

The background of the slide is filled with various light green, 3D-rendered geometric shapes. These include a large sphere on the left, a smaller sphere in the upper right, a cube in the top left, a rectangular frame in the upper center, a cone on the right, a wavy line in the top right, a cylinder in the lower left, a torus (donut shape) in the lower center, and several other smaller spheres and cones scattered throughout. The shapes are rendered with soft shadows, giving them a sense of depth and volume.

Swarm

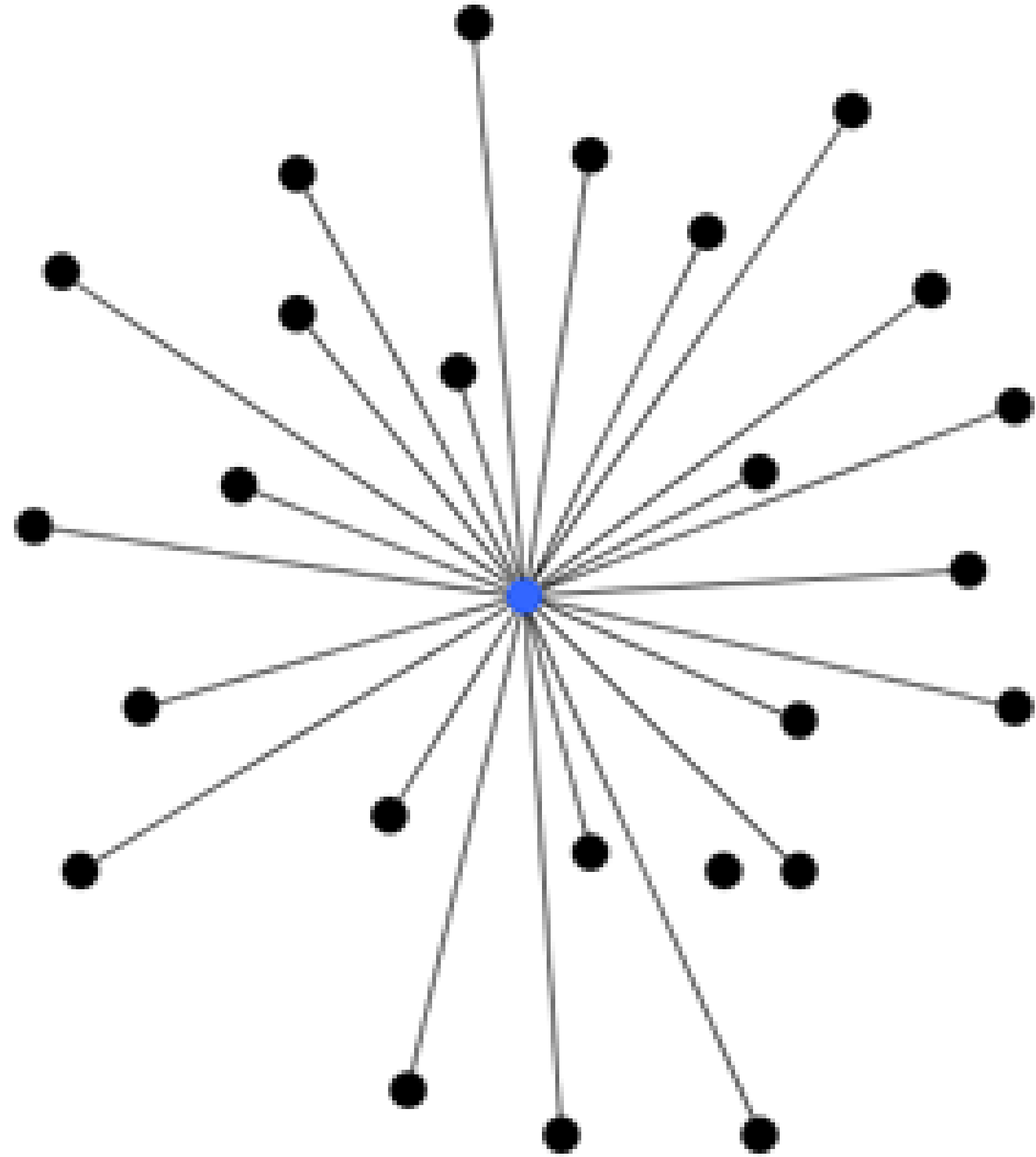
A distributed storage by ethereum.

