Redhouse Gases – A manifesto for re-decentralization

Jon Crowcroft
University of Cambridge
jon.crowcroft@cl.cam.ac.uk

Gareth Tyson
Queen Mary University of London
g.tyson@qmul.ac.uk

Richard Mortier
University of Cambridge richard.mortier@cl.cam.ac.uk

December 16, 2020

Abstract

In this chapter, we reflect on re-decentralisation. This represents very personal views, particularly potentially revisionist history, and we encourage people who were around for any of this time to disagree.

When we use the word centralisation, we are referring to ownership and management. Of course, there are other players in the supply chain, hardware vendors, government, civil society, but let us concentrate on this service aspect in this work. In the somewhat early days of the Internet (e.g. 1980), we saw it as a decentralised system that contrasted with the central telephone systems. The phone systems had until then largely been national monopolies. Whilst some of them were distributed in terms of technical management and operations, they were centralised in terms of ownership and administration. The Internet was decentralised at the network layer, and later in terms of assigned numbers and traffic.

Twenty years later, post world-wide-web and cloud and since the start of the smart phone revolution, from 2000 onwards, we see a massive swing towards centralisation of the Internet in almost all of its layers.

This has consequences, some, but not all of which are bad. It is time to think about re-decentralisation and how we might address some of the unfortunate consequences. We look at this through the lens of Human-Data Interaction.

1 Introduction

Why do we care about data centralisation? One reason is that the concentration of data leads to a concentration of wealth, and the tremendous temptation to make money from that through rent, rather than the production of novel goods and services[5]. Rather than attacking digital capitalism, like Piketty, we would like to see a modest push-back to a place where more reward is gained for creativity than simply getting richer as a side effect of being rich.

Let's take one example to kick off - that of Private Property. What is the way the digital world impacts on this from a centralisation perspective? We're used to seeing signs that say *Private, Keep Out.* Usually this refers to land around houses. But let's take a much smaller everyday object as an example: Now, imagine you rented your shoes rather than buying and owning them outright. And imagine, every now and then, they send you targeted adverts saying "You're wearing the soles out walking like that"; "Time to get re-soled."; "You're not walking enough steps"; "You're doctor may need to know" "That walks a bit gay"– "Why not try this LGBT dating site?". What were the privacy properties that changed and so confounded our expectations? The relationship between you and the organisation that rents the shoes is extensive, and intrusive, compared to the termination of the relationship between you and the seller, when you outright buy the shoes.

Another dimension of privacy refers to visibility – the possibility of surveillance, and the awareness of that possibility (sousveillance). How many people can see you? It is some not especially new, despite claims "What's behind the Arras, mum?"; or indeed, whispers in ancient Rome or dramatic irony in ancient Greek drama. the very idea of a persona. We present differently to different people, because we're not all the same[3]. This means that privacy is contextual with regards to both the subject and the observer. You'll note here we have deliberately conflated two meanings of privacy, because they don't necessarily actually differ - privacy can refer to exclusivity (of ownership or of visibility).

What are sensible or reasonable privacy proprietaries, or desiderata for such exclusivity? One is the principle of least astonishment - that a mirror isn't two-way. Another is that while property is about ownership; but data is more subtle – not just of things, but of personal space, and how it can be seen, reflected, refracted, obfuscated, deleted, and so on. Hence we can suggest that online (cyber) space should be no different from data itself - control-over-use is what matters. This is particularly the case as natural social structures mean that a centralised view is reductionist. This chapter therefore explores the role that centralisation has played in the management of data and its subsequent implications for user privacy.

2 HDI & re-decentralisation

A key concern that this work is trying to address is the way that all kinds of networks have a tendency to drift towards centralised data and compute, and hence power, despite the fact that we have the tools to re-build systems in a peer-to-peer way. There are multiple examples now (Guifi, Mastodon) that point the way towards how to sustain them.

