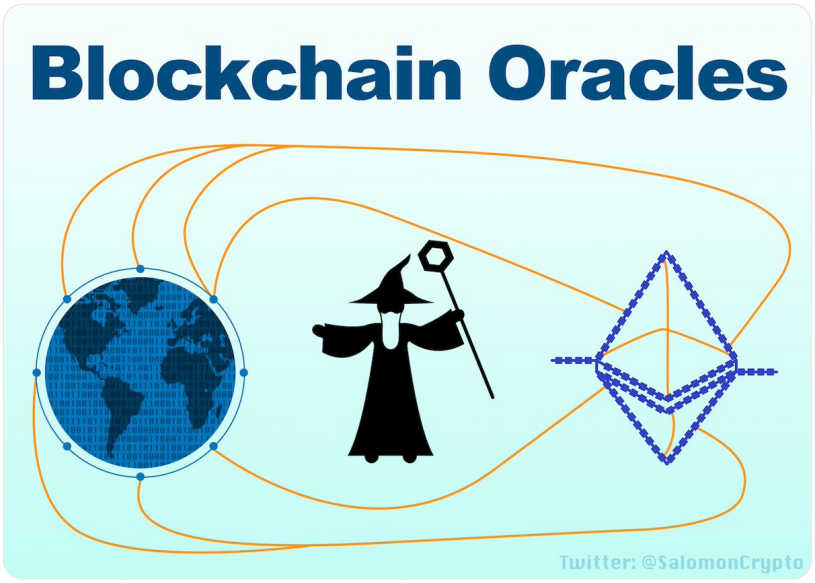
Haym Salomon @SalomonCrypto
Aug 10 · 21 tweets · SalomonCrypto/status/1557211373288898560

(1/20) @ethereum, Oracles and @chainlink: The Communication Layer of Web3

How do smart contracts get info from outside the Ethereum blockchain? How can a protocol interact with a web2 service? How will The World Computer integrate with The Real World?

This thread has answers!



(2/20) The problem: blockchains cannot pull in or push out data to any external system (by design). They are isolated networks - a computer without Internet.

Without internet? In 2022?!??!!

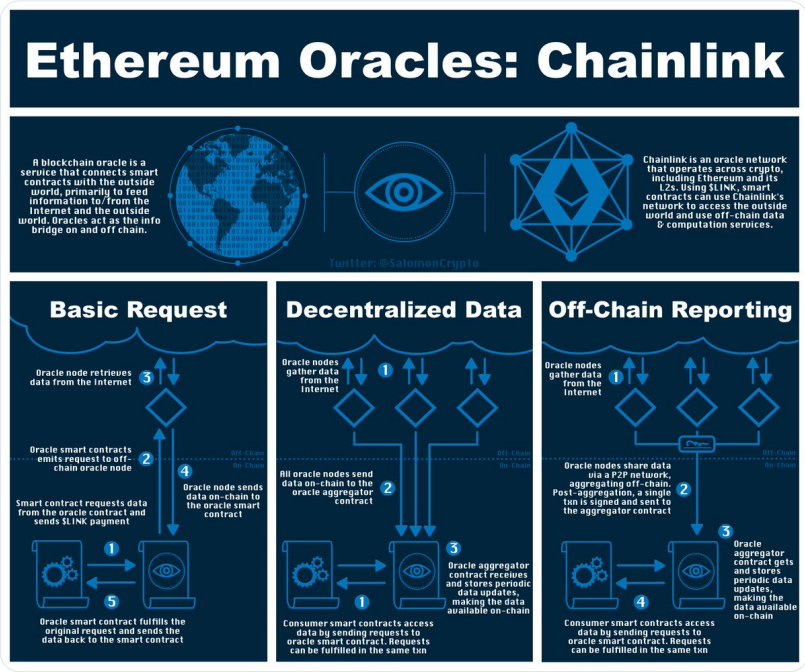
What Is the Blockchain Oracle Problem? Why Can't Blockchains Solve…

A blockchain oracle is a piece of middleware that facilitates communication between blockchain-based smart contracts and off-chain systems.

https://blog.chain.link/what-is-the-blockchain-oracle-problem/

(3/20) Fear not! While not yet solved, the blockchain oracle problem is being dismantled. Thanks to the @chainlink team, The World Computer is no longer disconnected!