by Abhishek Kedia

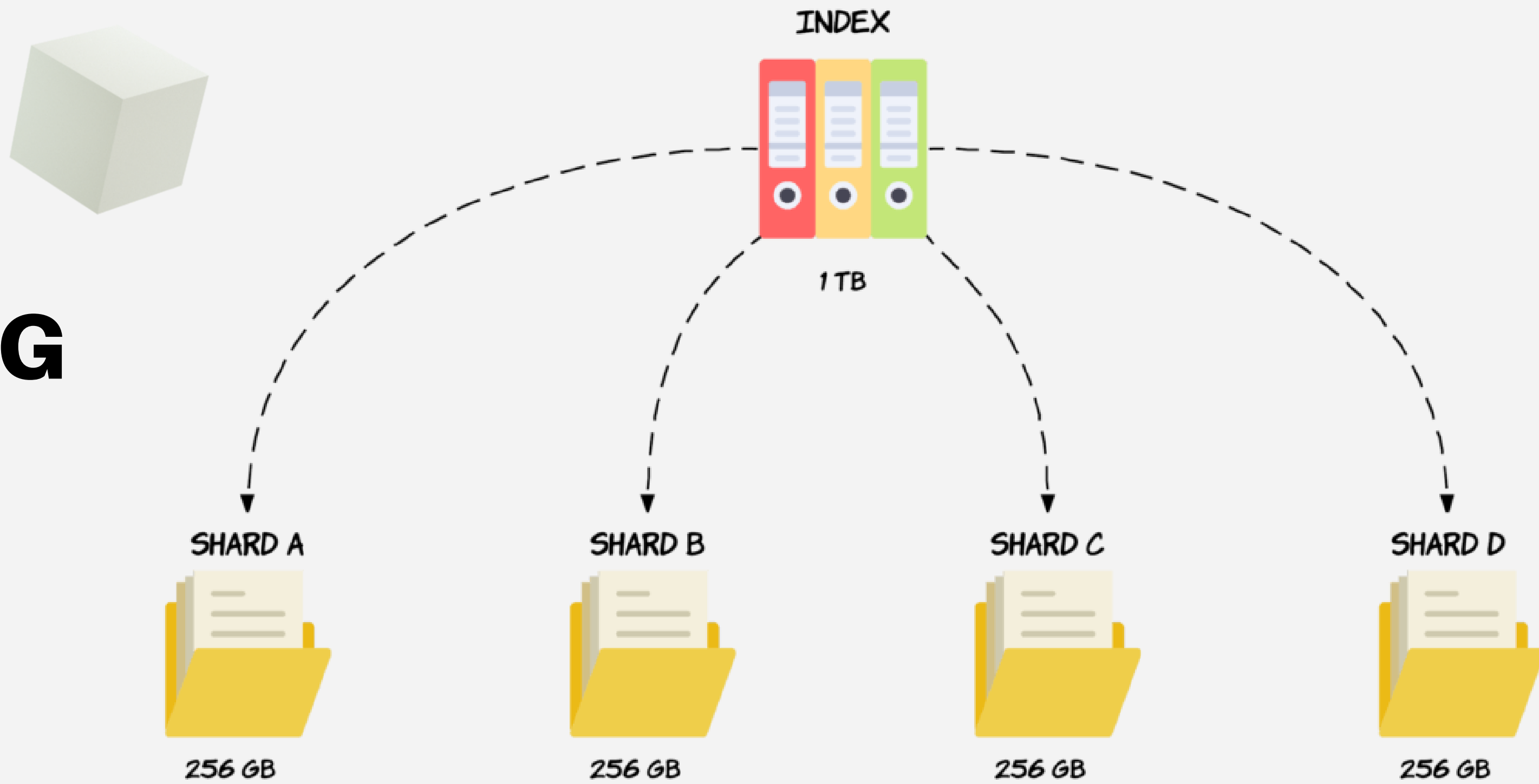
