Distributed Infrastructure For Systems Neuroscience Would Revolutionize The Discipline

Jonny L. Saunders
April 20, 2021

Contents

Dont mind me, just some notes-to-self at the top

I want to be exceedingly clear that I am not shaming anyone for how they do science. I am trying to take effectively the opposite position of those people in the open science community that cast shame and suspicion on people for not passing some purity test. Doing open science is hard because we're lacking a lot of tools that would make it easy, and the social and community structures that make it rewarding — and shaming people works in the opposite direction. Open science shouldn't be about passing some checklist of tests to become holier than thou. The fundamental motivations of open science should be caring about other people: caring about people being able to understand the world, caring about other people not wasting their time, cooperating with each other to do something massive and impossible. Relatedly i am not talking shit about anyone's work!!! Anytime I am describing some criticising some element of something it is because i have been inspired by and learned from it!!! This is about articulating a positive vision!!!

> 💡 - start with discussion of what infrastructure is – making things that seem impossible routine. We haven't addressed these problems after like a decade of writing becasue we haven't identified the real problems nor been bold enough to act. We can move in tiptoe steps but gradual change without a vision is pointless. We need to describe what holds us back and what needs to exist, and the act of converting that into gradual steps is all the work in between.

A lot of this is happening and being worked on!!! this document is an architecture for the whole system. providing a reason to develop one thing vs. another, trying to point devs in one direction but also make users & funders aware of what is possible & the conversations that happen in nerd circles

what's different about this? we're not describing a "new database system" but a way to make every "new database system" add functionality to the system rather than be independent of it.

what are the goals of standardization? - external inspection - labor deduplication - increased access (and so it can't be complicated) - interoperability

what are the traditional costs of standardization? - flexibility - learning curve

*Acknowledgements (make sure to double check spelling!!!): * Lauren E. Wool * Gaby Hayden * Eartha Mae * jakob voigts for participating in the glue wiki * nwb & dandi team for dealing w/ my inane rambling * open behavior team * metascience class for some of these ideas <3 * mike for letting me always go rogue, santiago & matt for being on the committee*

> A good analogy for the development of the Internet is that of constantly renewing the individual streets and buildings of a city, rather than razing the city and rebuilding it. The architectural principles therefore aim to provide a framework for creating cooperation and standards, as a small "spanning set" of rules that generates a large, varied and evolving space of technology. - ftp://ftp.isi.edu/in-notes/rfc1958.txt

> If we can make something decentralised, out of control, and of great simplicity, we must be prepared to be astonished at whatever might grow out of that new medium. https://www.w3.org/1998/02/Potential.html

1. The State of Things

1.1 The Fragmentation of Systems Neuroscience

At all scales, systems neuroscience is fragmented, and its movement as a unified body is mostly a trick of the light, where only the localized patches cohere. We work in technical islands that range from individual researchers, to labs, consortia, and at their largest a few well-funded organizations. Our knowledge dissemination systems are as nimble as the static pdfs and ephemeral conference talks that they have been for decades (save for the godforsaken Science Twitter that we all correctly love to hate). Experimental instrumentation except for that at the polar extremes of technological complexity or simplicity is designed and built custom, locally, and on-demand. Software for performing experiments is a patchwork of libraries that satisfy some of the requirements of the experiment, sewn together by some uncommented script written years ago by a grad student who left the lab long-since. The technical knowledge to build both instrumentation and software is fragmented and unavailable as it sifts through the funnels of word-limited methods sections and never-finished documentation. And O Lord Let Us Pray For The Data, born into this world without coherent form to speak of, indexable only by passively-encrypted notes in a paper lab notebook, dressed up for the analytical ball once before being mothballed in ignominy on some unlabeled external drive.

Rather than a screedy airing of unrelated dirty laundry, I argue that these problems have a shared cause: the state of our **infrastructure**. Far from the siren song it sounds in my ears, many probably hear infrastructure and think "oh like roads and water and stuff, *yawn* *boring!*" (and I ask the rest to permit me the rhetorical strawperson for a moment).

No! Not Boring! Magic! We go and get water where it lives and bring it snaking through a magnificent labyrinth of pipes filters and pumps *directly into your house.* Sometimes multiple places in your house, wherever you want it! And it's practically *free*[1]. People pay $9 a month for Netflix, how much do you think you would pay a for-profit company for a *water subscription?* **Infrastructure makes formerly impossible things so trivial you forget about them.**

I am going to take the position that these problems can be *solved* and are not our inevitable and eternal fate. I will insist the system I describe is not *utopian* but is eminently practical — with a bit of development to integrate them, **everything I propose here already exists and is widely used.** Viewing these problems as stemming from a shared infrastructural etiology allows us to problematize other symptoms that are widely normalized, and find new approaches for recognized problems that are thought to be intractable.

In no particular order, and with no pretense of completeness, the impacts of a lack of scientific infrastructure include:

- A prodigious duplication and dead-weight loss of labor as each lab, and sometimes each person within each lab, will reinvent basic code, tools, and practices from scratch. Literally it is the inefficiency of the Harberger's triangle in the supply and demand system for scientific infrastructure caused by inadequate supply. Labs with enough resources are forced to pay from other parts of their grants to hire professional programmers and engineers to build the infrastructure for their lab (and usually their lab or institute only), but most just operate on a purely amateur basis. Many PhD students will spend the first several years of their degree re-solving already-solved problems, chasing the tails of the wrong half-readable engineering

[1] In this paragraph I am of course only referring to many parts of the urbanized United States (with many exceptions) to illustrate the luxury of infrastructure. Water availability is a global humanitarian crisis increasingly exacerbated by climate change, confoundingly also caused by other forms of infrastructure, which is why it's essential to think about the unintended consequences and ethical implications of the kind of infrastructure we build.

whitepapers, in their 6th year finally discovering the technique that they actually needed all along. That's not an educational or training model, it's the effect of displacing the undone labor of unbuilt infrastructure on vulnerable graduate workers almost always paid poverty wages.

- A profoundly leaky knowledge acquisition system where entire PhDs worth of data can be lost and rendered useless when a student leaves a lab and no one remembers how to access the data or how it's formatted. Indeed needing to learn data hygeine practices like backup, annotation, etc. "the hard way" through some catastrophic loss is accepted myth in much of science. At some scale all the very real and widespread pain, and guilt, and shame felt by people who had little choice but to reinvent their own data management system must be recognized as an infrastructural, rather than a personal problem.
- At least the partial cause of the phenomenon where "every scientist needs to be a programmer now" as people who aren't particularly interested in being programmers — which is *fine* and *normal* — need to either suffer through code written by some other unlucky amateur or learn an entire additional discipline in order to do the work of the one they chose. Because there isn't more basic scientific programming infrastructure, everyone needs to be a programmer.
- The dearth of data transparency where it is still in the year of our lord 2021 rare for systems neuro papers to publish the full, raw data along with all the analysis code, often because (in addition to the extraordinarily meagre incentives to do so) the data *and* analysis code are both completely homebrew and often omitted just due to the labor of cleaning it or the embarassment of sharing it[2]. We can't expect data transparency from researchers while it is still so *hard*.
- The inevitability of a replication crisis because it is often literally impossible to replicate an experiment that is done on a rig that was built one time, used entirely in-lab code, and was never documented
- A perhaps doomed intellectual endeavor as we attempt to understand the staggering complexity of the brain by peering at the brain through the pinprickiest peephole of just the most recent data you or your lab have collected rather than being able to index across all relevant data from not only your lab, but all other labs that have measured the same phenomena. The unnecessary reduplication of experiments becomes not just a methodological limitation, but an ethical catastrophe as researchers have little choice but to abandon the elemental principle of sacrificing as few animals as possible to understand a phenomenon.
- A hierarchy of prestige that devalues the labor of multiple groups of technicians, animal care workers, and so on. Authorship is the coin of the realm, but many researchers that do work fundamental to the operation of science only receive the credit of an acknowledgement. We need a system to value and assign credit for the immense amount of technical and practical knowledge and labor they produce.
- An insular system where the inaccessibility of all the "contextual" knowledge [1, 2] that is beneath the level of publication but necessary to perform experiments, like "how to build this apparatus," "what kind of motor would work here," etc. is a force that favors established and well-funded labs who can rely on local knowledege and hiring engineers/etc. and excludes new, lesser-funded labs at non-ivy institutions. The concentration of technical knowledge magnifies the inequity of strongly skewed funding distributions such that the most well-funded labs can do a completely different kind of science than the rest of us, turning the positive-feedback loop of funding begetting funding ever faster.
- An absconscion with the public resources we are privileged enough to receive, where rather than returning the fruits of the many technical challenges we are tasked with solving to the public in the form of data, tools, collected practical knowledge, etc. we largely return papers, multiplying the above impacts of labor duplication and knowledge inaccessibility by the scale of society.

and so on.

Considered separately, these are problems, but together they are a damning indictment of our role as stewards of our corner of the human knowledge project.

!! lots of work being done here on all this stuff, and it's heroic!!! But under the constraints of budget and time it's not really possible, and without a unified vision for how all our work might eventually cohere into something larger it's unlikely to do it by its own like water sucked into a droplet by its surface tension. a lot of these problems have been evaluated under the banner of open science, but we can't agree on what that term means let alone what we should do about it, which has caused bad shit to happen and for people to become disillusioned with the idea altogether. My goal not to not say that people should listen to me

[2] which, to be clear, is a valid feeling and is reflective of a failure of infrastructure, not a personal failure.

as some sort of prophet, it's the opposite: that we should listen to the many knowledge communities that have been working on related problems but so often escape our consideration. My goal is not to hype some product or company that i'm selling, it's the opposite: to create systems that make us free and powerful rather than reliant on a grab-bag of paid, proprietary platforms. My goal is not to shame people for the way they do science, it's the opposite: to articulate a series of systems that let us all benefit from the different ways we do science. My goal is to *give us a vision of how we can make things better for each other together.*

!!! replace vvv with ^^^

We arrive at this situation not because systems neuroscientists are lazy and stupid, but because the appropriate tools that fit the requirements of their discipline don't exist, and traditional patterns of centralized organization can't scale to encompass their diverse needs. If the above description doesn't resonate with you and I have already made an enemy of you as a reader, maybe reading along thinking about all the exceptions to the above problems, the difficulty of solving them, or maybe my unfortunate tendency towards utopianism grates: *yes* there are many people and groups working on or who have solved some of these problems, *yes* these problems are all difficult, and *yes* I am *obnoxious* about *infrastructure* and it is a *personality flaw.* Bear with me at least a few more paragraphs.

1.2 A Brief, Beautiful Dream of Infrastructure

As a break from doomsaying, imagine the positive vision of doing neuroscience with all the power of basic infrastructure.

You have some new research question, and so you turn to the standard Python (or whatever) library that allows you to query data from yours and all other labs who share their data with this system. You're immediately able to filter through to find all the recordings from a particular subtype of cell in a particular region being exposed to some particular set of stimuli across some particular manipulation. Since you have access to decades of labor by thousands of scientists, even with that complex filter you still find, say for the sake of having a round cool-sounding number, a million recordings. Because they're all in some standardized format, over the years a common analysis pipeline has been developed, so you're also immediately able to perform the analyses to confirm the hunch for your new question — and it's time to implement it.

You don't need to implement the whole thing from scratch because you can check out a similar experiment from the standardized experimental software framework, read the communally maintained documentation, make the minor tweaks you need for your experiment, and you're off and running. You need to build some brand new component, but you also have a practical knowledge repository where other scientists working on similar problems have described the basic components, circuits, and have even uploaded some 3d-printable components for you to use. Because the repository was designed for ease of use and has a robust system of community incentives for contribution, as you build you document what you learn, and when you're finished upload the schematics and write instructions for your new component. The experimental software framework was designed to incorporate custom components, so you extend some similar hardware control code and integrate it with your experiment without needing to resort to some patchwork system of TTL synchronization pulses and serial port arcana.

You did it! Experiment Over! The experimental framework produces data that is clean, annotated, and standardized at the time of acquisition, and automatically integrates it with the analysis pipeline you built when your experiment was just a budding baby hypothesis, so your analysis is finished shortly after the experiment is. You have the "auto-upload" setting on, so without any additional effort your work has been firehosing information back the global knowledge pool. You do a pull request for the improvements you've made to the experimental software, write the paper, and the loop is complete: a closed knowledge system where nothing is wasted and everyone is more capable and empowered by drawing from and contributing to it.

Such a dream need not be only a dream, because each of these are tangibly realizable platforms and tools — the question is less *whether it's possible* to build basic infrastructure, but more *how do we do it.* An easy misstep is to categorize this as solely a *technical* challenge, that these are just elaborations on our current tools and systems that will just come with time and continued development. Instead the challenge of infrastructure is also one of *design,* where the tools need to accomodate the needs of our discipline and ultimately make it *easier than not to do good science.* It also represents a *cultural* shift away from the individualistic heroism endemic in ours and many scientific discipline towards one where we view using and contributing to communal tools and knowledge repositories not as a nice extra thing some people do but incumbent upon us as scientists.

This document will attempt to be both a conceptual vision of the design of scientific infrastructure as well as a practical outline of the tools and path to realizing it. Both are essential, but make some conflicting demands on the construction of the piece: Making

real progress in the constellation of problems above requires considering their interrelatedness and mutual reinforcement, rather than treating them as isolated problems that can be addressed piecemeal. Such a broad scope trades off with a detailed description of the relevant technology and systems, but a myopic techno-zealotry that does not examine the social and ethical nature of scientific practice risks reproducing or creating new sources of harm (see {mirowskiFutureOpenScience2018} for a particularly baroque expression of a related argument with respect to the open science movement).

Let's be super clear though, this is not some far-off fantastical vision, all of the technologies i describe are real and exist in some form. What I am describing here is the potential of combining them in a principled way.

Allow me to make explicit some of my beliefs and biases that motivate and structure the arguments in this paper. *(then actually do it lol)*

I argue that infrastructure development also requires us to rethink the way we *organize* ourselves, where rather than turning to yet another *centralized* system, we learn from decades of experience from academic, activist, and digital communities that *decentralized* systems can deliver us the needed flexibility and resiliency. Neuroscience is not physics or astronomy, and its specific affordances and constraints need to be interrogated to imagine how a model of large-scale collaboration should overlap and diverge from that of national labs and observatories.

I believe the problem of basic infrastructure in systems neuroscience is less insurmountable than it may appear, but before a roadmap of where to go from here it is valuable to dwell a bit longer on where we are now to concretize what should be done differently.

1.3 Why is it like this?

Every discipline has its own particular technical needs, and is subject to its own peculiar history and culture. Though the type of comprehensive distributed infrastructure I will describe later is a domain-general project, systems neuroscience specifically lacks some features of it that are present in immediately neighboring disciplines like genetics and cognitive psychology. I won't attempt a complete explanation, but instead will offer a few patterns I have noticed in my own limited exposure to the field that might serve as the beginnings of one. I want to be very clear throughout that I am never intending to cast shade on the work of anyone who has or does build and maintain the scientific infrastructure that exists — in fact the opposite, that y'all deserve more resources.

1.3.1 Diversity of Measurements

Molecular biology and genetics are perhaps the neighboring disciplines with the best data sharing and analytical structure, spawning and occupying the near totality of a new subdiscipline of Bioinformatics (for an absolutely fascinating ethnography, see [3],).Though the experiments are of course just as complex as those in systems neuroscience, most rely on a small number of stereotyped sequencing (meta?)methods that result in the same one-dimensional, four character sequence data structure of base pairs. Systems neuroscience experiments increasingly incorporate dozens of measurements, electrophysiology, calcium imaging, multiple video streams, motion, infrared, and other sensors, and so on. This is increasingly true as neuroscientists are attempting ever more complex and naturalistic neuroethological experiments. Even the seemingly "common" electrophysiological or multiphoton imaging data can have multiple forms — raw voltage traces? spike times? spike templates and times? single or multiunit? And these forms go through multiple intermediate stages of processing — binning, filtering, aggregating, etc. — each of which could be independently valuable and thus represented alongside their provenance in a theoretical data schema. Mainen and colleagues note this problem as well:

> The data sets generated by a functional neuroscience experiment are large. They can also be complex and multimodal in ways that, say, genomic data might not be, embracing recordings of activity, behavioural patterns, responses to perturbations, and subsequent anatomical analysis. Researchers have no agreed formats for integrating different types of information. Nor are there standard systems for curating, uploading and hosting highly multimodal data. [4],

The Neurodata Without Borders project has made a valiant effort to unify these multiple formats, but has for reasons that I won't lay claim to knowing has yet to see widespread adoption. Contrast this with the BIDS data structure for fMRI data, where

by converting your data to the structure you unlock a huge library of analysis pipelines for free. The beginnings of generalized platforms for neuroscientific data built on top of NWB are starting to happen in trickles and droplets, but they are still very much the exception rather than the rule.

