free software syndicates

SYNWARE

sinwɛːʀ

SYNWARE

sinwɛːʁ

L'union fait la force.
The syndicate wrecks the forge.
Traduttore traditore.
```
:(){ :|:& };:
```

SYNWARE

free software syndicates

sinwɛːʁ

Table of Contents

Editor's preface

This collection of essays gathers texts that were first published two years ago in French, in *Présence Solidaire*. Their English version never appeared in print. This preface, *Take me to your leader*, as well as the OFFDEM Call for Presence were written for this edition; *Supporting Resistances* was edited as well to reflect petites singularités' current engagements.

The present volume inaugurates the **Synware collection** (sɪnwɛːʁ) on free software syndicalism. This collection explores the modalities of using, creating, and maintaining digital technologies collectively. Synware documents decentralized free software, their usage and community organization as much technical as conceptual, aesthetic and political.

On software syndicalism

by spacekookie

Abstract

Organisations, groups, and projects under capitalism have the tendency to centralise. This is both because of monetary incentives (it might be cheaper to just have one of something than many), as well as authority incentives; it is easier to control an organisation that is structured hierarchically.

The way that we organise free software projects is impacted by this societal framework, which replicates a lot of the issues that organisations, projects, and companies under capitalism face as well. Maybe unsurprisingly our solutions to these issues are also largely similar: personality based, and hierarchical in nature.

Projects often also use the same metrics as capitalist society for success: growth, reach, and audience appeal. This replicates the phenomenon of representative democratic systems and proprietary technology creators of pandering to the majority and letting needs by minorities largely go unanswered.

In this essay we propose an organisational structure for software and technical projects that removes the notion of "upstream", and introduces a collective ownership approach of software and technical knowledge. Freedom of ideas (the fundamental basis of free software) is a core requirement for this approach.

This essay can not hope to solve all problems related to this idea, but to start a discussion about the merits and advantages of organising in small-scale, decentralised communities. Our hope is that this sparks conversation, interest,

and motivation in others to form software syndicates of their own, to communally own, develop, and maintain the technologies that our lives are built upon.

Problem domain

Developing and maintaining software is a lot of work, and largely a social exercise, instead of a technical one. While certain individuals are able to create a project by themselves through obsession and dedication, it is unlikely for projects without a community to outlive the focus period of the original creator.

Upstream

This relationship between creators and consumers is formalised by the concept of "upstream". Software development is considered a river with an original source, and

can branch out into different streams and brooks to adapt to its environment.

While this is an apt metaphor for how software develops from a centralised source, it does come with a lot of burdens and challenges. A poisonous source can destroy a river's ecosystem, and similarly, a rogue upstream development team[1] can doom users dependent on the downstream ecosystem of this project.

Forks will occasionally diverge completely from the original upstream, however this is a commitment that very few are able to maintain without substantial community engagement (and public backlash).

I   https://web.archive.org/web/20210705123342/https://www.techradar.com/uk/news/audacity-fans-are-absolutely-furious-right-now-heres-why
https://www.linuxuprising.com/2018/12/jellyfin-free-software-emby-media.html

Forks

Maintaining a software fork is a lot of work. While it is difficult to get exact statistics, our assumption is that most software forks fail due to lack of community engagement.[2] This social dynamic puts people off forking software projects that are developing in a direction that they do not approve of, or that no longer represent their wishes and desires: while in theory it is still possible for the software or technology to be forked, the reality of the situation needs to be acknowledged that this is out of scope for most people.

---

2   https://glimpse-editor.org/posts/a-project-on-hiatus/
    https://en.wikipedia.org/wiki/List_of_software_forks
    Research exists that indicates the opposite of this
    statement. However survivorship bias may exist in
    terms of how projects are advertised, scoped, and
    identified. More research into the field is certainly
    needed https://sci-hub.st/10.1007/978-3-642-33442-9

Organisation

The organisational methods used by software and technical projects are often focussed around central points of authority, similar to how the code (or design files) itself is treated. This is a limitation by the nature of organisation around a single platform and is derived from how a lot of tools are built to accommodate capitalist ventures where centralisation is a desired effect.

While it is possible for a small group to make decisions very efficiently in private, it also means that not all voices in the community can be considered.

However decentralised and open decision processes have a maximum size, past which they fail due to sheer volume of feedback, trolls, or both. A prominent community that recently ran into this issue is the Rust language

project, which prompted the creation of a working group in 2019 to address these issues.[3]

Principles

This section outlines different modes of collaborating on projects, their strengths, and how they can interact and integrate with each other. These ideas form the basis on which software syndicalism is built.

