



Reimagining Open Educational Resources

Stephen Downes

National Research Council Canada

Open Education Conference

November 10, 2020

<https://www.downes.ca/534>

This session introduces and demonstrates content-addressable resources for education, a set of tools and processes for the creation and storage of learning resources in a distributed peer-to-peer network.



Learning Outcomes

You will be able to:

- describe how content addressing works
- describe the use of content addressing to enable a secure and distributed resource network
- create and add their own open educational resources to the network
- access and reuse resources from the network
- appreciate how content addressing provides an alternative to license-based OER

Issues for Open Educational Resources

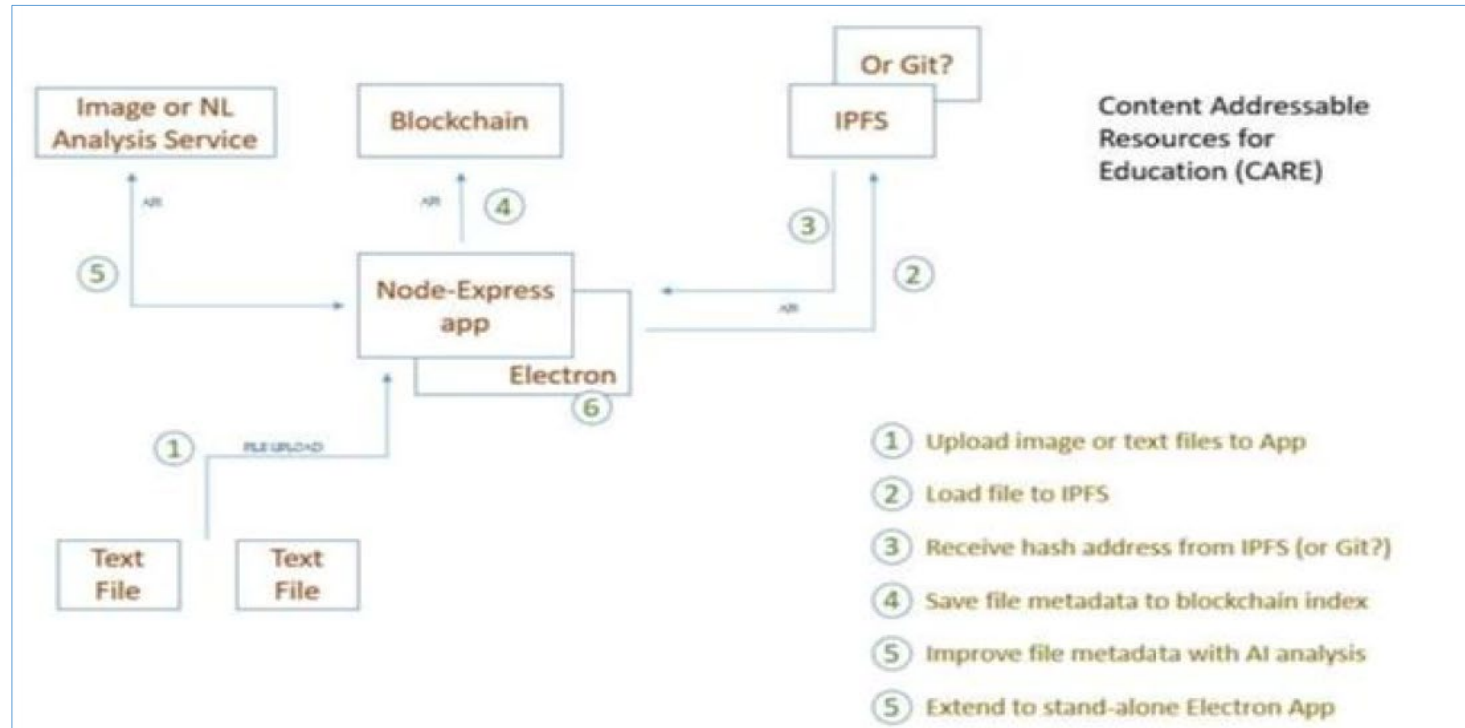
- Licensing alone is not enough

“CC licenses and tools are proving indispensable in certain domains, most notably in the education and research sectors. However, they do not entirely address mainstream content sharing on the internet today. Indeed, most sharing occurs on proprietary platforms designed to keep users within their own systems.”

- Resources are no longer community-based

“Our existing licenses and tools do not fully address the collateral damage caused by exploitative, decontextualized, unethical, and antisocial reuse of shared content. We cannot turn a blind eye.”

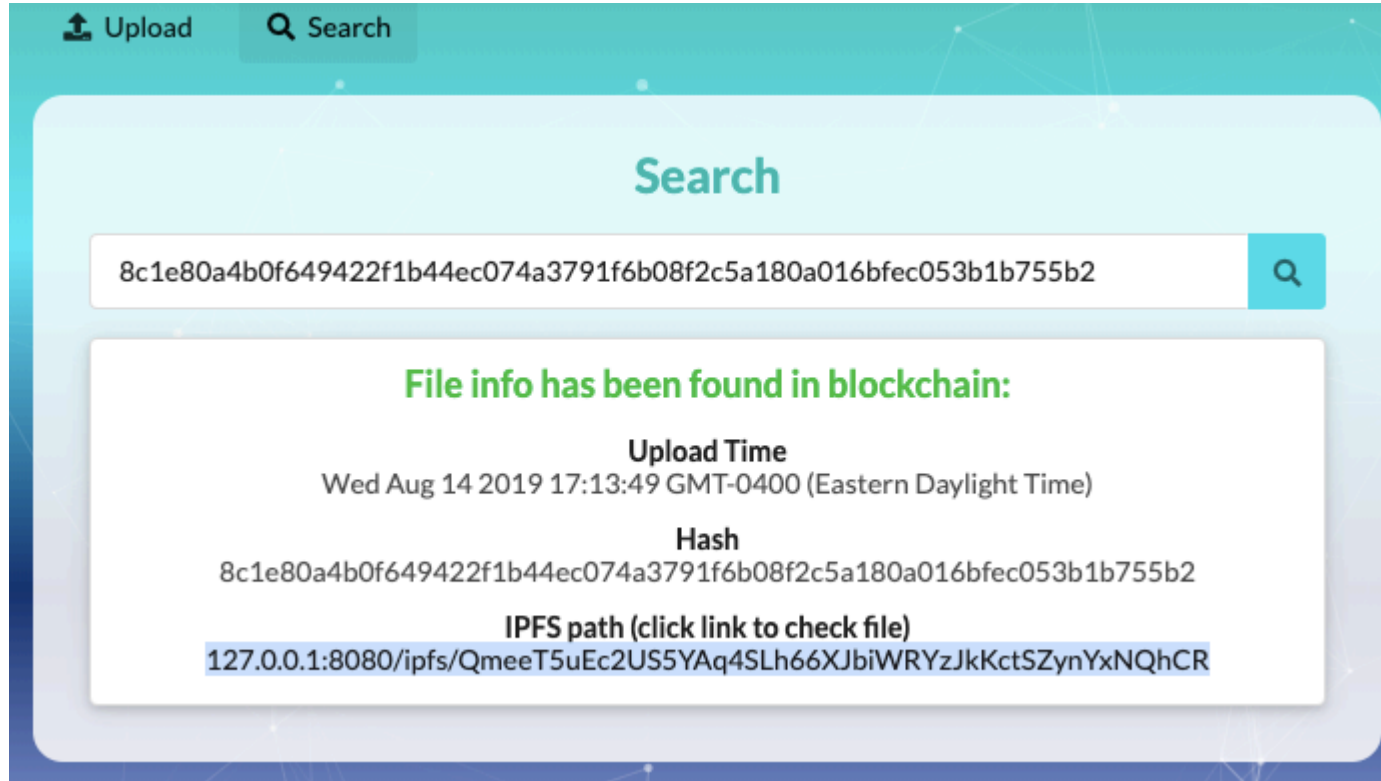
Content Addressable Resources for Education