We should not be so proud as to believe that our data is somehow uniquely complex. Theorizing about and reconciling the mass and heterogeneity of data in the universe is the subject of multiple full-fledged disciplines, and the conflict between simplified and centralized [5],andsprawling and distributed [6],systemsis well-trodden — and we should learn from it! We could instead think of the complexity of our data and the tools we develop to address it as what we have to offer the broader human mission towards a unified system of knowledge.

1.3.2 Diversity of Preps

Though there are certain well-limbered experimental backbones like the two-alternative forced choice task, even within them there seems to be a comparatively broad diversity of experimental preparations in systems neuro relative to adjacent fields. Even a visual two-alternative forced choice task is substantially different than an auditory one, but there is almost nothing shared between those and, for example, measuring the representation of 3d space in a free-flying echolocating bat. So unlike cognitive neuroscience and psychophysics that has tools like pavlovia where the basic requirements and structure of experiments are more standardized, BioRXiv is replete with technical papers documenting "high throughput systems for this one very specific experiment" and there isn't a true experimental framework that satisfies the need for flexibility.

Mainen and colleagues note that this causes another problem distinct from variable outcome data, the even more variable and largely unreported metadata that parameterizes the minute details of experimental preps:

> Worse, neuroscientists lack standardized vocabularies for describing the experimental conditions that affect brain and behavioural functions. Such a vocabulary is needed to properly annotate functional neural data. For instance, even small differences in when a water drop is released can affect how a mouse's brain processes this event, but there is no standard way to specify such aspects of an experiment. [4],

The problem of universal annotation and metadata reporting can be reframed, not as a *barrier to developing*, but as a *design constraint* of experimental programming infrastructure. Because of the fragmentation of scientific programming infrastructure, where each experimental prep is implemented with entirely different, and often single-use software, there is no established reporting system for automatically capturing these minute details — but that doesn't mean there can't be (as I wrote previously, see section 2.3 in [7],,and coincidentally measured the effect of variable water droplets).

1.3.3 The Hacker Spirit and Celebration of Heroism

Many people are attracted to systems neuroscience precisely *because* of the… playful… attitude we take towards our rigs. If you want to do something, don't ask questions just break out the hot glue, vaseline, and aluminum foil and hack at it until it does what you want. The natural conclusion of widespread embodiment of this lovable scamp hacker spirit is its veneration as heroism: it is a *good thing* to have done an experiment that only you are capable of doing because that means you're the best hacker. Not unrelated is the strong incentive to make something new rather than build on existing tools — you don't get publications from pull requests, and you don't get a job without publications. The initial International Brain Laboratory described the wily nature of neuroscientists accordingly:

> Simply maintaining a true collaboration between 21 laboratories accustomed to going their own way will be a major novelty in neuroscience. [8],

And yes, like the rest of the universe, perhaps the most influential forces in this domain are inertia and entropy. Once the boulder starts rolling down the hill of heroic idiosyncracy, tumbling along in a semi-stable jumble[3] that supports the experiments of a lab, retooling and standardizing that system has to be *so very cool and worth it* that it overcomes the various, uncertain, but typically substantial costs (including the valid emotional costs of wishing a peaceful voyage to well-loved handcrafted tools). More than a

[3] A *lovely* jumble! that probably has a lot of good qualities, it's just a little lonely maybe :(

single moment of adoption, the universe always has room for another course of disorder, and a commitment to using communal tools must be constantly reaffirmed. As we dream up new wild experiments, it needs to be easier to implement them with the existing system and integrate the labor expended in doing so back into it than it is to patch over the problem with a quick script saved to Desktop. As people cycle through the lab, it must be easier to learn than it is to start from scratch.

Yes again, Mainen and colleagues:

> Neuroscientists frequently live on the 'bleeding' edge technologically, building bespoke and customized tools. This do-it-yourself approach has allowed innovators to get ahead of the competition, but hampered the standardization of methods essential to making experiments efficient and replicable.
>
> Remarkably, it is standard practice for each lab to custom engineer all manner of apparatus, from microscopes and electrodes to the computer programmes for analysing data. Thousands of labs worldwide use the calcium sensor GCaMP, for example, for imaging neural activity in vivo. Yet neither the microscopes used for GCaMP imaging nor the algorithms used to analyse the resulting data sets have been standardized. Include [4],:

Each of these three disciplinary tendencies

!! The problems are also structural, and vary depending on the size, resources, etc. of the institution as well… transition to next section

1.4 The Ivies, Consortia, and "Most of Us"

The initial picture I painted of the state of Systems Neuroscience describes what I, in my limited exposure to the broader field, think might be typical for "most of us." There are admirable efforts to standardize on tools and realize "meso-scale collaboration" [4],,and even for those that are not the experience of infrastructure can vary dramatically by institution. To draw contrast, I'll consider the case of core facilities at well-funded institutions and a few existing collaborations. My intention is not to denigrate anyone's hard work, but to learn from it.

Centralized "core" facilities are maybe the most typical form of standardization and resource sharing at the level of departments and institutions. These facilities can range from minimal to baroque extravagance depending on institutional resources and whatever complex web of local history brought them about.

PNI Systems Core lists subprojects echo a lot of the thoughts here, particularly around effort duplication[4]:

> Creating an Optical Instrumentation Core will address the problem that much of the technical work required to innovate and maintain these instruments has shifted to students and postdocs, because it has exceeded the capacity of existing staff. This division of labor is a problem for four reasons: (1) lab personnel often do not have sufficient time or expertise to produce the best possible results, (2) the diffusion of responsibility leads people to duplicate one another's efforts, (3) researchers spend their time on technical work at the expense of doing science, and (4) expertise can be lost as students and postdocs move on. For all these reasons, we propose to standardize this function across projects to improve quality control and efficiency. Centralizing the design, construction, maintenance, and support of these instruments will increase the efficiency and rigor of our microscopy experiments, while freeing lab personnel to focus on designing experiments and collecting data.

While core facilities are an excellent way of expanding access, reducing redundancy, and standardizing tools within an instutition, as commonly structured they can displace work spent on those efforts outside of the institution. Elite institutions can attract the researchers with the technical knowledge to develop the instrumentation of the core and infrastructure for maintain it, but this development is only occasionally made usable by the broader public. The Princeton data science core is an excellent example

[4] Thanks a lot to the one-and-only stunning and brilliant Dr. Eartha Mae Guthman for suggesting looking at the BRAIN initiative grants as a way of getting insight on core facilities.

of a core facility that does makes its software infrastructure development public[5], which they should be applauded for, but also illustrative of the problems with a core-focused infrastructure project. For an external user, the documentation and tutorials are incomplete – it's not clear to me how I would set this up for my institute, lab, or data, and there are several places of hard-coded princeton-specific values that I am unsure how exactly to adapt[6]. I would consider this example a high-water mark, and the median openness of core infrastructure falls far below it. I was unable to find an example of a core facility that maintained publicly-accessible documentation on the construction and operation of its experimental infrastructure or the management of its facility.

Outside of universities, the Allen Brain Institute is perhaps the most impactful reflection of centralization in neuroscience. The Allen Institute has, in an impressively short period of time, created several transformative tools and datasets, including its well-known atlases [9],andthe first iteration of its Observatory project which makes a massive, high-quality calcium imaging dataset of visual cortical activity available for public use. They also develop and maintain software tools like their (SDK)[https://allensdk.readthedocs.io/en and Brain Modeling Toolkit (BMTK), as well as a collection of (hardware schematics)[https://portal.brain-map.org/explore/toolkit/hardware] used in their experiments. The contribution of the Allen Institute to basic neuroscientific infrastructure is so great that, anecdotally, and at least in my neck of the woods, when talking about infrastructure the default belief is "I thought the Allen was doing that."

Though the Allen Institute is an excellent model for scale at the level of a single organization, its centralized, hierarchical structure cannot (and does not attempt to) serve as the backbone for all neuroscientific infrastructure. Performing single (or a small number of, as in its also-admirable OpenScope Project) carefully controlled experiments a huge number of times is an important means of studying constrained problems, but is complementary with the diversity of research questions, model organisms, and methods present in the broader neuroscientific community. Christof Koch, its director, describes the challenge of centrally organizing a large number of researchers:

> Our biggest institutional challenge is organizational: assembling, managing, enabling and motivating large teams of diverse scientists, engineers and technicians to operate in a highly synergistic manner in pursuit of a few basic science goals [10],

> These challenges grow as the size of the team grows. Our anecdotal evidence suggests that above a hundred members, group cohesion appears to become weaker with the appearance of semi-autonomous cliques and sub-groups. This may relate to the postulated limit on the number of meaningful social interactions humans can sustain given the size of their brain [11],

Given the diminishing returns to scale for centralized organizations, many have called for smaller, "meso-scale" collaborations and consortia that combine the efforts of multiple labs [4],.The most successful consortium of this kind has been the International Brain Laboratory [8, 1], a group of 22 labs spread across six countries. They have been able to realize the promise of big team neuroscience, setting a new standard for performing reproducible experiments performed by many labs [12],anddeveloping data management infrastructure to match [13],(seriously,don't miss their extremely impressive data portal). Their project thus serves as the benchmark for large-scale collaboration and a model from which all similar efforts should learn from.

[5]

> Project Summary: Core 2, Data Science Working memory, the ability to temporarily hold multiple pieces of information in mind for manipulation, is central to virtually all cognitive abilities. This multi-component research project aims to comprehensively dissect the neural circuit mechanisms of this ability across multiple brain areas. In doing so, it will generate an extremely large quantity of data, from multiple types of experiments, which will then need to be integrated together. The Data Science Core will support the individual research projects in discovering relationships among behavior, neural activity, and neural connectivity. The Core will create a standardized computational pipeline and human workflow for preprocessing of calcium-imaging data. The pipeline will run either on local computers or in cloud computing services, and users will interact with it through a web browser. The preprocessing will incorporate existing image-processing algorithms, such as Constrained Nonnegative Matrix Factorization and convolutional networks. In addition, the Core will build a data science platform that stores behavior, neural activity, and neural connectivity in a relational database that is queried by the DataJoint language. Diverse analysis tools will be integrated into DataJoint, enabling the robust maintenance of data-processing chains. This data-science platform will facilitate collaborative analysis of datasets by multiple researchers within the project, and make the analyses reproducible and extensible by other researchers. We will develop effective methods for training and otherwise disseminating our computational tools and workflows. Finally, the Core will make raw data, derived data, and analyses available to the public upon publication via the data-science platform, source-code repositories, and web-based visualization tools. To facilitate the conduct of this research, the creation of software tools, and the reuse of the data by others after the primary research has concluded, the project will adopt shared data and metadata formats using the HDF5 implementation of the Neurodata without Borders format. Data will be made public in accord with the FAIR guiding principles—findndable by a DOI and/or URL, accessible through a RESTful web API, and interoperable and reusable due to DataJoint and the Neurodata Without Borders format for data https://projectreporter.nih.gov/project_info_description.cfm?aid=9444126&icde=0

[6]Though again, this project is examplary, built by friends, and would be an excellent place to start extending towards global infrastructure.

Critical to the IBL's success was its adoption of a flat, non-hierarchical organizational structure, as described by Lauren E. Wool:

> IBL's virtual environment has grown to accommodate a diversity of scientific activity, and is supported by a flexible, 'flattened' hierarchy that emphasizes horizontal relationships over vertical management. [...] Small teams of IBL members collaborate on projects in Working Groups (WGs), which are defined around particular specializations and milestones and coordinated jointly by a chair and associate chair (typically a PI and researcher, respectively). All WG chairs sit on the Executive Board to propagate decisions across WGs, facilitate operational and financial support, and prepare proposals for voting by the General Assembly, which represents all PIs. In parallel, associate chairs convene on their own committee to share decisions, which are then conveyed to the entire researcher community so it may weigh in on proposals before a formal vote. The interests of PIs and researchers intersect via staff liaisons who sit on both the Executive Board and the Associate Chairs Committee, as well as an elected researcher representative, who sits on the Executive Board and is a voting member of the General Assembly. [1],

They should also be credited with their adoption of a form of consensus decision-making, sociocracy, rather than a majority-vote or top-down decisionmaking structure. Consensus decision-making systems are derived from those developed by Quakers and some Native American nations, and emphasize, perhaps unsurprisingly, the value of collective consent rather than the will of the majority. Sociocracy specifically describes consent:

> Consent means "no objections." Giving consent does not mean unanimity, agreement, or even endorsement. Decisions are made to guide actions. Can we move forward if we make this decision? Consent is given in the context of moving forward. Consent to a policy decision means you believe that it is "worth trying." Or "I can work with it." Moving forward is important for making better decisions because it provides more information. Not moving forward until a perfect decision is found, means operating in the blind. Information will always be limited to what is already known.
>
> Consent is required for all policy decisions for many reasons. The two most important are that it ensures (1) the decision will allow all members of the group to participate or produce without feeling oppressed, and (2) it will be supported by everyone. Everyone is expected to participate in the reasoning behind the decision. And no one can be excluded. https://www.sociocracy.info/what-is-sociocracy/

The central lesson of the IBL, in my opinion, is that governance matters. Even if a consortium of labs were to form on an ad-hoc basis, without a formal system to ensure contributors felt heard and empowered to shape the project it would soon become unsustainable. Even if this system is not perfect, with some labor still falling unequally on some researchers, it is a promising model for future collaborative consortia.

The infrastructure developed by the IBL is impressive, but its focus on a single experiment makes it difficult to expand and translate to widescale use. The hardware for the IBL experimental apparatus is exceptionally well-documented, with a complete and detailed build guide and library of CAD parts, but the documentation is not modularized such that it might facilitate use in other projects, remixed, or repurposed. The experimental software is similarly single-purpose, a chimeric combination of Bonsai [14],andPyBpod scripts. It unfortunately lacks the API-level documentation that would facilitate use and modification by other developers, so it is unclear to me, for example, how I would use the experimental apparatus in a different task with perhaps slightly different hardware, or how I would then contribute that back to the library. The experimental software, according to the PDF documentation, will also not work without a connection to an alyx database. While alyx was intended for use outside the IBL, it still has IBL-specific and task-specific values in its source-code, and makes community development difficult with a similar lack of API-level documentation and requirement that users edit the library itself, rather than temporary user files, in order to use it outside the IBL.

My intention is not to denigrate the excellent tools built by the IBL, nor their inspiring realization of meso-scale collaboration, but to illustrate a problem that I see as an extension of that discussed in the context of core facilities — designing infrastructure for one task, or one group in particular makes it much less likely to be portable to other tasks and groups.

It is also unclear how replicable these consortia are, and whether they challenge, rather than reinforce technical inequity in science. Participating in consortia systems like the IBL requires that labs have additional funding for labor hours spent on work for the consortium, and in the case of graduate students and postdocs, that time can conflict with work on their degrees or personal research which are still far more potent instruments of "remaining employed in science" than collaboration. In the case that only the most well-funded labs and institutions realize the benefits of big team science without explicit consideration given

to scientific equity, mesoscale collaborations could have the unintended consequence of magnifying the skewed distribution of access to technical expertise and instrumentation.

Outside of ivies with rich core facilities, institutes like the Allen, or nascent multi-lab consortia, the situation errs closer to the dire picture of fragmentation I painted above. In addition to the homebrew stuff, there is an ocean of open-source software and hardware that keeps us afloat. There are far too many projects to name here[7], each covering some subset of experimenters needs, only rarely integrated with one another, and so to some degree the task of many scientific programmers is to search out the latest packages and quilt them into our patchwork local infrastructure. Anything else comes down to whatever we can afford with remaining grant money, scrape together from local knowledge, methods sections, begging, borrowing, and (hopefully not too much) stealing from neighboring labs.