There are then technical challenges for providing the applications, resilience and availability of central systems, but in fact the solutions are similar to those already employed in large data centers. In the end, decentralised systems will provide lower latency, lower energy consumption, and higher privacy to the users. Hence they can provide more agency, more legibility and more negotiability to their communities. That said, the specifics of this will vary with the given use case. Guifi suggests that incentives to maintain sharable resources are essential. BitTorrent realized this some time back. In contrast, Mastodon exemplifies an altruistic model, with many instance operators dedicating time and resources for free.

Next we look at some underlying high level principles to apply in bringing these various pieces together and motivate our solutions.

2.1 Core HDI principles

In the original Human Data Interaction (HDI) manifesto, we outline the core principles for the Interaction between Humans and Data. These principles were not developed in a vacuum. Rather, they were driven by the need to combat the (sometimes unintentional) misuses that occurred in recent digital systems (cloud, internet, government data on population).

Agency A.k.a. power: who has rights for what?

- basic: capacity to act
- strong: power to act on others

Negotiability How do we change attributes associated with data, including:

- permissions - read, write/modify, execute, delete, etc
- ownership, give, copy, take, make public
- Even rules?

Legibility a.k.a. transparency: do we understand data about us; and who knows what about us? sometimes referred to as *sousveillance*.

These principles emerged in-part due to the pressures created by centralisation, which has undermined these rights for many users. For example, in terms of Agency, it is often unclear who controls the data, and how individual may regain this control.

2.2 Data isn't property

Obvious: I give you a copy, I still have mine & To copy data about me costs nearly zero at least economically. It obviously is not anything like our classical inherited ideas of the most basic property of property: cost of ownership. Less obvious: I change execute rights to program – do I change other use rights? Data isn't even a type of property like just IP (Intellectual Property) since it may have involved virtually no effort (intellectual or otherwise) to create. It might even be a waste product[1]. DOme venture capitalists have ventured this idea to, e.g. `https://a16z.com/2019/05/09/data-network-effects-moats/`.

3 The Rise and Fall of Decentralisation

The original reason that the Internet happened to be decentralized is lost in the mists of time. The usual assumption derives from Paul Baran's document about survivability and packet-switched, dynamically routed, geographically distributed networks. Some people dispute that that was the real intention, but certainly looking at Dave Clark's Sigcomm 1988 paper on the architectural design principles for the Internet, he makes the case. The avoidance of central coordination is a big design feature. The practical deployments of such systems, however, can result in pressures towards centralisation. These might be for technical (e.g., scalability), management (e.g. control) or economic (e.g. mergers) reasons. A number of Internet related infrastructures started decentralized, yet have faced these pressures:

IP The Internet Protocol (not the Intellectual Property) uses datagrams. The series of tubes that make up the Internet carry information in the form of packets. The innovation that these units of data carrying only a few thousands of bits should be the way computers communicated was originally made by Donald Davis of the Post Office in the UK. However, an additional step was made after Paul Baran's RAND report, that the network of computers, and the computers on the network, should not share each others fate in the event of the failure (or destruction) of one or the other. This led to the decentralised design of so-called *connectionless protocols* connected by *stateless routers.* There are then three uncoupled tasks, end-to-end communication over the network, and hop-by-hop forwarding by routers, based on a distributed periodic computation of routes over which to forward data between end systems. The is no central coordinator for any of these three tasks.[1]

DNS and CA The Domain Name System is largely distributed, although the top level of the namespace is centrally managed. The cetificate authority (CA) that is the root of trust for signing off on the validity of secure sites or HTTPS accesss started centralised. CA transparency is an project to mitigate consequential problems(`https://www.certificate-transparency.org/`). The presence of multiple browser products somewhat offsets some

[1] Sometimes, geographic constraints force some degree of centralisation. These often are revealed when a local failure leads to more global consequences – e.g. the Baltimore tunnel fire.

of the dangers, but high profile security breaches (e.g. DigiNotar) have highlighted the risks.