**What is
distributed
storage?**

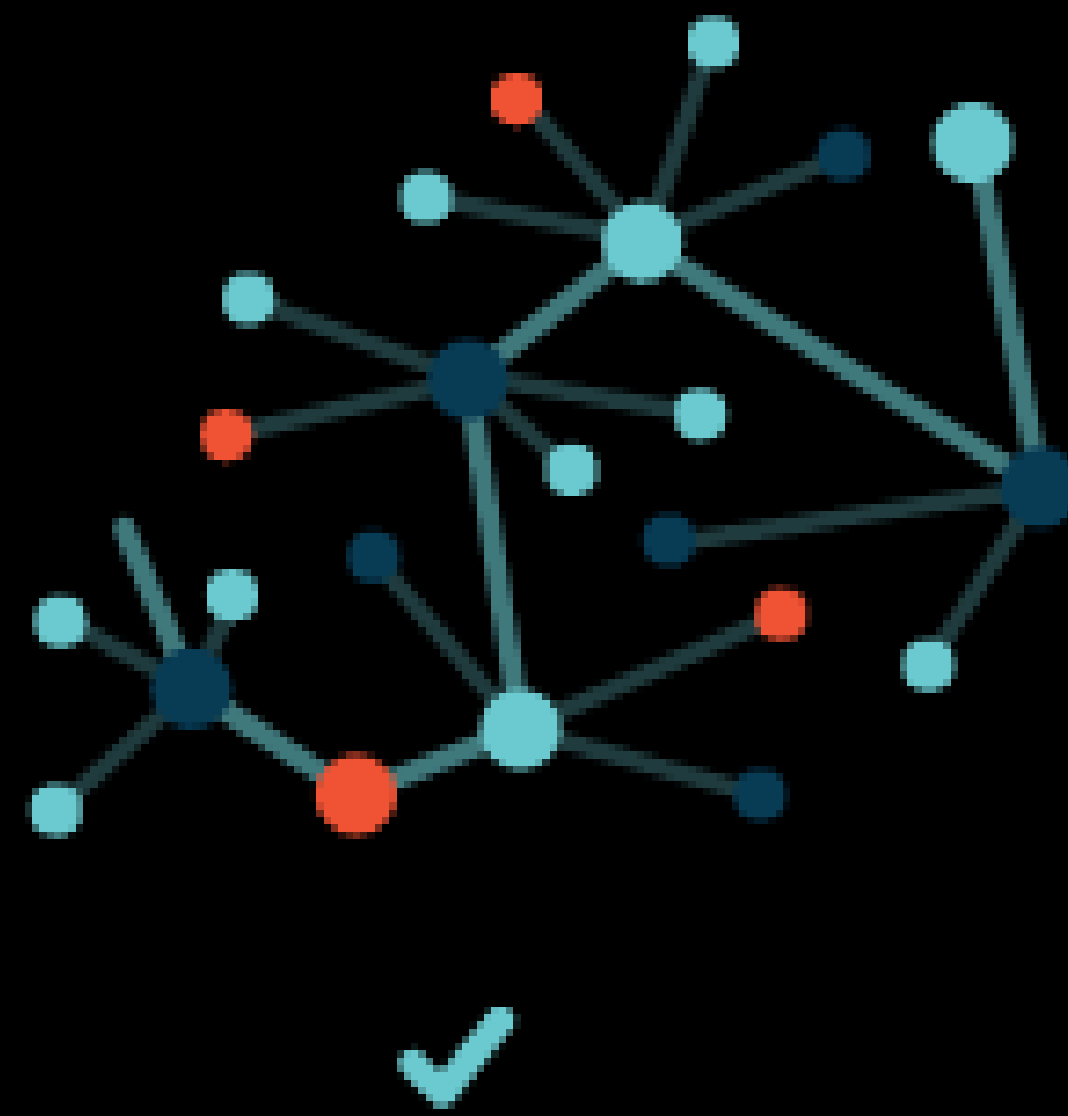
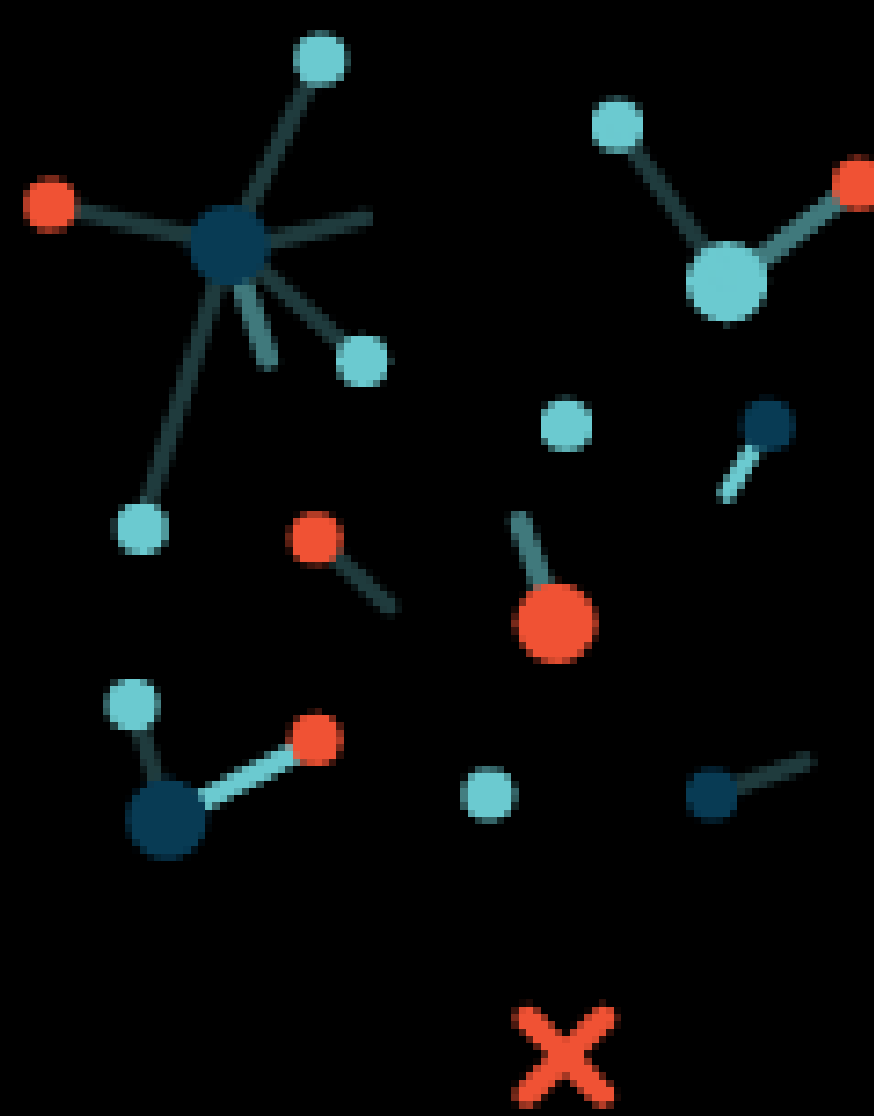
