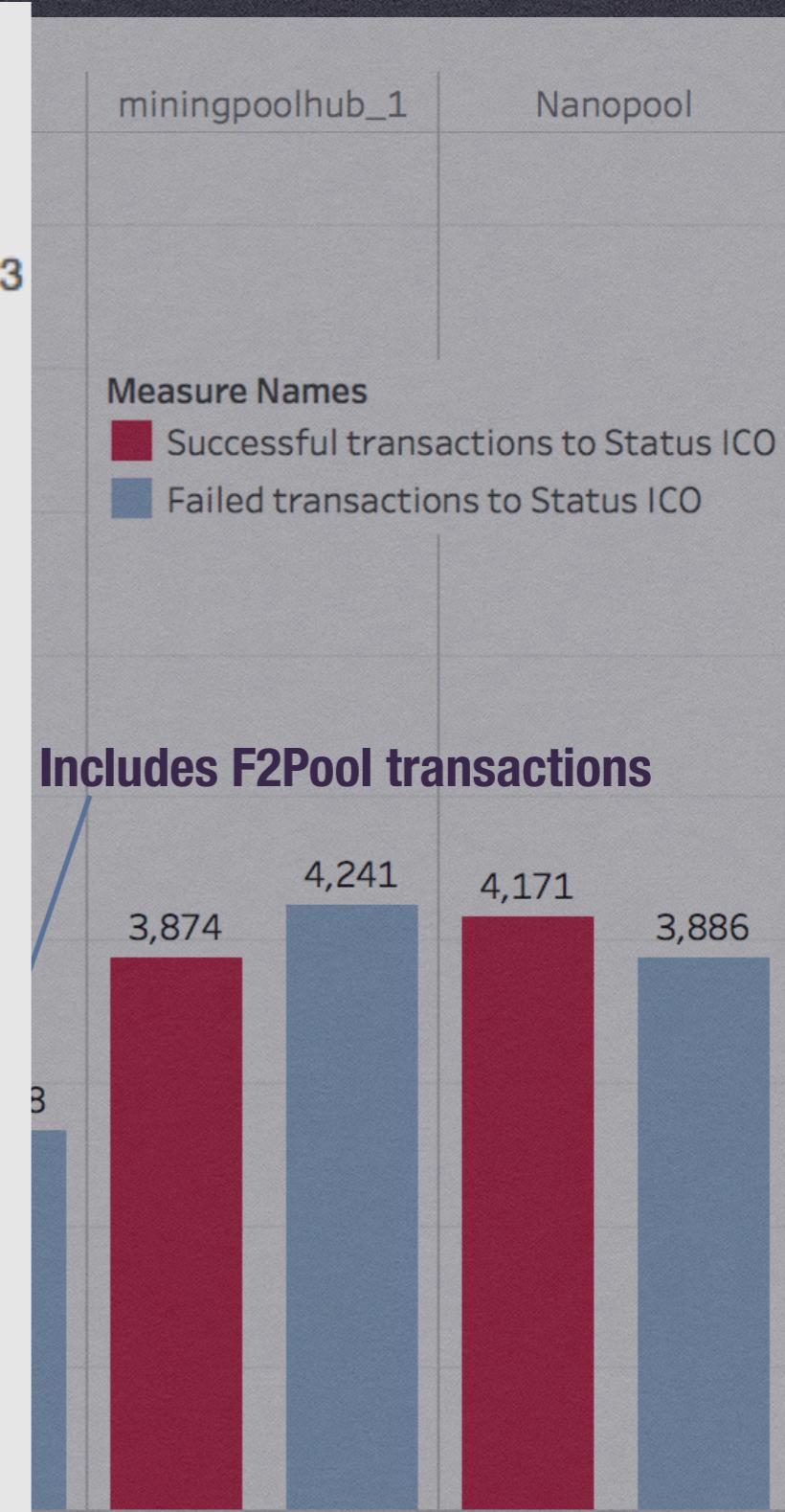
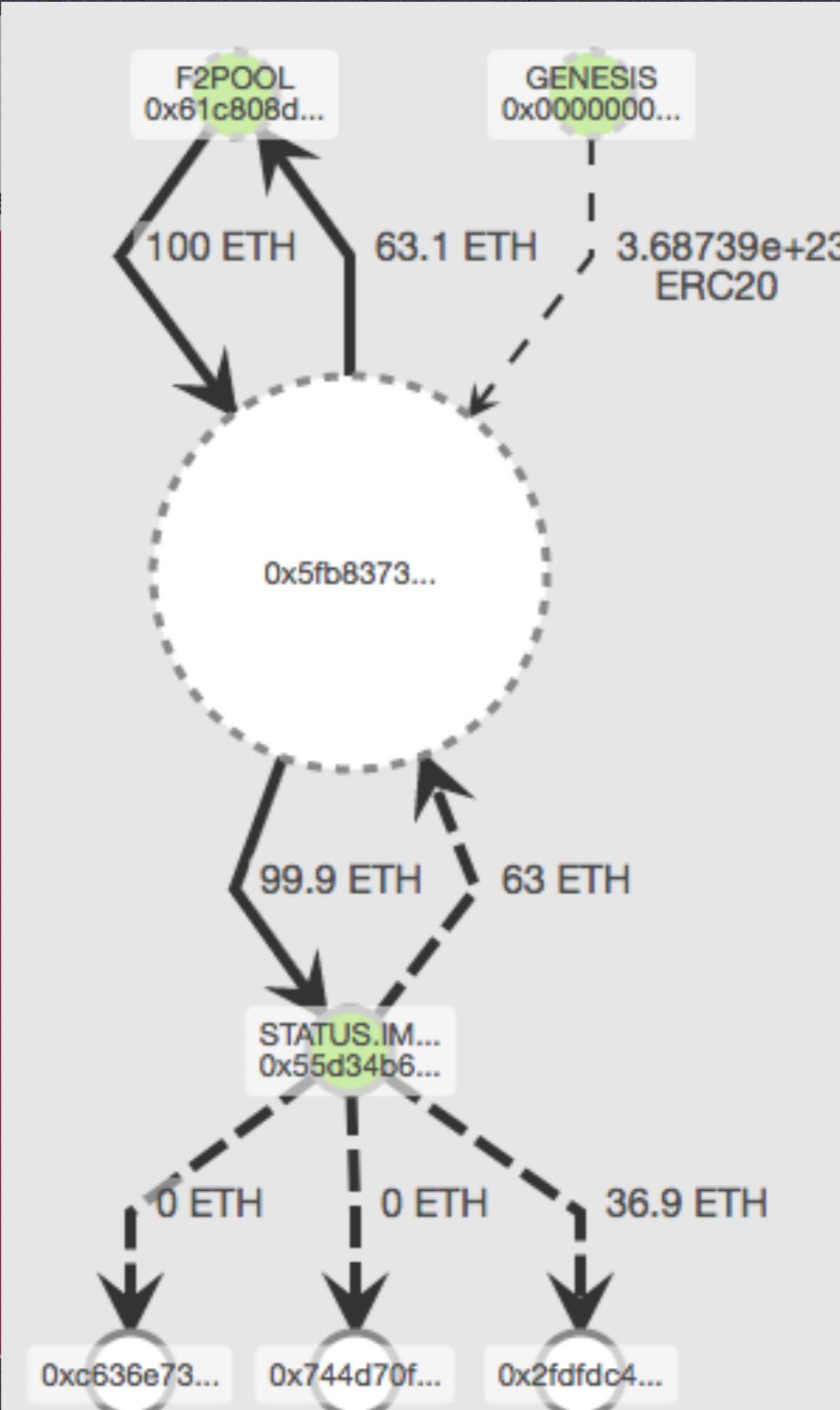
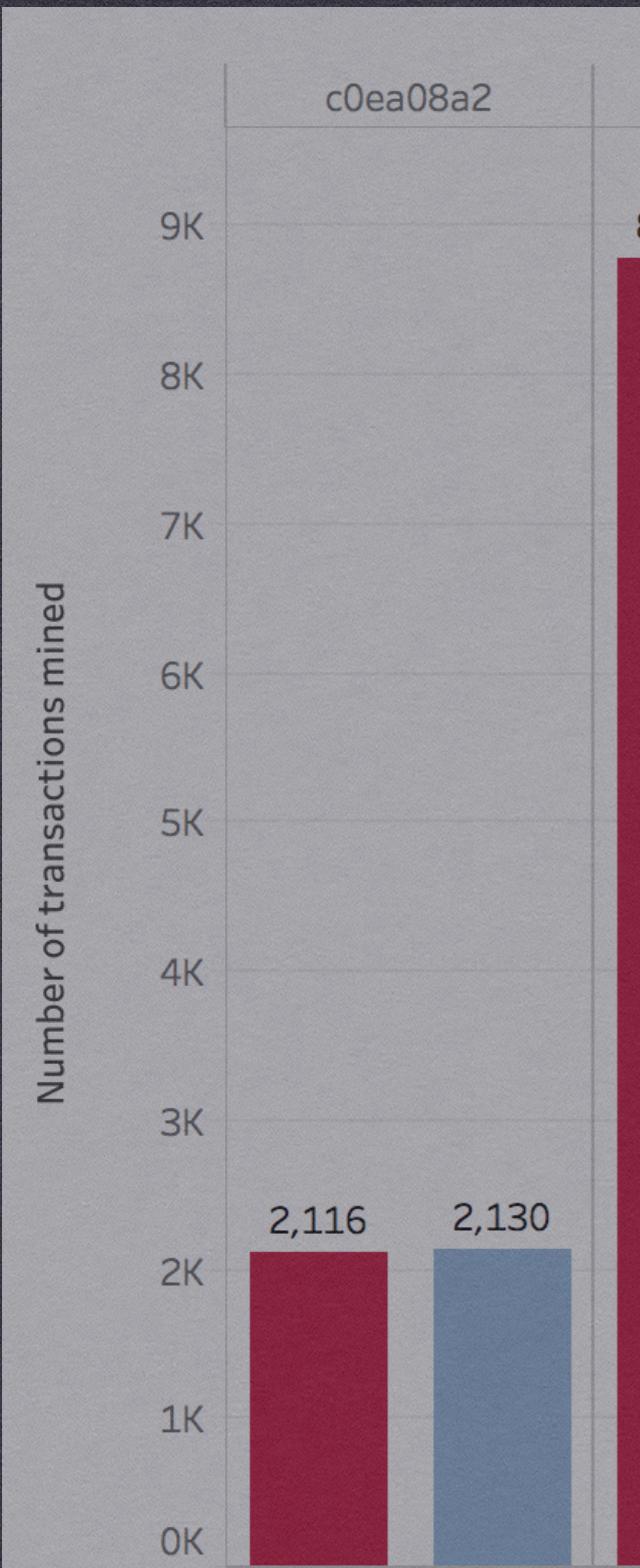
WHAT WE SAW



STATUS ICO <>> F2POOL
MISSING TRANSACTIONS?

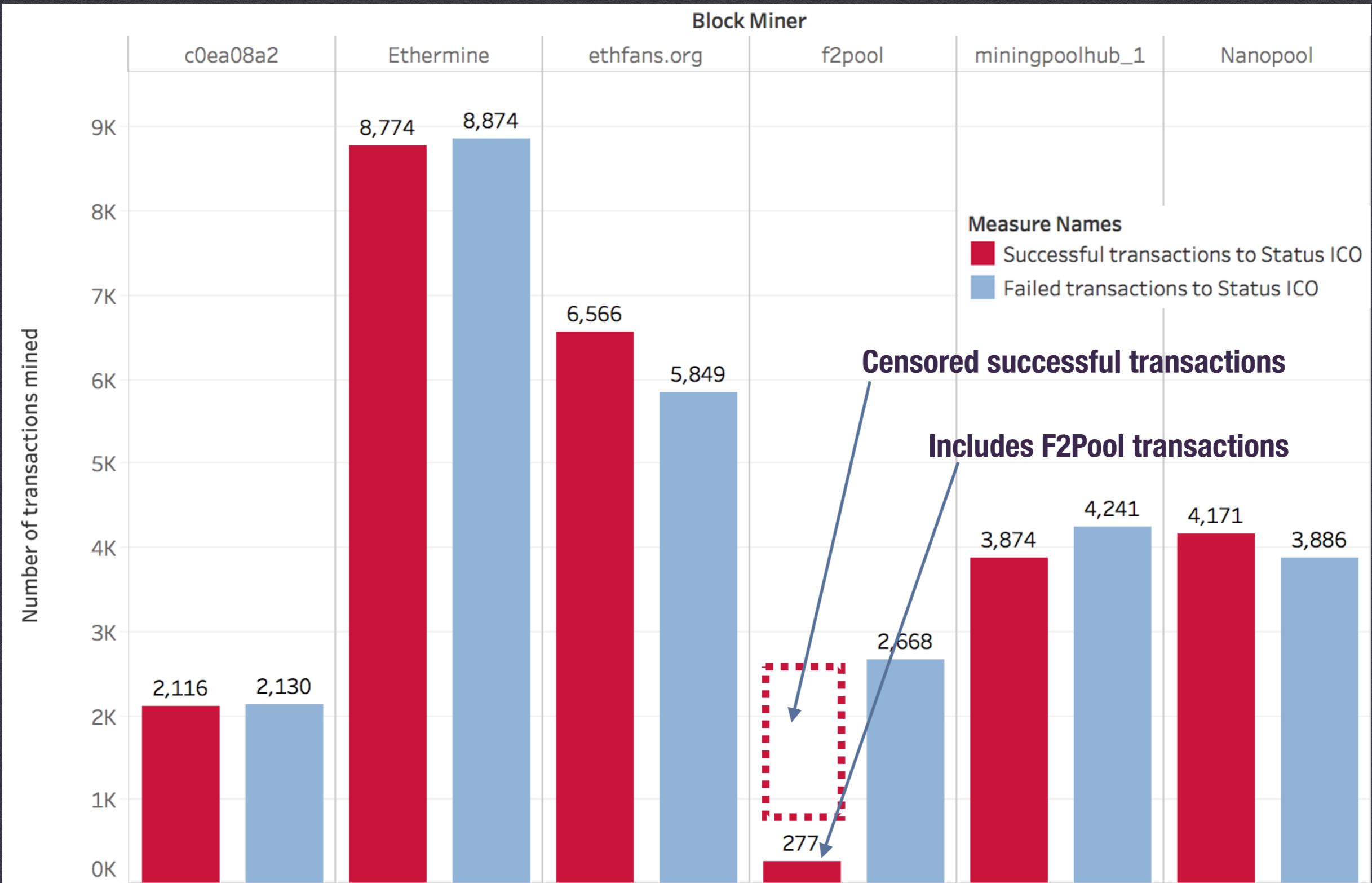


STATUS ICO <>> F2POOL
SELECTIVE MINING?