BGP The Internet is a network of networks. Each network can be run completely independently, including using its own (distributed) route computation. To interconnect these networks, then, requires forwarding, on the basis of another level of route computation. This is provided by the Border Gateway Protocol, which is, naturally, distributed. It supports connection of autonomous systems (as they are known – roughly equivalent to an Internet Service Provider), and there are two basic connection policies: peering and customer/provider. Two trends over recent years have also led to the emergence of Internet Exchange Points, where multiple ISPs connect, and to a *flattening* of the logical/business/topological organisation of the Internet in terms of the numbers of customer/provider hops it is between sources and destinations. It is not clear if this means, on balance, that the internet has become more, or less, centralised as Internet Exchange Points replace major tier-1 transit networks.

Multicast In the late 1980s, the capability of sending IP datagrams from one source to multiple recipients selectively was added. This is known as multicast. The original multicast scheme was decentralised in operation, in that there was very little coordination function. Even address assignment was at least partially free-for-all. Over time, the ability to sand from any source to any group came to be regarded as dangerous, and a more restrictive form of IP multicast emerged, with Single Source and Source-specific schemes seeing approval. Multicast is still used very little, despite having significant cost savings in bandwidth demand for some large scale content providers. Instead, Content Distribution Networks (CDNs) often employ a mixture of application level multicast and content caches for time shifted user demand.

ICN More recently still, recognizing the vast amount of content on the internet was both replicated and distributed by these CDNs, researchers proposed and deployed systems known as Named Data Networking (NDN) or Information Centric Networking (ICN). NDN is a an extension to the IP network layer to recognize that the sender and recipient demands may look something like those met well by CDNs, and therefore we need to recognize caches as a first-class part of the network architecture and its protocols. NDN delivers no central control of these caches, instead relying on distributed mechanisms outside of the scope of the network to manage their placement.

Community Mesh In the 1990s, it became clear that communities wished to build wireless networks, perhaps using the emergent WiFi standards, to connect users together directly, without the need of a wired (or wireless) infrastructure provider. Hence ad hoc and community mesh wireless networks emerged. Ad hoc networks are really the most extreme case.

Community networks, as exemplified by Guifi are planned and coordinated by the community. In that sense, they are no different from an ISP, except that typically ownership is collective and subject to different governance models. For instance, some may insist that all access is free, whereas otherds may allow resale.

Blockchain Most recently, we have seen the most extreme application of ideas of decentralisation in the form of cryptocurrencies such as Bitcoin. These make use of a distributed ledger technology such as Etherium, Hyperledger; these can be permissionless (no access control to the ledger) or permissioned (access control requiring a key distribution mechanism). Permissionless systems need mechanisms to prevent free riding and pollution, but if they support a currency, they need to scale their mechanism to the value of their virtual economy. This appears to be ill-affordable for any realistic global use. However, Bitcoin sure made people think. On the other hand, the number of miners is now really quite small and highly clustered.

We've now gone from the lowest level of the network stack (transmission of signal on community mesh wireless) the highest level (economic value). Next we look at the evolution of communications applications.

EMail The early days of electronic mail actually predate the Internet by some way. In the 1970s, before computer connectivity was a thing, we had pretty widespread telephone networks (around 600M land-lines, with connectivity, such that any of those phones could call any other one). Bell labs devised Unix, and being part of the telephone company, connected computers running Unix to each other using acoustic couplers, modems and the Unix Unix Copy Program (UUCP). This supported file transfer, and simple e-mail, but more interestingly it supported a structured, threaded hierarchy of news articles, which people around the world went on contributing to for 3 decades. Usenet news was a mix of social media, blogging and content distribution. Most people have sadly forgotten the lessons it bought in terms of consensus for control of content (e.g. moderated lists). Shortly after that time, the early version of the Internet mail system emerged, which was also decentralised. It was ready to roll out when Internet connectivity came along (as well as newsgroups simply switching over to using that connectivity instead of phone calls).

WWW One of the several strokes of genius of Tim Berners-Lee in creating the World Wide Web, was to understand that each web server was autonomous. This allowed asynchronous development of content, and even technology. The pivot for this was the simple idea that links between web sites are uni-directional, so that it is not necessary to have any central coordination of adding new links to old sites on a new site, or later updating old sites to link to new ones. Of course, centralisation has since happened in the form of many services. For example, social networks are subject to network effects, which make it difficult for new sites to be established.