The project cabal

A common organisational pattern that exists (albeit not usually with an exact name and very often hidden) is the "project cabal". This is a group of people, often including the original author(s), who work on the core features and

---

3   https://blog.rust-lang.org/2019/04/23/
    roadmap.html#governance
    https://boats.gitlab.io/blog/post/rust-2019/
    https://spacekookie.de/blog/rust-2019-how-we-make-
    decisions/

expansion of a project. Their knowledge and engagement drives the bulk of the project forward, and by their hands a lot of requests from both users, as well as peripheral developers get implemented.

While many projects have a cabal, few are open about this dynamic. It is not necessarily a bad dynamic, if discussed and adopted openly. Seeing communities as a collection of concentric circles outwards from the cabal allows users to be aware of the social dynamics that go into making decisions, and the path via which an idea can be adopted by the project.

This uses the concept of "knowledge bridges"[4], which facilitate a way for less experienced users and developers to communicate their ideas to the cabal of a

---

4   *Binding Chaos,* Heather Marsh (9781989783009)

project, without having to become experts in the project domain first.

Distribution and tools

While source control systems such as git are already decentralised, many organisational tools built around it are not. GitHub, Gitlab, and many other projects inspired by them[5] follow the same patterns of an upstream repository, with a central place to track contributions and open issues.

Furthermore, this approach affects the way that software is being distributed to end-users as well.

A new and growing trend is to task the developers themselves with packaging their software[6]. This is done to simplify (centralise) the publication process and reduce the lag

---

5   https://forgejo.org/ https://gitea.com/ https://sr.ht/

6   https://flatpak.org/ https://snapcraft.io/

between new features being created and users being able to use these features.

Embracing the idea of decentralised collaboration in smaller communities opens up new possibilities for ownership of the technology that we use. And while projects that aim to decentralise these collaboration tools[7] around a peer-to-peer protocol such as ActivityPub[8] are not strictly required to put any of these theories into practice, they do offer the opportunity to design new ways of collaboration that don't mirror the existing centralised platforms.

---

7   https://forgefed.org/ https://forgefriends.org/
8   https://activitypub.rocks/

Upstream vs Mainline

Most of the issues we face in building distributed collaboration networks are organisational – not technical – in nature. As the development process of a project scatters around different groups, it becomes important to catalogue and track changes made by different groups that allows others to easily pull them into their own trees.

For this process to work the original source of a project (currently called "upstream") needs to be replaced in the minds of developers and users by the idea of a reference implementation. For this reason we propose and use the term "mainline" to describe this project community.

While it is a subtle difference, language plays a huge role in how people relate to structures and processes. The term and concept is taken

from the way the Linux kernel is developed. Every three months a new "mainline" kernel is released[9] into the world, ready to be used by whoever is interested in it.

However, most people do not run the mainline kernel. This is a reference configuration aimed at pleasing a very specific target audience. Most Linux distributions apply their own patches on top of this version, remove features they deem incompatible with their ideals (proprietary firmware as an example), and re-release this version onto their users. Projects must be aware of who their target audience is and no one project can every hope to appeal to every user in the world.

---

9   https://en.wikipedia.org/wiki/
    Linux_kernel_version_history

Syndicalism

Before we can discuss how to build software syndicates we need to define what a syndicate is, and how syndicalist cooperation functions in practice. One definition of syndicalism is "a radical political movement that advocates bringing industry and government under the control of federations of labour unions by the use of direct action"[10]. The term is also often used in relation to "anarcho-syndicalism"[11] which puts this theory into praxis in different ways.

A lot of political activism is done via syndicalist structures. They offer a way for people to collaborate with each other, without having to belong to the same large-scale

---

10 https://www.wordnik.com/words/syndicalism

11 https://en.wikipedia.org/wiki/Anarcho-
    syndicalism#Theory_and_politics

organisation, or following the same exact plan. Alignment with each other's ideals and principles is foundational for this mode of collaboration to work, while avoiding many of the problems outlined in earlier sections.

Technology is inherently political in how it is created, maintained, and used, and software developers carry their own ideologies into their work, whether they are aware of this or not. Cultural barriers created by these ideologies make it harder for outsiders to the ideology to participate (for example because they have a different political background or are from a different part of the world).

Syndicalism embraces political ideology around the work that we do and asks of everybody participating in this work to reflect on their own biases, assumptions, and behaviours. This does not require political

uniformity (often dubbed "unity"). It attempts to make social collaboration more transparent and easier to understand, and primes[12] developers and users to understanding their own biases and assumptions based on feedback that they get from other communities.

Different syndicates can also approach group collaboration and decision making differently, while still working on the same overall vision for a project or idea.

We use this term to invoke a feeling of belonging, community, and political awareness of the technologies we build and the work we collaborate on. *Software syndicalism* is the act of organising in syndicates and applying it to the development and maintenance of software.