STATUS ICO <>> F2POOL

TRACE OF F2POOL TRANSACTIONS



STATUS ICO <>> F2POOL
CENSORING OTHER TRANSACTIONS

STORY 2: FOMO3D



UNIVERSITÉ
Concordia
UNIVERSITY

CONSENSYS
Diligence

someone else is

EXIT SCAMMING



350.6794 

+ Pre-Seed: 0.2092 

= Total: 350.8886 

23:04:15

This is your key, there are many like it, but this one is yours

EXITSCAM.ME → FOMO3D

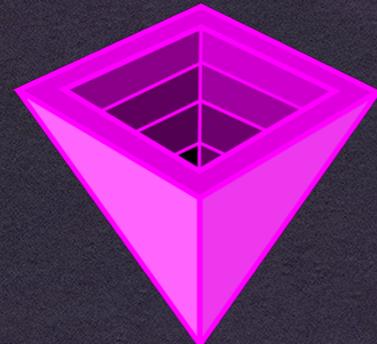


UNIVERSITE
Concordia
UNIVERSITY

CONSENSYS
Diligence

someone else is

- * A countdown timer
- * Every ticket purchase increases the timer by 30 seconds
- * The last ticket when the timer reaches 00:00:00 wins the pot



350.6794



+ Pre-Seed: 0.2092



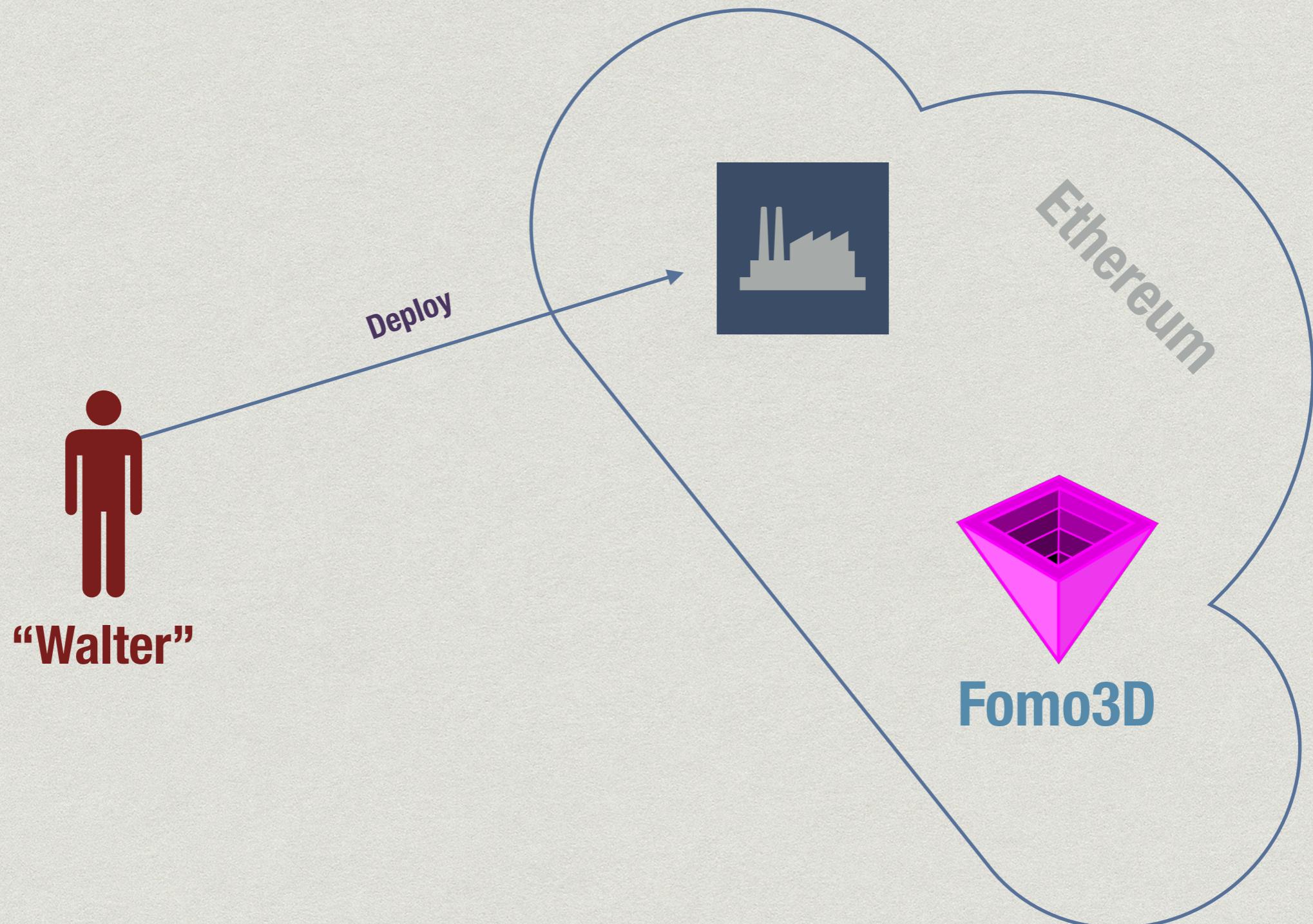
= Total: 350.8886

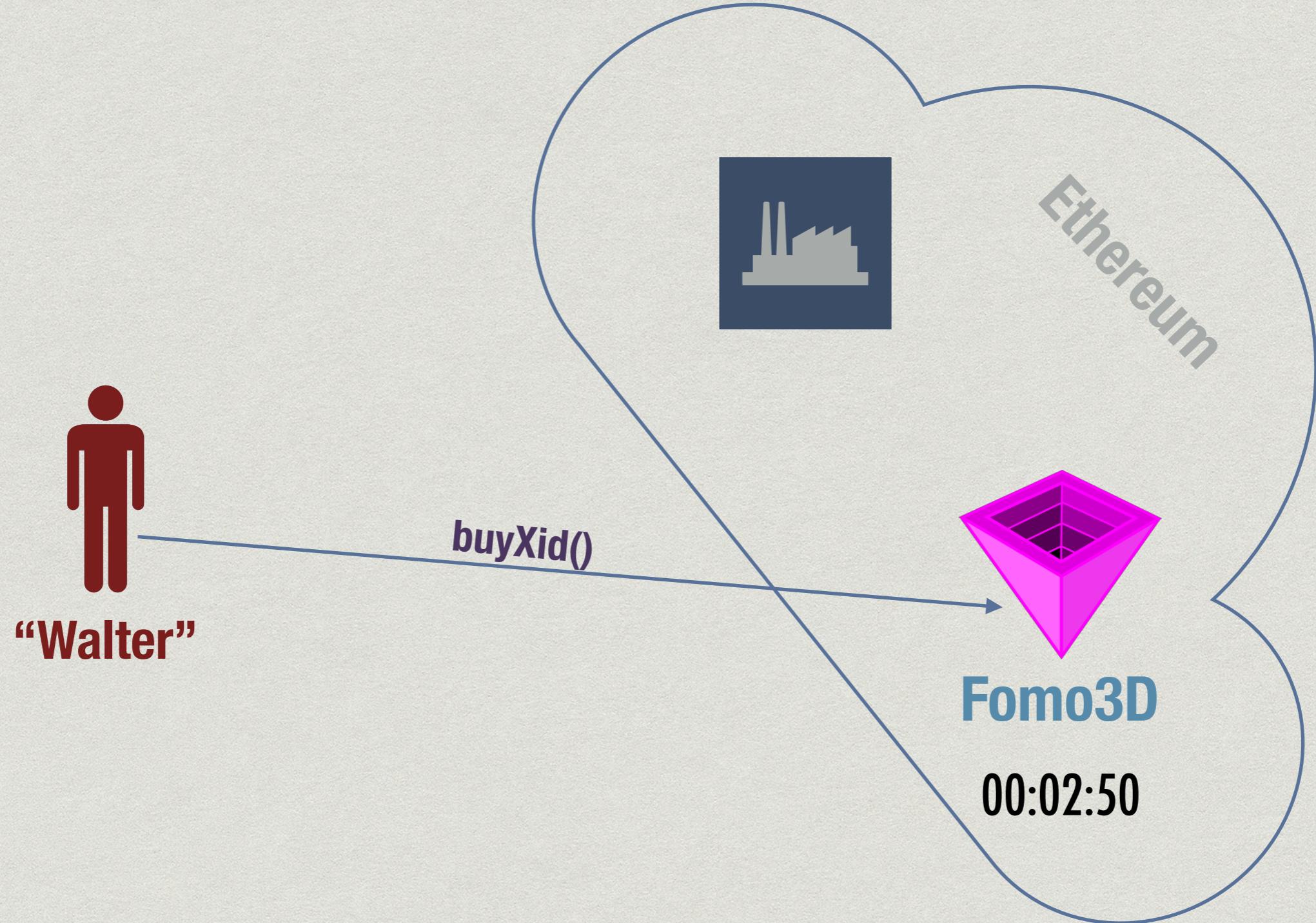


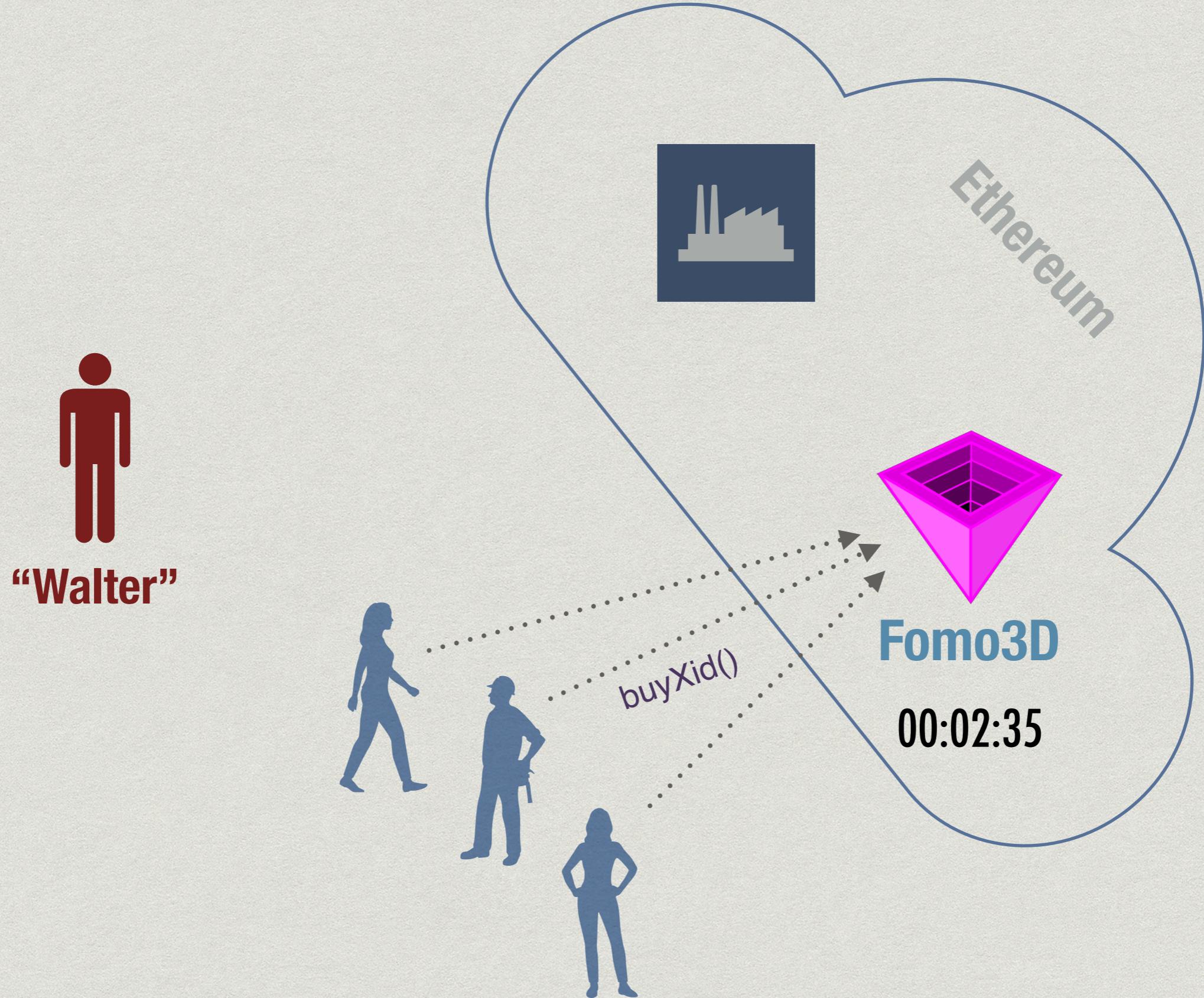
23:04:15

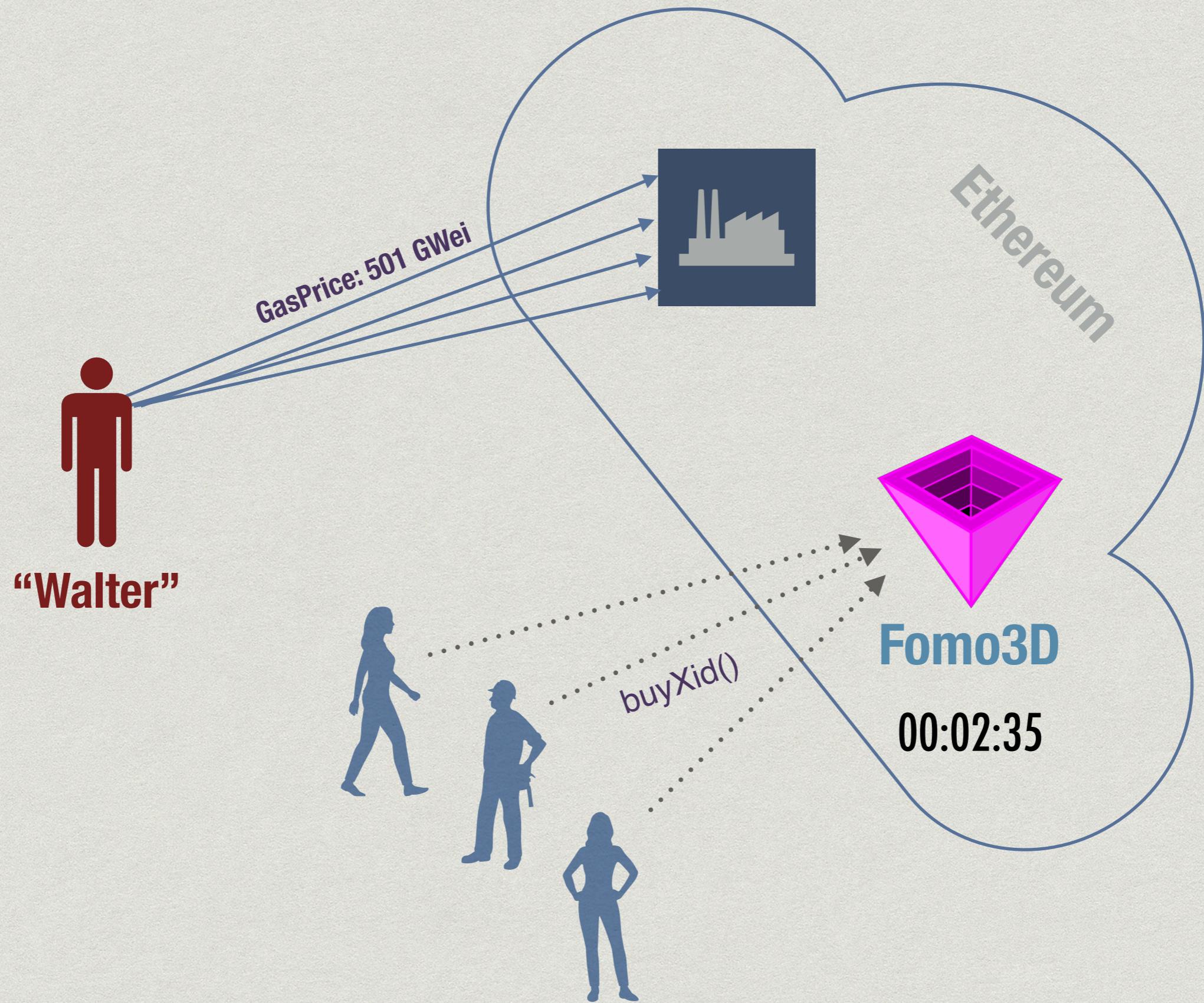
This is your key, there are many like it, but this one is yours

EXITSCAM.ME → FOMO3D











“Walter”

Block 6191904

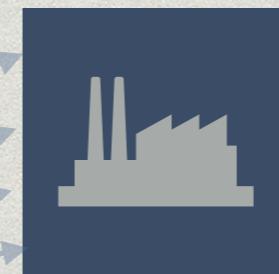
2018-08-22 06:49:57, ts:1534920597

Average gas price: 190.0 Gwei

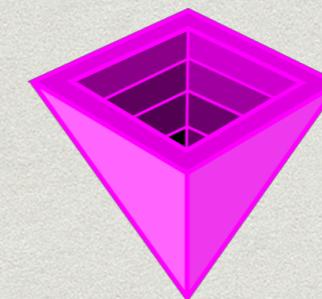
Idx	From	To	Hash	ETH sent	Gas Price [Gwei]	Gas Limit	Gas Used	ETH spent on gas	ABI Call	Events
0	0xF03...1f2	0x18e...801	0x7d1...4cf	0	190.0	4,200,000	4,200,000	0.798018		
1	0x87C...4eF	0x18e...801	0x8db...9d2	0	190.0	3,600,000	3,600,000	0.684013		
2	0xf6E...059	0x18e...801	0x79a...1aa	0	190.0	200,000	200,000	0.038		