- CARE are content-addressable, they are stored and access in the web as a whole
- CARE are also associated with each other in an Open Resource Graph (ORG)
- CARE can be *cloned* and *edited* by any user

<https://github.com/Downes/CARE-project>

Content Addressable Resources for Education

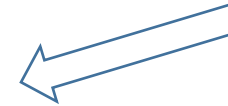


- CARE are content-addressable, they are stored and access in the web as a whole
- CARE are also associated with each other in an Open Resource Graph (ORG)
- CARE can be *cloned* and *edited* by any user

<https://github.com/Downes/CARE-project>

Encryption Algorithms

Algorithms



These are open source!

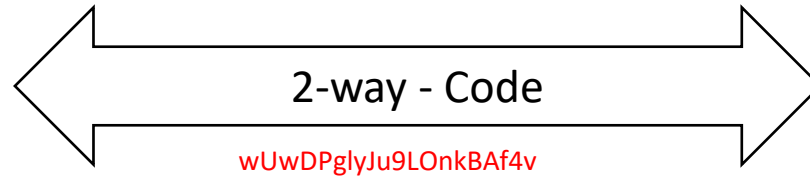
Algorithm and variant		Output size (bits)	Internal state size (bits)	Block size (bits)	Rounds	Operations
MD5 (as reference)		128	128 (4 × 32)	512	64	And, Xor, Rot, Add (mod 2 ³²), Or
SHA-0		160	160 (5 × 32)	512	80	And, Xor, Rot, Add (mod 2 ³²), Or
SHA-1						
SHA-2	<i>SHA-224</i>	224	256 (8 × 32)	512	64	And, Xor, Rot, Add (mod 2 ³²), Or, Shr
	<i>SHA-256</i>	256				
	<i>SHA-384</i>	384	512 (8 × 64)	1024	80	And, Xor, Rot, Add (mod 2 ⁶⁴), Or, Shr
	<i>SHA-512</i>	512				
	<i>SHA-512/224</i>	224				
	<i>SHA-512/256</i>	256				
SHA-3	<i>SHA3-224</i>	224	1600 (5 × 5 × 64)	1152	24 ^[5]	And, Xor, Rot, Not
	<i>SHA3-256</i>	256		1088		
	<i>SHA3-384</i>	384		832		
	<i>SHA3-512</i>	512		576		
	<i>SHAKE128</i>	<i>d</i> (arbitrary)		1344		
	<i>SHAKE256</i>	<i>d</i> (arbitrary)		1088		

https://en.wikipedia.org/wiki/Secure_Hash_Algorithms

One-way Encryption

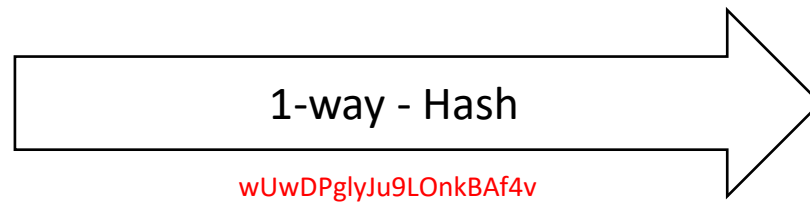
Algorithms

“Come on over!”



“Come on over!”

“Come on over!”

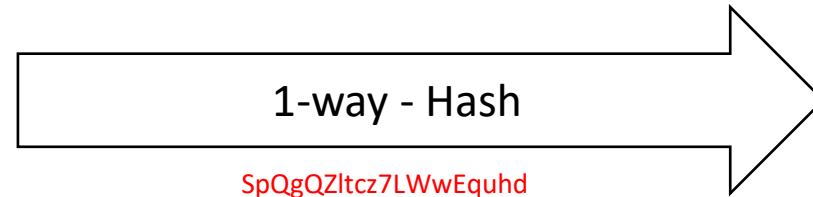


wUwDPglyJu9LOnkBAf4v

Uniqueness

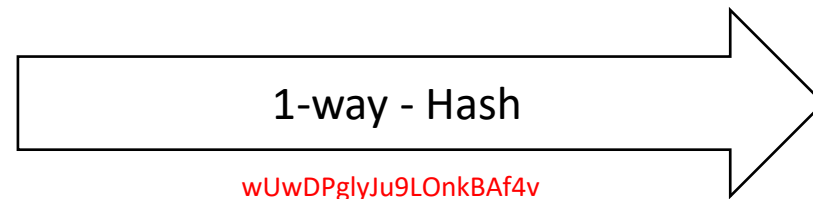
Algorithms

“Come on over now!”