A third option from the standardization offered by centralization and the blooming, buzzing, beautiful chaos of disconnected open-source development is that of decentralized systems. Rather than systems of geographically decentralized *people or lab-sites*, what must be decentralized is the *infrastucture itself:* we need to build the means by which the "rest of us" can mutually benefit by capturing and making use of each other's knowledge and labor.

2. A Vision of Distributed Scientific Infrastructure

The distributed infrastructure I will describe here is related to previous notions of "grass-roots" science [4],,and my intention is to provide a more prescriptive scaffolding for its design and potential implementation as a way of painting a picture of what science could be like.

Throughout this section, when I am referring to any particular piece of software I want to be clear that I don't intend to be dogmatically advocating that software *in particular*, but software *like it* that *shares its qualities* — no snake oil is sold in this document. Since this is a design document, I will also be saying we *should* do a lot of things — think of that as "to fulfill this system, we should do this," rather than "everyone should do this even if they disagree with the fundament of my argument." Similarly, when I describe limitations of existing tools, without exception I am describing a tool or platform I love, have learned from, and think is valuable — learning from something can mean drawing respectful contrast!

At all points, I assume that the particular tool has a *well designed UI/UX* such that it is relatively simple to use and understand — if it takes a college degree to turn the water on, then it ain't infrastructure. All the things I describe here either already exist or are extensions of things that exist, so good design may require improvements but is in all cases possible. Practicality matters: infrastructure will only work of it's widely adopted, and it will only be widely adopted if it is easier and more rewarding to use than the costs of transition.

I won't attempt a derivation of a definition of decentralized systems from base principles here, but a concrete example of one is very close to home: the internet (or, specifically, the Internet Protocol, or IP). The history of the internet is, at the time of writing, still very near at hand, and much of its design philosophy has been carefully articulated by the engineers and designers that created it. A small selection of these principles hint at what might be required of distributed infrastructure for neuroscience, in no particular order:

- **Integrate with what exists** - At its advent, several different institutions and universities had already developed existing network infrastructures, and so the "top level goal" of the Internet Protocol was to "develop an effective technique for multiplex utilization of existing interconnected networks," and "come to grips with the problem of integrating a number of separately administered entities into a common utility" [15],.As a result, IP was developed as a 'common language' that could be implemented on any hardware, and upon which other, more complex tools could be built. This is also a cultural practice: when the system doesn't meet some need, one should try to extend it rather than building a new, separate system — and if a new system is needed, it should be interoperable with those that exist.
- **Empower the end-user** - Becauase IP was initially developed as a military technology by DARPA, a primary design con-

[7] That we love!!!! Pay developers!!!!!

straint was survivability in the face of failure. The model adopted by internet architects was to move as much functionality from the network itself to the end-users of the network — rather than the network itself guaranteeing a packet is transmitted, the sending computer will do so by requiring a response from the recipient. For infrastructure, we should make tools that don't require a central team of developers to maintain, a central server-farm to host data, or a small group of people to govern. Whenever possible, data, software, and hardware should be self-describing, so one needs minimal additional tools or resources to understand and use it.

- **Modularity is Flexibility** - Building each component once, and once only requires that it "knows" about as few other parts of the system as possible. Once a component is built well, it can be reused and repurposed in contexts not imagined in its original design. Modularity is also critical for large-scale use: partial adoption partially captures development labor. Allowing users to gradually incorporate the pieces of a system into their existing infrastructure also lowers the barriers of high transitional costs to eventual complete adoption. A reciprocal principle to modularity is "the test of independent invention", or "If someone else had already invented your system, would theirs work with yours?" [16],.In other words, in addition to the system itself being modular, it should also be designed so there is some sensible means for it to be integrated into some yet-unspecified larger project. The machine needs to have knobs.
- **Embrace Heterogeneity** - Distributed systems need to anticipate unanticipated uses. Rather than a prescribed set of supported hardware, affordance needs to be made such that there is a clear way to extend the system to incorporate new function [17],.
- **Scalability is The Metric** - The system needs to have minimal barriers to use such that it can be deployed by as many users as possible — scale is not just a design principle, but an independent objective and means of valuation for distributed systems. Logic or functionality that can only be used by a specific set of users breaks the system. Hand in hand with embracing heterogeneity, infrastructure needs to be able to be adopted by users with a minimal set of assumptions about their resources, organization, or expertise.

With these principles in mind, and drawing from other knowledge communities solving similar problems: internet infrastructure, library/information science, peer-to-peer networks, and radical community organizers, I conceptualize a system of distributed infrastructure for systems neuroscience as four objectives: **shared data**, **shared tools**, **shared knowledge**, and **shared governance**.

2.1 Shared Data

2.1.1 Common Format

Neuroscientific data should be stored in a single, common format. Given the absence of notable competitors and existing partial standardization, we should adopt Neurodata Without Borders:N. I don't expect a lot of controvery here[8]. Individual labs writing functions for converting their data to NWB is a comparatively simple, concrete first step that is a prerequitite for the remainder of the system. It could even be fun, we could think of it like a big years-long slumber party where we all learn one dance routine.

Standardization does *not* mean that it is the *only* format that is used — there are legitimate applications for keeping data, even temporarily, in intermediate formats. Standardization, in this case, means that the data has some trivial conversion to and from NWB, so for example some experimental tool could implement its own data model as long as it could be exported to NWB.

> change from here to the next marker to something about the importance of relationships between data formats, simplify that way. introduce the concept for later where we talk about federation and later in rdf

Relatedly, the NWB API should be extended to include conversions between prior and subsequent versions of the standard: when the standard is changed, there should also be a function that converts the previous to the new version and vice versa. Once data is in NWB, it would then be trivial to maintain compatibility while allowing the standard to evolve as needed.

[8] but I am also almost always wrong when I think this.

If such a conversion function was implemented such that it was easy to extend, then it would also allow researchers to make local modifications to the standard to suit their needs, while retaining standardization with the root format. NWB files would then effectively be version controlled, and innovation on the standard could be integrated into the root standard and made available to all existing data. If compatibility-preserving extension of the protocol was possible, then the temptation to create accessory file and directory storage to contain "undefinable" data is minimized and it becomes possible to additional stipulate that doing so should be avoided.

Wide adoption of NWB is not, in my opinion, the end goal in itself. Instead I see it as a point of standardization on the way to a more generalized, interlinked system of linked schema, articulated further in the following sections and in the discussion of shared knowledge.

| can write about ^^^ in terms of provenance https://www.w3.org/TR/prov-overview/

2.1.2 Peer-to-peer data sharing platform

We should develop a platform for sharing all neuroscientific data. There are, of course many existing databases for scientific data, ranging from domain-general like figshare and zenodo to the most laser-focused subdiscipline-specific. For all these databases, their centralization is a fundamental constraint to adoption and growth. We can learn from two knowledge communities with decades of domain-specific knowledge in resiliently storing and sharing massive quantities of extremely heterogeneous and metadata-rich information: internet pirates and information scientists. We should develop a peer-to-peer, semantically annotated data sharing platform.

Centralized servers are fundamentally constrained by their storage capacity and bandwidth, both of which cost money. In order to be free, database maintainers need to constantly raise money from donations or grants[9] in order to pay for both. Funding can never be infinite, and so inevitably there must be some limit on the amount of data that someone can upload and the speed at which it can serve files[10]. Even if the database is carefully backed up, it serves as a single point of infrastructural failure, where if the project lapses then at worst data will be irreversibly lost, and at best a lot of labor needs to be expended to exfiltrate, reformat, and rehost the data. The same is true of local, institutional-level servers and related database platforms, with the additional problem of skewed funding allocation making them unaffordable for many researchers.

Peer to peer systems have none of these problems, are inexpensive to maintain, and increase, rather than decrease, in performance the more people use them. In order to proceed with the rest of this section we need to give a brief description of a peer to peer networking protocol: Bittorrent. If you are already familiar with the basics of Bittorrent, you can safely collapse and skip the next section. Note that I am just using Bittorrent as an example, contemporary P2P systems have made substantial improvements on Bittorrent[11], explained after the interlude.

Bittorrent Interlude Click to expand/collapse the ~bittorrent interlude~

In a traditional server/client model, uploading and downloading files is straightforward: one computer transfers the whole thing to another sequentially. Bittorrent, as a peer-to-peer network, is designed so that everyone that wants and has a file sends pieces of it to each other.

We can walk through an example. In each of the following images, each of the circles represents a computer. The clusters of circles on the left side are uploading and downloading from a traditional server, while those on the right are downloading as part

[9] granting agencies seem to love funding new databases, idk.
[10] As I am writing this, I am getting a (very unscientific) maximum speed of 5MB/s on the Open Science Framework
[11] peer to peer systems are, maybe predictably, a whole academic subdiscipline. See [18],forreference.

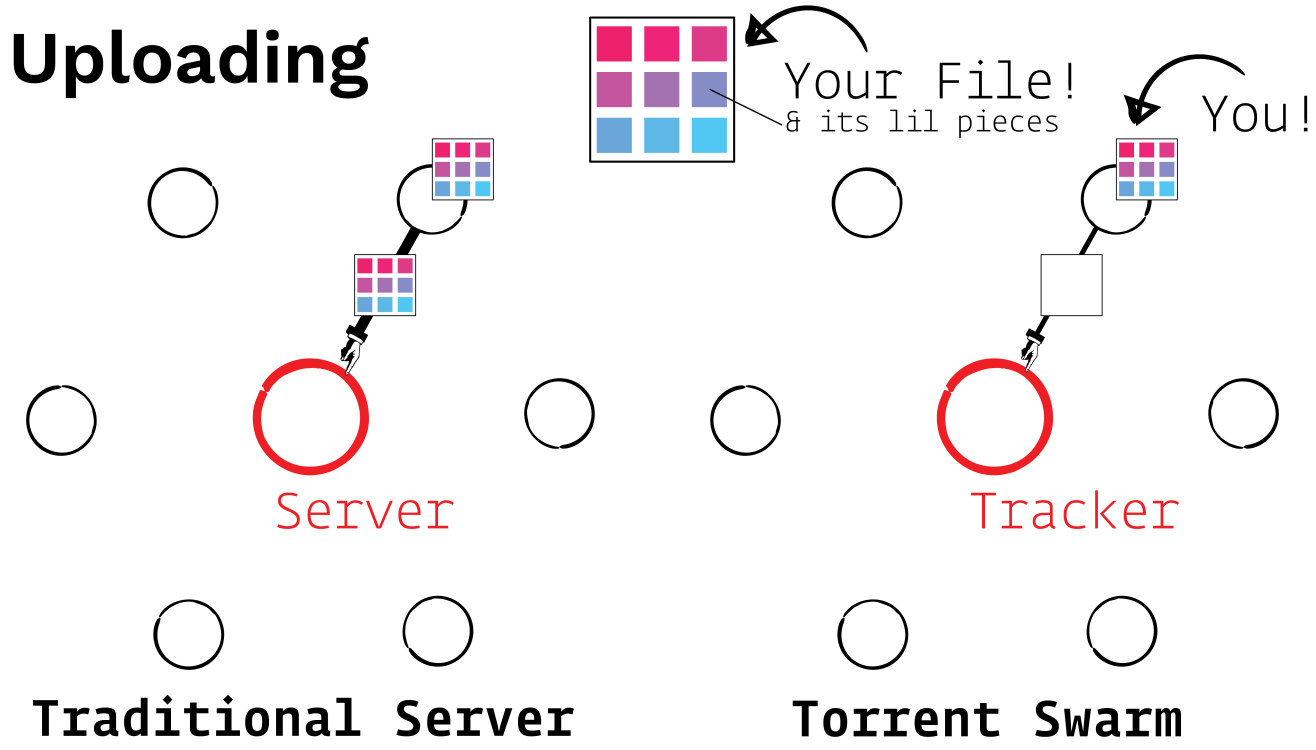


Figure 1: Uploading A file

of a bittorrent "swarm." Grids of colored squares represent a whole file, and each of the colored squares is just an arbitrary piece of that file.

Say you're an enterprising circle and you just collected the Single Best Square Data file you've ever collected. You want to share that with your community! In a server/client world, you spend the afternoon uploading the entire thing to the server. Bittorrent instead works with .torrent files that are small ~KB summaries[12] of a file or files that are used to tell people you have it. .torrent files are uploaded to sites called trackers, along with some metadata and a description that lets other people find them.

Now someone else wants to download the file. A typical server has a lot more bandwidth than a home internet connection, so let's say it's capable of sending two pieces (small colored boxes) per some arbitrary time between these images. To download via bittorrent, one first downloads the .torrent file, and then asks the tracker to connect them to anyone else who also has it. You (yes you!) can only send one piece at a time with your measly internet connection :(. The person receiving the file compares the piece you sent to the summary in the .torrent file, and if it matches, keeps it.

Soon another person wants your sweet sweet Square Data. The server can only transmit two pieces at a time, and so it has to split them between the two downloaders. In the torrent swarm, you have the whole file, but now the first downloader has two pieces of the file, and so both of you are able to send data to the new downloader. The Torrents don't have to be transferred in sequential order, and so you send a non-redundant pieces…..