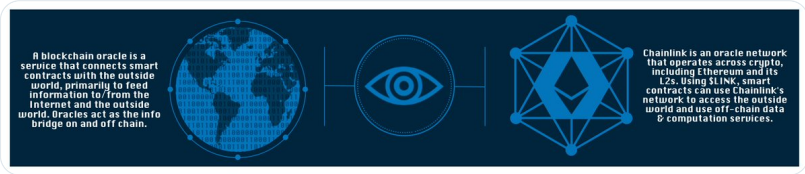
The challenge is now increasing bandwidth, latency and reliability.



(4/20) A blockchain oracle is an entity that sits between @ethereum and the outside world, moving information between the two.

An oracle is made up of two parts:

- Oracle Smart Contracts
- Oracle External Interface (Node)



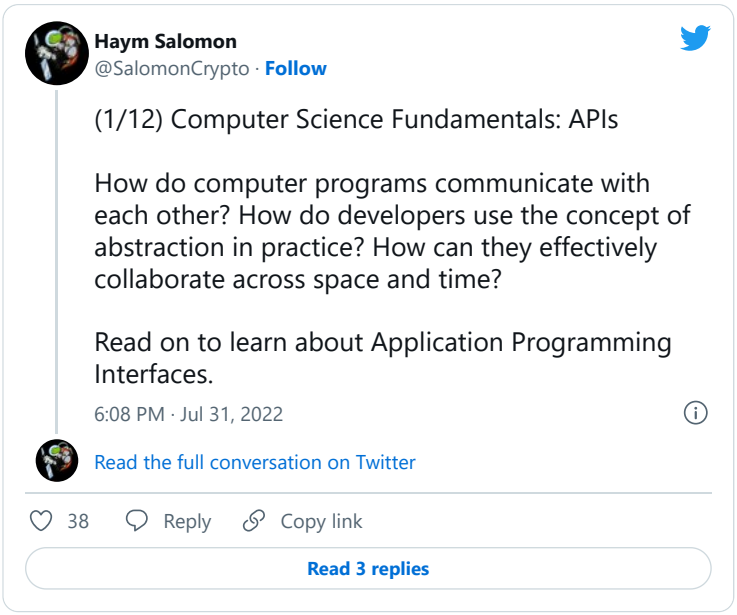
(5/20) Oracle smart contracts are the code that exist within @ethereum, processing requests and coordinating between internal and external clients.

(6/20) Oracle nodes are the computers and servers that are in communication with @ethereum.