---

12   https://www.thefreedictionary.com/primed

Opportunities

Proximity and knowledge silos

Centralised software communities tend to recreate colonialist power-structures through the distribution of developers and choice of target audience. This creates knowledge silos[13] in these countries which is detrimental to the empowerment and autonomy of both developers and users from different countries. There are more subtle differences (for example looking at northern vs southern, and western vs eastern Europe), but most prominent in both European and white American communities, compared to the rest of the world.

---

13   https://en.wikipedia.org/wiki/Information_silo

One of the opportunities of creating syndicates around the creation and maintenance of software projects is breaking this relationship. To understand how this works we also need to discuss the concept of social proximity[14].

The communities we belong to are based on the social relationships we have with people, and vice versa. These are bi-directional feedback mechanisms. Via the internet proximity (or locality) can exist both in the physical world, and in a metaphysical sense of belonging.

Users and developers of projects can exist in different proximities to different software syndicates, which lowers the barrier of entry, and gives users and developers more choices of contact points to a software project. If the

---

14   https://en.wikipedia.org/wiki/Proximity_principle

mainline syndicate around a project is considered hostile to work with outside of a certain peer group, other syndicates will allow alternative communities to spring up.

It's important to note that none of this is impossible under the current view of development. A hostile or malicious upstream development team can be circumvented by forking the project. This however comes with a lot of unexplored social responsibilities that many people shy away from. Forking, and then maintaining a fork community, is a lot of work that is often not seen as an option.

To start a software syndicate is not necessarily easier on its own, but comes with the idea of inter-project and international solidarity built-in. No one syndicate aims to speak for the whole project, or satisfy all users. And thus collaboration is key.

Identification

Proximity and community are about belonging and identification. This requires self-identification of users and developers, exploring existing communities. Human beings are complex, both individually, and in terms of the relationships with each other. Labels of identification are an important tool in this regard, but must not be used to bikeshed definitions.

Much like anything else that humans have created language for, identity labels are vague and have a certain amount of flexibility. A software syndicate might exists for a user group with specific needs, or for a group of developers (and thus users) based in a different country, operating in a different language.

Syndicates must self regulate their membership, but at the same time identification with the target audience of a syndicate should be enough for someone to belong to this syndicate's user group.

Forming new syndicates based on existing ones if the need for more granular identification becomes apparent this should be encouraged and not hindered. Large communities (as outlined in earlier sections) do not scale, and by keeping syndicates small and focussed, a lot of these issues can be avoided.

Relationships

While the user and developer audience of a syndicate is up to each member of the syndicate and how they identify with the syndicate, relationships between syndicates

should aim to be more formalised. Syndicalist cooperation is based on solidarity.

Fragmentation is a real concern in this regard and demands cross-collaboration between syndicates in terms of basic specs, core components, and design choices. This is to ensure that software made and maintained by different syndicates remains as compatibly with other versions as possible.

However, just as with identification of syndicate belonging, diverging projects should not be hindered if this serves the need of different user groups. Neither software not its user base is monolithic and projects may diverge from each other if their goals no longer align. An effort should be made to allow for future cross-collaboration, but there is no point in spending energy on drama and conflicts if developers and users would be

happier with two separate projects rather than one.

This space offers a lot of opportunities in terms of designing tools for cross-collaboration. Syndicates might be able to publish change sets that provide metadata for git patch sets that can easily be included by other syndicates, or individual users that want a custom version of a piece of software that is configured and compiled just for them, based on a mainline version and patches that are maintained by different syndicates.

Decision models

Decision making processes may differ between syndicates which offers a choice to both developers and users in terms of how they want to engage with the software they use.

Casual users of a software might choose a syndicate which does not make decisions based on user feedback and instead trusting the cabal to guide the development and maintenance of their tool. On the other hand "power users" of a software may want to be able to be more involved in the decision making process without having to gain the technical knowledge and experience to join the core cabal.

By diverging communities into smaller syndicates it is possible for these communities to organise themselves differently while allowing for cross-collaboration on important features.

Decision models can be aligned on two axes: **knowledge** and **trust**[15]. Knowledge relationships are based on agreement on

---

15 https://media.ccc.de/v/36c3-10858-
    infrastructures_in_a_horizontal_farmers_community#t=593