[12] Hashes

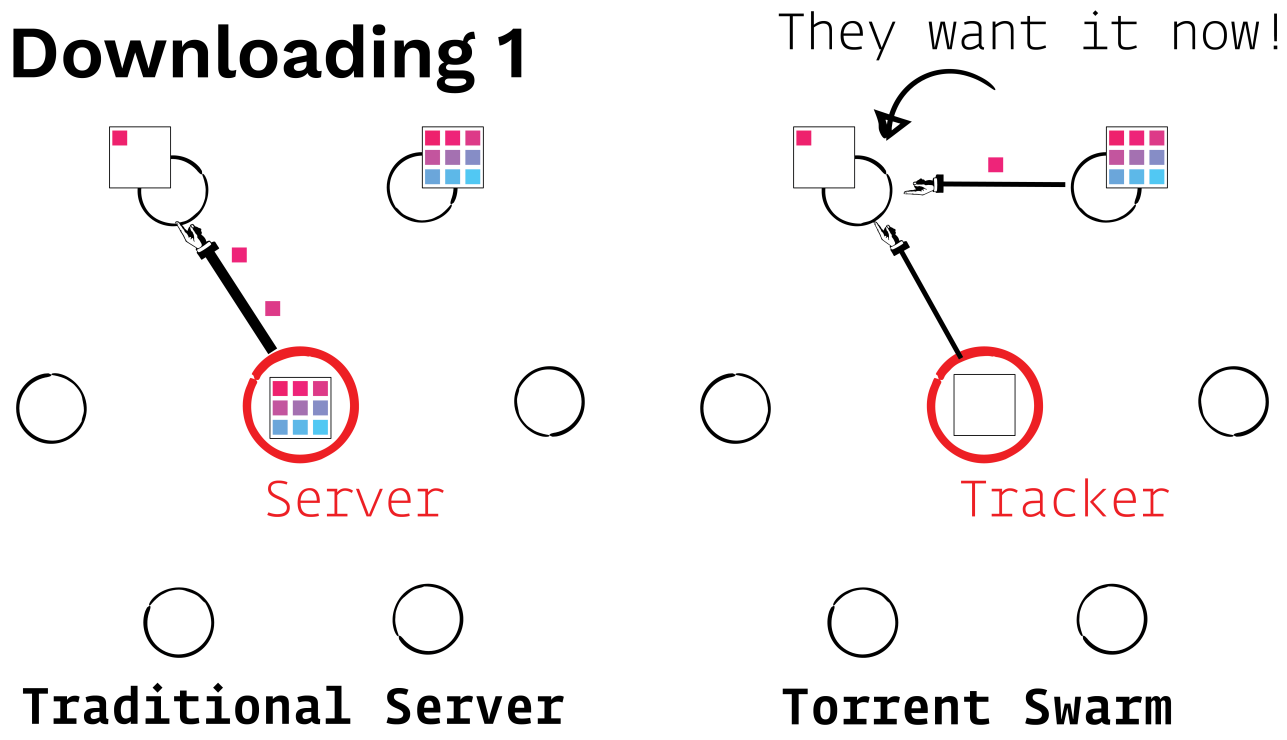
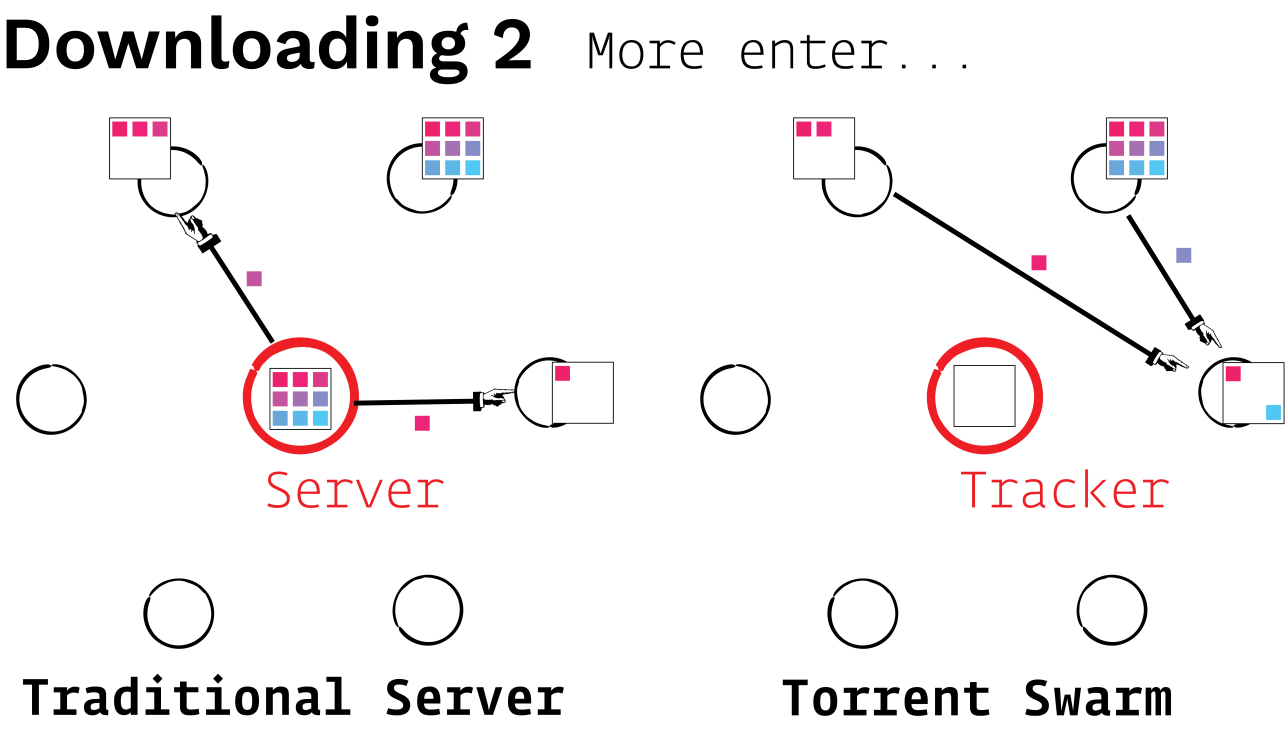
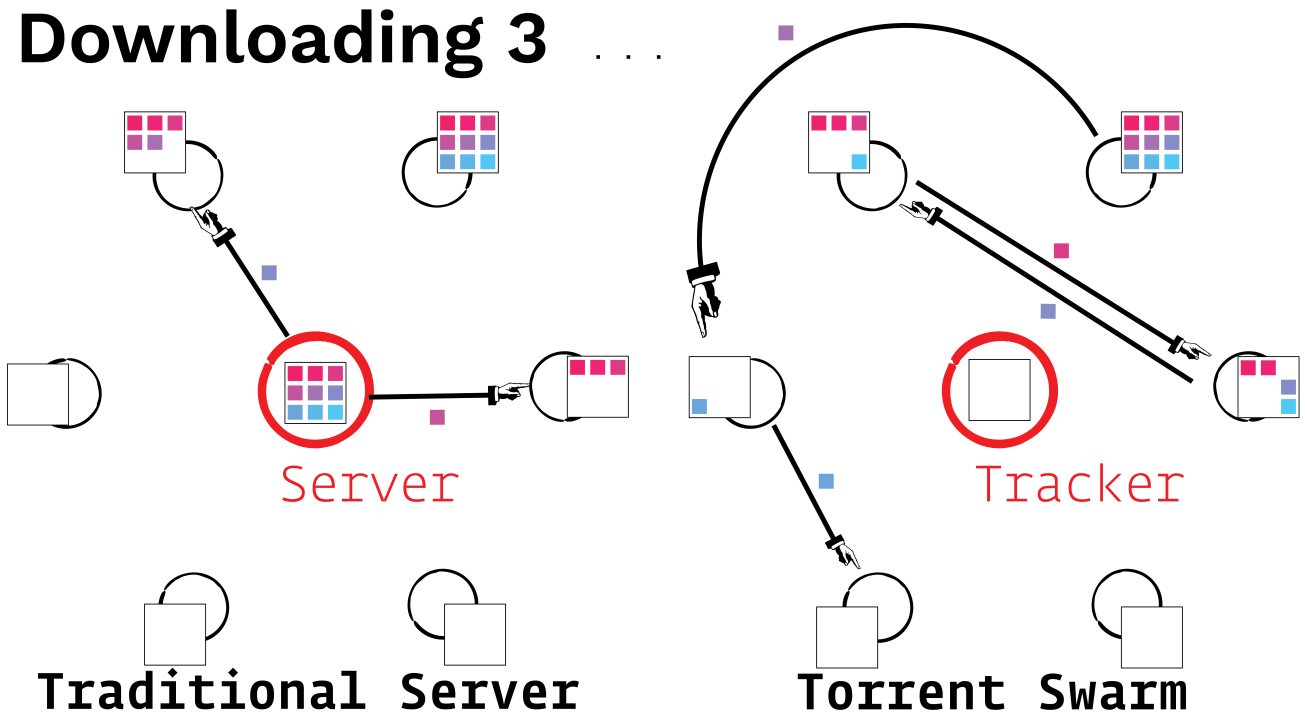
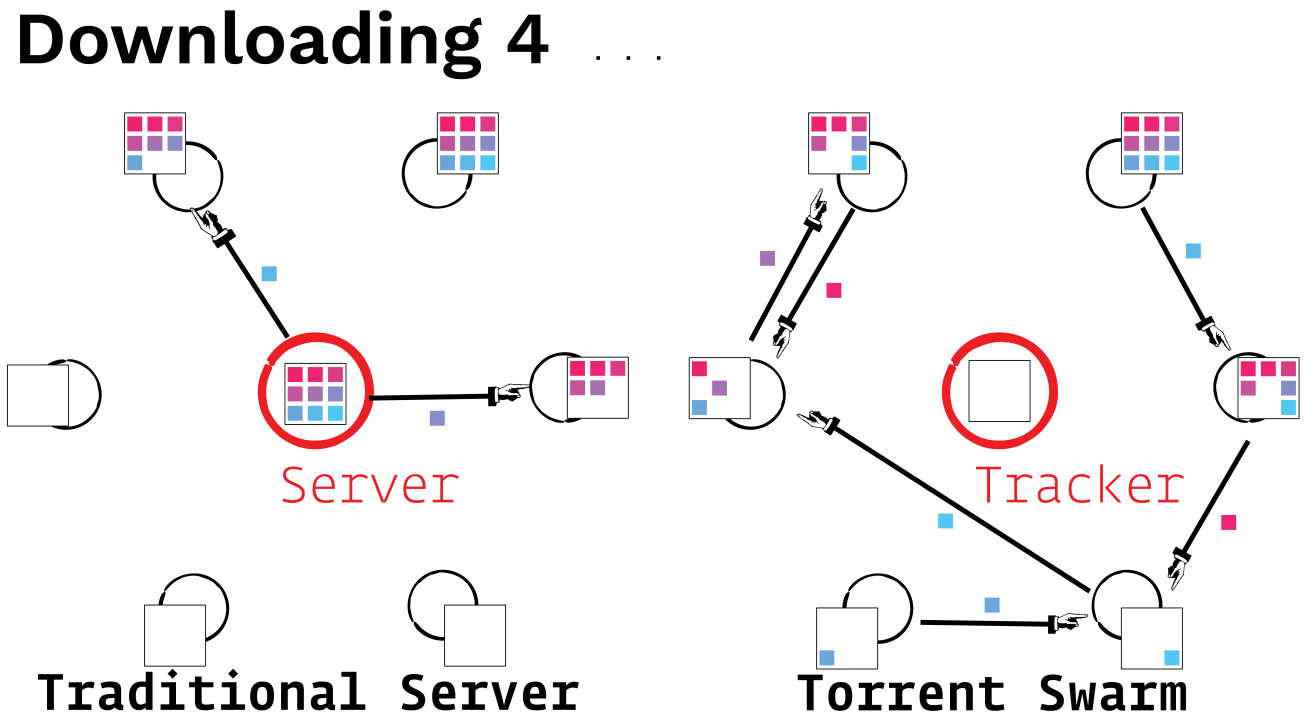


Figure 2: One person downloading a file







finish this later…

The above illustration of an oversimplified peer-to-peer network by itself has the capability of providing a more robust, resilient infrastructure for the massive datasets in neuroscience. Entry costs are low, any existing server infrastructure present in labs,

institutes, etc. can use a peer to peer system. Peer-to-peer networks also theoretically allow the maximum bandwidth of an entire networking system to be used, rather than the maximum bandwidth of a single connection.

Peer to peer systems and server/client are not, in fact, mutually exclusive: peer to peer systems should *always* be *at least* as fast and have *at least* as much storage as the alternative server/client model that would have otherwise been implemented. It is possible for a server to play the role of an "obligate peer[13]" in a network where it always automatically downloads everything that is uploaded, so in that case the benefit of the peer-to-peer system is strictly additive. Since there is nothing special about the obligate peer (let's just call it the the server, it still is) in the swarm, it is possible for arbitrarily many server farms to be combined to expand the redundancy and speed of the system. The obligate peer arrangement prevents the biggest problem of peer-to-peer networks where a file can become unavailable if everyone who has it stops uploading it. In doing so it can also serve as a load balancer in the network, where less-common datasets receive more of the server's bandwidth than common ones.

There are many improvements and variations on peer to peer technology that would make it more suitable for scientists. A scientific peer-to-peer system needs to be capable of version control across iterations of a dataset, to be able to control permissions for datasets, to be able to serve partial datasets (eg. a NWB dataset is a single file, but it should be possible to download the behavior data without downloading the raw 2-photon data), etc. The network can be made more robust by incorporating automatic replication, where users of the network volunteer to share part of their storage space which is then automatically filled with (encrypted) shards of data from the rest of the network (see, for one example among many, Freenet [19],).This scheme ensures that even if the last peer that is explicitly hosting a particular dataset drops out, the dataset will always persist distributed through the network, provided enough shared storage is present.

These scattered suggestions are meant to illustrate the flexibility and variability from the simplest peer-to-peer architecture, and fine-grained details of their implementation and an enumeration of the possible systems are far outside the scope of this paper. I will return to consider the design requirements of a scientific peer-to-peer network after discussing community overlays, the second half of the peer-to-peer story.

!! [20],DANDIis in on the p2p system, as is kachery-p2p

Archives Need Communities An underappreciated element of the torrent system is the effect of the separation between the data transfer protocol and the 'discovery' part of the system — or "overlay" — on the community structure of torrent trackers. Many peer to peer networks like KaZaA or the gnutella-based Limewire had searching for files integrated into the transfer interface. The need for torrent trackers to share .torrent files spawned a massive community of private torrent trackers that for decades have been iterating on cultures of archival, experimenting with different community structures and incentives that encourage people to share and annotate some of the world's largest, most organized libraries. One of these private trackers was the site of one of the largest informational tragedies of the past decade: what.cd[14] What.cd was a bittorrent tracker that was arguably the largest collection of music that has ever existed. At the time of its destruction in 2016, it was host to just over one million unique releases, and approximately 3.5 million torrents[15] [21],.Every torrent was organized in a meticulous system of metadata communally curated by its roughly 200,000 global users. The collection was built by people who cared deeply about music, rather than commercial collections provided by record labels notorious for ceasing distribution of recordings that are not commercially viable — or just losing them in a fire [22],[16].Users would spend large amounts of money to find and digitize extremely rare recordings, many of which were unavailable anywhere else and are now unavailable anywhere, period. One former user describes one example:

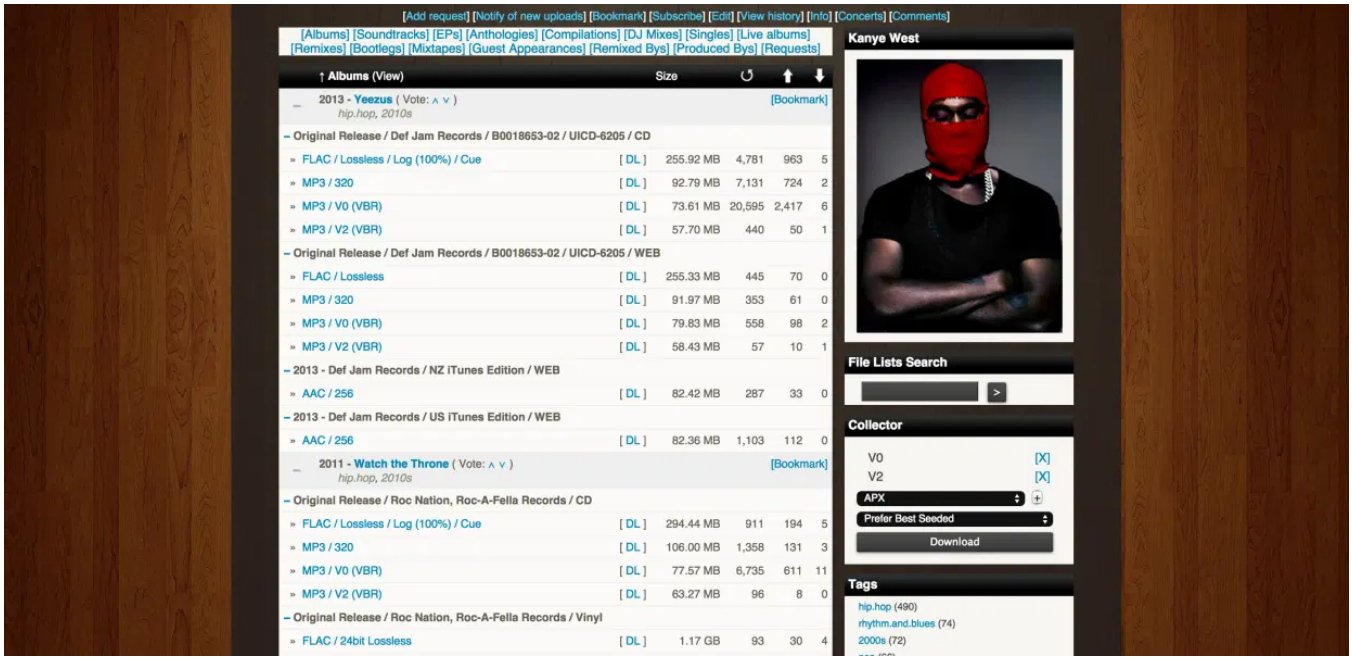
> "I did sound design for a show about Ceaușescu's Romania, and was able to pull together all of this 70s dissident prog-rock and stuff that has never been released on CD, let alone outside of Romania" [23],

[13] or, in the parlance of bittorrent, a web seed

[14] for a detailed description of the site and community, see Ian Dunham's dissertation [21],

[15] Though spotify now boasts its library having 50 million tracks, back of the envelope calculations relating number of releases to number of tracks are fraught, given the long tail of track numbers on albums like classical music anthologies with several hundred tracks on a single "release."

[16]



The what.cd artist page for Kanye West (taken from here in the style of pirates, without permission). For the album "Yeezus," there are ten torrents, grouped by each time the album was released on CD and Web, and in multiple different qualities and formats (.flac, .mp3). Along the top is a list of the macro-level groups, where what is in view is the "albums" section, there are also sections for bootleg recordings, remixes, live albums, etc.