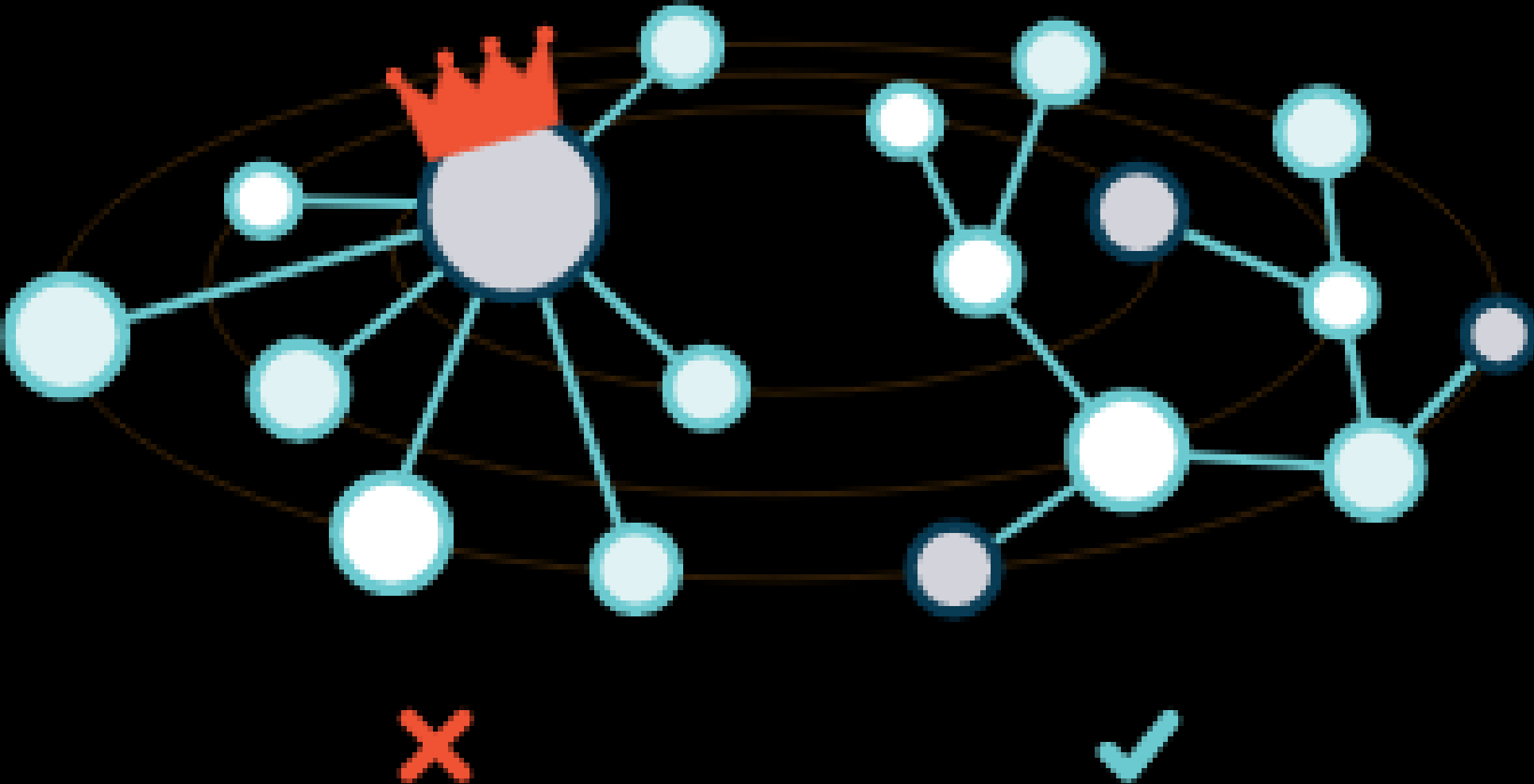