Peer-to-peer (P2P) protocols Again, in opposition to costly content services, peer-to-peer protocols based in distributed hash tables (DHTs), or else in unstructured gossip/search algorithms were constructed. Out of a plethora of different approaches, two successful technologies are now in wide spread use for key-value stores (Kademlia), and for content distribution without central management (BitTorrent). The latter technology is also known as swarms for the way that different blocks of data can be sources from different peers. In some studies, it has been shown that this approach to content distribution achieves optimality in terms of network resource use. Nevertheless, commercial ISPs often opposed this, and pressure cause the BitTorrent protocol designers to introduce a *less than best effort* approach so that their traffic was always lower priority than *regular* content distribution users. BitTorrent implementors also introduced an incentive-matching protocol based on a simple tit-for-tat token system, to reduce the prevalence of free-riders and polluters. P2P systems have also had some drift towards fewer nodes (like the Skype Supernode) carrying out a larger proportion of the effort.

Xenoservers Fully distributed computation was put on the map by SETI@Home, although many people may not be aware that the incept of the Xen hypervisor was originally as a way to share home PC users' compute resources, thus really being the first form of edge-cloud.

Social nets As mentioned above, UUCP supported newsgroups which functioned partly as an early form of social media. Nowadays, we have seen the emergence of digital giants implementing highly centralised social networks, but this did lead to a backlash which has seen attempts to re-decentralise social networking, through an unsuccessful Diaspora, and the more successful Mastodon, of which more later.

Physical world network platforms Ride hailing and room rental services claim to be transformative in that they replace chains of taxi firms and hotels with a simple platform that lets anyone with a car or a spare room join in the business. However, while the physical world resource ownership is decentralised, the service platforms are almost always thoroughly centralised. Claims are made that this makes life easier for the customer in terms of finding the service, and trust/recommendation management. This claim can be tensioned against the fact that there's lots of abuse anyhow. The centrality is really more merely branding.

It has been observed that part of the evolutionary process here is driven by *involuntary* centralisation. Even if you create your own platform, and decentralise its operations, you can still be experience centralisation via unavoidable dependencies, e.g. certificate authorities, high-centrality transit ASes. And of course, you need to interconnect both socially and technically. So, inevitably web mail, content distribution networks and social networks that are now centralised have to be accommdated — the *network effect* of a service that isn't a network at one (service ownership) level, distorts the reality of others that are networks at all levels. It's a powerful attractor.

4 Challenges

Decentralisation brings with it a number challenges.

Availability There are two challenges to providing high availability from decentralised computation and storage, both caused by the asymmetry of many, if not most, broadband access links, based on ADSL, Cable Modem or Cellular data lines. First the uplink capacity is usually significantly lower than downlink. The design assumption was that home and small officer users are more consumers than producers (e.g. watching videos). This can be changed (it is a trade off, but not usually user selectable). The other facet of home links is simply mean time between failure. Measurements vary but typical outages of hours can occur.[2]. Exposing control over this balance (e.g. via an open API) would allow individuals to specialise their connectivity for their own needs.

As the Internet upgrades, we see a move to more fiber-to-the-home deployment in developed regions, or 5G networks, where the capacity is much higher, so uplink speeds can be significant, and availability and latency may also significantly improve. Nevertheless, we have a great deal of heterogeneity in the system and there will always be much slower links, and less reliable machines in the home.

The usual way to obviate this is through replication. Indeed, centralised cloud services already implement replication, within data centers and between data centers over the wide area network. And CDNs (as discussed above) have very large scale replication of content in the form of content caches. The same algorithms can be applied in the edge-cloud, the fully decentralised case. Indeed, recent research on consensus algorithms (e.g. flexible Paxos) allow tuning of the protocol so that updates to state (mutable content or computation) are applied consistently across copies, even in the wide area. We can even tune for copies running on disparate sites. Of course, there are intermediate design choices where the *edge* is not on the end-user's devices, but instead is just inside the network – in the case of cellular data networks, this could be at the cell tower. This has performance advantages for mobile devices, at the risk of handing back more control to the telcos.