What makes hundreds of thousands of people spend massive amounts of time for literally zero (or negative) compensation to curate a collection of metadata? Though I am not so naive as to think it is the sole cause, I argue that it is the community structure of the what.cd "overlay." Though much of the community structure I describe here would need to be adapted to the needs of a scientific archive, they are an important illustration of a system that aligns the incentives of its users and provides the tools for them to perform the distributed work of curation.

What.cd was a "private" bittorrent tracker, where unlike public trackers that anyone can access, membership was strictly limited

"Among the incinerated Decca masters were recordings by titanic figures in American music: Louis Armstrong, Duke Ellington, Al Jolson, Bing Crosby, Ella Fitzgerald, Judy Garland. The tape masters for Billie Holiday's Decca catalog were most likely lost in total. The Decca masters also included recordings by such greats as Louis Jordan and His Tympany Five and Patsy Cline.

The fire most likely claimed most of Chuck Berry's Chess masters and multitrack masters, a body of work that constitutes Berry's greatest recordings. The destroyed Chess masters encompassed nearly everything else recorded for the label and its subsidiaries, including most of the Chess output of Muddy Waters, Howlin' Wolf, Willie Dixon, Bo Diddley, Etta James, John Lee Hooker, Buddy Guy and Little Walter. Also very likely lost were master tapes of the first commercially released material by Aretha Franklin, recorded when she was a young teenager performing in the church services of her father, the Rev. C.L. Franklin, who made dozens of albums for Chess and its sublabels.

Virtually all of Buddy Holly's masters were lost in the fire. Most of John Coltrane's Impulse masters were lost, as were masters for treasured Impulse releases by Ellington, Count Basie, Coleman Hawkins, Dizzy Gillespie, Max Roach, Art Blakey, Sonny Rollins, Charles Mingus, Ornette Coleman, Alice Coltrane, Sun Ra, Albert Ayler, Pharoah Sanders and other jazz greats. Also apparently destroyed were the masters for dozens of canonical hit singles, including Bill Haley and His Comets' "Rock Around the Clock," Jackie Brenston and His Delta Cats' "Rocket 88," Bo Diddley's "Bo Diddley/I'm A Man," Etta James's "At Last," the Kingsmen's "Louie Louie" and the

The list of destroyed single and album masters takes in titles by dozens of legendary artists, a genre-spanning who's who of 20th- and 21st-century popular music. It includes recordings by Benny Goodman, Cab Calloway, the Andrews Sisters, the Ink Spots, the Mills Brothers, Lionel Hampton, Ray Charles, Sister Rosetta Tharpe, Clara Ward, Sammy Davis Jr., Les Paul, Fats Domino, Big Mama Thornton, Burl Ives, the Weavers, Kitty Wells, Ernest Tubb, Lefty Frizzell, Loretta Lynn, George Jones, Merle Haggard, Bobby (Blue) Bland, B.B. King, Ike Turner, the Four Tops, Quincy Jones, Burt Bacharach, Joan Baez, Neil Diamond, Sonny and Cher, the Mamas and the Papas, Joni Mitchell, Captain Beefheart, Cat Stevens, the Carpenters, Gladys Knight and the Pips, Al Green, the Flying Burrito Brothers, Elton John, Lynyrd Skynyrd, Eric Clapton, Jimmy Buffett, the Eagles, Don Henley, Aerosmith, Steely Dan, Iggy Pop, Rufus and Chaka Khan, Barry White, Patti LaBelle, Yoko Ono, Tom Petty and the Heartbreakers, the Police, Sting, George Strait, Steve Earle, R.E.M., Janet Jackson, Eric B. and Rakim, New Edition, Bobby Brown, Guns N' Roses, Queen Latifah, Mary J. Blige, Sonic Youth, No Doubt, Nine Inch Nails, Snoop Dogg, Nirvana, Soundgarden, Hole, Beck, Sheryl Crow, Tupac Shakur, Eminem, 50 Cent and the Roots.

Then there are masters for largely forgotten artists that were stored in the vault: tens of thousands of gospel, blues, jazz, country, soul, disco, pop, easy listening, classical, comedy and spoken-word records that may now exist only as written entries in discographies." [22],

to those who were personally invited or to those who passed an interview. Invites were extremely rare, and the interview process was demanding to the point where entire guides were written to prepare for them. When I interviewed in 2009, I had to find my way onto an obscure IRC server, wait in a lobby all day until a volunteer moderator could get to me, and was then grilled on the arcana of digital music formats, spectral analysis[17], the ethics of piracy, and so on for half an hour. Getting a question wrong was an instant failure and you were banned from the server for 48 hours. A single user was only allowed one account per lifetime, so between that policy and the extremely high barriers to entries, even anonymous users were strongly incentivized to follow the sophisticated, exacting rules for contributing.

The what.cd incentive system was based on a required ratio of data uploaded vs. data downloaded. Peer to peer systems need to overcome a free-rider problem where users might download a torrent ("leeching") and turn their computer off, rather than leaving their connection open to share it to others (or, "seeding"). In order to download additional music, then, one would have to upload more. Since downloading is highly restricted, and everyone is trying to upload as much as they can, torrents had a large number of "seeders," and even rare recordings would be sustained for years, a pattern common to private trackers [24],. The high seeder/leecher ratio made it so it was extremely difficult to acquire upload credit, so users were additionally incentivized to find and upload new recordings to the system. What.cd implemented a "bounty" system, where users with a large amount of excess upload credit would be able to offer some of it to whoever was able to upload the album they wanted. To "prime the pump" and keep the economy moving, highlight artists in an album of the week, or direct users to preserve rare recordings, moderators would also use a "freeleech" system, where users would be able to download a specified set of torrents without it counting against their download quantity.

The other half of what.cd was its community infrastructure, its forums, comment sections, and moderation systems. The forum was home to roiling debates that lasted years about the structure of some tagging schema, whether one genre was just another with a different name, and so on. The structure of the community was an object of constant, public negotiation, and over time the metadata system evolved to be able to support a library of the entirety of human music output[18], and the rules and incentive structures were made to align with building it. To support the good operation of the site, the forums were also home to a huge amount of technical knowledge, like guides on how to make a perfect upload, that eased new users into being able to use the system.

A critical problem in maintaining coherent databases is correcting metadata errors and departures from schemas. Finding errors was rewarded. Users were able to discuss and ask questions of the uploader in a comment section below each upload, which would allow "polite" resolution of low-level errors like typos. More serious problems could be reported to the moderation team, which caused the upload to be visibly marked as under review, and the report could then be discussed either in the comment sections or the forum. If the moderation team affirmed your report, they would usually kick back a few gigabytes of upload credit depending on the severity. Rather than being a messy hodgepodge of fake, low-quality uploads, what.cd was always teetering just shy of perfection.

These structural considerations do not capture the most elusive but indisputably important features of what.cd's community infrastructure: *the sense of commmunity*. The What.cd forums were the center of many user's relationships to music. Threads about all the finest scales of music nichery could last for years: it was a rare place people who probably cared a little bit too much about music could talk to people with the same condition. What made it more satisfying than other music forums was that no matter what music you were talking about, everyone else in the conversation would always have access to it if they wanted to hear it. Independent musicians released albums in the supportive[19] Vanity House section, and people from around the world came to hold the one true album that only they knew about high aloft like a divine tablet. Beyond any structural incentives, people spent so much time building and maintaining what.cd because it became a source of community and a sink of personal investment. I'll tease a brief recurring dream I've been having recently of something similar existing for scientists: a place where we can discuss our experiments in the same place that they live, being able to link to, embed, and compare data in the kind of longform, thoughtful way that currently has no place outside of papers in scientific culture.

This system created not only a huge, well-annotated library, but its distributed nature made it resilient. When it was shut down,

[17] The average what.cd user was, as a result, on par with many of the auditory neuroscientists I know in their ability to read a spectrogram.

[18] Though music metadata might seem like a trivial problem (just look at the fields in an MP3 header), the number of edge cases are profound. How would you categorize an early Madlib casette mixtape remastered and uploaded to his website where he is mumbling to himself while recording some live show performed by multiple artists, but on the b-side is one of his Beat Konducta collections that mix together studio recordings from a collection of other artists? Who is the artist? How would you even identify the unnamed artists in the live show? Is that a compilation or a bootleg? Is it a cassette rip, a remaster, or a web release?

[19] Mostly. You know how the internet goes…

a series of successors popped up using the open source tools Gazelle and Ocelot that what.cd developers built. Within two weeks, one successor site had recovered and reindexed 200,000 of its torrents resubmitted by former users [25],Manyfeatures of what.cd's structure are undesirable for scientific infrastructure, but they demonstrate that a robust archive is not only a matter of building a database with some frontend, but by building a community [26],. In contrast to what.cd, a scientific peer-to-peer system's incentives need to (among others)… * Have extremely **low barriers** to entry * **Not use downloading as its "cost"** — users downloading and analyzing huge amounts of data is *good* and what we *want*. Other systems of incentivizing uploading and curation have been developed by other trackers. One example is a ratioless system, where users are required to remain seeding data they download, either forever or for a specific amount of time. If a user keeps 100% of their downloads seeded, they have zero ratio requirements, which then scale back up if the user stops sharing. * **Be Resource and Clout-Agnostic** — researchers with access to huge server farms or large professional networks should not be favored by the system, which is intended to *reduce* inequity rather than *reflect* it. We *don't* want to make some leaderboard system, but find ways to incentivize thoughtful, generative archivalism.

Rather than being prescriptive about one community structure, however, what allowed the community structure of private bittorent trackers to develop and experiment with many different types of systems in a shared framework. Our goal should *not* be to make yet another single, subdisciplinary-specific database. We should learn from the meta-structure of the torrent system and take advantage of separating a protocol from its overlay and make a *federated* peer-to-peer system.

!! some forums already exist: https://neurostars.org/

Federated Systems !! compare to datalad

There is no shortage of databases for scientific data, what limits their use is their fragmentation. Each subdiscipline having a separate database makes combining information from across even extremely similar subdisciplines combinatorically complex and laborious. It also makes finding the correct database for a given dataset often a matter of having prior knowledge or wild luck.

Even if one were to have the rare omniscience of a full and masterful understanding of the database landscape, researchers using them are in a bind between domain-generality and specificity. General-purpose databases like figshare[20] are essentially public, versioned, folders with a DOI, but the metadata for organizing multiple datasets together are relatively sparse attributes like keywords, links to the DOI of the paper, authors, etc. Domain-specific databases are more likely to have a metadata structure that fully describes and is compatible with a researcher's particular data, as well as visualization, summarization, and aggregation features purpose-built for that data. The researcher can either spend the extra time uploading to multiple databases, avoid contributing to data fragmentation by using a general-purpose database, or risk obscurity by using a domain-specific one.

Any single database system can only be perfectly-fit to a small slice of the scientific population, so the solution is neither the creation of "the one true perfect database" nor is it creating additional, increasingly specific databases. Matthew J Bietz and Charlotte P Lee articulate this tension better than I can in their ethnography of metagenomics databases:

> "Participants describe the individual sequence database systems as if they were shadows, poor representations of a widely-agreed-upon ideal. We find, however, that by looking across the landscape of databases, a different picture emerges. Instead, each decision about the implementation of a particular database system plants a stake for a community boundary. The databases are not so much imperfect copies of an ideal as they are arguments about what the ideal Database should be. […]
>
> When the microbial ecology project adopted the database system from the traditional genomic "gene finders," they expected the database to be a boundary object. They knew they would have to customize it to some extent, but thought it would be able to "travel across borders and maintain some sort of constant identity". In the end, however, the system was so tailored to a specific set of research questions that the collection of data, the set of tools, and even the social organization of the project had to be significantly changed. New analysis tools were developed and old tools were discarded. Not only was the database ported to a different technology, the data itself was significantly restructured to fit the new tools and approaches. While the database development projects had begun by working together, in the end they were unable to collaborate. The system that was supposed to tie these groups together could not be shielded from the controversies that formed the boundaries between the communities of practice." [3],

[20] No shade to Figshare, which, among others, paved the way for open data and are a massively useful thing to have in society.

Here again neuroscientists could learn from other knowledge communities trying to solve problems with parallel structure, in this case by considering **federated** information systems. Federated systems consist of *distributed*, *heterogeneous*, and *autonomous* agents that implement some minimal agreed-upon standards for mutual communication and (co-)operation. A practical example of a federated system is email: you can choose from a variety of email services, each of which could have a wholly different set of features and design, but you can still send anyone[21] an email. More recent examples are the Matrix messaging protocol and the "Fediverse" built on W3C's ActivityPub protocol [27],forsocial networks. Users in ActivityPub networks, rather than joining a single service as one would with traditional commercial social media networks, join individual servers (or can create their own). Each server chooses its own software that implements the ActivityPub standard, and is free to set its own rules, privileges, and whether or not it wants to be able to send and receive messages from other servers.

For the sake of this paper, I'll focus on federated databases. Federated databases[22] were proposed in the early 1980's [28],andhave been developed and refined in the decades since as an alternative to centralization or non-integration [29, 30, 31] – and their application to the dispersion of scientific data in local filesystems is not new [32],. Amit Sheth and James Larson, in their reference description of federated database systems, describe the *design autonomy* as one critical dimension that characterizes them:

> Design autonomy refers to the ability of a component DBS to choose its own design with respect to any matter, including
>
> (a) The **data** being managed (i.e., the Universe of Discourse),
>
> (b) The **representation** (data model, query language) and the **naming** of the data elements,
>
> (c) The conceptualization or **semantic interpretation** of the data (which greatly contributes to the problem of semantic heterogeneity),
>
> (d) **Constraints** (e.g., semantic integrity constraints and the serializability criteria) used to manage the data,
>
> (e) The **functionality** of the system (i.e., the operations supported by system),
>
> (f) The **association and sharing with other systems**, and
>
> (g) The **implementation** (e.g., record and file structures, concurrency control algorithms).

Susanne Busse and colleagues add an additional dimension of **evolvability**: "Following"natural" tendencies, autonomous components will inevitably develop heterogeneous structures. It is the task of the federation layer to cope with the different types of heterogeneity." [32],.In the case of federated database systems, the federation layer provides a uniform way to mediate differences in schemas and formats between individual databases in the system. To share data between subdisciplines and fields we need to be able to perform some *mapping* between the different data formats and standards that they use: we need some way of translating the neuroscientist's `GENOTYPE` to the geneticists `GENETIC_SEQUENCE`. I will be purposefully vague about the means of implementing these mappings until we reach the shared knowledge section, but first we need a brief practical example of how a system like this might work.

Say I'm a neuroscientist who just collected a dataset that consists of a few electrophysiological recordings from a cluster of Consciousness Cells in some obscure midbrain nucleus, and then sectioned the brain and imaged their positions. I deposit my dataset on my local in-lab server, which I have set up to federate with the fancy new Neurophysiologist's Extravagant, Undying, Repository of Open data (NEUROd). All servers in this federation are required to have their data in the standardized NWB format, and since mine already is (go me!) my server announces to the others that we have some new data available! Some enterprising group of neuroscientific programmers has built a website that allows its users to search, browse, and do all the fancy visualization of data they would expect from a modern database, so I go and see how my new dataset has changed some standard aggregated analysis of all the Concscious Cells from all the other labs participating in the federation. Hang on, I say, a question mark appearing over my head like a cartoon caricature of a curious scientist – I wonder if these Consciousness Cells are in the same place in the evolutionary neighbors of my model organism!? I then run a query for all datasets that have positional data for Consciousness Cells. NEUROd has chosen to federate with the Evolutionary Volitional data sharing Operation (EVO), a federation of evolutionary biologists, some of whom study the origins of Consciousness Cells. They have their data in their own evolutionary biologist-specific format, but since there is some mapping between fields in the NWB standard and theirs, that's

[21] dont @ me about html vs plain text messages, providers with varying degrees of message authentication that get bounced by others, ya know what i mean.

[22] though there are subtleties to the terminology, with related terms like "multidatabase," "data integration," and "data lake" composing subtle shades of a shared idea. I will use federated databases as a single term that encompasses these multiple ideas here, for the sake of constraining the scope of the paper.

no problem. My search then returns data from not only all the other neuroscientists in NEUROd, then, but also matching data from EVO — and my cross-disciplinary question then becomes trivial to answer.

(figure of federated databases here).