osolmaz.com

GasPrice: 501 Gwei



Ethereum



Fomo3D

00:02:20



“Walter”

Block 6191905

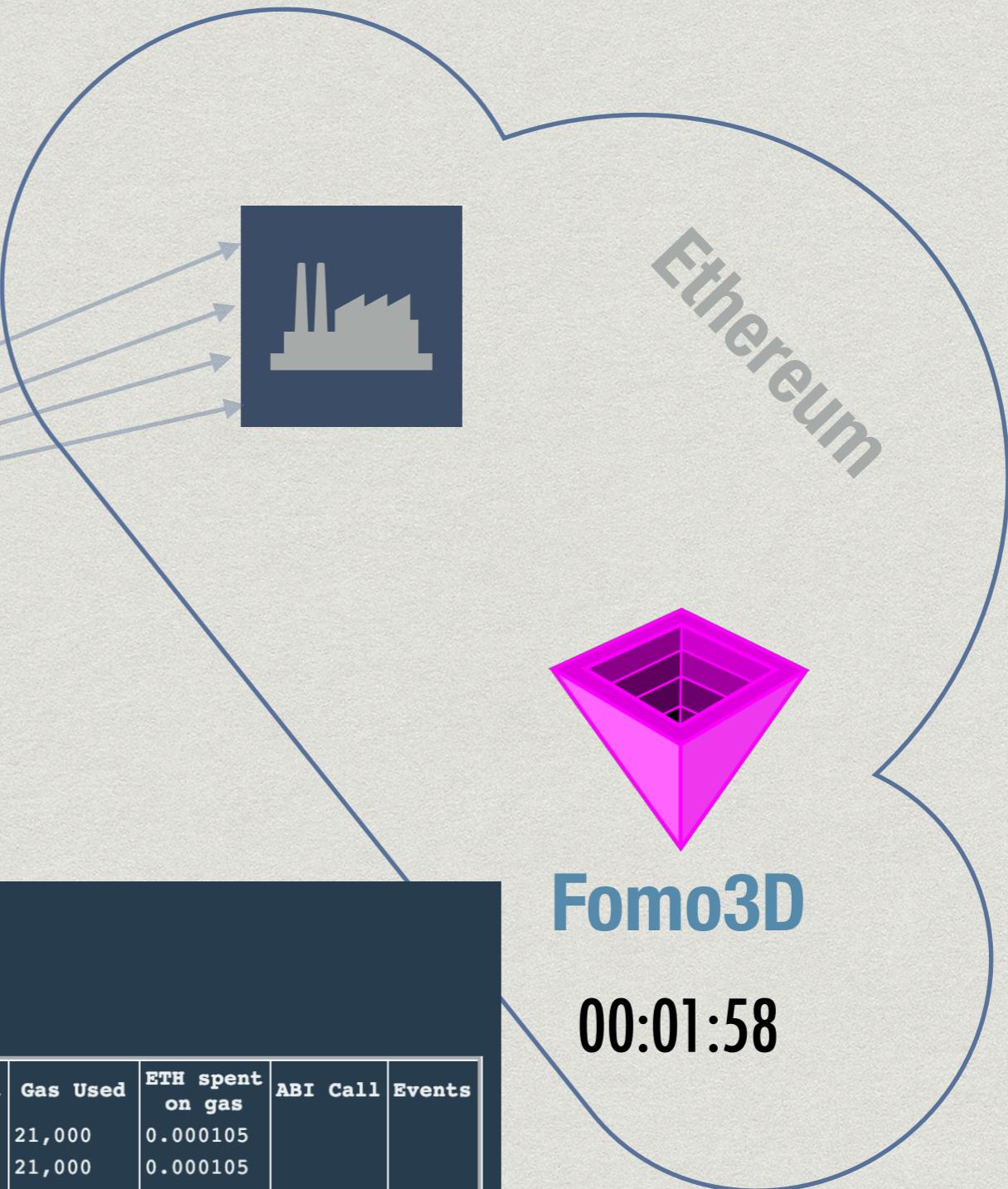
2018-08-22 06:50:36, ts:1534920636

Average gas price: 228.3 Gwei

Idx	From	To	Hash	ETH sent	Gas Price [Gwei]	Gas Limit	Gas Used	ETH spent on gas	ABI Call	Events
0	0xb29...347	0x359...B23	0xd54...a47	0.0396017	5.0	100,000	21,000	0.000105		
1	0xb29...347	0xF65...Fa8	0xe76...d52	0.0195326	5.0	100,000	21,000	0.000105		
2	0xb29...347	0x48d...aFe	0x78b...f5b	0.0095638	5.0	100,000	21,000	0.000105		
3	0x9DA...0cF	0x18e...801	0xbc8...319	0	501.0	4,800,000	4,800,000	2.40482		
4	0x7Dd...c4c	0x18e...801	0xd9d...2b7	0	501.0	2,700,000	2,700,000	1.35271		
5	0x00c...776	0x18e...801	0x7c3...2e0	0	501.0	400,000	400,000	0.2004		
6	0xA10...e25	0xb9e...9b0	0xb27...c2e	6.99832	80.0	21,000	21,000	0.00168		
				7.06702	1598.01	8,221,000	7,984,000	3.95993		

osolmaz.com

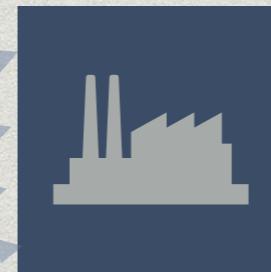
GasPrice: 501 Gwei





“Walter”

GasPrice: 501 GWei



Block 6191906

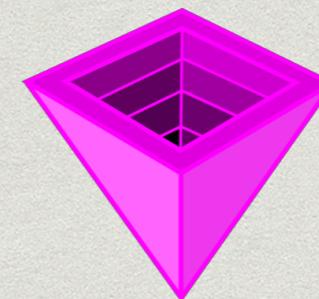
2018-08-22 06:50:45, ts:1534920645

Average gas price: 501.0 Gwei

Idx	From	To	Hash	ETH sent	Gas Price [Gwei]	Gas Limit	Gas Used	ETH spent on gas	ABI Call	Events
0	0xF03...1f2	0x18e...801	0xb97...8e4	0	501.0	4,200,000	4,200,000	2.10422		
1	0x87C...4eF	0x18e...801	0x96f...1b0	0	501.0	3,600,000	3,600,000	1.80361		
2	0xf6E...059	0x18e...801	0x897...2b3	0	501.0	200,000	200,000	0.1002		