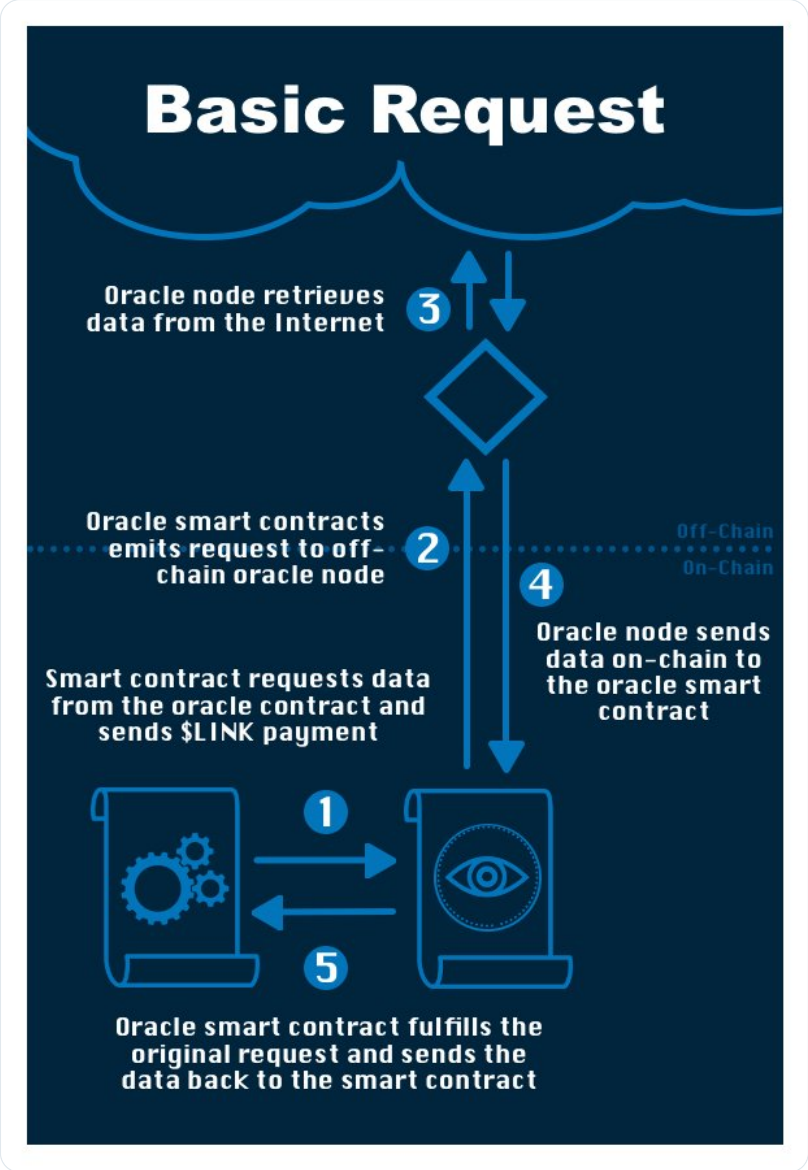
Nodes are centralized entities capable of incredibly quick and cheap computation. They are also connected to external data feeds and can publish that data on-chain.

(7/20) Oracle nodes and smart contracts achieve this two way communication by staying closely in sync and actively watching each-other across the on/off-chain barrier.

Entities on both sides have someone they can use to pass information back and forth via a robust set of APIs.



(8/20) The Basic Request Model is a generalized structure for passing data on/off-chain. A client smart contract will send the oracle smart contract the API request and a $LINK payment. The oracle network bridges the message, making changes and providing data where it's needed.



(9/20) Let's say a protocol is providing synthetic exposure to the price of gold and needs a feed of the price of gold from the real world. It could use a basic request to retrieve the data.

Unfortunately, the reality is that oracle work is very expensive.

(10/20) Furthermore, while basic requests uses @chainlink's technology to send messages in and out of @ethereum, they are still only as safe and secure as the nodes they connect to.

What happens if a malicious oracle node changes the price of gold before it publishes it?

(11/20) It's not just bad guys. What about a disaster scenario: server gets hacked, data center gets flooded, some weird configuration bricks the node forever.