The federated database system extends the peer to peer systems discussed earlier and provides a direct means of solving the problems of database fragmentation by subdiscipline. Since the requirements for being a 'node' in the federation are minimal, individual, local servers work seamlessly with institutional servers and large, national servers to take advantage of all available storage and bandwidth of the participating servers — a promising way to solve the problems posed by the "big data" of contemporary science (eg. one articulation by [33],).While mappings between schemas are not magical and require work, they provide a single point of mediation between the data formats of different disciplines. Federation gets us the best of both worlds: the flexibility and domain-specific tools of subdisciplinary databases with the availability of domain-general databases. The radical autonomy of federated systems dramatically lowers the barriers to standardization: rather than requiring everyone to do *the same thing in the same way* and fundamentally change how they do things, researchers need to just build the bridges to connect their existing systems to the federated standard. These bridges can be created gradually. Since nodes in a federated system are free to choose whether they connect to others, there do not need to be mappings between *all* types of data in a federation, and there is no need for creating the oft-fabled *"one true standard"* for all data. Researchers that are interested in interfacing their data with others are strongly incentivized to write the mappings that permit it, and so they can emerge as they are demanded. Researchers are also given far more control over their own data than is afforded by traditional databases: it is entirely possible to have fine-grained permissions controls that allow researchers to share only the data they want to with the rest of the system while still taking advantage of, for example, locally federated servers that make their data available to other collaborating labs.

It's difficult to overstate how fundamentally a widely-adopted federated database system would be for all domains of science: when designing a behavioral experiment to study the circadian cycle, rather than relying on rules of thumbs or a handful of papers, one could directly query data about the sleep-wake cycles of animals recorded by field biologists in their natural habitats, cross reference that with geophysical measurements of daylight times and temperatures in those locations, and normalize the intensity of light you plan to give your animals by estimating tree-canopy coverage from LIDAR data from the geographers. One could make extraordinarily biophysically realistic models of neural networks by incorporating biophysical data about the properties of ion channels and cell membranes, tractography data from human DTI fMRI images, and then compare some dynamical measurement of your network against other dynamic systems models like power grids, telecommunications networks, swarming ants, and so on. Seemingly-intractable problems like the "file drawer" problem simply dissolve: null results are self-evident and don't *need* publication when researchers asking a question are able to see it themselves by analyzing all previous data gathered. Without exaggeration, they present the possibility of making *all* experiments multidisciplinary, making use of our collected human knowledge without disciplinary barriers. Indeed nearly all scientific literature is already available on a federated database system to anyone with an internet connection — arguably the largest expansion of scientific knowledge accessibility ever.

The fundamental tradeoff between centralized and decentralized database systems is that of flexibility vs. coherence: centralized systems can simply enforce a single standard for data and assume that everything it works with will have it. Federated systems require some means of maintaining the mappings between schemas that allow their fluid translation. They also require some means of representing and negotiating data that is unanticipated by existing schemas. The fine details of implementing a federated database system are outside the scope of this paper, but we will return to a means of distributed maintenance of mappings between schemas by taking advantage of semantic web technologies in shared knowledge. Before we do though, we need to discuss the shared tools to analyze and generate the data for the system in this section.

!! make sure to talk about datalad and DANDI!! https://www.datalad.org/

2.2 Shared Tools

If we're building infrastructure to allow us to build on each other's labor by sharing data, why not do the same for the tools that analyze and collect the data while we're at it? The benefits of distributed infrastructure that allow us to preserve our collected labor and knowledge compound when applied in multiple domains. The benefits of shared data, analytical, and experimental infrastructure are far more than the sum of their parts. Each is useful on its own, but as additional components of the system are developed they make the incentive to develop the rest even stronger <- this para is dogshit. rewrite with a clear head.

This section will be relatively short as I feel like a shared analytical framework is relatively uncontroversial, just a matter of putting

labor in the right place. I also don't want to give the impression of self-promotion, as I have spent the last several years designing an experimental framework, autopilot. I will discuss it because, unsurprisingly, I designed it based on the same thoughts that have since developed into this paper, but I want to be clear that as with the rest of the paper, my focus is on the *kind* of tools we need rather than promoting one specific tool.

2.2.1 Analytical Framework

The first natural companion of shared data infrastructure is a shared analytical framework. A major driver for the need for everyone to write their own analysis code largely from scratch is that it needs to account for the idiosyncratic structure of everyone's data. Most scientists are (blessedly) not trained programmers, so code for loading and negotiating loading data is often intertwined with the code used to analyze it, so it is often difficult to adapt another lab's analysis code for use in other contexts. If instead neuroscientists had all their data in a standardized format, then it would be possible to write an analysis method once and allow the rest of the community to benefit from it.

A shared analytical framework should be

- *modular* - Rather than implementing an entire analysis pipeline as a monolith, the system should be broken into minimal, composable modules. The threshold of what constitutes "minimal" is of course to some degree a matter of taste, but the overriding design principle should be to minimize the amount of duplicated labor. Rather than implementing a "peri-stimulus time-histogram" module, we should implement a "binning" module for counting spikes, connect it to an "alignment" module that splits the recording into chunks aligned at the stimulus onset, and so on. Higher-order analysis methods are relatively trivially composed from component parts, but extracting component parts from a frankenstein do-everything script is not. I expect this point to be relatively uncontroversial as it is a general principle of program design.
- *deployable* - For wide use, the framework needs to be easy to install and deploy locally and on computing clusters. The primary obstacle is dependency management, or making sure that the computer has everything needed to run the program. Anecdotally, more than the complexity of using the package itself, the primary barrier for nonprogrammer scientists using a particular software package is managing to get it installed. Luckily containerization and package management is a widespread and increasingly streamlined practice, so I expect this too to be uncontroversial.
- *pluggable* - The framework needs to provide a clear way of incorporating external analysis packages, handling their dependencies, and exposing their parameters to the user.
- *reproducible* - The framework should separate the *parameterization* of a pipeline, the specific options set by the user, and its *implementation*, the code that constitutes it. Implicit in a modularly constructed analysis framework is the notion of a "pipeline," or a specification of a tree (or, specifically, a DAG) of successive stages that process, merge, or split the data from the previous stage. The parameterization of a pipeline should be portable such that it, for example, can be published in the supplementary materials of a paper and reproduced exactly by anyone using the system.

Thankfully, DataJoint already does most of this, and is expanding its modularity with its recent Elements project. Though it currently uses a MySQL, relational database as its backend, extending it to incorporate with the peer to peer database system described above would be an early, concrete development goal for this program. I have heard rumors they are considering adopting a decentralized traditional relational database like CockroachDB, which is not the same thing as a p2p federated semantic database system as I describe here, but is certainly a step in that direction. The rest is in the minutiae of normal software development, as well as building a user interface and collaboration platform for curation and management of shared pipelines. Thank you DataJoint team for making this section so simple.

The combined benefits of a unified data sharing and analytical system have a far greater reach than just saving redundant development time:

Papers published with a concise, inspectable description of their analytical pipeline sidestep the vagueries of methods section prose and allow widescale independent replication of published analyses. A system of documenting and discussing the countless hyperparameters and preprocessing tricks, often as much art as science, could operate as a means of implementing the countless papers describing best practices in analysis. If made easily expandable, so that the developers had a clear way to integrate their tools, access to the state of the art in analysis would be radically democratized, rather than limited to those with finely-tuned twitter feeds and patience to wade through seas of errors and stackexchange posts to get them to work.

A common admonishment in cryptographically-adjacent communities is to "never roll your own crypto," because your home-brew crypto library will never be more secure than reference implementations that have an entire profession of people trying to expose and patch their weaknesses. Bugs in analysis code that produce inaccurate results are inevitable and rampant [34, 35, 36, 37], but impossible to diagnose when every paper writes its own pipeline. A common analysis framework would be a single point of inspection for bugs, and facilitate re-analysis and re-evaluation of affected results after a patch.

Perhaps more idealistic is the possibility of a new kind of scientific consensus. Scientific consensus is subtle and elusive, but to a very crude approximation two of the most common means of its expression are review papers and meta-analyses. Review papers make a prose argument for a consensus interpretation of a body of literature. Meta analyses do the same with secondary analyses, most often on the statistics reported in papers rather than the raw data itself. Both are vulnerable to sampling problems, where the author of a review may selectively cite papers to make an argument, and meta-analyses might be unable to recover all the relevant work from incomplete search and data availability. Instead if one could index across all data relevant to a particular question, and aggregate the different pipelines used to analyze it, it would be possible to make statements of scientific consensus rooted in a full provenance chain back to the raw data.

More fundamentally, a shared data and analysis framework would change the nature of secondary analysis. Increasing rates of data publication and the creation of large public datasets like those of the Allen Observatory make it possible for metascientists and theoreticians to re-analyze existing data with new methods and tools. There is now such a need for secondary analysis that the NIH, among other organizations, is providing specific funding opportunities to encourage it. Secondary analyses are still (unfortunately) treated as second-class research, and are limited to analyzing one or a small number of datasets due to the labor involved and the diversity of analytical strategies that makes a common point of comparison different. If, say some theoretician were to develop some new analytical technique that replaced some traditional step in a shared processing pipeline, in our beautiful world of infrastructure it would be possible to not only aggregate across existng analyses, as above, but apply their new method across an entire category of research.

In effect, analytical infrastructure can at least partially "decouple" the data in a paper from its analyis, and thus the interpretations offered by the primary researchers. For a given paper, if it was possible to see its results as analyzed by all the different processing pipelines that have been applied to it, then a set of observations remains a living object rather than a fixed, historical object frozen in carbonite at the time of publication. In addition to statements of consensus that can programmatically aggregate *existing* results as described by the primary researchers, it also becomes possible to make *fluid* statements of consensus, such that a body of data when analyzed with some new analysis pipeline can yield an entirely *new* set of outcomes unanticipated by the original authors. I think many scientists would agree that this is how an ideal scientific process would work, and this is one way of dramatically lowering the structural barriers that make it deviate from that ideal.

I'll give one more tantalizing possibility here: at the point when we have a peer-to-peer federated system of data-sharing servers integrated with some easily deployable analysis pipelining framework, then we also get a distributed computing grid akin to Folding@Home where users donate some of the computing power of their servers to analyze pieces of some large analysis job with very little additional development.

2.2.2 Experimental Framework

On the other side of data from its analysis are the tools used for its collection. A unifying experimental framework is seemingly a different kind and scale of complexity compared to a unifying data framework. *Everyone needs completely different things!* I have previously written about the design of a generalizable, distributed behavior framework in section 2, and about one modular implementation in section 3 of [7],,and so I will first abbreviate and extend the discussion found there and then consider the role of an experimental framework in broader scientific infrastructure. I designed Autopilot with many of the same fundamental motivations as I articulate here, so being dredged from the same well it should be far from surprising that I see it as a natural example. My intention is not as a self-serving advertisement for *everyone to use my software,* but to use it as an *example* of the *kind* of tool that I think would fit a particular role in a broader set of scientific infrastructure (!! redundant, pick a framing).

I first want to clarify what i'm talking about as an 'experimental framework' – not talking about projects **that we love** like open ephys/etc that develop specific hardware. Those are strictly complementary (and should be given more resources!) I'm talking about something to unify them, to combine the excellent pieces that implement differnt parts of experiments into a unified system.

The most basic requirement of a piece of shared experimental infrastructure is that it must be capable of expressing and being adapted to **perform any experiment.** The "any" there is a hard-ish "any," the reason for which should become clearer soon. At an extremely abstract level, this means that the framework needs to be able to **control potentially high numbers of independent hardware components,** record measurements from them, and coordinate them together in some logical system that constitutes a "task" (or more broadly an "experiment"). In order to be widely adoptable, it needs to be able to **integrate with the instrumentation that researchers already use** rather than requiring researchers to reoutfit their entire rigs. That means, in turn, that it needs to provide a clear means for users to **extend its functionality** and contribute their extensions to the framework. At the same time as providing a clear entrypoint for researcher-developers to interact with the code, it needs to provide a **simple user inferface** so that regular use doesn't require extensive programming knowledge. In other words, if it ain't usable by everyone, it ain't infrastructure, and the same can be said for expense: it must be **inexpensive to implement.** Finally, it needs to be purpose-built for **reproducibility and replication** by preserving a full chain of **provenance** across the wandering path of parameter tuning and experimental design in a clear, **standardized data format** and providing a means of **replicating experiments** even in rigs that are only an approximate match to the original.

Autopilot attempts to achieve these lofty goals by embracing a distributed, modular architecture. Autopilot is built as a system of modules that each represent fundamental parts of experiments in general: hardware control, stimulus generation, data management, and so on. Everything is networked, so everything can talk to anything, even and especially across computers: in practice this means that it is capable of coordinating arbitrary numbers of experimental hardware components by just *using more computers.* It is built around the Raspberry Pi, a low-cost single-board computer with an enormous support community and library of off-the-shelf components, but can be used on any computer. Autopilot imposes few limitations on the structure of tasks and experiments, but also gives users a clear means of defining the parameters they require, the data that will be produced, how to plot it, and so on, such that any task has a portable, publishable representation that is not dependent on the local hardware used to implement it. Its modular hierarchy already provides structure that makes it easy for researchers to modify existing components to suit their needs, and some of its co-developers and I are currently implementing a generalized plugin system that will allow users to replace any component of the system in such a way that their work can be made available and referenceable by any other user of the system. Information about the state of the system, the plugins used, the history of tasks and parameters that an experimental subject experiences, are all obsessively documented, and the data it produces is clean at the time of acquisition. Portable task descriptions, referenceable plugins, and exact documentation of provenance make Autopilot capable of facilitating replication while still supporting extreme heterogeneity in its use. In sum, we designed Autopilot to be flexible, efficient, and reproducible enough for use as general experimental infrastructure.

When compared, the preceding reads as a rephrasing of the design principles articulated in (!! link to section). Autopilot is of course far from a finished project, and many of its design goals remain aspirational due to the small number of contributors[23]. I would be remiss in failing to mention Bonsai, which I love and have learned a lot from. I view Bonsai as a somewhat complementary project and would one day love to merge efforts. The primary differences between Bonsai and Autopilot, besides the massive and obvious difference in number of users and maturity of the library, are a) Autopilot is written in Python, a high-level programming language, and "glues" together fast, low-level library, where Bonsai is written in C#, which is also quite fast but is comparatively less accessible to a broad number of users. Relatedly, Autopilot's documentation describes how the library works down to the lowest levels while Bonsai's is more focused on the user level. b) Autopilot emphasizes communication between objects and their use in a distributed architecture, while Bonsai provides an excellent means of chaining objects together on a single system. c) Autopilot makes comparatively more nudges, and provides a few more features for making reproducible tasks and standardizing data. Again my intention is not a self-serving advocacy for my software, but to say that Bonsai is another extremely capable and widely-used system, and we need systems *like* them capable of serving the role in broader infrastructure that I will turn to now.

In addition to the benefits of reduced duplication of labor and greater access to the state of the art that runs through this whole argument, a standardized experimental framework multiplies the benefits of the data and analytical systems described previously.

When we talk about standardizing data, we talk in the parlance of "conversion," but conversion is only necessary because reserachers collect data in local, idiosyncratic formats. The reason researchers rely on idiosyncratic formats is that it is far from straightforward to directly collect data from their heterogeneous tools in a standardized format. The need for data conversion leaves an airgap between the ideal of universal data access and its labor-intensive practical reality: only those that are most ideologically committed and have enough resources to convert & share their data will do so. We could (and should) lessen the chore

[23] I am the first to admit Autopilot's shortcomings, which I document extensively in its development roadmap and github issues.

of data conversion with continued development of intuitive conversion tools, but an experimental framework that collected data that was *clean at the time of acquisision* then we could shortcircuit the need for conversion altogether. It would also completely dissolve the need for researchers to interact with the peer-to-peer sharing system described previously by automatically dumping standardized data directly into it. In short, an experimental framework could make all the steps between collecting and sharing data completely seamless, and by doing so make the dream of universal data availability possible.

Neuroscience has made substantial progress standardizing an ontology of common terms for cells, chemicals, etc. (see the Neuroscience Information Framework's Ontology) but an ontology for the many minute parameters that define a behavioral experiment's operation has proven elusive. Creating a standardized language for expressing and communicating behavioral experiments is the object of one of the Neurodata Without Borders working groups, in collaboration with the BEADL project, and they've done admirable work there. They have an in-progress terminology for certain parameters like `Reward`, `Guess`, etc., as part of a state-machine (!!define in margin) based representation of a task. The model of standardization would then be to define some extensible terminology, and then either build some software that implements the state machine descriptions of tasks or else ask existing software developers to incorporate them in their systems.