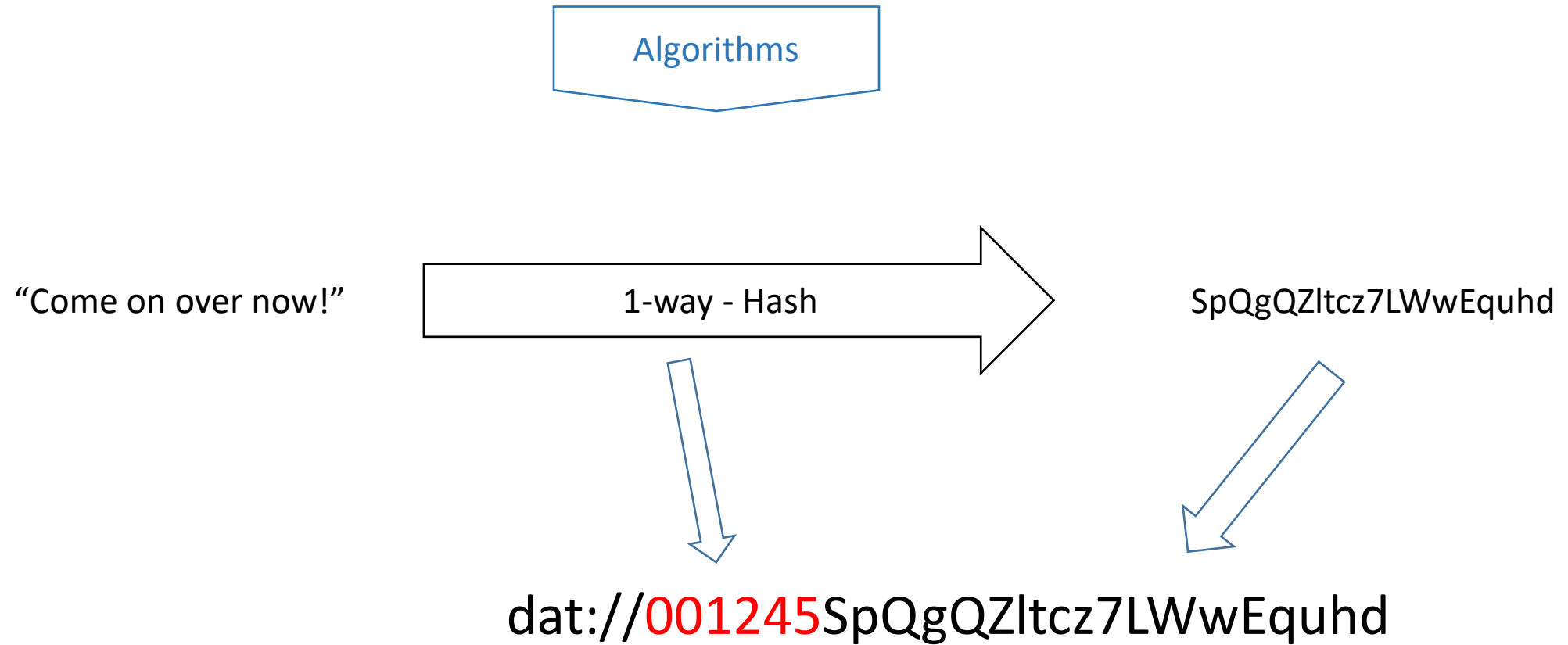
SpQgQZItcz7LWwEquhd

“Come on over!”



wUwDPglyJu9LONkBAf4v

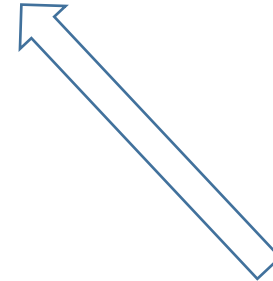
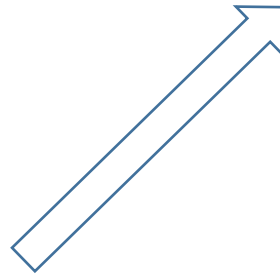
Content-based Address



Revisions

wUwDPglyJu9LOnkBAf4v ⇒ SpQgQZItcz7LWwEquhd

2FdgvdlC7sDv7G1Z7pCNz



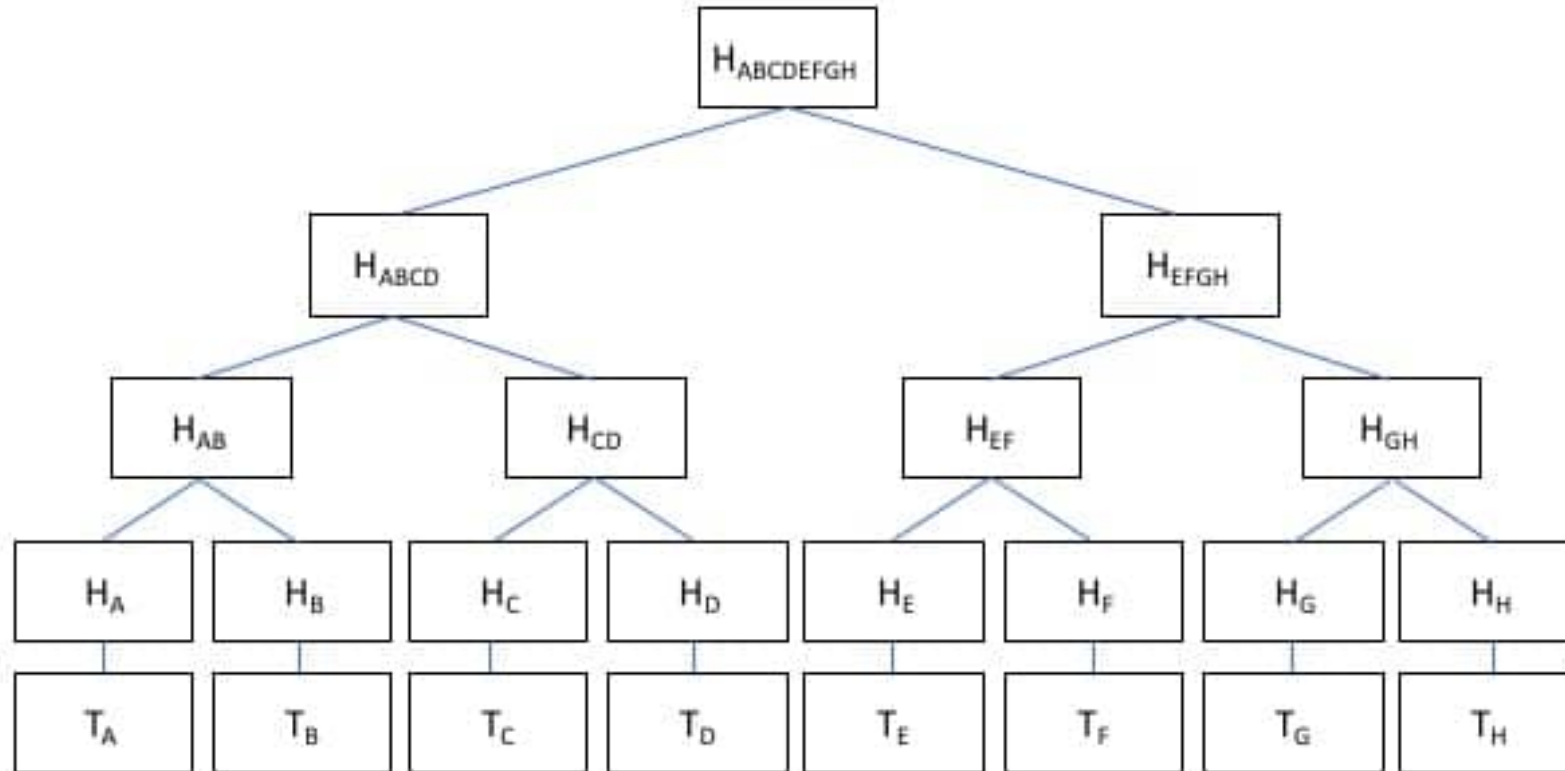
“Come on over!”

wUwDPglyJu9LOnkBAf4v

“Come on over now!”