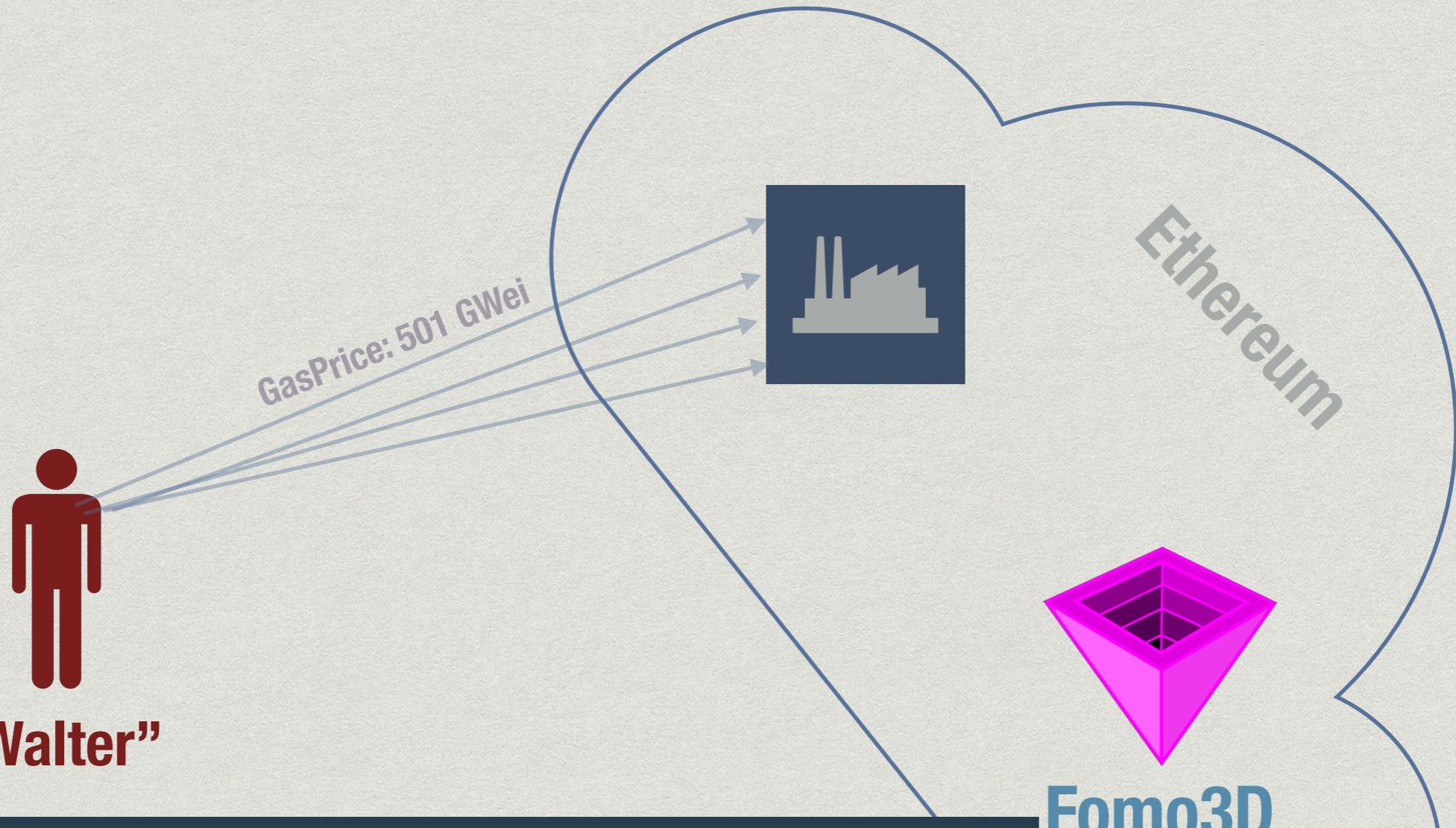
osolmaz.com

Ethereum



Fomo3D

00:01:12



Block 6191909

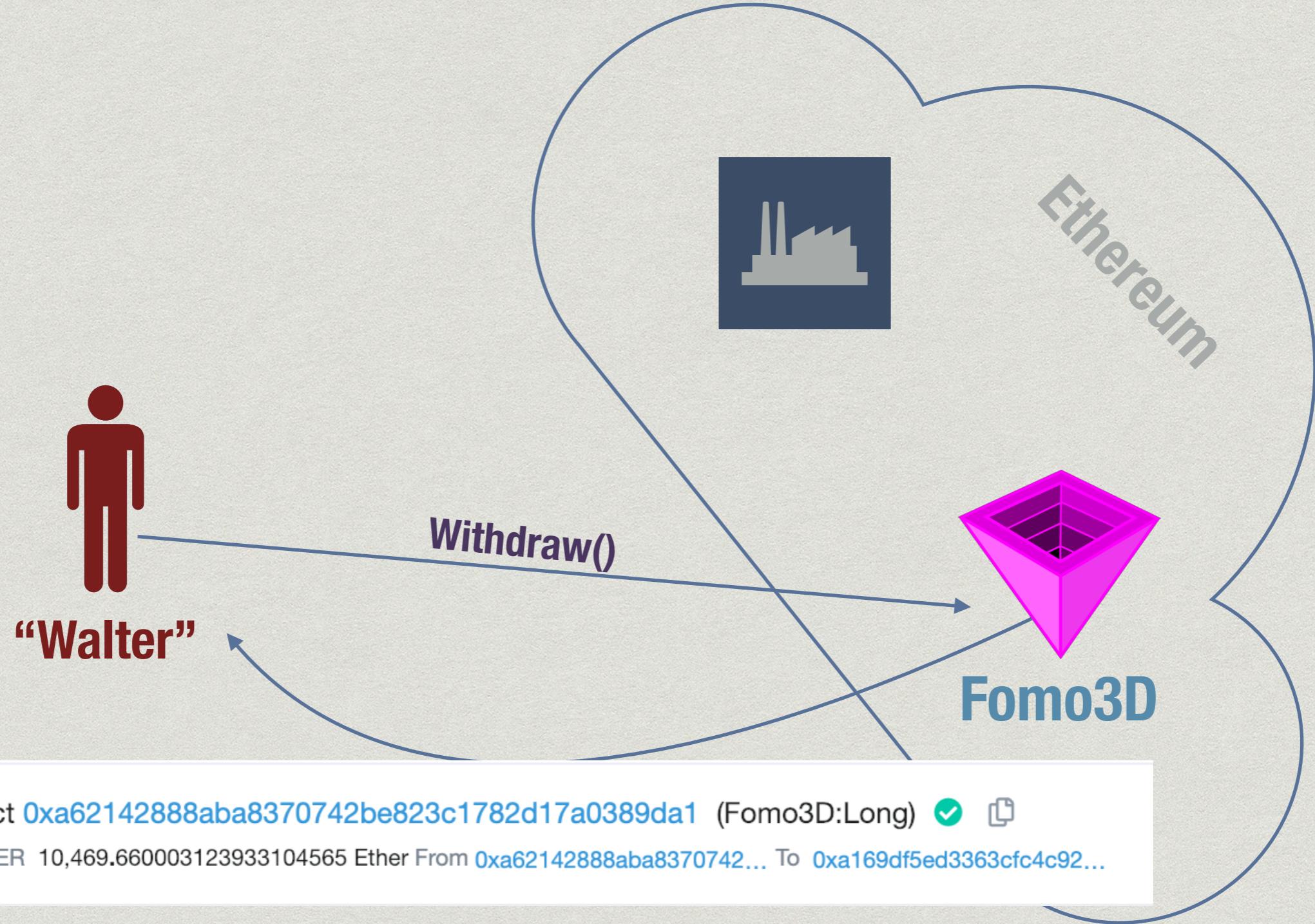
2018-08-22 06:51:17, ts:1534920677

Average gas price: 93.0 Gwei

Idx	From	To	Hash	ETH sent	Gas Price [Gwei]	Gas Limit	Gas Used	ETH spent on gas	ABI Call	Events
0	0x32A...370	0xA62...Da1	0xa14...012	0.00560162	5562.2	379,000	304,750	1.69508	buyXaddr	onBuyAndDistribute
1	0xC96...590	0x18e...801	0xf47...9ca	0	501.0	2,200,000	37,633	0.0188542		
2	0xb1D...aEF	0x18e...801	0xe4c...edb	0	501.0	1,400,000	37,633	0.0188542		
3	0x18D...A9A	0x18e...801	0xf3a...995	0	501.0	800,000	37,633	0.0188542		
4	0x00c...776	0x18e...801	0xeb2...100	0	501.0	400,000	37,633	0.0188541		
5	0xf6E...059	0x18e...801	0x8c2...b23	0	501.0	200,000	37,633	0.0188541		

osolmaz.com

Fomo3D
00:00:00



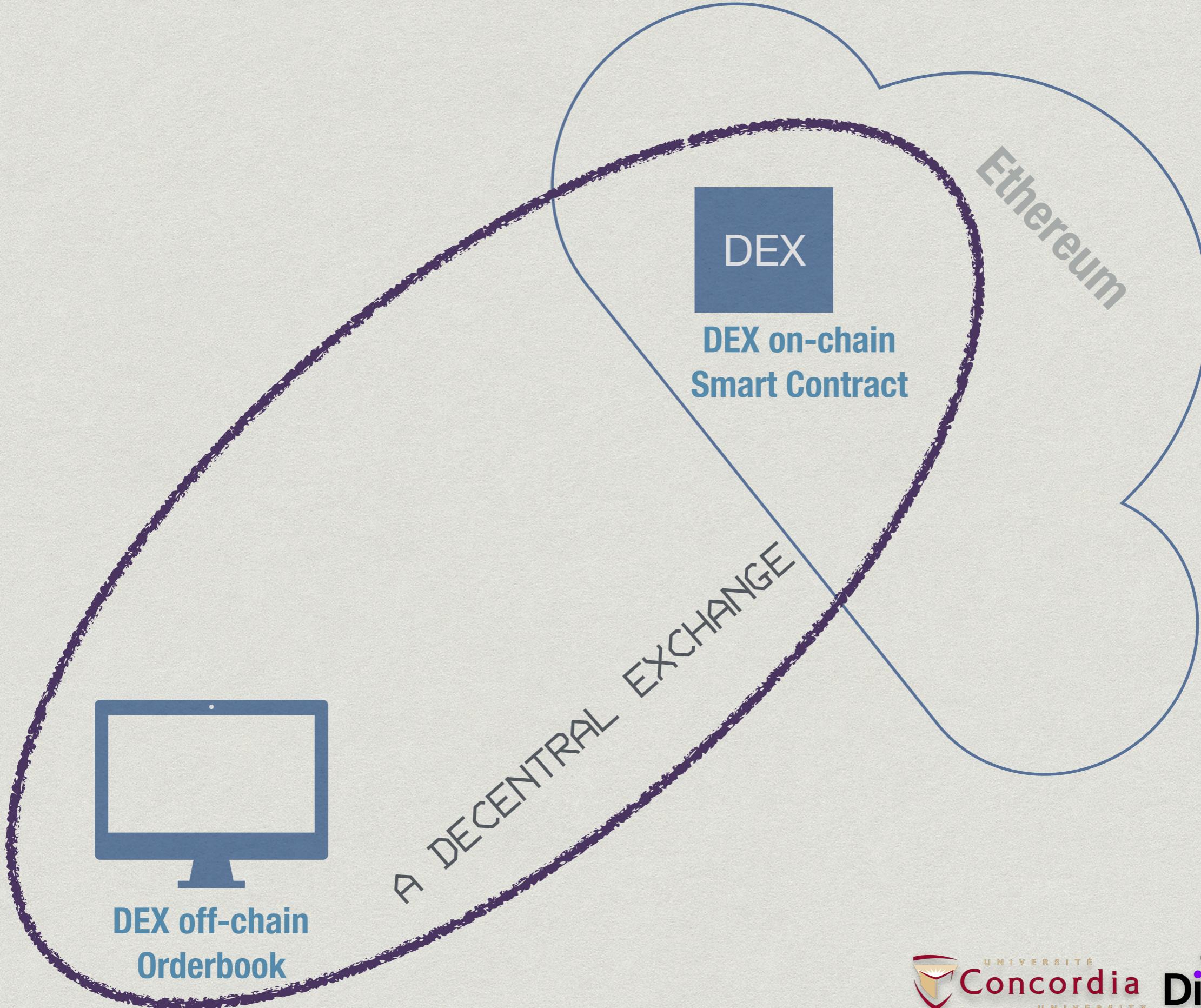
STORY 3: DECENTRAL EXCHANGES

Cancellation Griefing



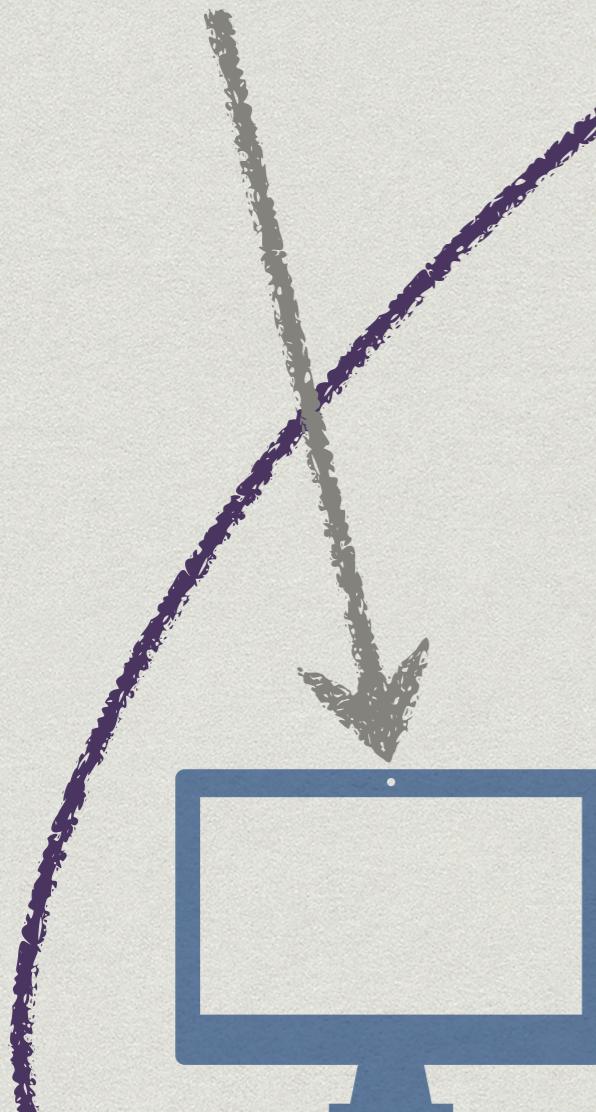
UNIVERSITE
Concordia
UNIVERSITY

CONSENSYS
Diligence



Orders (bid/ask)