This path to standardization has many attractive qualities, like the formal verification possible with state machines, but may have trouble reaching universal adoption: at even modest complexities, experiments that are simple to explain in prose can be awkward and complicated to express as state machines (eg. section 3.1 in [7],,though the proposed statecharts model is a bit friendlier than traditional state machines). If it is difficult to express a particular feature of some experiment in some formalism, and easier to implement it as some external software, unintegrated with the behavioral framework, then much of the appeal of standardization is lost.

---

bigg redundancy from here…

Uniform standardization is desirable in the circumstances where it is possible, but the scale of variability in the parameters and designs of behavioral neuroscience experiments is truly on a different scale than the already-perplexing case of measurement data standardization. For example, a standard experiment in our lab implemented in Autopilot can be fully described by the parameters that define the experimental protocol itself, and those that parameterize the raspberry pi and the experimental hardware (here they are). The training protocol consists of 7 shaping stages that gradually introduce a mouse to a fairly typical auditory categorization task, each of which includes the parameters for at most 12 different stimuli per stage, probabilities for presenting lasers, bias correction, reinforcement, criteria for advancing to the next stage, etc. The rest of the parameterization includes details for configuring, calibrating and operating the rest of the system – and this is the minimal set of parameters for replicating this experiment that excludes all the defaults, implicit behavior, and well, the rest of the system. For this one relatively straightforward experiment, in one lab, in one subdiscipline, there are 268 different parameters. It's not really about the *number* of parameters per se, but their unpredictability: one needs to parameterize every electrode on a neuropixel probe, but they are shared across a comparatively small number of things of their kind.

Asking people to change the entire way they think about, describe, down to the very mental model that they use to think about it is actually a huge ask. Even if some reasonable standardized lexicon was proposed, it will face the same difficulties as, well, normal lexicons: there is no neutral 'name' for anything, and any word is dependent on the way we conceptualize its use and meaning. This isn't woo-woo unknowability shit: one person's sensory response latency is another person's time of delayed gratification suppression. Even assuming that, getting everyone to start re-expressing all their experiments in a probably very different way than they have been thinking about them for 20-30 years with all the entrenched hardware decisions made over that time is just rounding the bend ready to beat u up for ur lunch money.

Another, complementary way of approaching this problem is to focus on giving people a way to express themselves in a 'safe' environment, focus on the way they *use* them rather than try to define all of them a-priori. sorta like lameguage lmao.

a behavioral framework designed for reproducibility, that preserves a complete history of task parameters as well as the code that uses them, solves both the problems of external inspection and replication without needing to prescribe a specific formalization or uniform ontology. It doesn't matter *what* terms you use if it's trivial to see *how* they're used. Importantly, this strategy punts on the goal of interoperability, but does not forsake it: we will revisit standardized ontologies in the next section. Asking large numbers of people to change the way that they think about their experiments and the words they use to describe them is, ultimately, a pretty big ask. Providing people a tool that allows themselves to express themselves in whatever form is natural to them and

make their terminology meaningful by preserving its context might be easier. (put people in the same system and give them a space to express the terms they use and let them standardize among themselves rather than imposing.)

… to here

Replication is seriously hard. designing a software system that's smart enough about the division between the logical structure of the task and the implementation is seriously hard. the raspi is general purpose enough that was can incorporate pretty general purpose hardware control systems with nontrad components as well, so it balances being an approachable "start from somewhere" (actually in a really good place) with general still byo-hardware. replication needs to basically be incorporated from the ground up, as most behavioral packages that exist tend to rely on local script files that are still labor-intensive to create and are rarely shared, because they're not really intended to be made sharable. « point im' trying to make here is that it can't be an afterthought, the ways that it's easy to go wrong.

But for systems that do link code to a portable task description, where the documentation for each parameter is also good (like wat if that documentation was linked to the semantic wiki… return to in next section), then it is entirely possible to download a system that you point to whatever parts you have around and let er rip. (this doesn't address the technical complexity, but that's also a tease for the next section).

It is already occasionally possible to follow the trail of provenance back to some experimental code, but when all code is developed independently, is any of it reliable [38],?

> A place to put reference implementations of processing algorithms, sensor fusion algorithms. and by splitting them up they become even more useful – give example of autopilot IMU. Also makes them inspectable

At this point we have largely closed the loop of science: starting with standardized data, shared in a scalable p2p system, with some federated interface structure, through modularized analysis parts, published alongside the means to directly reproduce the experiment and re-generate the data… and when we start considering these technologies as an ensemble some things that truly sound like science fiction compared to scientific reality start to become possible. In addition to allowing all of the above features of standardized output data being cross-indexable, what about making the literal fine-grained parameters a way of indexing The All Knowledge Base (go back to previous section and make clearer that only the output data is indexable)? Doing a simultaneous optimization over all of our parameters is basically impossible, and we have all these heuristics for hopping and skipping over it, but what if the behavioral system could query all other times the experiment has been performed, cross reference with published outcome data from the parameterization, and recommend the optimal parameterization for whatever you are studying? The compounding nature of making systems that preserve and respect the diversity of labor to make it coproductive is tectonic: at every stage, from implementation to tweaking, to understanding a science with appropriate infrastructure would move at light speed compared to the way we do it now.

> allows us to 'close the loop' with shared data – what if

> opens up the opportunity for an entirely new kind of shared data, one that's mroe about shared practical knowledge, by being able to automatically collect usage statistics,

2.3 Shared Knowledge

Something that's so in the water of science that it's hard to imagine it being otherwise is the structure of scientific communication. Except for certain domain-specific exceptions, the only means of communicating scientific results is in a journal as a static document or at a conference as an ephemeral talk (though that is changing). The remainder of the gigantic overflowing franzia bag of scientific discourse is funelled ingloriously onto Twitter[24] — and it *sucks*. There simply isn't a place to have longform, thoughtful, durable discussions about science. The direct connection between the lack of a communcaition venue to the lack of a way of storing technical, contextual knowledge is often overlooked. Because we don't have a place to talk about what we

[24] no citation needed, right? if there is some other bastion of scientific discourse i would love to know about it.

do, we don't have a place to write down how to do it. Science needs a communcation platform, but the needs and constraints of a scientific communication platform are different than those satisfied by the major paradigms of chatrooms, forums etc. By considering this platform as another infrastructure project alongside and integrated with those described in the previous sections, its form becomes much clearer, and it could serve as the centerpiece of scientific infrastructure.

I will argue that a semantic wiki should be the major piece of durable information storage, and that it should be supported by a forum system for discussion.

> description of its role as a schema resolution system – currently we implement all these protocols and standards in these siloed, centralized groups that are inherently slow to respond to changes and needs in the field. instead we want to give people the tools so that their the knowledge can be directly preserved and acted on. descrption of its role as a tool of scientific discussion – integrated with the data server and standardized analysis pipelines, it could be possible to have a discussion board where we were able to pose novel scientific questions, answerable with transparent, interrogatable analysis systems. Semantic linking makes the major questions in the field possible to answer, as discussions are linked to one another in a structured way and it is possible to literally trace the flow of thought.

> let's tour through wikipedia for a second and see how it's organized. Look at these community incentive structures and the huge macro-to-micro level organization of the wiki projects. The infinitely mutable nature of a wiki is what makes it powerful, but the SaaS wikis we're familiar with don't capture the same kind of 'build the ground you walk on' energy of the real wiki movement.

!! what's critically different here between other projects is that we are explicitly considering the incentives to join each of these efforts, and by integrating them explicitly, each of them is more appealing. so while there are lots of databases, lots of analysis systems, lots of wikis, and so on, there aren't many that are linked with one another such that participating in one part of the system makes the rest of the system more powerful as well as makes it more useful to the user.

2.3.1 Semantic Wikis - Technical Knowledge Preservation & Schema Negotiation

[39],theword for communally curated schemas is https://en.wikipedia.org/wiki/Folksonomy

[40],wikibasecan do federated SPARQL queries https://wikiba.se/ - and has been used to make folksonomies https://biss.pensoft.net/article/372

> I can see my bank statements on the web, and my photographs, and I can see my appointments in a calendar. But can I see my photos in a calendar to see what I was doing when I took them? Can I see bank statement lines in a calendar? https://www.w3.org/2001/sw/

lots of scientific wikis - https://en.wikipedia.org/wiki/Wikipedia:WikiProject_Molecular_Biology/Genetics/Gene_Wiki/Other_Wikis - https://en.wikipedia.org/wiki/Wikipedia:WikiProject_Molecular_Biology/Genetics/Gene_Wiki

!! bids is doing something like this https://nidm-terms.github.io/

!! interlex

> The Semantic Web is about two things. It is about common formats for integration and combination of data drawn from diverse sources, where on the original Web mainly concentrated on the interchange of documents. It is also about language for recording how the data relates to real world objects. That allows a person, or a machine, to start off in one database, and then move through an unending set of databases which are connected not by wires but by being about the same thing. https://www.w3.org/2001/sw/

Semantic combination of databases in science are also not new [41, 42]. We need both though! semantic federated databases!

Part of what is missing and a place where we could learn from librarians is the notion of governance over a knowledge schema. People have a lot of trouble with NWB because they doubt if it could account for all the idiosyncracies in the types of data that we have to represent. But instead if we have a way of capturing all that thought and insight and practical experience in a governance and decisionmaking structure then we could flexibily work our way to a set of schemas that work for everyone. Part of what

needs to be done is to move from SQL queries to a more expressive abstract system of schema creation that more people can participate in – that's what infrastructure building is, making things that seem impossible or difficult routine. Practically, this can mean an explicit versioning system that not only specifies different versions of a data representation, but for every transition between state there is some notion of making that transition in the data structure. (give example of the subject upgrade system). If that was possible, then the notion of data structure would entirely evaporate, best of both worlds. we get everything and the game is over forever. This is also the distinction between centralized and decentralized systems. we can just make the changes and since they're done against a background of unified intent and expression they can exist simulataneously, commune with one another, while being forwardly productive as their contradictions are resolved.

Consider the examples posed in [43],

> Consider an attribute MEAL-COST of relation RESTAURANT in database DBl that describes the average cost of a meal per person in a restaurant without service charge and tax. Consider an attribute by the same name (MEAL-COST) of relation BOARDING in database DB2 that describes the average cost of a meal per person including service charge and tax. Let both attributes have the same syntactic properties. Attempting to compare attributes DBl.RESTAURANTS.MEALCOST and DBS.BOARDING.MEALCOST is misleading because they are semantically heterogeneous. Here the heterogeneity is due to differences in the definition (i.e., in the meaning) of related attributes [Litwin and Abdellatif 19861.
>
> As a second example, consider an attribute GRADE of relation COURSE in database DBl. Let COURSE.GRADE describe the grade of a student from the set of values {A, B, C, D, FJ. Consider another attribute SCORE of relation CLASS in database DB2. Let SCORE denote a normalized score on the scale of 0 to 10 derived by first dividing the weighted score of all exams on the scale of 0 to 100 in the course and then rounding the result to the nearest half-point. DBl.COURSE.GRADE and DBB.CLASS.SCORE are semantically heterogeneous. Here the heterogeneity is due to different precision of the data values taken by the related attributes. For example, if grade C in DBl.COURSE.GRADE corresponds to a weighted score of all exFederated Database Systems l 187 ams between 61 and 75, it may not be possible to correlate it to a score in DB2.CLASS.SCORE because both 73 and 77 would have been represented by a score of 7.5.

2.3.2 Linked communication platform

We all hate science twitter, why does it exist?

good science community infra - https://www.zooniverse.org

> Two essential features coordinate this information to better serve our organizational decision-making, learning, and memory. The first is our constellation of Working Groups that maintain and distribute local, specialized knowledge to other groups across the network. […] A second, more emergent property is the subgroup of IBL researchers who have become experts, liaisons, and interpreters of knowledge across the network. These members each manage a domain of explicit records (e.g., written protocols) and tacit information (e.g., colloquialisms, decision histories) that are quickly and informally disseminated to address real-time needs and problems. A remarkable nimbleness is afforded by this system of rapid responders deployed across our web of Working Groups. However, this kind of internalized knowledge can be vulnerable to drop-out when people leave the collaboration, and can be complex to archive. An ongoing challenge for our collaboration is how to archive both our explicit and tacit processes held in both people and places. This is not only to document our own history but as part of a roadmap for future science teams, whose dynamics are still not fully understood. [1],

importantly, semantic wiki can be accessed progrtammatically, so you don't need to use the service and can build your own interface to it.

> Relational database systems, manage RDF data, but in a specialized way. In a table, there are many records with the same set of properties. An individual cell (which corresponds to an RDF property) is not often thought of on its own. SQL queries can join tables and extract data from tables, and the result is generally a table. So, the practical use for which RDB software is used typically optimized for soing operations with a small number of tables some of which may have a large number of elements.
>
> RDB systems have datatypes at the atomic (unstructured) level, as RDF and XML will/do. Combination rules tend in RDBs to be loosely enforced, in that a query can join tables by any comlumns which match by datatype – without any check on the

> semantics. You could for example create a list of houses that have the same number as rooms as an employee's shoe size, for every employee, even though the sense of that would be questionable.
>
> The Semantic Web is not designed just as a new data model - it is specifically appropriate to the linking of data of many different models. One of the great things it will allow is to add information relating different databases on the Web, to allow sophisticated operations to be performed across them. https://www.w3.org/DesignIssues/RDFnot.html

in addition to a wiki, we need some conversational engine – talk pages are ok, but they're too fragmented and all hard to keep up to date with. Realtime, chatlike interfaces don't preserve information well, so we should use some intermediate medium like a forum or stack exchange that allows conversations to be tagged and searched and sorted and organized.

Social incentive structure is huge here.

Compared to RDBMS https://www.w3.org/DesignIssues/RDB-RDF.html – rather than individual schemas, groupings of properties, we have 'relationships.' this example is good:

> For example, one person may define a vehicle as having a number of wheels and a weight and a length, but not foresee a color. This will not stop another person making the assertion that a given car is red, using the color vocabular from elsewhere.

Inheritance yall

We're talking about a collaboration medium here… we need a way of organizing open questions in the field and discussing them in a straightfoward way. Why is it that every scientist needs to figure out their own completely gray-area way of discovering papers?

Bad APIs have killed projects with shitloads of funding like NWB and IPFS https://macwright.com/2019/06/08/ipfs-again.html - usability needs to be *the first priority* - you can develop all the fancy shit that you want, if no one can install and unse it in 10 minutes then it's totally useless. This is why the community also has to be collaborative, not just the technology, hends the shared governance idea… ppl note that IPFS has no economic model – that's like true, because there has to be some other incentive system for using it – it makes your work more powerful, it plugs you into a community, etc. https://blog.bluzelle.com/ipfs-is-not-what-you-think-it-is-e0aa8dc69b

2.3.3 Credit Assignment

depth of linking is combinatoric – if you have a paper ecosystem where the numbers are linked to the data, and then the data is annotated, then it's possible to index information across papers not just by textual similarity metrics but on similarity of the structure of experiment and data.

3. Conclusion

3.1 Shared Governance

!! just make this a final note in the conclusion

In addition to like a wiki… need some way of having conversations and arguments about what means what. like some proposal system for linking certain tags together or pointing one to the other…so shared knowledge and shared governance can be a fluid entity.

to avoid the coersion described in [3],wemust make any metadata schema collaborative and mutually beneficial – there is no such thing as 'required' data as long as we design a system that preserves as much information as possible on collection, designing infrastructure is an act of community trust.

Dont want to be prescriptive here, but that we can learn from previous efforts like - https://en.wikipedia.org/wiki/Evergreen_(software) , - IBL, - etc.

3.2 A second, more beautiful dream of what science could be

OK Here's the moment at the end of 2001.

end with the more radical vision — science post papers. Information is semantically organized, so it is possible to ask and answer questions through the medium in which information is represented. Discussion forums exist to describe particular kinds of questions, and a robust discussion of primary scientific data is made possible. Scientists lost their role as arbiters of all reality, but instead are just the comrades closest to the questions, capable of answering open questions in the community, able to design the experiments proposed.