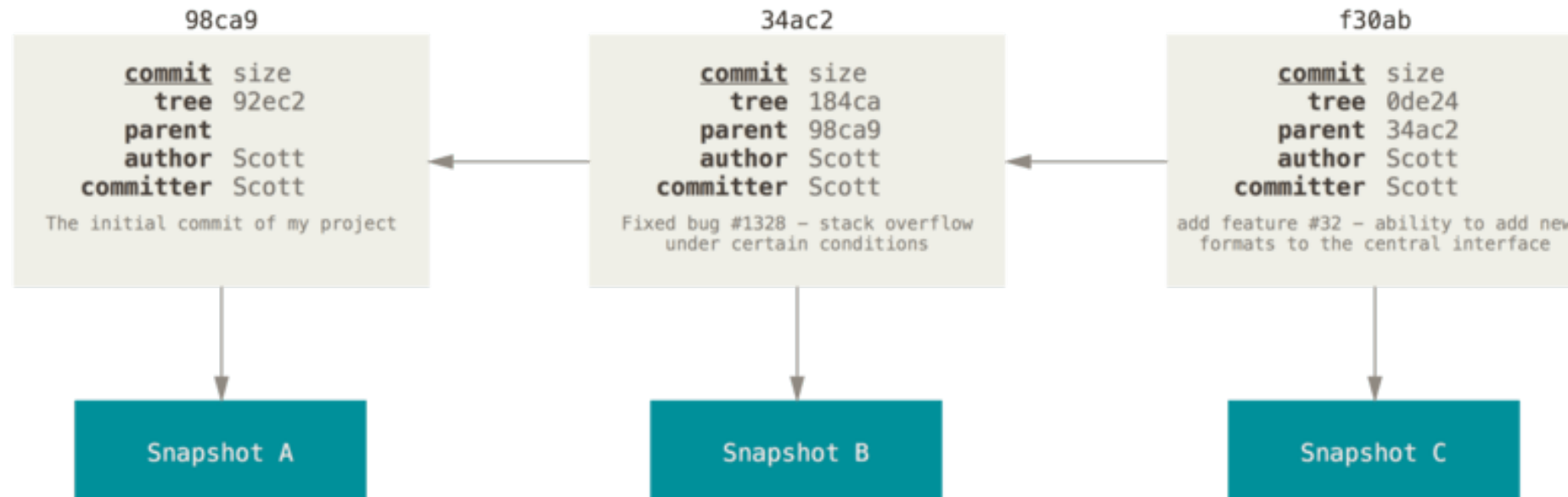
SpQgQZItcz7LWwEquhd

Merkle Chain



<https://btc-investor.net/merkle-tree-hashing-blockchain/>

How Git Uses Merkle Chains



<https://git-scm.com/book/en/v2/Git-Branching-Branches-in-a-Nutshell>

Hash trees are also used in the IPFS, Btrfs and ZFS file systems; Dat protocol; Apache Wave protocol; Git and Mercurial distributed revision control systems; the Tahoe-LAFS backup system; Zeronet; the Bitcoin and Ethereum peer-to-peer networks; the Certificate Transparency framework; and a number of NoSQL systems such as Apache Cassandra, Riak, and Dynamo. https://en.wikipedia.org/wiki/Merkle_tree

Peer-to-Peer Networks



Each node is a server. Some are big, some are small. They each connect to some, but not all, of the other servers, so nobody is overloaded.

https://www.kindpng.com/imgv/iTJwhxT_connection-png-page-network-effect-transparent-png/

Peer-to-Peer Networks



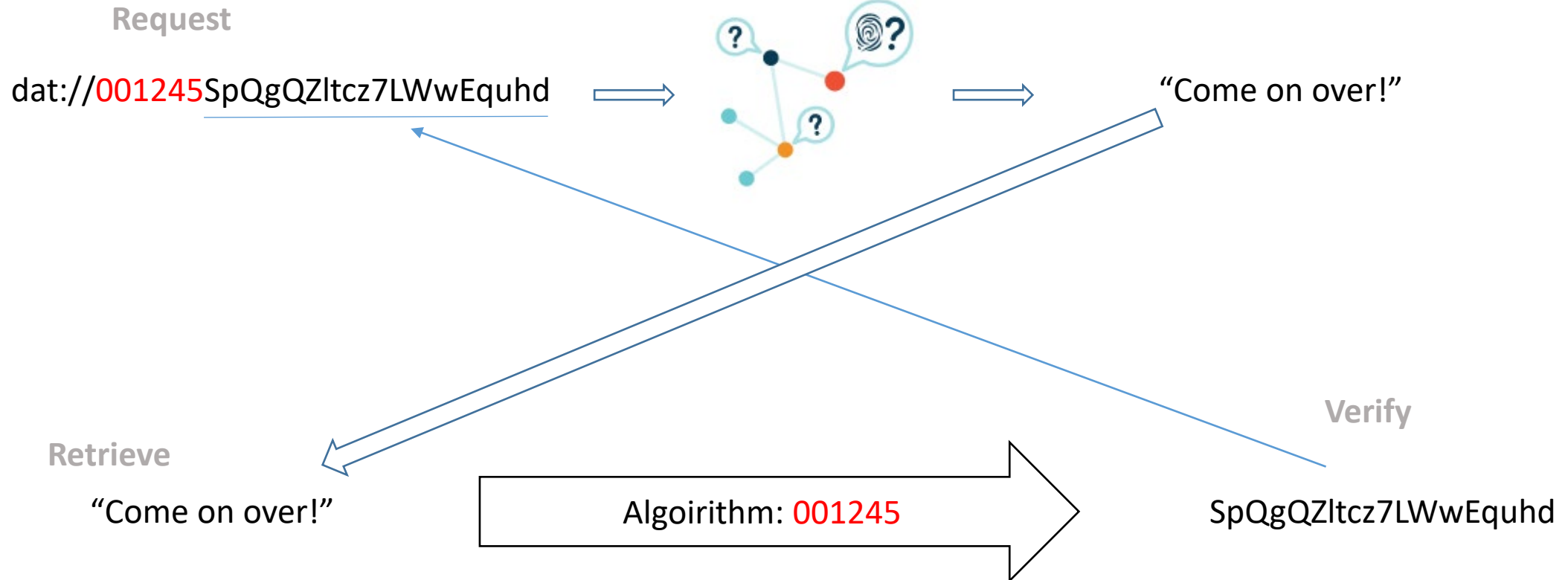
Each network node stores only content it is interested in, plus some indexing information that helps figure out which node is storing what.