- **Fast**
- **Low fees**



**DEX off-chain
Orderbook**

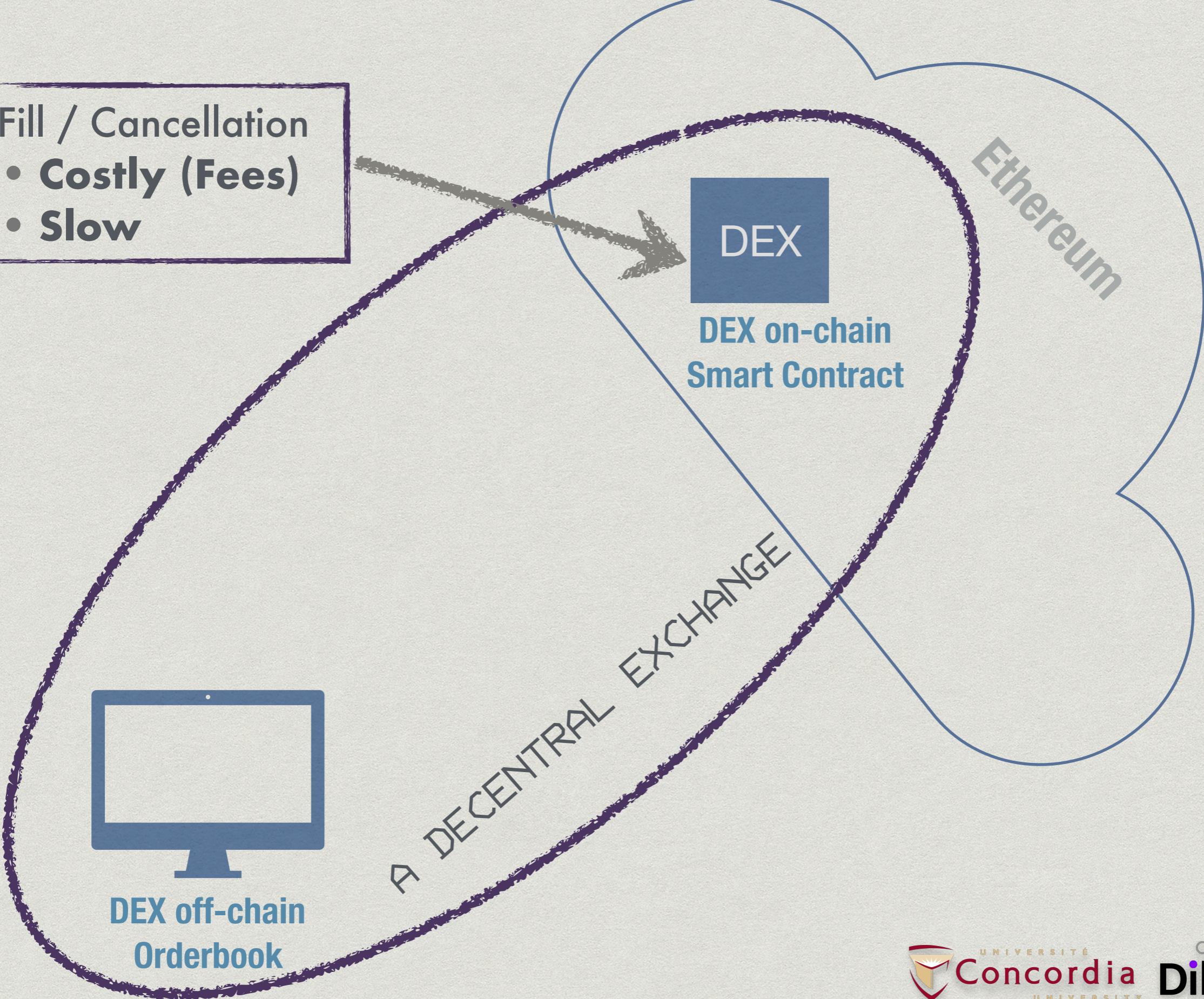
A DECENTRAL EXCHANGE

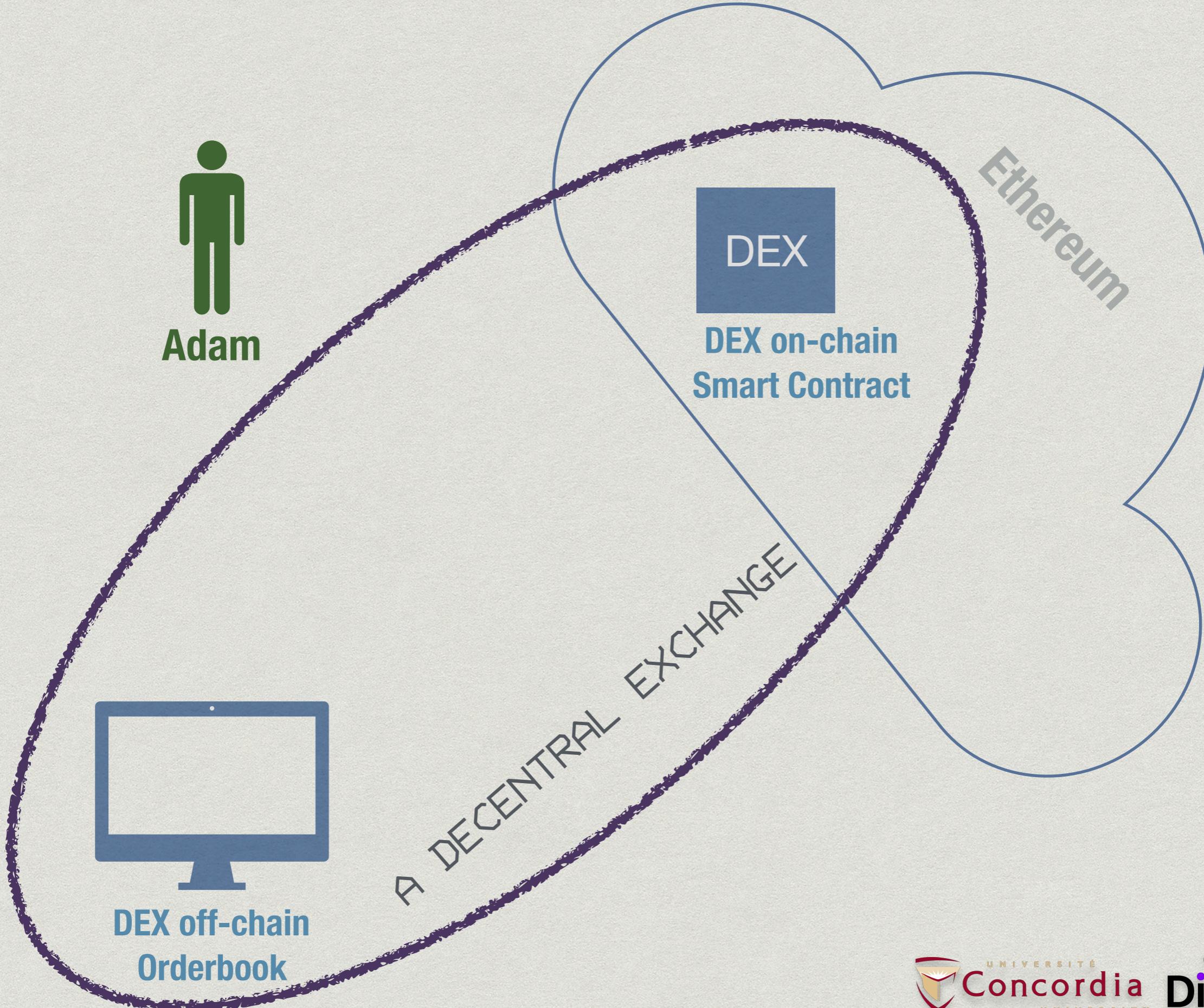


**DEX on-chain
Smart Contract**

Fill / Cancellation

- **Costly (Fees)**
- **Slow**

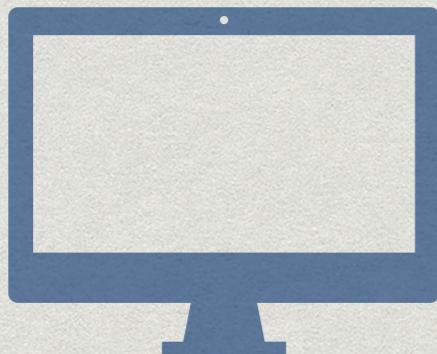






Adam

Buy(1000 UET)