Incentives While there's a motive for people to take part in mutual exchange, this may fail in a world with Byzantine behaviours. Enforcement through tit-for-tat, as mentioned above, is one approach. Others, notably Guifi,

[2] Another minor annoyance is the presence of Network Address Translators, which make inbound connections to the home more complex as NAT traversal protocols have to be used, but these are widely deployed and understood so that applications like Skype can work.

are simply not free, but are collective, not for profits. These have proved relatively stable. Distributed ledgers for crypto currencies have relied on various unsustainable proof-of-work approaches. Nor has that avoided collusion attacks. This remains a difficult area.

At a higher level, there are the incentives to the innovator, whether they want to scale-out their system fast, and therefore likely take advantage of centralised substrates (the cloud), or they are prepared for a long, slow lead time, and let a decentralised system grow more organically. Remember, the Internet experienced exponential growth all the way, but the first 20 years were pretty small compared to the telephone and TV networks of the world.

Integrity Early peer-to-peer file sharing systems suffered from poisoning and pollution attacks, where rivals would bring up servers that made demands but offered no resource, or offered content with the same name, but meaningless actual media. It is a challenge to build integrity checks without invoking an oracle or centralised authoritative server.

Identity The are a variety of attacks on identity in decentralised systems with counterfeit and replicated accounts (e..g Sybil attacks) being frequent. Self-sovereign identity seems like a promising direction, especially where the features or facets of an id being verified by the service are of client-request specific attributes only.

How does engineering travel to meet the HDI challenges?

control of access control Default should afford equal power – to data subject, source, originators

power Agency – can I share or delete data?

Trade Negotiability – can I charge, refuse, revoke access

Comprehension Legibility – Do I know and understand all this?

The ability to of users to realise the above challenges largely depends on the nature of how and where data is stored and managed.

Where is data?

Where data lies This may matter, but it may not. For example, if we keep data encrypted at rest, in transfer and during processing (see below) then where the keys are is what really matters technically. That said, GDPR/cloud/jurisdiction/redress all seem to point to the idea that where the data storage and processing happen (same thing in law) is important.

The redress point matters because people make mistakes (deliberate or accidental) so you'd like some mechanism for recompense. and incentives to minimise mistakes. This inevitably will involve law unless you take data in to your own hands.

Otherwise Power asymmetry and coercion will not be your friends.

How long Retain data, or use trusted 3rd party

What if we enclave or escrow with someone we pay who isn't even curious? Changing the business model may also be a viable approach. Removing the temptation to better target advertising may be a big step to not needing edge cloud/storage. Or a mix, where some of the replicas of your data and computation are paid service providers.

5 Avoiding Recidivism & Roadmap

We next discuss ways of avoiding relapsing into centralisation. The are already a number of good band-aids, discussed, for example, in the Royal Society report on readiness and limits of Privacy Enhancing Technologies. These are relevant as it addresses the secure processing of personally sensitive data. So we care not only about what you process, but how, and where, and who can see the source data, intermediate results and outputs, then a number of tools exist for this if the data is to be gathered to some servers, whether in a private data center or public cloud Safe havens (including enclaves), Fully Homomorphic Encryption, and Secure Multi-Part Computation.

However, there is also a growing movement to leave the data where it is and use techniques known as edge cloud and federated learning, so that only results (aggregate statistics or models) are acquired. Two past projects in this space are: the EPSRC funded Databox project, which led to the BBC adopting the Databox to provide an open platform for third party accountable access to processing, e.g., viewing statistics for broadcast content, without revealing individual specifics; The Digital Economy program funded the HAT project, which look at the markets for personal data. This led to the creation of the (VC funded) startup Dataswift which provides a commercial, open platform for secure processing personal data. This is similar to Databox (indeed shares some creative ancestry) but focuses on providing commercial value, while retaining similar legal, ethical and security goals.