When you look up a file to view or download, you're asking the network to find the nodes that are storing the content behind that file's hash.

<https://ipfs.io/#how>

Secure Access



A Range of Applications



- Content-addressable resources could be used in combination with other graph-based technologies to create such features as:
 - learner-generated content
 - activity records, and
 - digital badges
- Because these records are stored and linked as one-way encrypted data, they are private and secure.
- Participants can determine for themselves whether any course-related activity is shared to a wider audience.

- The authors recently developed and used these resources in a MOOC, which will be shared. Participants can see how content-addressable resources could be used in combination with other graph-based technologies to create such features as learner-generated content, activity records, and digital badges. Because these records are stored and linked as one-way encrypted data, they are private and secure. Participants can determine for themselves whether any course-related activity is shared to a wider audience.

E-Learning 3.0 Course



E-Learning 3.0

Course Outline

Sign up for E-Learning 3.0 2019

[Register Here](#)

-1. Getting Ready

Connectivism is based on the idea that knowledge is essentially the set of connections in a network, and that learning therefore is the process of creating and shaping those networks. To get started we'll look at what to do to set up and how to learn in a connectivist course.

0. E-Learning 1 and 2

The premise of this course is that we are entering the third major phase of the world wide web, and that it will redefine online learning as it has previously. The first phase of the internet as it was originally developed in 1994, based on the client-server model, and focused on pages and files. The second phase, popularly called Web 2.0, created a web based on data and interoperability between platforms.

1. Data

This week the course addresses two conceptual challenges: first, the shift in our understanding of content from documents to data; and second, the shift in our understanding of data from centralized to decentralized.

2. Cloud

The joke is that "the cloud" is just shorthand for "someone else's computer." The conceptual challenge is that it doesn't matter whose computer it is, that it could change any time, and that we should begin to think of "computing" and "storage" as commodities, more like "water" or "electricity", rather than as features of a type of device that sits on your desktop.