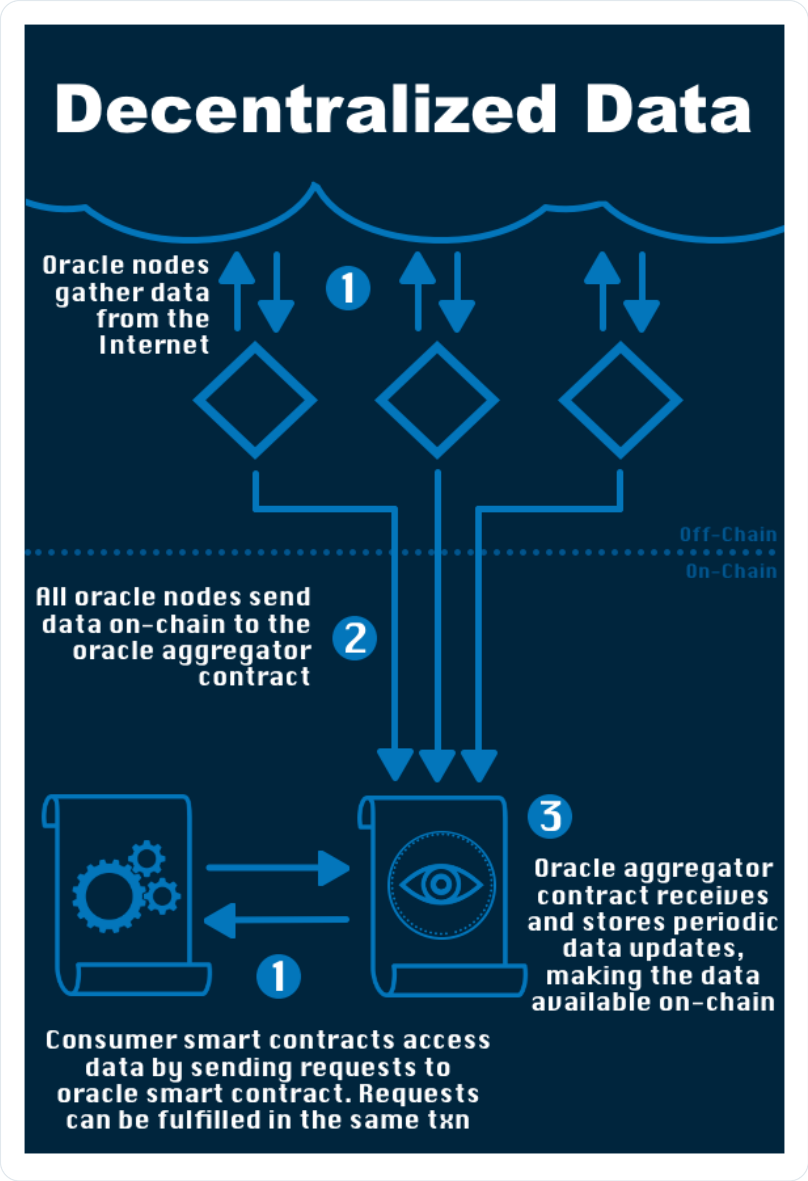
There's a reason the @ethereum community is so nuts about decentralization.

(12/20) Fortunately, @chainlink has an answer: Decentralized Oracle Network (DON)

A DON combines multiple independent oracle node operators and multiple reliable data sources to establish end-to-end decentralization.

(13/20) The Decentralized Data Model combines three layers of decentralization:

- data source, gathering data from multiple well-vetted feeds
- node, relying on a network of operators
- chain, publishing data across blockchain ecosystems



(14/20) Each feed is built and funded by the community of users who rely on accurate, up-to-date data in their smart contracts.

As more users rely on and contribute to a data feed, the quality of the data feed improves.

(15/20) Here is the list of data feeds available currently. If you click through you can see performance metrics (eg how often it is updated), users, networks, etc.