Despite these initial successes, there remain a number of technical challenges that emerge from decentralised of data processing. For example, the scale and heterogeneity of network and processing resources, set membership revelation, and possible intercept of intermediate results yielding access to recent training data. Further lines of exploration that could underpin for federated edge process include:

Trusted third parties Rather than keep data local or sending it to the center, pay a third party to work with it - this is the basis for Privad and HAT

Reputation systems How do you know your peers are good? Either pay them, or pay them back by reputational damage. This is reactive, which means it is too late for a particular misbehaviour- the intention is that statistically, it acts as a disincentive. Note even blockchain supports a notion of trust -for example Bitcoin does not require people to fully check each step in a transaction chain at that point in time - they may defer the *six confirmations rule* til the end of the chain.

Permissioned blockchain or secure multiparty stores Keep your data in a distributed ledger - OK, so long as you don't want to delete it.

Federated machine learning/AI slice the data and the processing so that each shard cannot determine things about either the data or the processing. It can also be done with FHE or MPC, but that can be more costly in processing overhead or latency.

One of the drivers for re-decentralisation arises out of the tendency of profit maximising capitalist systems to over-reach - the phrase "honest but curious" is supposed to capture this notion, that someone will always seek to extract all the possible value out of your data, and will also be rent seeking. This is not intentionally evil unless you think capitalism is evil. You are the means of (data) production and the cloud factory owners own you. Hence we have a spectrum of aggregated value and motivation ranging from the highest to the lowest: The ; the cloud; the data center; the CDN; the provider; the Edge.

6 A decentralised Example – Id-as-a-service

What is Id-as-service? One of the earliest examples was Microsoft's Passport system, that would verify customer's identity to other services. A physical passport obviously entitles the holder/subject to a number of rights of access (e.g. typically to their own country, but also since it carries additional information such as age, to act as a verification of that fact. The physical passport is made so that it is hard to forge (perhaps watermarked, nowadays also holding a chip with encrypted biometric data) or alter (tamper evident). Digital passports can be similar, however, they (like national mints for currency) rely on a central authorities, and therefore require every user to trust that agency. As we need to verify our digital rights to more and more digital services from many organisations, the central passport idea becomes less and less tenable. Not only this it is represents a substantial risk if it is hacked. It holds too much power and too much responsibility. It is too tempting a target.

Lets re-factor the system design, starting with what the basic service offers: A client presents a key, gets one or more values back. An example key is a biometric (pass phrase, fingerprint, iris, face, palm, TBA) plus possible parameter (age verify, bank account number). The biometric is subject to a robust one-way (hash) function in a way that results in a unique value (despite noise in the biometric scanned value), that can be used as the key to do the lookup of the value associated with that attribute, without the service having to store the actual biometric, or be able to map back to it. Example response values could be binary, "is over 21", or an integer, e.g. bank a/c, or access right "is entitled to NHS care" etc.

So what are the security considerations - lets look at these in decreasing order of naivity:

- The client side should run with security up to and possibly including client user context such as knowing who can see display/location etc.

- The network should be at least TLS. The server side should keep all data encrypted.

- We could run the server in an enclave (SGX, Trustzone etc). A problem is that these are frequently compromised, but we can just use anyway (i.e. confidential cloud) with relatively low performance penalty. but use as well as other privacy tech as part of defense in depth. Enclaves also potentially provide attestation which can also be useful but might depend on a single authority (has to be trusted).

- We could run the server for key/value lookup using FHE. A problem is performance - look up rate would be pretty low throughput.

- We could run server with data sliced/disaggregated and use MPC to do match key to value. A problem: has some latency challenges, but not computationally bad, so scales out.

- We could distribute data over many cloud services and federate.

- Could run a fully distributed bespoke system (possibly non virtualised/not cloud. One candidate doe this is a distributed ledger system(DLT); one added benefit of this is integrity (tamper proof).

- Ledgers can be fully p2p (permissionless) or depend on an access control system (which itself could be distributed or centralised) (permissioned). Mutable data has to be kept off chain, or some new idea applied. DLT also support computation (as part of transactions) and IBM have proposed adding MPC as part of these computations – this seems promising.

- There's a slight circularity here which is the sign on system for a permissioned system requires authorisation. So if the permissioned blockchain is supporting id-as-a-service, who provides the id for the sign-on? Note the entities using the service are as likely to need Id-as-a-service as the subjects (e.g. bank manager is a person too). Permissioned systems also are mainly using authorisation for write/append access. We'd need to enforce read access permissions (and see differential privacy and argument below for trawl problem).