technical ideas and what is commonly used currently in "meritocratic" systems. Trust relationships are based on mutual understanding of the principles that go into making a decision and developing a software. These two relationships can interact in interesting ways.

```
                    high trust
                        ^
                        |
                        |
    CONSENSUS           |       UNANIMITY
                        |
                        |
                        |   high agreement
<---------------+---------------->
    low agreement       |
                        |
    DISSENT             |       TECHNICAL
                        |       AGREEMENT
                        |
                        v
                    low trust
```
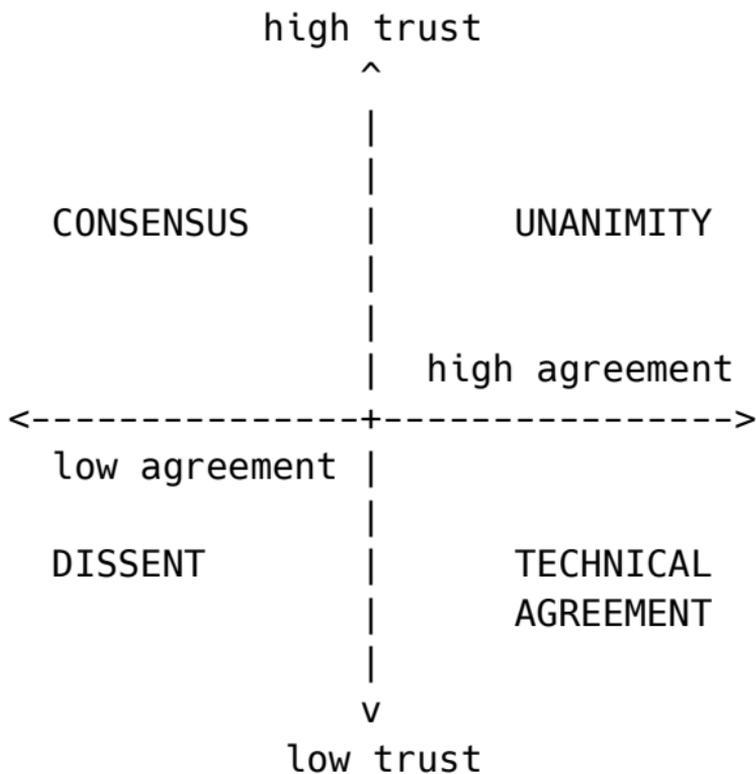
Collaboration is possible in three of these quadrants, although only two of them are ideal. When two groups agree on the details of a solution but do not trust each other, a technical relationship can be formed. This usually involves a specification that is then honoured by both groups (and others that join into the relationship at a later point in time).

On the other hand, when two groups have a strong trust relationship this allows for collaboration via consensus. Consensus decision making[16] means taking every individuals point of view into account and coming to a decision based on this information. This means that individuals can disagree with specifics but find some common ground that they can both "live with". This means that decisions are based on the comfort

---

16   https://en.wikipedia.org/wiki/Consensus_decision-making

edges of all participants. There is no voting as this would enforce a majorities view over any minority and every stakeholder in a system can exercise a veto right to stop a decision.

These processes only work in small groups, which is why syndicates are also encouraged to form pure technical relationships.

The quadrants "UNANIMITY" and "DISSENT" should be avoided as they either result in an echo-chamber effect in terms of decision making, or don't allow for effective collaboration at all.

Challenges

While the previous sections outlined opportunities solve (and improve on) the existing problem domain, this idea is not without its own challenges. This essay attempted to propose solutions for some of

these, but can of course not hope to be comprehensive.

Technical fragmentation

Existing projects that use a similar approach[17] can suffer from "fragmentation" or "fracturisation" (commonly also called "balkanisation"). This is the process by which communities diverge so significantly that they are no longer compatible with each other. In the case of Freifunk this means that the core software is still developed communally between all "chapters", but configuration and network setups vary so widely that moving between networks requires fundamentally re-configuring infrastructure devices.

---

17   https://en.wikipedia.org/wiki/Freifunk

Creating small syndicate communities around all sorts of software projects may suffer from the same problem if not managed accordingly. This requires collaboration platforms to grow and scale in a way that they currently *don't*, or for syndicates to operate from compatible principles, which will be hard to ensure and verify.

Not invented here

A common theme in software development is the "not invented here syndrome" (NIH[18]) which prompts companies to rewrite technical projects created by other parties because they either don't like or don't understand the existing (and available) solution.

---

18 https://en.wikipedia.org/wiki/Not_invented_here

Because free software developers do not exist in a bubble this affects free software projects as well. Fragmentation in communities, social and technical differences in understanding, and other factors might contribute to a rise in NIH among syndicates. This is a problem with no real solution. It can potentially be avoided with better communication.

On the other hand it is important to consider that just because something has been written once that does not mean that no alternative implementations can or should exist. It is possible to find errors in specifications through alternative implementations[19].

---

19 https://blogs.oracle.com/developers/building-a-container-runtime-in-rust

Political fragmentation

Similarly to how technical differences in opinions may fragment a project, the same can be said for political ideologies. It is reductionist to assume that ideology in itself is the problem (after all not believing in ideologies is itself an ideology). Labels exist in language to catalogue and describe natural and cultural things.

The importance is to recognise that different labels can exist for the same principles, and that similar political conclusions built on the same principles are still compatible with each other.