DEX off-chain
Orderbook



DEX
DEX on-chain
Smart Contract

Ethereum



Adam



DEX off-chain
Orderbook



UNIVERSITÉ
Concordia
UNIVERSITY

CONSENSYS
Diligence



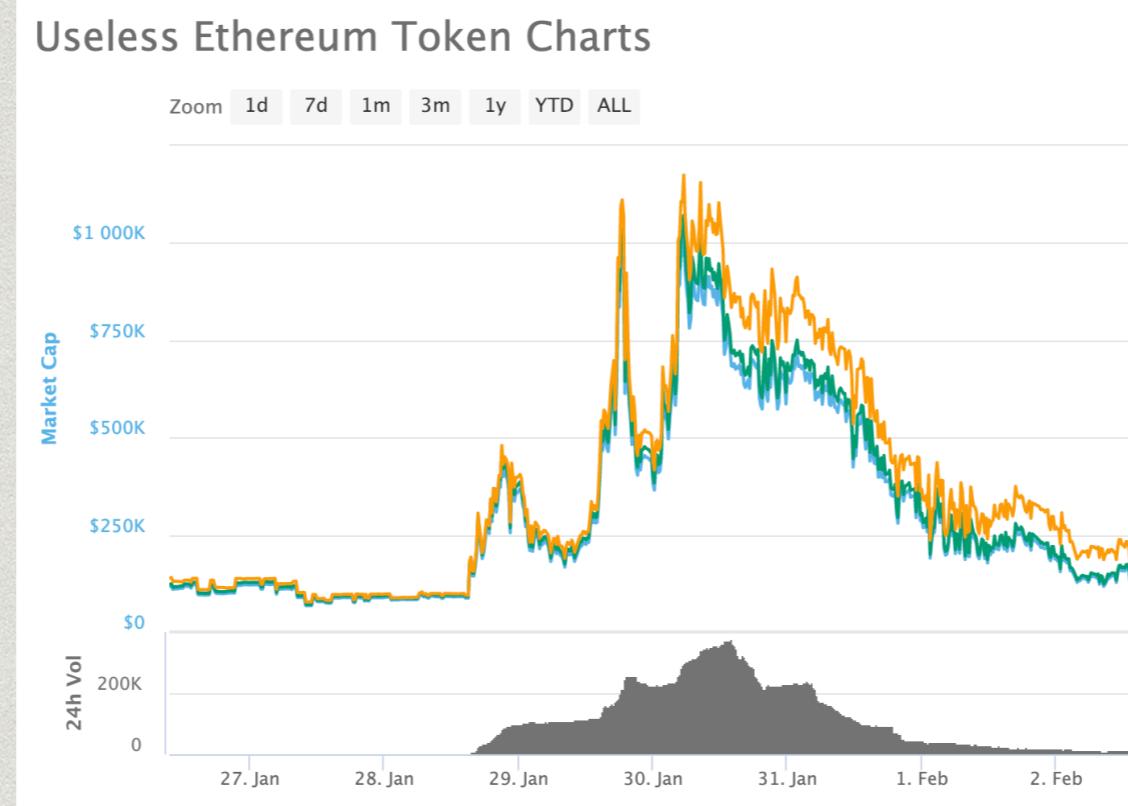
Adam



DEX off-chain
Orderbook

DEX

Ethereum





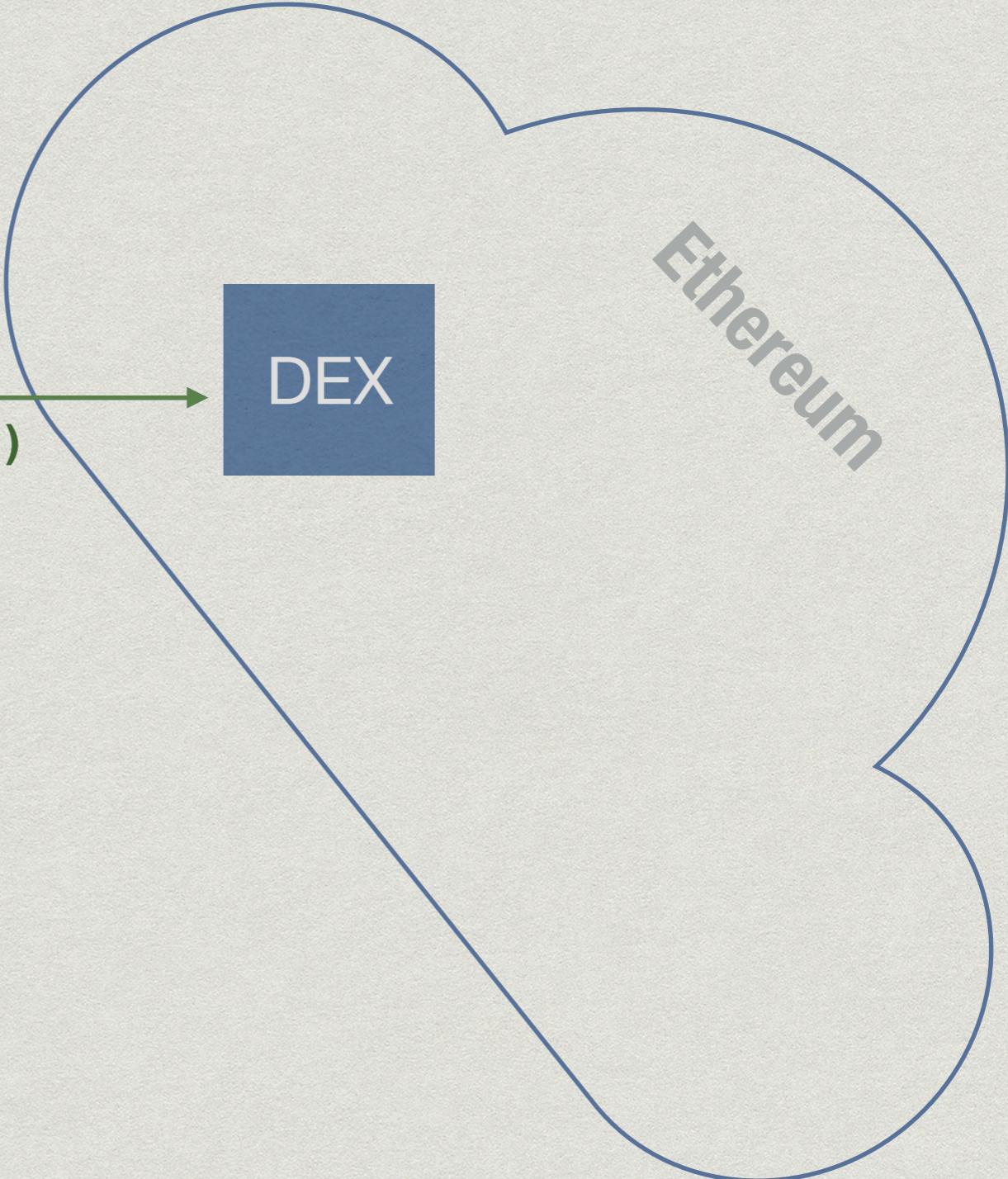
Adam

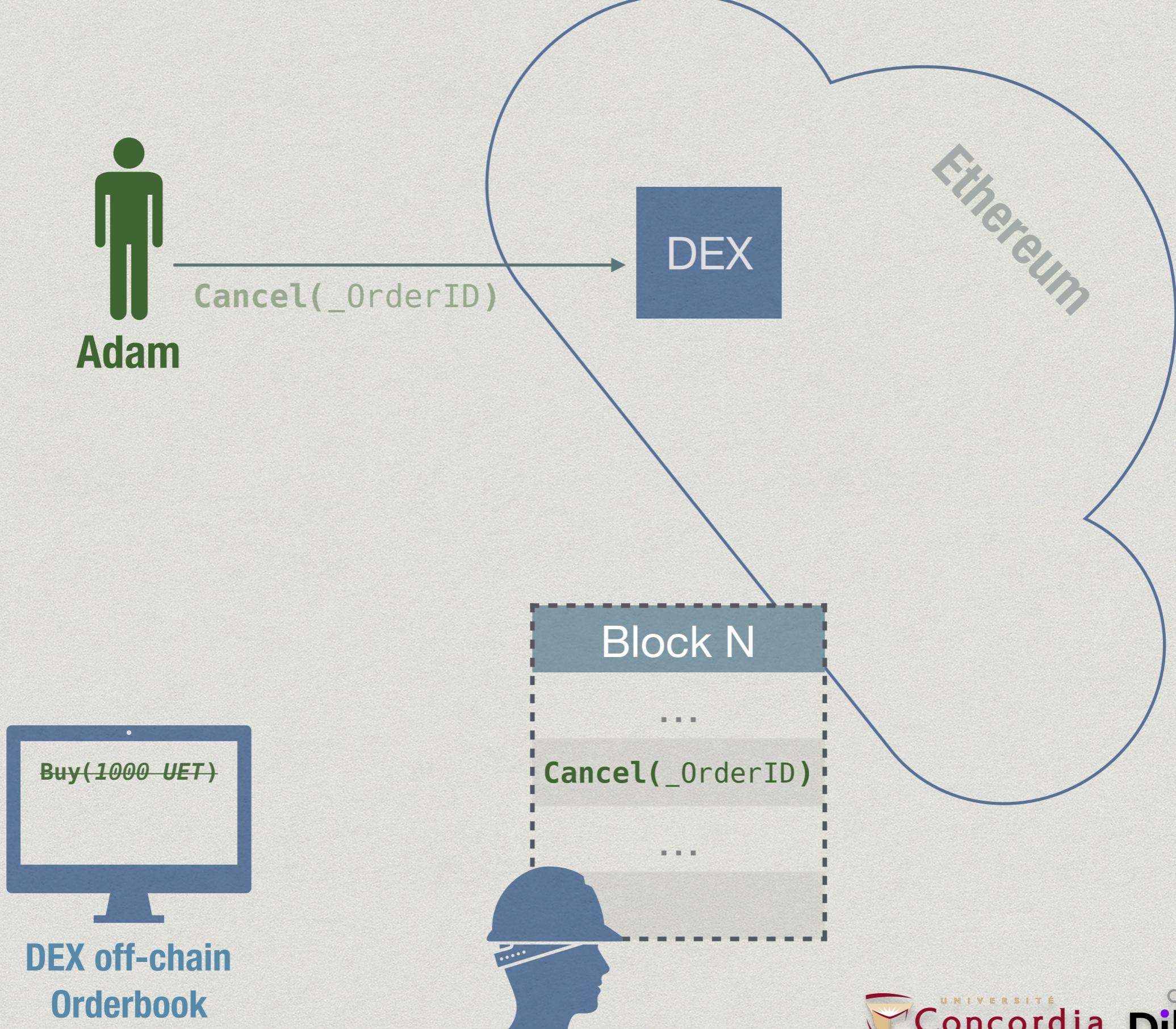
Cancel(_OrderID)

DEX



DEX off-chain
Orderbook







Adam



“Walter”



DEX off-chain
Orderbook

Cancel(_OrderID)



Ethereum



Adam

`Cancel(_OrderID)`



“Walter”

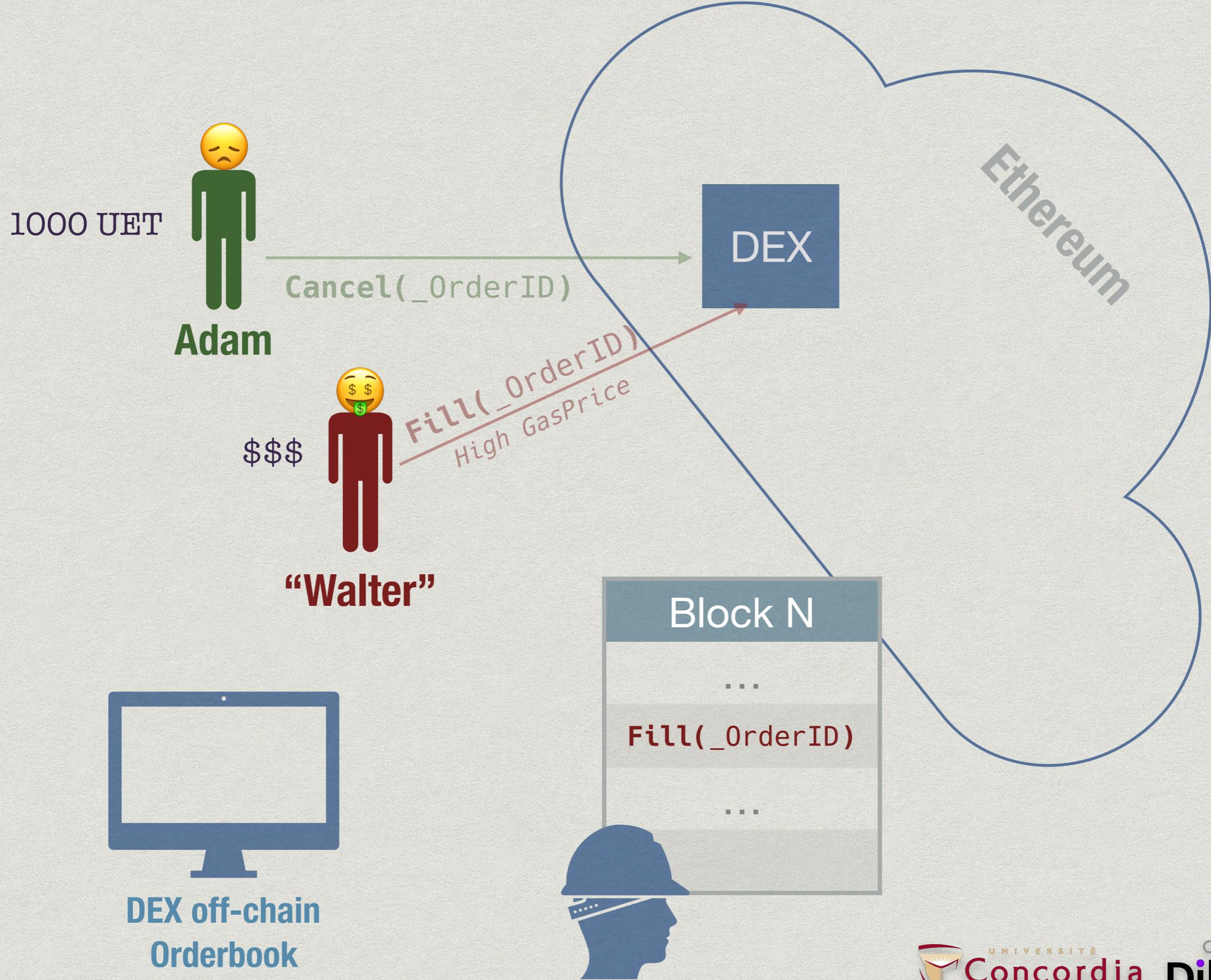
`Fill(_OrderID)`
High GasPrice

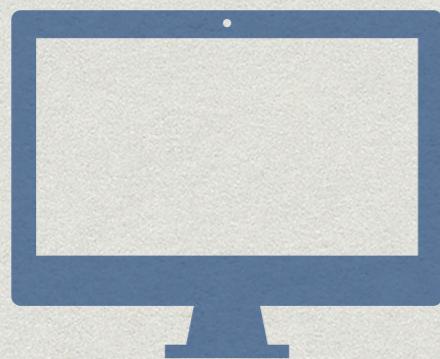
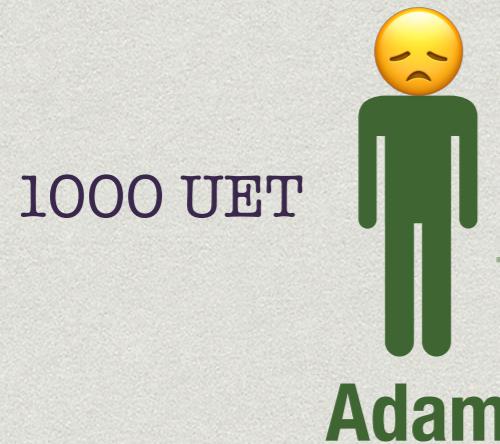
DEX