Centralised Storages



SHARDING

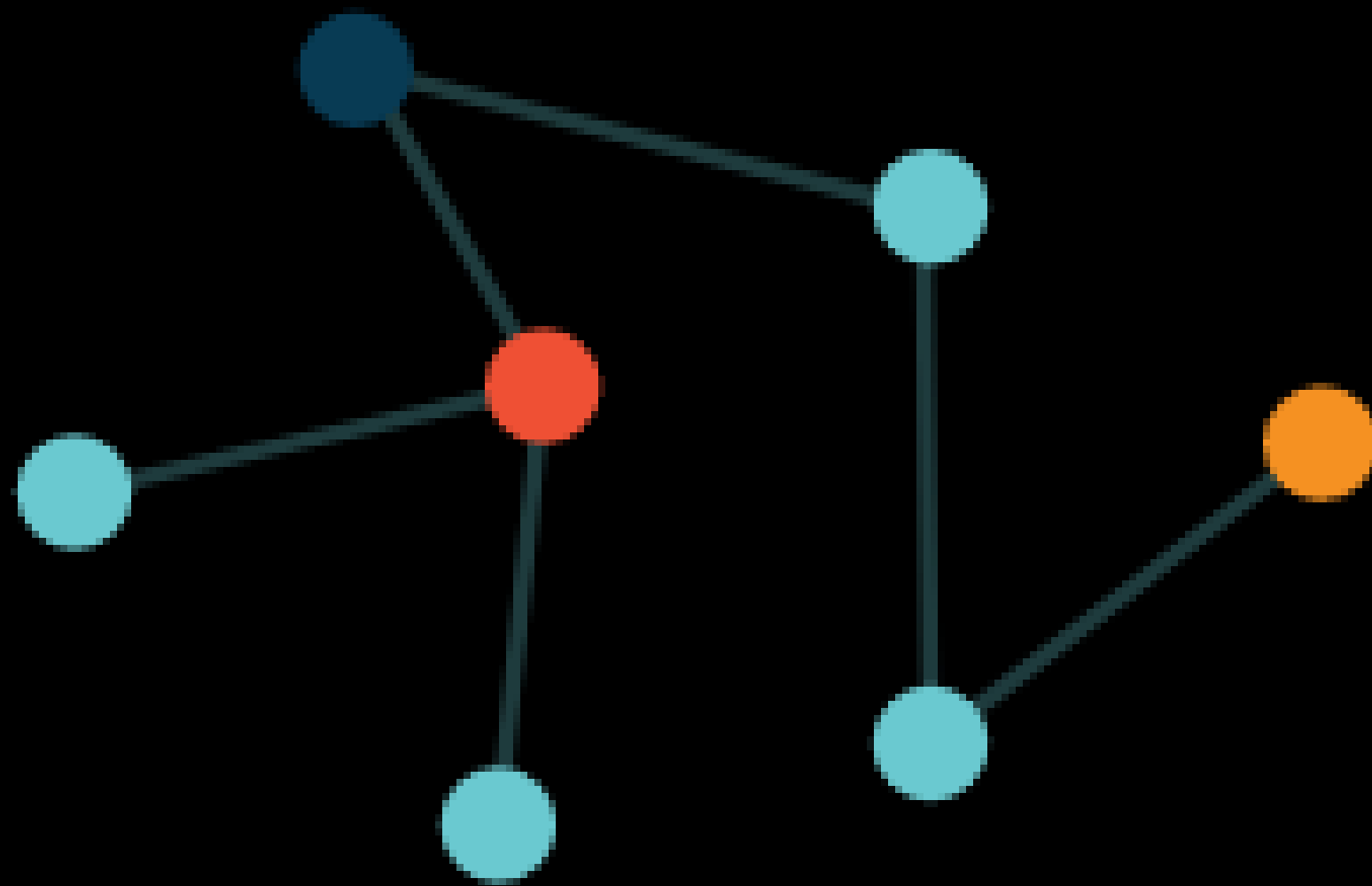






The Rise of distributed storages

Swarm is a censorship resistant,
permissionless, distributed storage and
communication infrastructure.



**How is it
different from
IPFS or
bittorrent?**

How does Swarm work?

Swarm defines 3 crucial notions:

Chunks

Chunks are pieces of data of limited size (max 4K), the basic unit of storage and retrieval in the Swarm. The network layer only knows about chunks and has no notion of file or collection.

Reference

A reference is a unique identifier of a file that allows clients to retrieve and access the content.