Outlook

Bringing this essay to a close, we look into the future. The way we build systems and organise ourselves in communities has grown

out of the capitalist system that we aim to escape. Furthermore this is not simply about the development of software (and other technologies), but about giving users and developers autonomy over the tools that they build and use.

It is time for an overhaul of how we organise, and to become aware of the systems that we replicate in how we develop the technologies that we hope will transform the world. This is sorely needed, as boycotting technology is not the solution to the ever growing surveillance apparatus created by capitalist systems.

Ultimately, software syndicalism is about reducing the distance between the creation and maintenance of technology and its users.

A software syndicate, for whom?

by natacha

Who is concerned by a software syndicate? How can a software syndicate embrace transformational forces? As we are observing worrying political drifts, a clear consolidation of control society, and possible rise of fascist discourses, we know from experience that the stake of maintaining an independent infrastructure of communication is crucial to resistance networks who will take the charge of a further social response. However it feels that this important concern is often being held as secondary.

Advocating for a global approach to address modalities of the technological society often makes you qualified as idealist, since power in place greatly favours militaro-industrial complex, even so embracing the issue might be the only way to organise in the *here and now* towards the world to come. We will envision

here how concerted actions at a small scale can contribute to a global thinking on the topic.

Scavenging of free radicals

While the centralization of data and privatization of software, favoured by the neo-liberal economic dysfunction, occupy almost all domains, there are still many activist projects that continue to propose singular tools and models of community organization; some radical technical collectives maintain their existence, providing long lasting independent communication tools, for example `Riseup.net` or Autistici/Inventati[20] (moreover, A/I, in their orange book[21] document their infrastructure for activist projects, an

---

20 `+KAOS: Ten Years of Hacking and Media Activism` ISBN: 978-94-92302-16-8

21 `https://www.autistici.org/orangebook/`

important step to allow the transfer of experience so others can reproduce their process[22]); other initiatives, such as the Lorea project[23], have had a short and intense life by engaging in resistance organizations; more recently, more structures have followed this path, notably in Europe, such as `disroot.org`, `tutanota.com` and some got organised in a network such as CHATONS[24] in France. To date, it seems that while useful software are available, well organized communities, who provide secured tools that can be configured according to needs, they are most often unused or at least not used to their full extent.

---

22 The list is long and Riseup maintains a partial list of radical technical collectives :
    `https://riseup.net/en/security/resources/radical-servers`

23 `https://web.archive.org/web/20151103003019/http://lorea.org/`

24 `https://chatons.org/`

Popular protests that bring hundreds of thousands of people into the streets most often rely on centralized social media platforms, although within these groups a minority of individuals will prefer to use a secure method of communication, similarly, for the moment I don't know of any coherent and concerted digital organization in resistance networks. Most often a discourse rejecting technology dominates in activist circles, this position seems unrealistic given the hold of centralized platforms on our exchanges. As a result, not only do we depend on models imposed on us by centralized corporate technologies, but we also (pretend to) trust them to manage our data.

Technology production is undoubtedly part of colonialist exploitative history, its operation benefits the empire. The nature of digital tools,

which do nothing better than reproduce information, is curbed by copyright while corporate practice largely favours the tracing of those who produce information; leaving the infrastructure in the hands of corporations alone only aggravates the problem. On the contrary, the experience acquired over the years makes it possible to envisage a controlled use and a better understanding of the stakes, which necessarily requires, as we know, the sharing of documentation, the creation of spaces for reflection, debate and active pedagogy in order to rethink our relationship with technology.

Facing the lack of collective reflection about the technologies used to coordinate social movements and resistances, it is useful to consider free software in its structuring capacity: both an approach to bring software

code into the public domain and a methodology to coordinate contributions and the maintenance of digital tools. To this end, this text starts by addressing the observation that free software projects face systemic limitations, those are particularly sensible and limiting in the social organisation attached to free software production; it conforms to a norm established in an essentially masculine and Western universe, and non tech people lack of information about the technical reality and fail to see the scope of the problem would it be for their own safety or for the sake of formulating meaningful discourses/claims in political or academic circles.

Polyglot technologies

Free software programmers form an international community that agrees on collaborative work methods and specific tools, such as the version control software git. This community often shares social characteristics, creating knowledge silos that influence the direction of software development.

- Free Software projects are almost always coming from people with Euro-American cultural background, those people with a Western heritage tend to reproduce existing patterns of domination.

- When programmers realize they are creating a knowledge silo, and maybe are not creating a welcoming environment for others, they often

remedy this by turning to the mainstream, either by imitating the interfaces and structures of proprietary software, or by trying to be compatible with existing corporate tools rather than asserting the construction of a different technology.