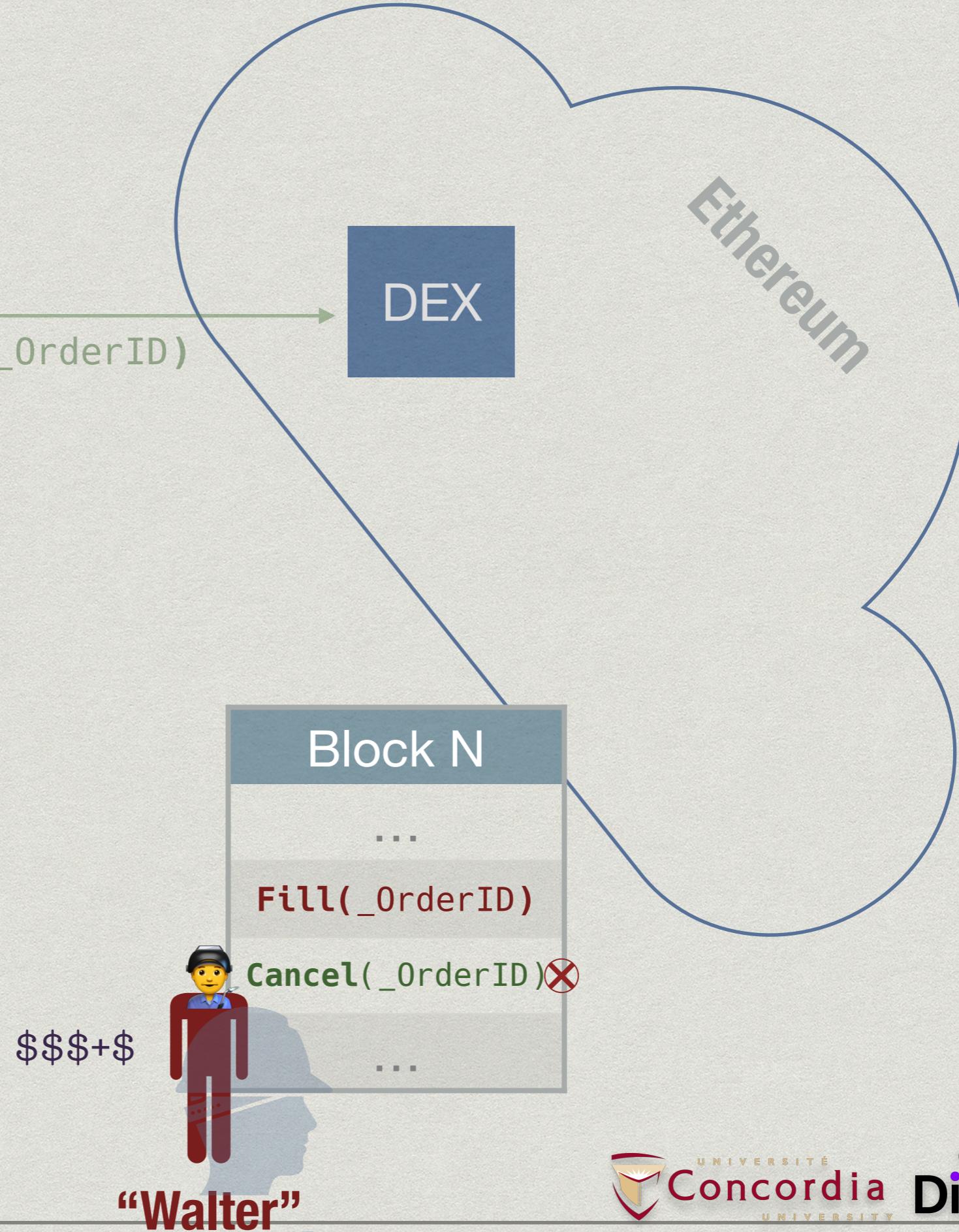
**DEX off-chain
Orderbook**

Ethereum





**DEX off-chain
Orderbook**





New York Stock Exchange
October 1929



WHAT DO THESE 3 STORIES HAVE IN COMMON?

(Traditional) Front-running is a course of action where someone:

- * Benefits from early access to market information about upcoming transactions and trades
- * typically because of a privileged position along the transmission of this information

Blockchain Front-running

- * Everyone (Full Nodes) in the network have access to “Privilege Information”
 - * Miners are in a more privileged position —> Order of transactions in a block they mine
- * Miners can be “bribed” by transaction fee / gasPrice A.K.A *Gas Auction*

Taxonomy of Front-running attacks

Attack Type	Description	Example
Displacement	Not important to the adversary for original function call to run after her function .	Domain Name Registration
Insertion	Important to the adversary for original function call to run after her function .	Buy(_price), _price > Best offer
Suppression	Run Function and delay original function call (Or any other)	Fomo3D

Taxonomy of Front-running attacks

Attack Type	Description	Example
Displacement	Not important to the adversary for original function call to run after her function.	Domain Name Registration
Insertion	Important to the adversary for original function call to run after her function.	Buy(_price), _price > Best offer
Suppression*	Run Function and delay original function call (Or any other)	Fomo3D

* A.K.A Block Stuffing

Variants	Description	Example
Asymmetric	Different function than the original caller	Cancellation Griefing
Bulk	Run Large set of functions	Capped ICO

Cases of Front-running in DApps

DApp Category	Names	Rank
Exchanges	IDEK	1
	ForkDelta, EtherDelta	2
	Bancor	7
	The Token Store	13
	LocalEthereum	14
	Kyber	22
Crypto-Collectible Games (ERC-721 [26])	0x Protocol	23
	CryptoKitties	3
	Ethermon	4
	Cryptogirl	9
	Gods Unchained TCG	12
	Blockchain Cuties	15
	ETH.TOWN!	16
	0xUniverse	18
	MLBCrypto Baseball	19
	HyperDragons	25
Gambling	Fomo3D	5
	DailyDivs	6
	PoWH 3D	8
	FomoWar	10
	FairDapp	11
	Zethr	17
	dice2.win	20
	Ether Shrimp Farm	21
Name Services	Ethereum Name Service	24

- * Top 25 DApps
- * Based on recent user activity
 - * DAppRadar.com
 - * September 2018
- * See the paper for detailed case studies

Key Mitigations

- 1. Transaction Sequencing**
- 2. Confidentiality**
- 3. Design Practices**

Transaction Sequencing

Blockchain itself removes the (miner's) ability to arbitrarily order transactions

- * First-in-first-out (FIFO) is generally not possible on a distributed network
- * Go-Ethereum implementation prioritizes transactions based on their gas price and nonce
- * Off-chain (e.g. Order books in 0x or EtherDelta)
- * Pseudorandom Sorting (e.g. Canonical Transaction Ordering Rule (CTOR) by Bitcoin Cash ABC)

Confidentiality

Limit the visibility of transactions

- * DApp interaction includes the following components:

1	Code of the DApp
2	Current state of the DApp
3	Name of the function being invoked
4	Parameters supplied to the function
5	Address of the contract the function is being invoked on
6	Identity of the sender.

Confidentiality

- * Privacy-Preserving Blockchains, similar to Dark pools in HFT
 - * (2,3,4)-confidential

1	Code of the DApp
2	Current state of the DApp
3	Name of the function being invoked
4	Parameters supplied to the function
5	Address of the contract
6	Identity of the sender.

Confidentiality

- * Privacy-Preserving Blockchains, similar to Dark pools in HFT
 - * (2,3,4)-confidential
- * **Commit and Reveal.**
 - * (3,4)- or (4)-confidentiality
 - * Namecoin, ENS
 - * Collateralized? Leaks information

1	Code of the DApp
2	Current state of the DApp
3	Name of the function being invoked
4	Parameters supplied to the function
5	Address of the contract
6	Identity of the sender.

Confidentiality

- * Privacy-Preserving Blockchains, similar to Dark pools in HFT

- * (2,3,4)-confidential

- * Commit and Reveal.

- * (3,4)- or (4)-confidentiality

- * Namecoin, ENS

- * Collateralized? Leaks information

- * Enhanced Commit and Reveal: LibSubmarine

- * (3,4,5)-confidentiality+

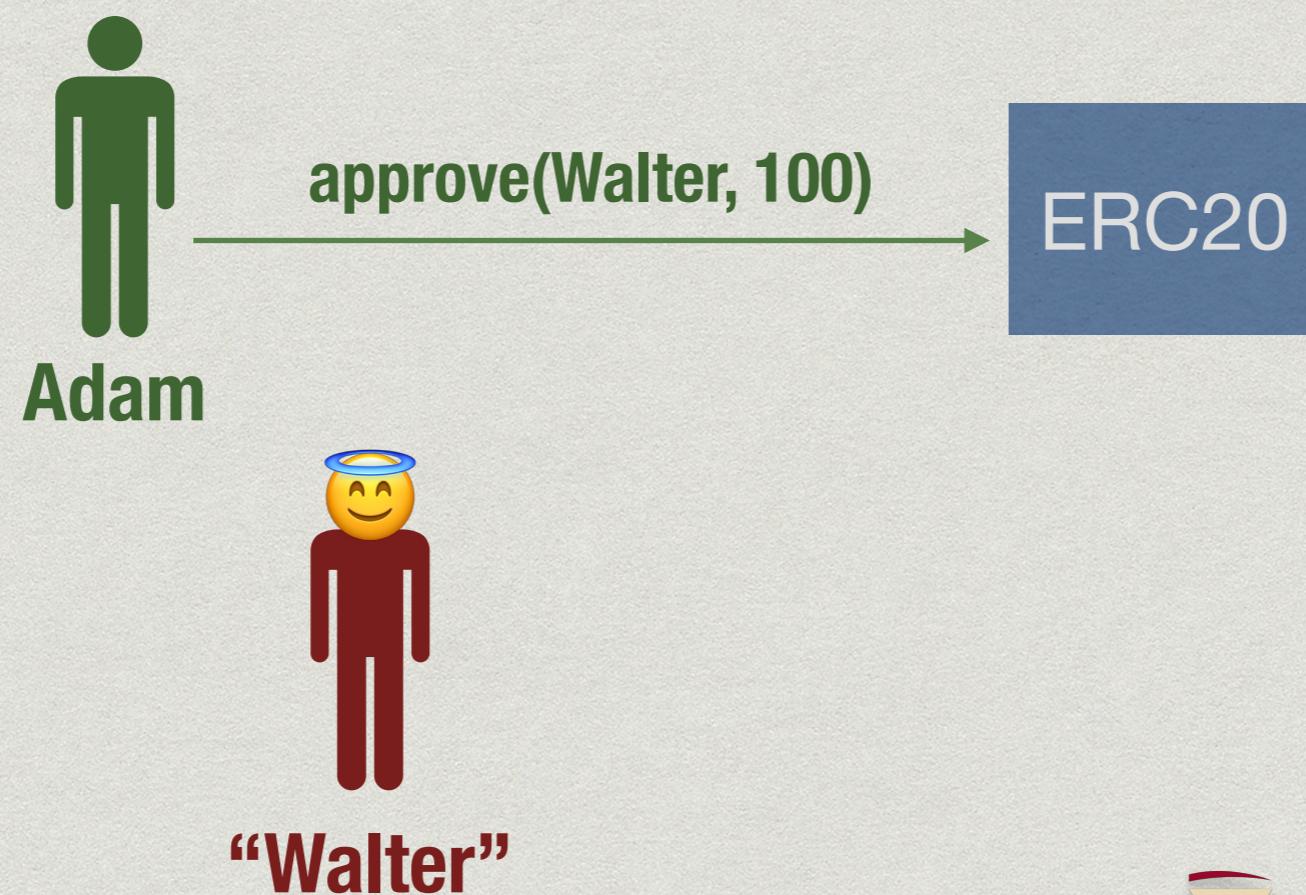
1	Code of the DApp
2	Current state of the DApp
3	Name of the function being invoked
4	Parameters supplied to the function
5	Address of the contract
6	Identity of the sender.

Design Practices

- * Assume front-running is unpreventable —> Remove any benefit from it
 - * remove the importance of transaction ordering or time
- * **Call market** design instead of a time-sensitive order book
- * ERC20 Allowance functionality, “`approve()`”, was not designed with front-running in mind.

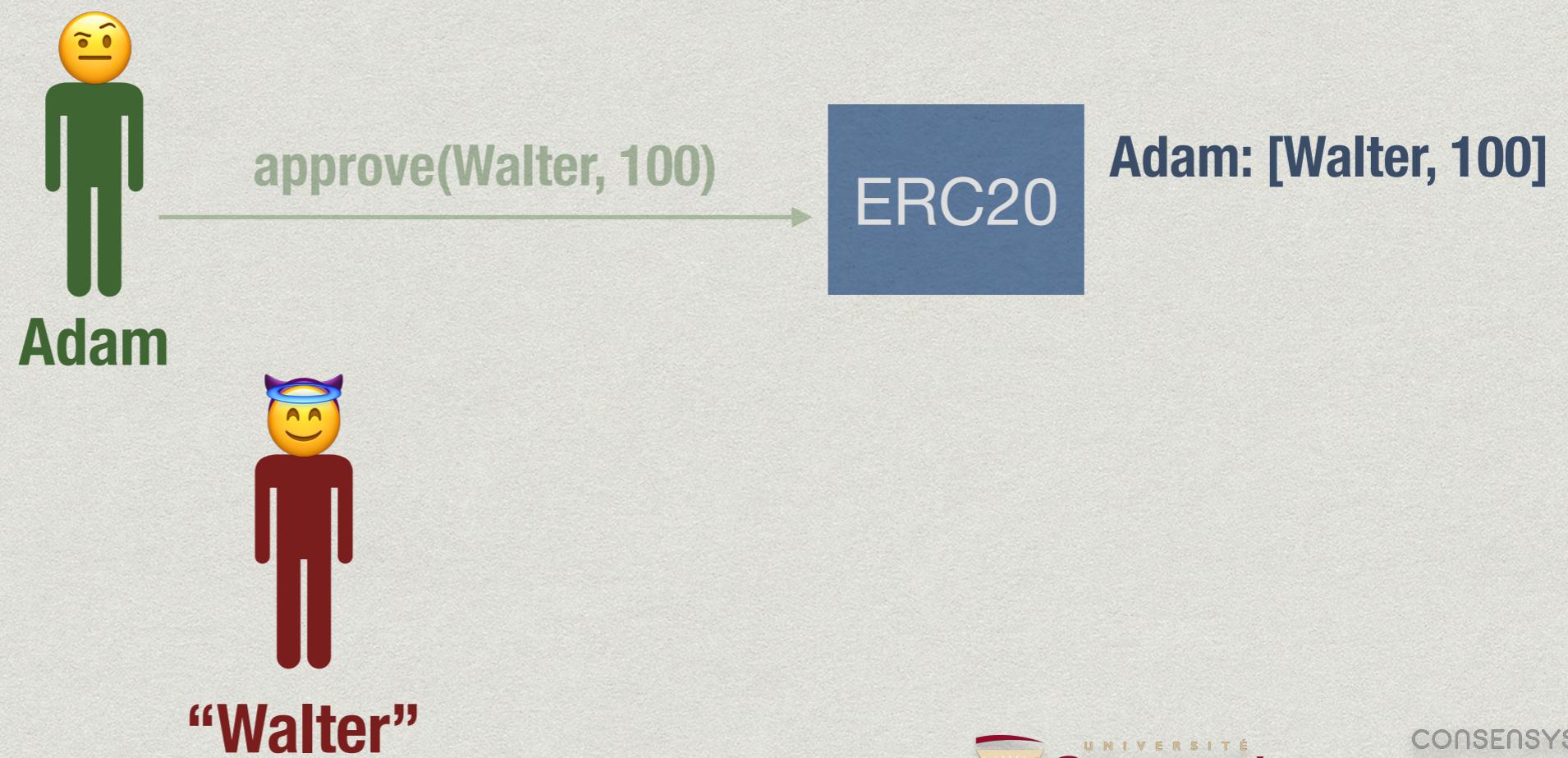
Design Practices (cont.)