Manifest

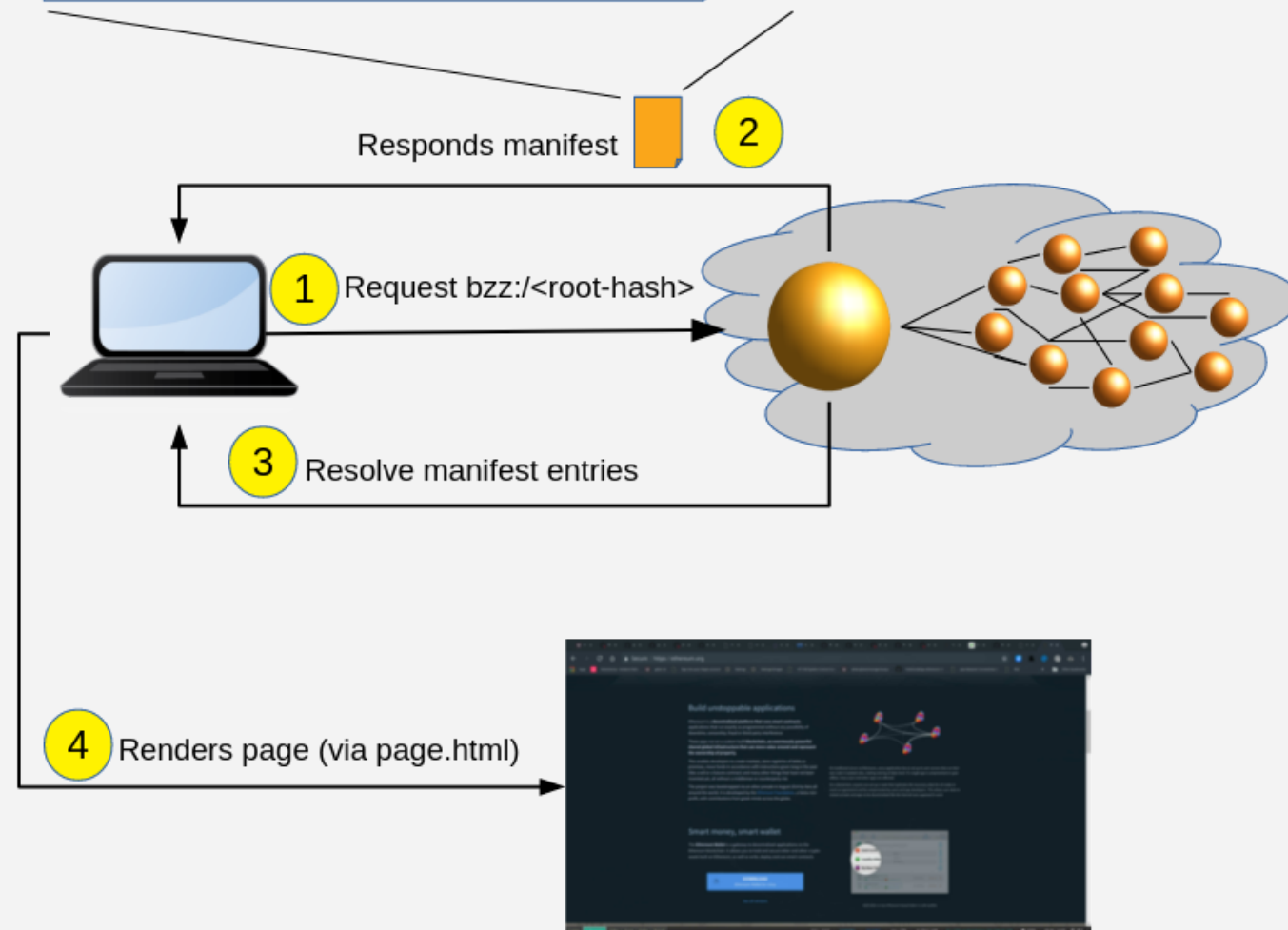
A manifest is a data structure describing file collections; they specify paths and corresponding content hashes allowing for URL based content retrieval.



root-hash: 6a4eb323... corresponds to manifest, represents a directory:

```
<hash: 8afec4234...> page.html  
<hash: 7d32a09df...> page.css  
<hash: ac43498e1...> page.js  
<hash: 5bb35788e...> img1.jpg  
<hash: 102ef1e28...> other.js  
<...> ...  
<...> ...
```

Page.html contains references as local directory entries:



The nodes constituting the Swarm all dedicate resources (diskspace, memory, bandwidth and CPU) to store and serve chunks.



**Who stores the
data?**



The process by which chunks get to their address is called syncing and is part of the protocol. Nodes that later want to retrieve the content can find it again by forwarding a query to nodes that are close to the content's address. Indeed, when a node needs a chunk, it simply posts a request to the Swarm with the address of the content, and the Swarm will forward the requests until the data is found.

Swarm is similar to a traditional distributed hash table (DHT).



Let's Try it.