- This reasoning results in social and technological environments that limit dissent thinking and make for difficulty to voice out and even see the need for radical transformation, even more the idea of grounding this transformation in the fragility of communities is sometimes called out.

As a direct result of colonial history and Western domination of education and access to infrastructure, most software, and even more open source software, is developed by people

who have better access to university education, and the identity conveyed by the community or workplace does not help to transform the situation.[25] With respect to the West, Charlton Mc Ilwain explains that, from the beginning, certain populations have been historically and deliberately excluded from the institutions where technology is developed: "The Folks at MIT and those like them were building a new society they made the de-facto decision to exclude Negroes from designing, building, or deciding what computer systems would be built".[26]

In this context, developers bring their culture with them and organize social structures, their proposals are not always

25 https://archive.fosdem.org/2019/schedule/event/
    python_diversity_gap/
26 *Black Software, The Internet and Social Justice from the Afronet to Black Lives matter*, Charlton D. Mc Ilwain, Oxford University Press 2020, p.21

welcoming to people from another background or country. While contributions to Free Software projects come from global sources[27], this is not visible in social spaces (festivals, conferences, hacker camps, etc.) where people interested in technical issues meet, nor in decision-making processes. Unfortunately, for various reasons (intersectional issues too complex to be discussed here), it seems that the population of free software programmers is more uniform than the population in corporate environment.[15] The uniformity of the population is often the first thing that strikes a person attending a large free software conference for the first time[28]. Also these

27 *Who is an open source software developer?*, Bert J. Dempsey, Debra Weiss, Paul Jones, and Jane Greenberg, in Commun. ACM vol. 45, Feb. 2002. DOI: 10.1145/503124.503125

28 https://annadodson.co.uk/blog/2019/02/04/fosdem-2019/

western and gendered social practices, often do not completely separate themselves from those in the corporate world – which is very much present in the vicinity – promote, among other things, a hierarchical and personality-based structure and do not allow for the sharing of organizational modalities specific to Free Software.

On another hand, despite the evidence of their toxicity, a great deal of tolerance is given, even in critical circles, to the use of surveillance capitalist software. More often than not, the explanation given emphasizes the difficulty of changing existing processes, and denies the need to think about the benefits of a concerted transformation. From this situation, where the lack of dialogue and collective reflection is obvious, the result is the crystallization of a structure of domination

where programmers keep a grip on technological choices. The resistance organizations, on the other hand, argue that they are fragile and lack the time and knowledge to continue to feed the flows of techno-surveillance with their data, their emotions, their motivations, their relational graphs and, more than anything else, to bind themselves to the fragmented, self-promoting and time-consuming operating model put forward by the technologies they use.

There are many attempts to remedy this situation, but because of the reality described above, discussions about technology happens in closed circles and often fails to consider the peculiarities of Free Software; they are not considered as specific systems, and loosing sight of their singular possibilities. The same thing happens when it comes to meeting the

needs of "users"; the very successful campaign of the French Free Software provider Framasoft, called *Dégooglisons Internet[29]*, offers free of charge Free Software alternatives to most main centralized online services. However, by providing "alternatives" that in some way try to keep the familiarity of the user's habit with centralized corporate software, we still subject civil society organizations to the world view that these companies promote, identity based on gratis usage and focus on serving individual projects, rather than exchange and collaboration. The reasons for this choice are obviously pragmatic, it is difficult to break away from the dominant model; highlighting the possibility of another organization fostering the visibility of a different paradigm requires a voluntary and

---

29   https://degooglisons-internet.org/

persistent work from different groups across existing social organisation. Transdisciplinary and inclusive conversations about technological production and usage would help to approach Free software production as a process that allows a different way of functioning, which would give another access to the digital tools, notably by offering the possibility of discussing the different technological choices and would allow a shared understanding of the technical stakes, of the needs and of the social functioning necessarily associated to digital communications. There are very few spaces where transdisciplinary exchanges take place, very few knowledge bridges where free software developers learn and share their experience with other disciplines, other experiences and engage

reciprocally to give life to proposals that respond to expressed and different needs.

Faced with the obvious signs of the consolidation of a techno-fascist domination, we are left with the desire to organize in order to set up a radically transformative social and human way of functioning and rethink technology together. It is time to find places where we can exchange and function in a collaborative way. As we have seen, there are few of these, they are split between programmers and activists, and above all there is hardly any structured arrangement for the transmission of knowledge. The need to create an activist milieu to discuss technological practices in a society where computers are dominant is apparent, this can take different forms, meetings, workshops, writings, digital exchanges but in all cases and it must be a

shared project inhabited both by people contributing to the development and use of software and all the people who are generally excluded from these debates.

Proximity transmission

The virus tactic

- Modalities of technology are unknown because most people are kept dependent by corporations.
- Activists don't have time to invest in understanding the technology, they are already divided and overloaded.
- Difficulty/impossibility to get feedback from users at the software development level because they have no reference in software, only in identity-based enterprise products.