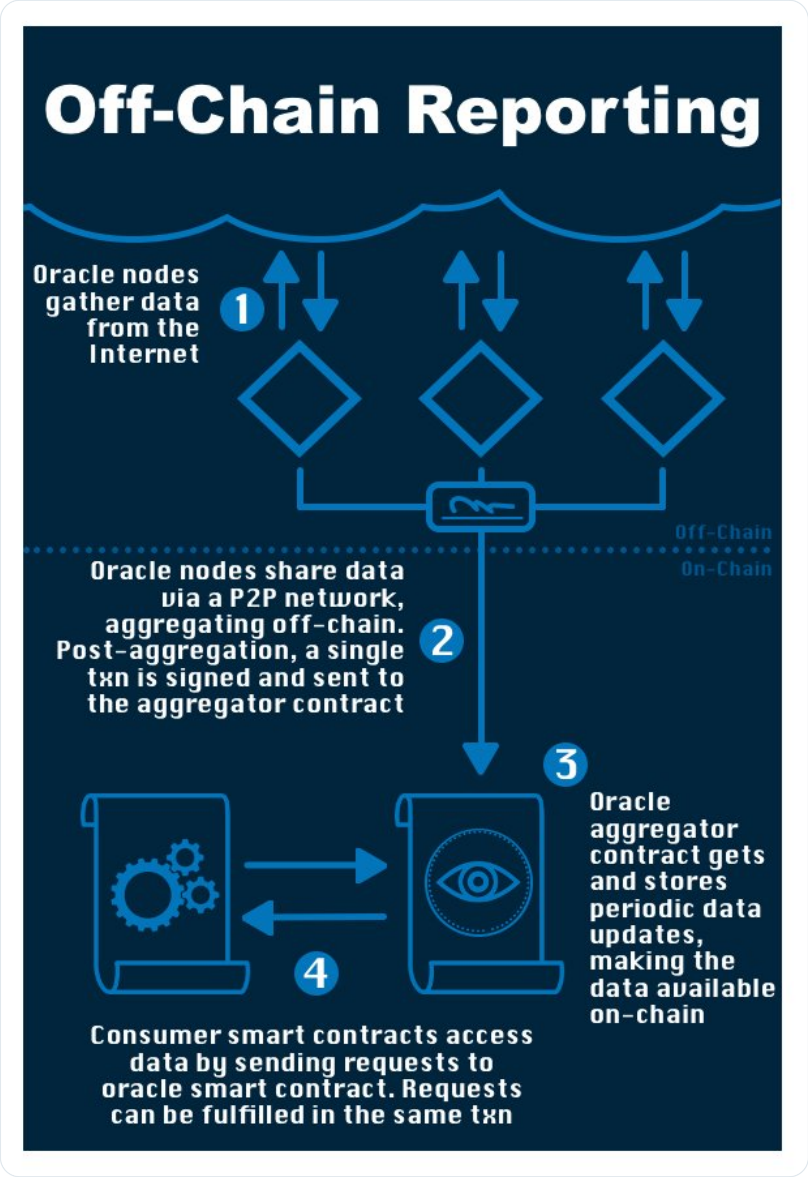
Decentralized Data Feeds | Chainlink
Access the most versatile list of decentralized pricing networks to build your smart contracts, powered by the Chainlink network
https://data.chain.link/

(16/20) While the decentralized data model has boosted the reliability and security of the oracle network, it has come at a high cost.

The most expensive part of oracle-work is paying for gas; in the vanilla decentralized data model, every node pays every time they publish.

(17/20) Fear not, the decentralized data model is just the beginning! On to the next evolution in oracle technology: Off-Chain Reporting (OCR).

OCR leverages the same decentralized principals, but moves the computational and gas-expensive work of aggregating data off-chain.



(18/20) In OCR, all nodes communicate using a peer to peer network. During the communication process, a lightweight consensus algorithm runs: each node reports its data observation and signs it. A single aggregate txn is then transmitted to the oracle smart contract.

> **A simple analogy**
>
> Imagine ordering 10 items from an online store. Each item is packaged separately and posted separately, meaning postage and packaging costs must be applied to each one, and the carrier has to transport 10 different boxes.
>
> OCR, on the other hand, packages all of these items into a single box and posts that. This saves postage and packaging fees and all effort the carrier associates with transporting 9 fewer boxes.

(19/20) OCR is a huge improvement on the decentralized data model. The oracle node network can scale exponentially bigger, increasing decentralization, security and reliability.

With OCR, there will always be only 1 txn per update, regardless of network size.

(20/20) 7 years ago, @VitalikButerin launched @ethereum, giving us The World Computer.

5 years ago, @SergeyNazarov, @streamOfCo & @chainlink begin connecting The World Computer to the Internet.

Today, we have the equivalent of dial-up...

Can you even imagine the broadband-era?

Like what you read? Help me spread the word by retweeting the thread (linked below).

Follow me for more explainers and as much alpha as I can possibly serve.

• • •