- * ERC20 Allowance functionality, “approve()”, was not designed with front-running in mind.



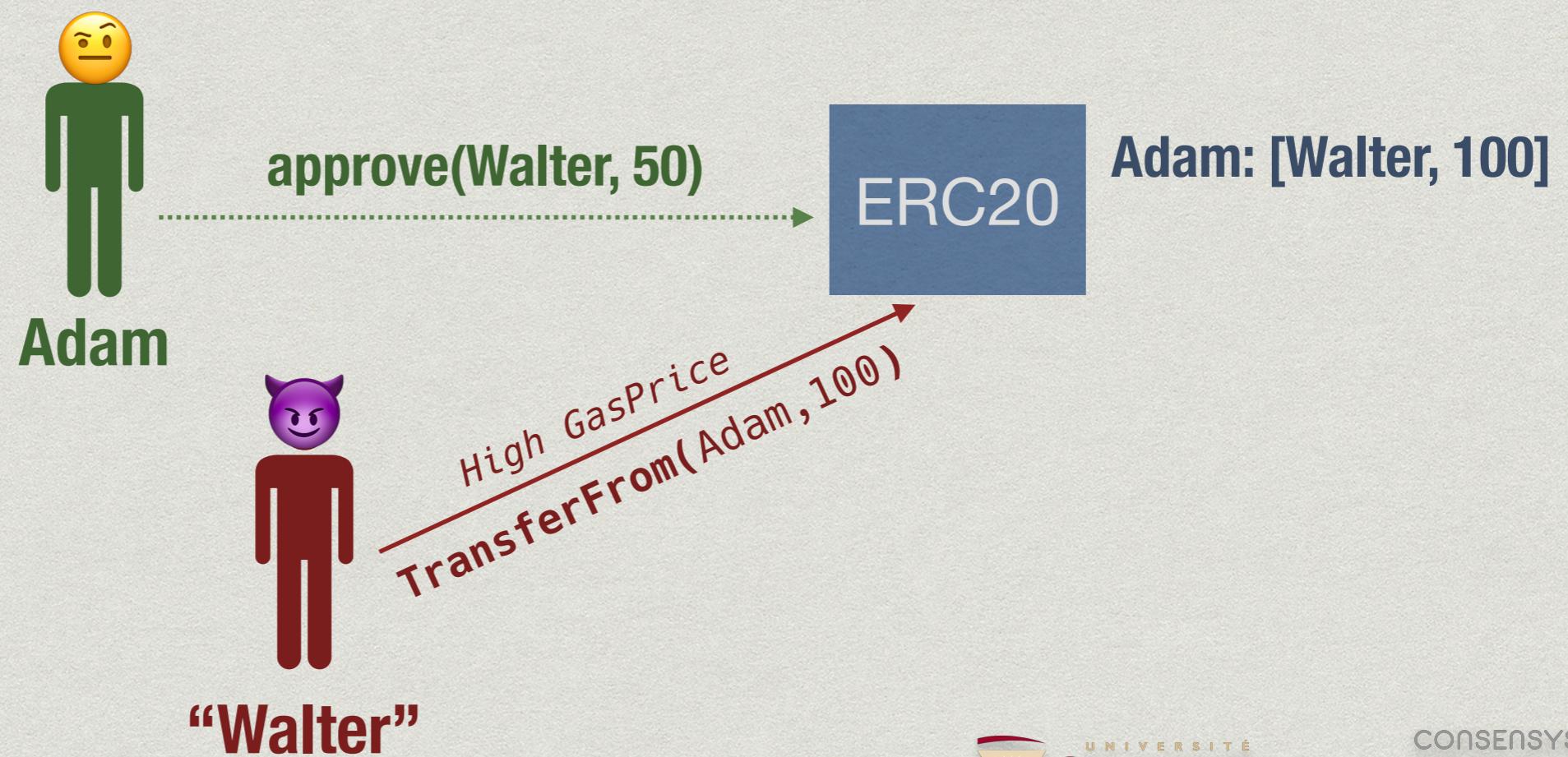
Design Practices (cont.)

- * ERC20 Allowance functionality, “approve()”, was not designed with front-running in mind.



Design Practices (cont.)

- * ERC20 Allowance functionality, “approve()”, was not designed with front-running in mind.

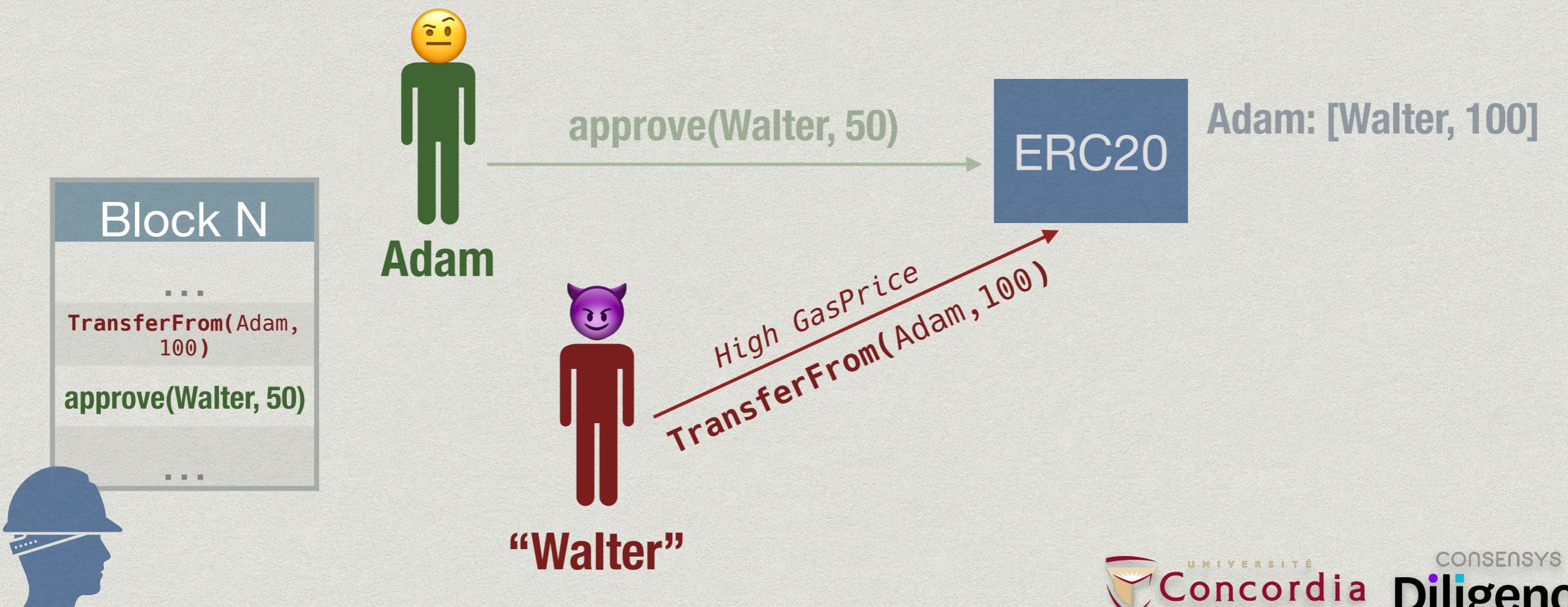


UNIVERSITÉ
Concordia
UNIVERSITY

CONSENSYS
Diligence

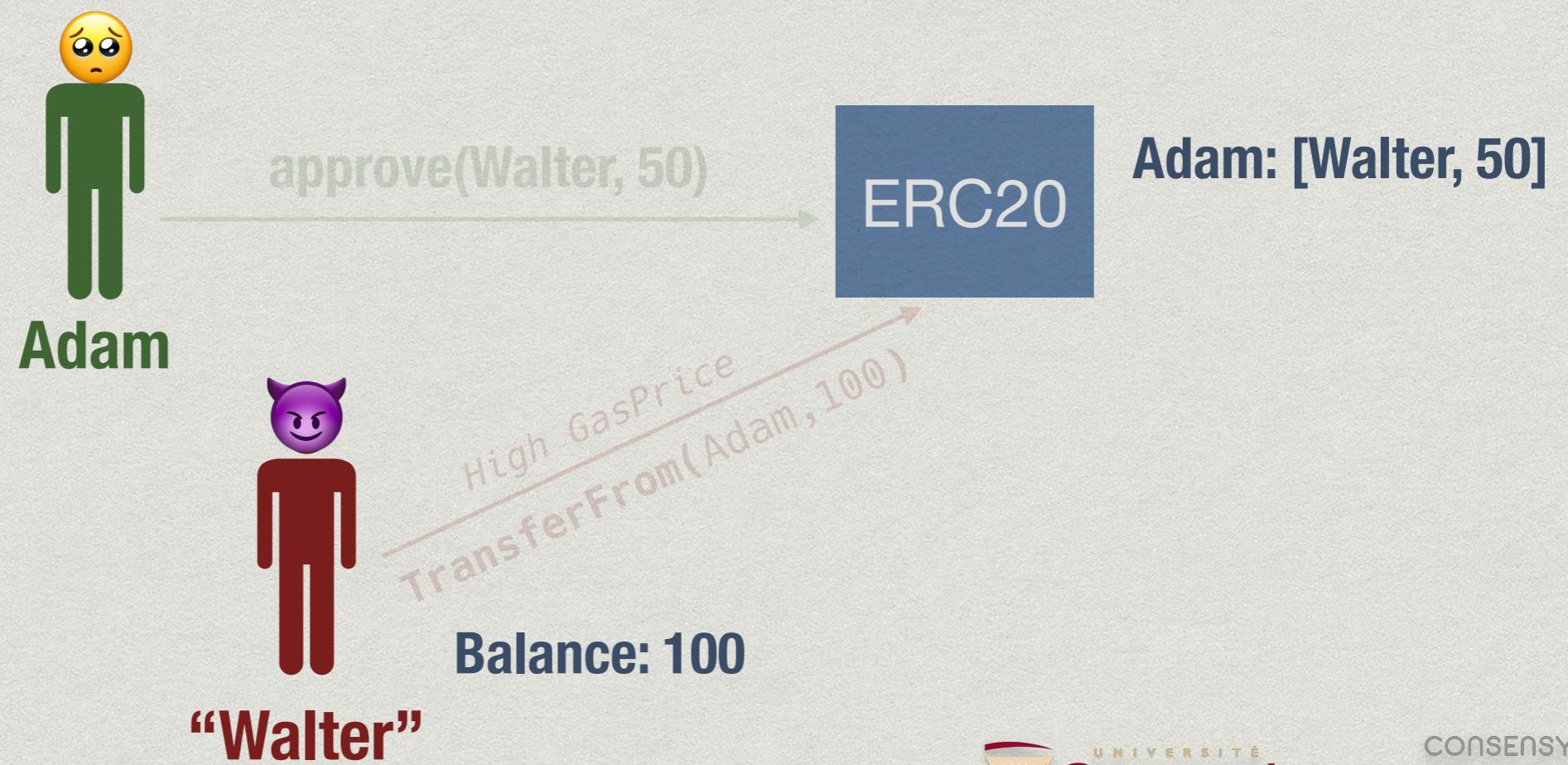
Design Practices (cont.)

- * ERC20 Allowance functionality, “approve()”, was not designed with front-running in mind.



Design Practices (cont.)

- * ERC20 Allowance functionality, “`approve()`”, was not designed with front-running in mind.
- * *`decreaseApproval()` / `increaseApproval()`* were proposed



Concluding Remarks

- * Front-running is a pervasive issue in Ethereum DApps
- * Increase awareness of these type of attacks
- * Usable DApp layer & Blockchain-level solutions
- * We highlight this as an important research area.

SOK: TRANSPARENT DISHONESTY FRONT-RUNNING ATTACKS ON BLOCKCHAIN.

THANK YOU

Take-home readings:

- The paper itself : <https://arxiv.org/abs/1902.05164>
- Front-Running - Insider Trading under the Commodity Exchange Act (1988) - Jerry W. Markham
- Flash Boys: A Wall Street Revolt
 - Youtube: "Brad Katsuyama - The Stock Market had become an Illusion"

This is a website



UNIVERSITÉ

Concordia
UNIVERSITY

SHAYAN.ES

CONSENSYS

Diligence