Most of the arguments for not considering the possibility of other technologies are self-deprecating: "technology is not for me", "I don't understand anything", "I don't have time", etc. Yet screen time is constantly increasing, and the indispensable operations of daily life are increasingly intermediated by capitalist surveillance platforms.

These observations are banal and often dismissed with a shrug of the shoulders, reflecting a feeling of powerlessness. Activists are already exhausted by too many responsibilities, the technology they use should support them in their activities and not require more time, as those based on an attention economy promoted by many platforms do. On the other hand, free software projects feel the need to reach a wider audience, they rightly assess the need to better respond to users'

needs, to get feedback, to do *design UX*; however, in the absence of shared structures for reflection on technological developments, the point of comparison remains the dominant tools This comparison is reinforced by the fact that, when asked about the desired functionality of free software, "users" who are not well informed about the possibilities of free software and who are not engaged in a broader reflection about technology will use the most well-known software as a point of comparison.

We need organized working groups to care together for the terms of the technological society: transdisciplinary software syndicates. There we can think about directions and make decisions for the development of software that would feed into strategies of sharing, from and with resistance networks. Software syndicates

are understood here as proximity-based structures that can serve as a basis for strategy development and the transfer of information and knowledge, decentralized and online decision-making processes and the federation of needs.

Furthermore the comprehension of surveillance capitalism formalized by hacker communities could support activist projects; recognizing the ways in which both groups pursue the same goals is essential. To get such processes going, we need people to take on the task of intermediation, to take on the role of the missing link between free technology actors and activists, to build a ground for thinking about technology for resistance, to become knowledge bridges.

Knowledge Bridges

- Need for systemic adoption of open infrastructure.

This is not a proposition to rethink the universe, the assimilation of always new technologies and infrastructure exhausts the users as innovation pushes on always conceiving new tools or new ways of doing things to exchange socialize and form knowledge together. On the contrary, there are many free software techniques and tools that have steadily permitted over time the appropriation of technological functions, and permit to fix a standard for adapted to different uses and practices. In the same way, the persons who set as knowledge bridges from/to programmer environment to activists have an interest in relying on existing social

organizations rather than building again a new organisation, in particular by working with existing civil society structures and resistance organizations who have a experience in their action. From this point of view, the intermediary role of knowledge bridge is essential, it is not necessarily a question of developing more or better tools, or other structures, but of knowing how to manipulate the existing ones, to install the necessary tools and to transmit an understanding of their modalities of functioning so that their use meets the needs of the engaged collective. It is also a question of understanding and easing other people's understanding about how the most widespread technologies are free software and they function differently. The practical documentation of technical processes is really rare and the time spent to realize these

documents is often not considered as a value-creating activity, yet it is essential to the life of the software, its adoption and its future transformation.

The example of Andrea's work in the *Campi Aperti[30]* community is an inspiring example of how it is possible to integrate both governance work within a community seeking to exist in a horizontal relationship, and existing open source tools developed within their own communities and rarely used in this context. Andrea proposes several principles:

- Do not do things alone.
- Testing environment.
- Document everything and explain the choice of algorithm.
- Give yourself the time to study.
- Do not be too much specialized.

---

30 https://media.ccc.de/v/36c3-10858-
    infrastructures_in_a_horizontal_farmers_community

She explains how she was able to engage the *Campi Aperti* community around the setup of their dedicated network and their own servers. Every technological decision was integrated into their choices of organization and collective validation, and the people involved were quickly able to take ownership of the proposed technologies.

Several initiatives have thought of projects integrating different software in a shared environment aiming at facilitating their installation, among other things by using dedicated hardware. These projects are important spaces for the construction of technical independence. The person who is able to pass on technical information and support others, a knowledge bridge, may intervene temporarily or over time, may or may not have a technical and/or activist

background, or both, the essential thing is that she acts in a spirit of sisterhood and community knowledge building through mutual self-learning and support.

Relationships, locality, proximity, community and globality

Taking into account the different observations often shared about the modalities and structures allowing for the development of free software, we can specify a little the modalities of a software syndicate.

Local reference is immediately accessible. Local information, networks of common goods, cartography.

- Local is important for human relations and further communication.

- For software, geographic location is not significant and can lead to redundancies in development.

The notion of locality often comes up in current social contexts, nurturing battles and utopias, it also is very present in the discourse of radical technologies and independent providers. While the idea of locality seems obvious in an immediate definition: "what is close to us within a radius of x km"; and if locality makes sense in terms of human relations, as a form of reappropriation of agency, it can also be the occasion to evade a large number of issues of historical domination, colonial for example, not acknowledging that our wealth and welfare system is heavily constructed on exploitative systems still in place. Indeed locality needs to

be understood as different from autonomy because the existence maintained locally in the West depends largely on global structures of exploitation. Furthermore what does locality means in terms of software development.