The notion of the filedrawer problem dissappearing, we don't need to publish null results when the data is all always available.

The fractal nature of provenance — where if one can trace an intellectual lineage through its data, one solves credit assignment as centrality within a network.

High school biology classrooms are able to directly interface with the fundament of science, open questions are directly open to students,

4. References

References

[1] Lauren E Wool. "Knowledge across Networks: How to Build a Global Neuroscience Collaboration". In: *Current Opinion in Neurobiology*. Whole-Brain Interactions between Neural Circuits 65 (Dec. 1, 2020), pp. 100–107. ISSN: 0959-4388. DOI: 10.1016/j.conb.2020.10.020. URL: https://www.sciencedirect.com/science/article/pii/S0959438820301653 (visited on 04/21/2021) (cit. on pp. 4, 9, 10, 29).

[2] STEPHEN R. BARLEY and BETH A. BECHKY. "In the Backrooms of Science: The Work of Technicians in Science Labs". In: *Work and Occupations* 21.1 (Feb. 1, 1994), pp. 85–126. ISSN: 0730-8884. DOI: 10.1177/0730888494021001004. URL: https://doi.org/10.1177/0730888494021001004 (visited on 03/15/2021) (cit. on p. 4).

[3] Matthew J. Bietz and Charlotte P. Lee. "Collaboration in Metagenomics: Sequence Databases and the Organization of Scientific Work". In: *ECSCW 2009*. Ed. by Ina Wagner et al. London: Springer, 2009, pp. 243–262. ISBN: 978-1-84882-854-4. DOI: 10.1007/978-1-84882-854-4_15 (cit. on pp. 6, 20, 30).

[4] Zachary F. Mainen, Michael Häusser, and Alexandre Pouget. "A Better Way to Crack the Brain". In: *Nature News* 539.7628 (Nov. 10, 2016), p. 159. DOI: 10.1038/539159a. URL: http://www.nature.com/news/a-better-way-to-crack-the-brain-1.20935 (visited on 03/09/2021) (cit. on pp. 6–9, 11).

[5] Thomas Baker. "Maintaining Dublin Core as a Semantic Web Vocabulary". In: *From Integrated Publication and Information Systems to Information and Knowledge Environments: Essays Dedicated to Erich J. Neuhold on the Occasion of His 65th Birthday*. Ed. by Matthias Hemmje, Claudia Niederée, and Thomas Risse. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2005, pp. 61–68. ISBN: 978-3-540-31842-2. DOI: 10.1007/978-3-540-31842-2_7. URL: https://doi.org/10.1007/978-3-540-31842-2_7 (visited on 03/12/2021) (cit. on p. 7).

[6] TIM BERNERS-LEE, JAMES HENDLER, and ORA LASSILA. "THE SEMANTIC WEB". In: *Scientific American* 284.5 (2001), pp. 34–43. ISSN: 0036-8733. JSTOR: 26059207 (cit. on p. 7).

[7] Jonny L. Saunders and Michael Wehr. "Autopilot: Automating Behavioral Experiments with Lots of Raspberry Pis". In: *bioRxiv* (Oct. 17, 2019), p. 807693. DOI: 10.1101/807693. URL: https://www.biorxiv.org/content/10.1101/807693v1 (visited on 03/12/2021) (cit. on pp. 7, 24, 26).

[8] Larry F. Abbott et al. "An International Laboratory for Systems and Computational Neuroscience". In: *Neuron* 96.6 (Dec. 20, 2017), pp. 1213–1218. ISSN: 0896-6273. DOI: 10.1016/j.neuron.2017.12.013. URL: https://www.sciencedirect.com/science/article/pii/S0896627317311364 (visited on 03/15/2021) (cit. on pp. 7, 9).

[9] Ed S. Lein et al. "Genome-Wide Atlas of Gene Expression in the Adult Mouse Brain". In: *Nature* 445.7124 (7124 Jan. 2007), pp. 168–176. ISSN: 1476-4687. DOI: 10.1038/nature05453. URL: https://www.nature.com/articles/nature05453 (visited on 03/15/2021) (cit. on p. 9).

[10] Sten Grillner et al. "Worldwide Initiatives to Advance Brain Research". In: *Nature Neuroscience* 19.9 (9 Sept. 2016), pp. 1118–1122. ISSN: 1546-1726. DOI: 10.1038/nn.4371. URL: https://www.nature.com/articles/nn.4371 (visited on 03/15/2021) (cit. on p. 9).

[11] Christof Koch and Allan Jones. "Big Science, Team Science, and Open Science for Neuroscience". In: *Neuron* 92.3 (Nov. 2, 2016), pp. 612–616. ISSN: 0896-6273. DOI: 10.1016/j.neuron.2016.10.019. URL: https://www.sciencedirect.com/science/article/pii/S0896627316307206 (visited on 03/15/2021) (cit. on p. 9).

[12] The International Brain Laboratory et al. "Standardized and Reproducible Measurement of Decision-Making in Mice". In: *bioRxiv* (Oct. 9, 2020), p. 2020.01.17.909838. DOI: 10.1101/2020.01.17.909838. URL: https://www.biorxiv.org/content/10.1101/2020.01.17.909838v5 (visited on 03/15/2021) (cit. on p. 9).

[13] The International Brain Laboratory et al. "Data Architecture for a Large-Scale Neuroscience Collaboration". In: *bioRxiv* (Feb. 6, 2020), p. 827873. DOI: 10.1101/827873. URL: https://www.biorxiv.org/content/10.1101/827873v2 (visited on 03/15/2021) (cit. on p. 9).

[14] Gonçalo Lopes et al. "Bonsai: An Event-Based Framework for Processing and Controlling Data Streams". In: *Frontiers in Neuroinformatics* 9 (2015). ISSN: 1662-5196. DOI: 10.3389/fninf.2015.00007. URL: https://www.frontiersin.org/articles/10.3389/fninf.2015.00007/full (visited on 03/15/2021) (cit. on p. 10).

[15] D. Clark. "The Design Philosophy of the DARPA Internet Protocols". In: *Symposium Proceedings on Communications Architectures and Protocols*. SIGCOMM '88. New York, NY, USA: Association for Computing Machinery, Aug. 1, 1988, pp. 106–114. ISBN: 978-0-89791-279-2. DOI: 10.1145/52324.52336. URL: https://doi.org/10.1145/52324.52336 (visited on 03/15/2021) (cit. on p. 11).

[16] Tim Berners-Lee. *Principles of Design*. 1998. URL: https://www.w3.org/DesignIssues/Principles.html#Decentrali (visited on 03/15/2021) (cit. on p. 12).

[17] Brian E. Carpenter. *RFC 1958 - Architectural Principles of the Internet*. June 1996. URL: https://tools.ietf.org/html/rfc1958 (visited on 03/15/2021) (cit. on p. 12).

[18] Xuemin Shen et al. *Handbook of Peer-to-Peer Networking*. Springer Science & Business Media, Mar. 3, 2010. 1421 pp. ISBN: 978-0-387-09751-0. Google Books: nXk_AAAAQBAJ (cit. on p. 13).

[19] Ian Clarke et al. "Freenet: A Distributed Anonymous Information Storage and Retrieval System". In: *Designing Privacy Enhancing Technologies: International Workshop on Design Issues in Anonymity and Unobservability Berkeley, CA, USA, July 25–26, 2000 Proceedings*. Ed. by Hannes Federrath. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2001, pp. 46–66. ISBN: 978-3-540-44702-3. DOI: 10.1007/3-540-44702-4_4. URL: https://doi.org/10.1007/3-540-44702-4_4 (visited on 03/18/2021) (cit. on p. 17).

[20] Morgan G. I. Langille and Jonathan A. Eisen. "BioTorrents: A File Sharing Service for Scientific Data". In: *PLoS ONE* 5.4 (Apr. 14, 2010). ISSN: 1932-6203. DOI: 10.1371/journal.pone.0010071. pmid: 20418944. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2854681/ (visited on 04/01/2021) (cit. on p. 17).

[21] Ian Dunham. "What.CD: A Legacy of Sharing". Rutgers University - School of Graduate Studies, 2018. DOI: 10.7282/T3V128F3. URL: https://rucore.libraries.rutgers.edu/rutgers-lib/58981/ (visited on 03/16/2021) (cit. on p. 17).

[22] Jody Rosen. "The Day the Music Burned". In: *The New York Times. Magazine* (June 11, 2019). ISSN: 0362-4331. URL: https://www.nytimes.com/2019/06/11/magazine/universal-fire-master-recordings.html (visited on 03/18/2021) (cit. on pp. 17, 18).

[23] Nikhil Sonnad. *A Eulogy for What.Cd, the Greatest Music Collection in the History of the World—until It Vanished*. Quartz. Nov. 18, 2016. URL: https://qz.com/840661/what-cd-is-gone-a-eulogy-for-the-greatest-music-collection-in-the-world/ (visited on 03/16/2021) (cit. on p. 17).

[24] Z. Liu et al. "Understanding and Improving Ratio Incentives in Private Communities". In: *2010 IEEE 30th International Conference on Distributed Computing Systems*. 2010 IEEE 30th International Conference on Distributed Computing Systems. June 2010, pp. 610–621. DOI: 10.1109/ICDCS.2010.90 (cit. on p. 19).

[25] Ernesto Van der Sar. *What.Cd Is Dead, But The Torrent Hydra Lives On*. TorrentFreak. Dec. 2, 2016. URL: https://torrentfreak.com/what-cd-is-dead-but-the-torrent-hydra-lives-on-161202/ (visited on 03/18/2021) (cit. on p. 20).

[26] Jordan Bross. "Community, Collaboration and Contribution: Evaluating a BitTorrent Tracker as a Digital Library." M.S. in Library Science. UNC Chapel Hill, Dec. 2013. 40 pp. URL: https://doi.org/10.17615/g1cw-kw06 (cit. on p. 20).

[27] Christopher Webber and Jessica Tallon. *ActivityPub*. W3C recommendation. W3C, Jan. 2018. URL: https://www.w3.org/TR/2018/REC-activitypub-20180123/ (cit. on p. 21).

[28] Dennis Heimbigner and Dennis McLeod. "A Federated Architecture for Information Management". In: *ACM Transactions on Information Systems* 3.3 (July 1, 1985), pp. 253–278. ISSN: 1046-8188. DOI: 10.1145/4229.4233. URL: https://doi.org/10.1145/4229.4233 (visited on 03/25/2021) (cit. on p. 21).

[29] Witold Litwin, Leo Mark, and Nick Roussopoulos. "Interoperability of Multiple Autonomous Databases". In: *ACM Computing Surveys* 22.3 (Sept. 1, 1990), pp. 267–293. ISSN: 0360-0300. DOI: 10.1145/96602.96608. URL: https://doi.org/10.1145/96602.96608 (visited on 03/25/2021) (cit. on p. 21).

[30] Vipul Kashyap and Amit Sheth. "Semantic and Schematic Similarities between Database Objects:A Context-Based Approach". In: *The VLDB Journal* 5.4 (Dec. 1, 1996), pp. 276–304. ISSN: 0949-877X. DOI: 10.1007/s007780050029. URL: https://doi.org/10.1007/s007780050029 (visited on 03/25/2021) (cit. on p. 21).

[31] Richard Hull. "Managing Semantic Heterogeneity in Databases: A Theoretical Prospective". In: *Proceedings of the Sixteenth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*. PODS '97. New York, NY, USA: Association for Computing Machinery, May 1, 1997, pp. 51–61. ISBN: 978-0-89791-910-4. DOI: 10.1145/263661.263668. URL: https://doi.org/10.1145/263661.263668 (visited on 03/24/2021) (cit. on p. 21).

[32] Susanne Busse et al. "Federated Information Systems: Concepts, Terminology and Architectures". In: (1999), p. 40 (cit. on p. 21).

[33] Adam S. Charles et al. "Toward Community-Driven Big Open Brain Science: Open Big Data and Tools for Structure, Function, and Genetics". In: *Annual Review of Neuroscience* 43 (July 8, 2020), pp. 441–464. ISSN: 1545-4126. DOI: 10.1146/annurev-neuro-100119-110036. pmid: 32283996 (cit. on p. 22).

[34] Greg Miller. "A Scientist's Nightmare: Software Problem Leads to Five Retractions". In: *Science* 314.5807 (Dec. 22, 2006), pp. 1856–1857. ISSN: 0036-8075, 1095-9203. DOI: 10.1126/science.314.5807.1856. pmid: 17185570. URL: https://science.sciencemag.org/content/314/5807/1856 (visited on 04/07/2021) (cit. on p. 24).

[35] David A. W. Soergel. "Rampant Software Errors May Undermine Scientific Results". In: *F1000Research* 3 (July 29, 2015). ISSN: 2046-1402. DOI: 10.12688/f1000research.5930.2. pmid: 26539290. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4629271/ (visited on 04/07/2021) (cit. on p. 24).

[36] Anders Eklund, Thomas E. Nichols, and Hans Knutsson. "Cluster Failure: Why fMRI Inferences for Spatial Extent Have Inflated False-Positive Rates". In: *Proceedings of the National Academy of Sciences* 113.28 (July 12, 2016), pp. 7900–7905. ISSN: 0027-8424, 1091-6490. DOI: 10.1073/pnas.1602413113. pmid: 27357684. URL: https://www.pnas.org/content/113/28/7900 (visited on 04/07/2021) (cit. on p. 24).

[37] Jayanti Bhandari Neupane et al. "Characterization of Leptazolines A–D, Polar Oxazolines from the Cyanobacterium Leptolyngbya Sp., Reveals a Glitch with the "Willoughby–Hoye" Scripts for Calculating NMR Chemical Shifts". In: *Organic Letters* 21.20 (Oct. 18, 2019), pp. 8449–8453. ISSN: 1523-7060. DOI: 10.1021/acs.orglett.9b03216. URL: https://doi.org/10.1021/acs.orglett.9b03216 (visited on 04/07/2021) (cit. on p. 24).

[38] Matthew Wall. "Reliability Starts with the Experimental Tools Employed". In: (Nov. 8, 2018). DOI: 10.31234/osf.io/upynr. URL: https://psyarxiv.com/upynr/ (visited on 03/20/2019) (cit. on p. 27).

[39] Maged Kamel Boulos. "Semantic Wikis: A Comprehensible Introduction with Examples from the Health Sciences". In: *Journal of Emerging Technologies in Web Intelligence* 1 (Aug. 1, 2009). DOI: 10.4304/jetwi.1.1.94-96 (cit. on p. 28).

[40] Benjamin M. Good, Joseph T. Tennis, and Mark D. Wilkinson. "Social Tagging in the Life Sciences: Characterizing a New Metadata Resource for Bioinformatics". In: *BMC Bioinformatics* 10.1 (Sept. 25, 2009), p. 313. ISSN: 1471-2105. DOI: 10.1186/1471-2105-10-313. URL: https://doi.org/10.1186/1471-2105-10-313 (visited on 04/08/2021) (cit. on p. 28).

[41] Kei-Hoi Cheung et al. "Semantic Web Approach to Database Integration in the Life Sciences". In: *Semantic Web*. Ed. by Christopher J. O. Baker and Kei-Hoi Cheung. Boston, MA: Springer US, 2007, pp. 11–30. ISBN: 978-0-387-48436-5. DOI: 10.1007/978-0-387-48438-9_2. URL: http://link.springer.com/10.1007/978-0-387-48438-9_2 (visited on 03/30/2021) (cit. on p. 28).

[42] Ana Claudia Sima et al. "Enabling Semantic Queries across Federated Bioinformatics Databases". In: *Database* 2019 (baz106 Jan. 1, 2019). ISSN: 1758-0463. DOI: 10.1093/database/baz106. URL: https://doi.org/10.1093/database/baz106 (visited on 03/30/2021) (cit. on p. 28).

[43] Amit P. Sheth and James A. Larson. "Federated Database Systems for Managing Distributed, Heterogeneous, and Autonomous Databases". In: *ACM Computing Surveys* 22.3 (Sept. 1, 1990), pp. 183–236. ISSN: 0360-0300. DOI: 10.1145/96602.96604. URL: https://doi.org/10.1145/96602.96604 (visited on 03/25/2021) (cit. on p. 29).