- Self-sovereign systems completely decentralise the Id service. However, thisis seerate from disaggregating or sharding the attrbutes associated with one (or more) Ids, so that control over those attributes (age, membership of club, employment, qualifications, health status, etc) is not in one place.

As discussed earlier, fully decentralised systems have a problem with trust and require another component/service to provide that e.g. proof of work, stake, community etc. Furthermore, in cases where the identity system is being used to verify association with exogeneous factors (e.g. checking if somebody should have access to a building), naturally, it becomes necessary to link to an external trust anchor. These are all known to have scaling or stability / risk challenges and no convincing solution is known. If they aren't good for currency, why should we trust them for Idenity systems? Isn't the bootstrap/circularity problem a showstopper? What is the root of trust?[3].

7 Future Work

We've talked of agency, negotiability, and legibility We talked of symmetry of power, resilience, availability and affordance.

To summarize what we've concluded:

- Just because we virtualise something...doesn't make it OK to treat it differently.
- Data can be payment too – She who can access my data, owes me.
- Think about your shoes – As you walk out of here

8 Further Reading

- Baran, Paul, On Distributed Communications: I. Introduction to Distributed Communications Networks. Santa Monica, CA: RAND Corporation, 1964. `https://www.rand.org/pubs/research\_memoranda/RM3420.html`
- Re-centralisation – Gareth Tyson's mastodon paper[6]
- Edge Cloud[2]
- BitTorrent: Swarms, Multipath, and resource pooling [4]
- Guifi thesis: Development and management of collective network and cloud computing infrastructures by Roger Baig Viñas. `https://dsg.ac.upc.edu/rogerb-phd`
- Xenoservers: Global public computing, by Evangelos Kotsovinos, `https://www.cl.cam.ac.uk/techreports/UCAM-CL-TR-615.pdf`
- Human Data Interaction `https://www.cl.cam.ac.uk/techreports/UCAM-CL-TR-837.pdf`
- Flexible Paxos: Distributed consensus revised, by Heidi Howard `https://www.cl.cam.ac.uk/techreports/UCAM-CL-TR-935.pdf`

[3] We can even envisage different real world parallel views of digital identity information, e.g. as displayed like this `https://www.geekwire.com/2020/delta-air-lines-debuts-crazy-parallel-reality-airport-experience-based-seattle-startups-technology/`

- Royal Society Report on PETS https://royalsociety.org/topics-policy/projects/privacy-enhancing-technologies/
- Databox:
https://www.bbc.co.uk/rd/projects/databox
- HAT:
https://warwick.ac.uk/newsandevents/pressreleases/16312m_hat_project/
- Dataswift:
https://dataswift.io/
- Multiple attacks on federated learning: https://emilianodc.com/publications/

References

[1] Crowcroft, J. The emergent property market. *Internet Policy Review 9 (1)* (2020).

[2] Crowcroft, J., Madhavapeddy, A., Schwarzkopf, M., Hong, T., and Mortier, R. Unclouded vision. In *Proceedings of the 12th International Conference on Distributed Computing and Networking* (Berlin, Heidelberg, 2011), ICDCN'11, Springer-Verlag, p. 29–40.

[3] Goffman, E. The presentation of self in everyday life.

[4] Key, P., Massoulié, L., and Towsley, D. Path selection and multipath congestion control. *Commun. ACM 54*, 1 (Jan. 2011), 109–116.

[5] Piketty, T., and Goldhammer, A. *Capital in the Twenty-First Century.* Harvard University Press, 2014.

[6] Raman, A., Joglekar, S., Cristofaro, E. D., Sastry, N., and Tyson, G. Challenges in the decentralised web: The mastodon case. In *Proceedings of the Internet Measurement Conference* (New York, NY, USA, 2019), IMC '19, Association for Computing Machinery, p. 217–229.

9 Acknowledgements

Thanks to Hamed Haddadi for discussion about some of the ideas in this article.