Course Outline

Course Newsletter

Activity Centre

-1. Getting Ready

0. E-Learning 1 and 2

1. Data

2. Cloud

3. Graph

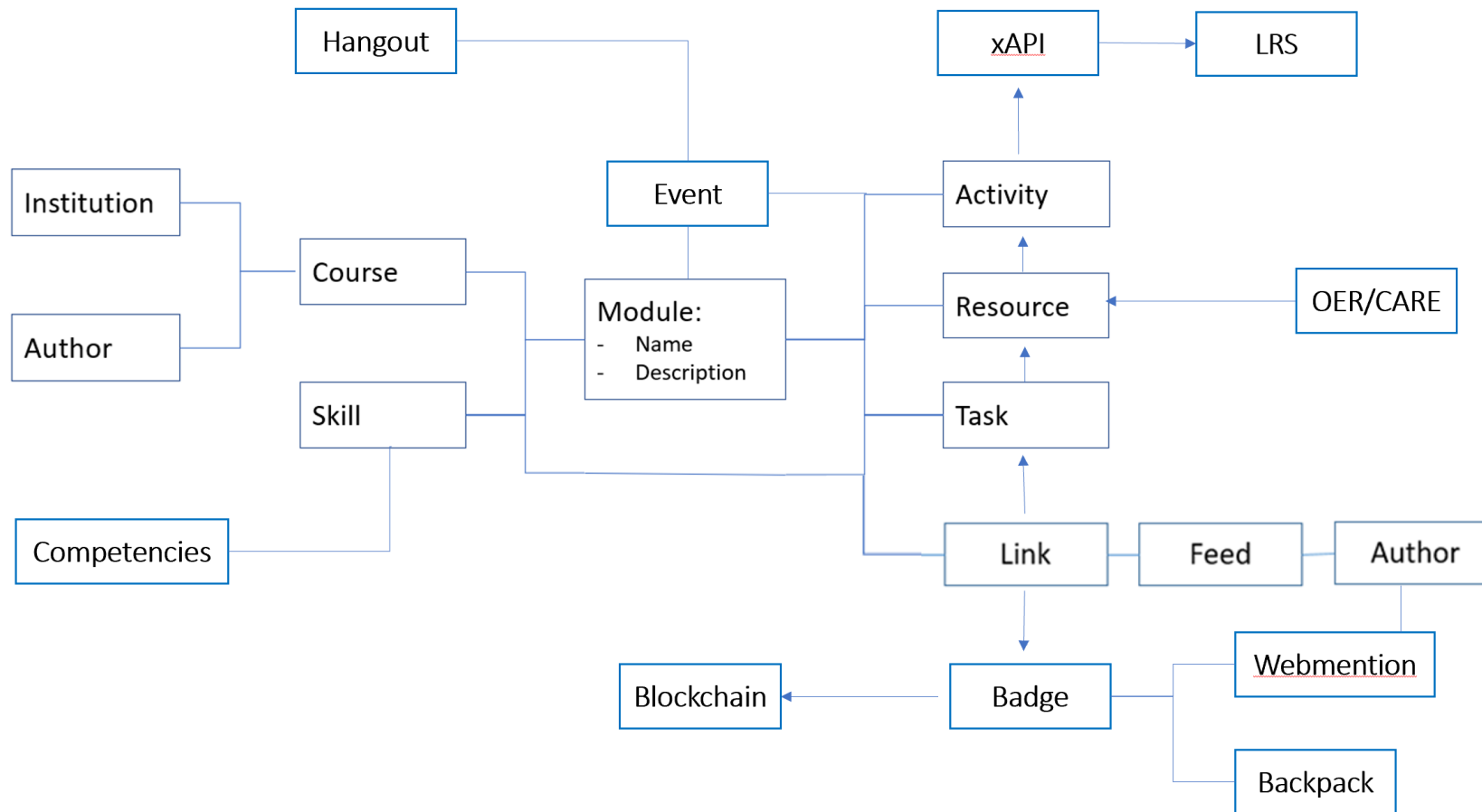
4. Identity

5. Resources

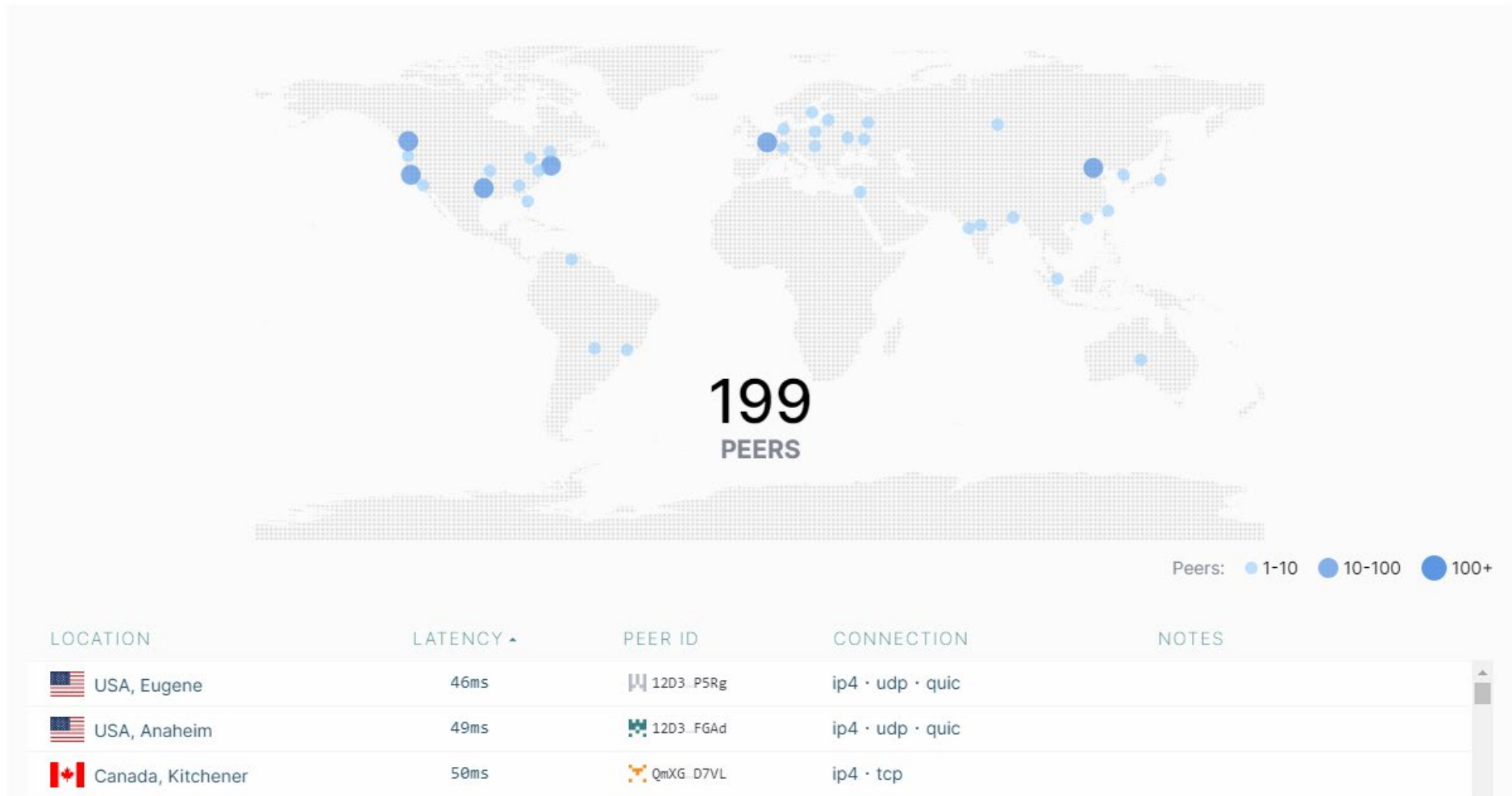
6. Recognition

https://el30.mooc.ca/course_outline.htm

EL30 - A Course as Linked Open Data

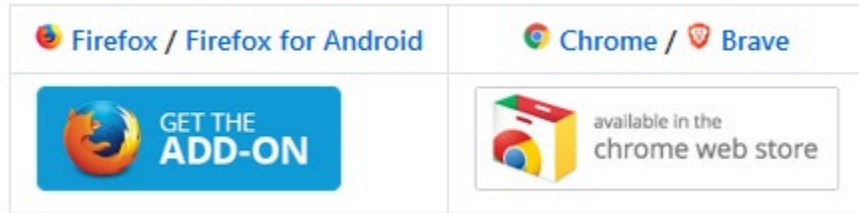


IPFS Desktop



<https://ipfs.io/#install>

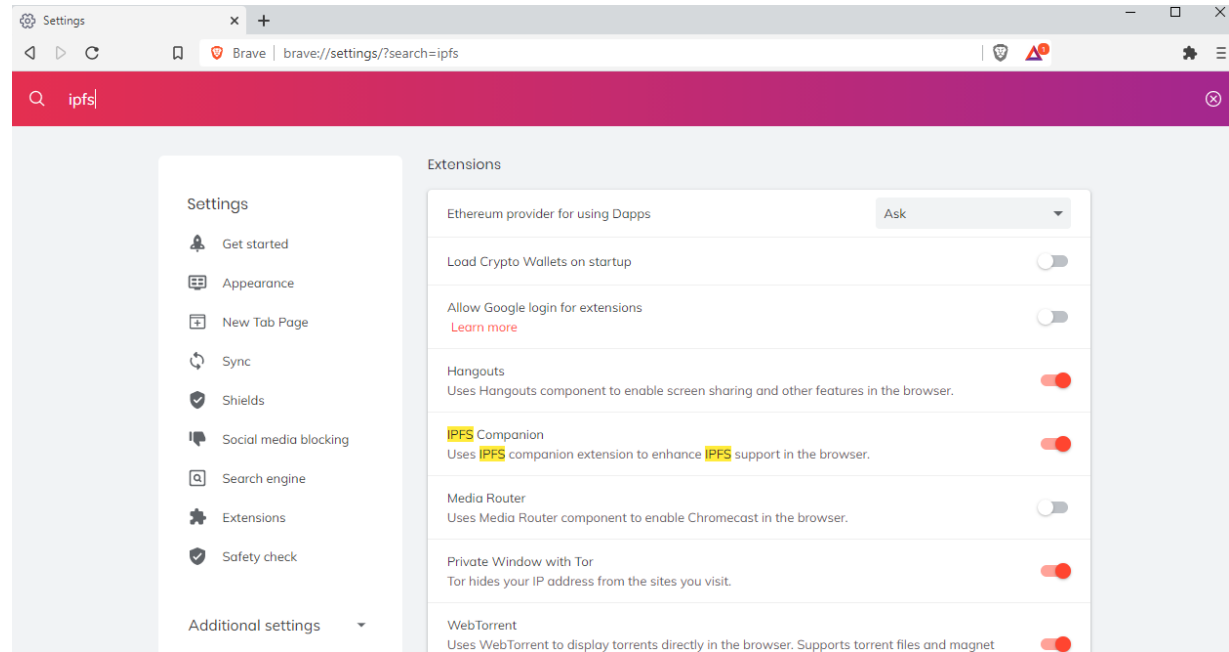
Browsing the IPFS Network



<https://github.com/ipfs/in-web-browsers>

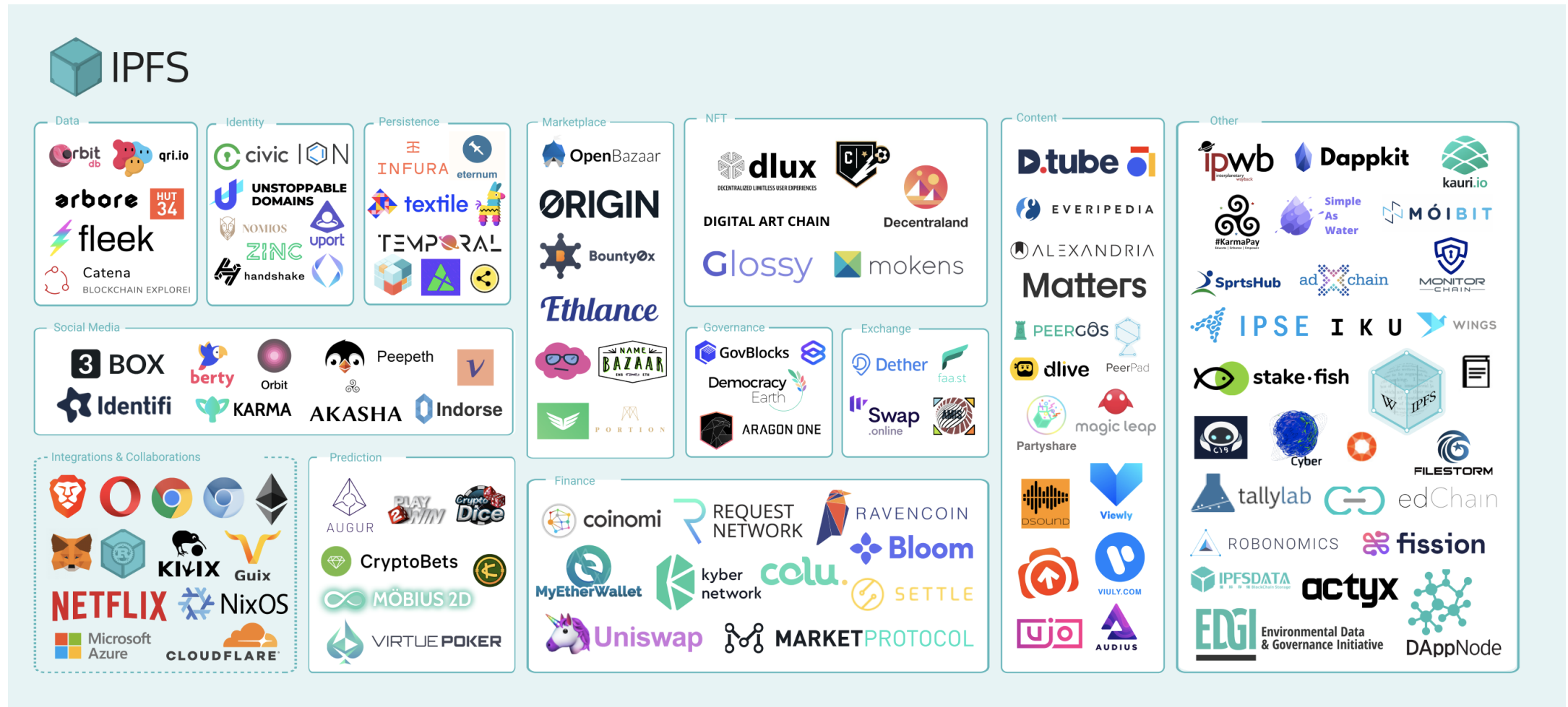
IPFS Web Hub

<https://ipfs.io/ipfs/QmdmQXB2mzChmMeKY47C43LxUdg1NDJ5MwCkMKxDu7RgQm>





<https://brave.com/>



Interplanetary File System (IPFS)



Example: Notebooks

jupyter spectrogram (autosaved) 

File Edit View Insert Cell Kernel Help | Python 3 

 Markdown 

Simple spectral analysis

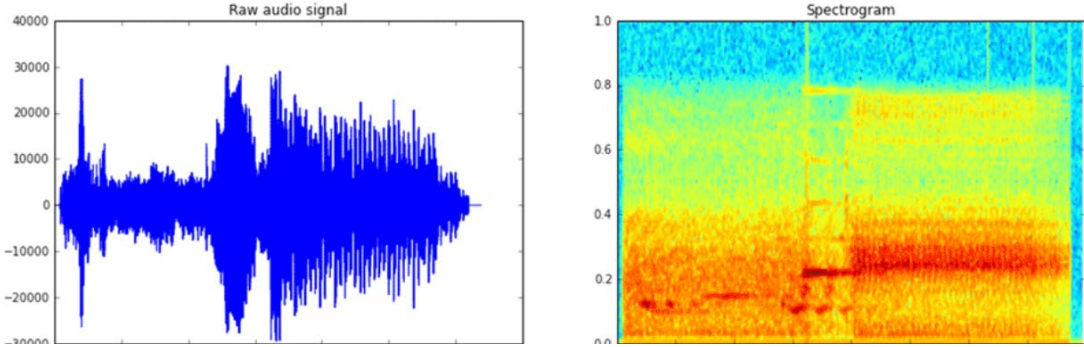
An illustration of the [Discrete Fourier Transform](#)

$$X_k = \sum_{n=0}^{N-1} x_n \exp\left(\frac{-2\pi i}{N} kn\right) \quad k = 0, \dots, N-1$$

```
In [2]: from scipy.io import wavfile
rate, x = wavfile.read('test_mono.wav')
```

And we can easily view it's spectral structure using matplotlib's builtin specgram routine:

```
In [5]: fig, (ax1, ax2) = plt.subplots(1,2,figsize(16,5))
ax1.plot(x); ax1.set_title('Raw audio signal')
ax2.specgram(x); ax2.set_title('Spectrogram');
```





<https://github.com/jupyter/jupyter/wiki/A-gallery-of-interesting-Jupyter-Notebooks#machine-learning-statistics-and-probability>

OpenLearn [Jupyter Books Remix, TM351 Notebooks in VM and Electron.](#)

<https://www.dataquest.io/blog/jupyter-notebook-tips-tricks-shortcuts/>

Applications

The screenshot shows the 'Awesome IPFS' website interface. At the top, there is a navigation bar with icons and labels for APPS, ARTICLES, DATASETS, SERVICES, TOOLS, and VIDEOS. Below the navigation bar is a search bar with the placeholder text 'Find your awesome app...'. The main content area is a grid of application cards, each with a title, description, size, and a link to the application. The cards are categorized by type, indicated by a colored header bar above each card.

- DATASETS**
 - Operating Systems Mirror & Archive (OSMA)**: Mirror and archive of 29 established operating systems with 192 releases. Size: 643GB. [/ipns/QmRd...tXONP](#)
 - WistfulBooks: LibriVox Audiobook Archive**: Free public domain audiobooks from LibriVox.org packaged into a single page that lets you listen to audiobooks in your browser. Size: 2.0 TiB. [/ipfs/QmXy...w3RYU](#)
- APPS**
 - edChain**: EdChain is a global network for education and careers. It uses blockchain and distributed web technologies to deliver educational content, even to those without broadband internet. 
 - qri**: Dataset version control, discovery and collaboration tools (free, open-source).
 - Peer Map Demo**: A map of IPv4 IPFS peers that uses win dow.ipfs. 
 - gogo.tattoo**: Gogo Tattoo Project uses IPFS, DLTs and other modern technologies to offer tattoo artists and wearers an unbreakable lifetime record of their portfolios. gogo.tattoo app already has a feature to
- TOOLS**
 - mahuta**: Mahuta is a plug and play service for your micro-service architecture allowing to collect, store and index data on IPFS and offering search functionalities (full text, query).
- ARTICLES**
 - Hands-on IPLD Tutorial in Golang Series**: January 4, 2020
- TOOLS**
 - Blockwatch**: Monitor the current block number of the Ethereum blockchain, and set alerts (stored locally using PouchDB in your browser) for when certain block heights are passed.
- APPS**
 - IPSE**: A search engine for the IPFS network.
 - Temporal**: Temporal is an easy to use API and platform for integrating IPFS and other distributed/decentralized storage technologies into enterprise applications
 - a js video player**

<https://awesome.ipfs.io/>

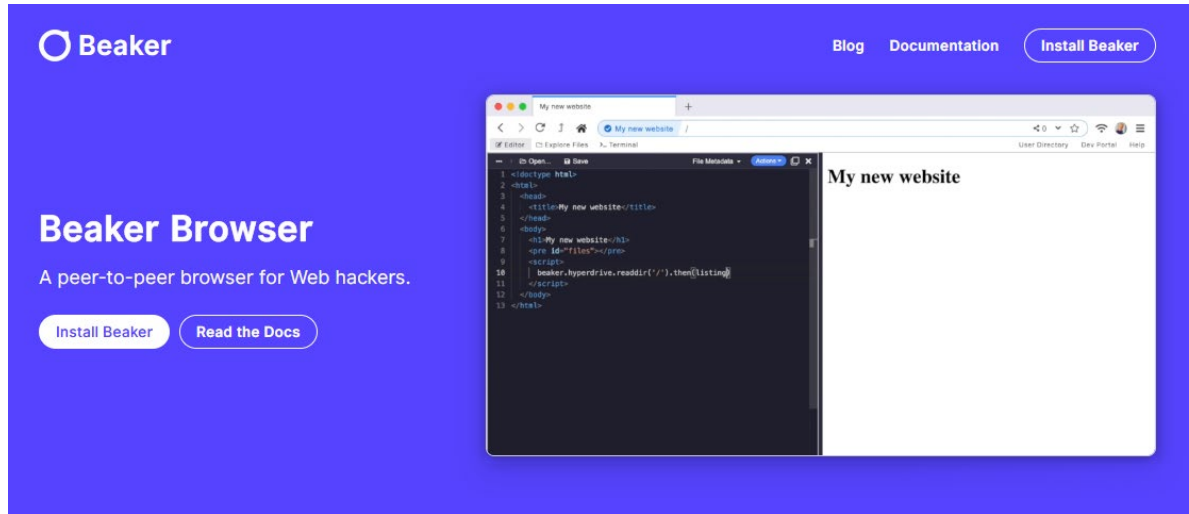
Dweb

- One significant current project implementing such a protocol is called Dweb (for 'distributed web' or 'decentralized web'). (Ayala, 2018)
- Based on the dat protocol, a mechanism for finding and distributing content
<dat://502bdf152d00a35f9785f78d107b9037b5eca9354bcf593e7b4995f9be97a614/>
- This address is in fact the dat:// address for the first *Content Addressable Resource for Education* (CARE)

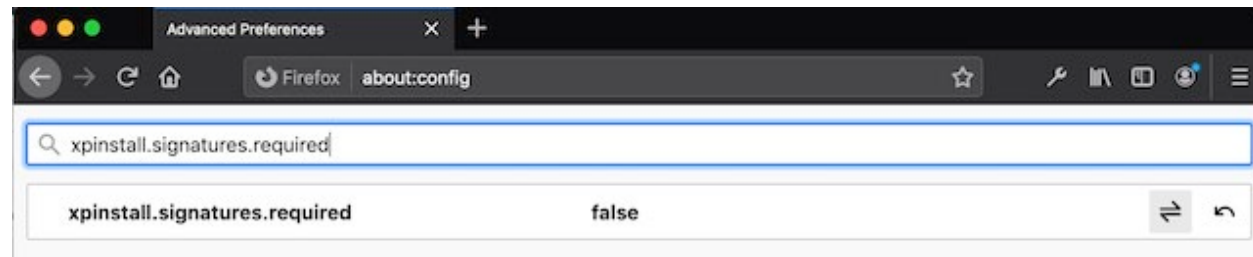
<https://hacks.mozilla.org/2018/07/introducing-the-d-web/>

<https://www.datprotocol.com/>

Browsing the Dweb Network



<https://beakerbrowser.com/>



<https://sammacbeth.eu/blog/2020/05/08/install-dat-for-firefox.html>

An important aspect of these resources is that they can be developed or modified by anyone. This supports not only content revision but also a common mechanism for community-based meta-tagging or content reviews and to, optionally, provide data on context and use. Thus participants will be able to appreciate how content-addressable resources can inhabit a rich ecosystem that provides an open alternative to published-based and controlled repositories.