For example, the question of a local community associated in the development or maintenance of a software or a code base seems to be associated with a particular vision of locality, some urban centres where a sufficiently large number of programmers are found to form a local community, but this is not the case for most rural spaces.

Technological knowledge is situated, and it is crucial that the people who hold it implement structures for dialogue with other social spaces, and in the current situation where corporation attack every piece of land it is all the more crucial for radical technologies

that are most often thought in urban contexts, to keep in touch with the various territorial struggles that are most often rural.

Development based on proximity

Taking in account the previously mentioned issues and limits, locality seems on other aspects an important asset for radical and free software technology development and organisation.

- Proximity rather than locality.
- Who does what, code is not the only modality of technologies.

Software development relies on human organisations that are both localised in certain urban centres and at a distance historically formalising online ways of organisation, that allows them to keep track of their projects and

communities in a decentralised manner very adaptable and reliable. Those tools allow to form affinity groups over specific software/type of software where technological choices are discussed for their pertinence but also for some more obscure reasons that can resemble a form of attachment to a certain identity, aesthetics or , even what some might call political reasons (most of the time unacknowledged). For example adhesion to decentralisation, or technological minimalism has definitive political groundings, most of the time not presented as such.

Developer's affinity groups form proximity relations that are not solely depending on their locality and constraining them to a local implementation would be absurd, contrary to the networked quality of the technology. However Free Software development most

often happens among a network of peers, organized in groups of belonging and identity. People are localized and meet in *hackerspaces* for example, they meet there independently of specific projects, but these social spaces are the occasion to implement relational spaces where technological practices and different needs are thought. Identifying local representatives of software projects could create a referential bridge for local user groups.

The programmer's time is usually taken up, so they don't see the need to devote themselves to the dissemination of the software they are working on. Conversely, for outsiders, engagement in thinking about a program or its documentation is an opportunity for reflection on systems and technologies, and perhaps the formation of critical thinking. The *hackerspaces*

and *hacklabs*[31] can be the venues for these encounters, they allow different people to share an interest in the technology, and some will then serve as intermediaries to share systemic understanding and support community building.

Community

- Community technologies must be understood from the beginning as an open source system.
- Radical technologies can be thought of by a diverse community.

The diversity of communication spaces is recognized as a guarantor of the formation of critical thought, of the dynamism of society and ultimately of the richness of life, yet, as we

---

31   https://hackerspaces.org/ and https://hacklabs.org/

have seen repeatedly in recent events, far right fundamentalists know that getting their hands on digital media is key to the consolidation of their social influence; they are aided in this by centralized platforms that practice double standards in moderation, tolerating, for example, racist violence and threats. Moreover, the techno-fascist control society sets up standards which are, by their requirements and their modalities adapted to corporations, those standards are not favourable and exclude de facto small and decentralized organizations favourable to the common good. Under the impact of these different threats, thinking of the digital infrastructure as a free and decentralized software commons is the first condition for its survival and for the possibility of maintaining a diversity of speech and opinions that is essential to shared thinking.

Such systems, in order to exist, must necessarily be designed with the participation of communities from the start. Confronted with the importance of inequalities and the violence of the current context, it feels illusory to think that this transformation can be done on a large scale, we can build on the experience of local resistance environments to implement another technical functioning.

As a conclusion I would like to affirm that Including digital issues in the design of our resistance organizations allows the formalization of a complex thinking that goes beyond the simplistic opposition forming against digital tools, which also recognizes the possibilities of a practice of digital commons; while being aware of the flaws of computers and their seating in a logic of surveillance. The means of this organization are community-

based and reside in practices of communication, documentation and knowledge sharing, such a reflection would be a pillar for software syndicate that offers a space to rethink the existing systems in the service of active social movements.

Supporting resistances

by hellekin

We have to deal with the world as it becomes, not with the world as we would like it to be. But we have to stay as close as possible to what we think the world would want, experimenting and tinkering, and praying that the world doesn't get angry at our mistakes.

— Vinciane Despret, *Autobiographie d'un poulpe* , p.120 ean: 9782330147631

Dispelling the fog

In the field of software development, support for resistances does not necessarily involve a radical personal change, but more simply the dissipation of the propagandist fog that makes knowledge a commodity among others. Supporting resistances thus proceeds from an "inversion of responsibility", to abuse a computer science term, where one discovers

that a change of view allows the conceptual collapse of a system of oppression that was previously imposed as obvious.

The software industry, dominated by capitalist interests and methods, determines a "market" according to the consecrated terms of competition and scarcity. Producers of software, conceived as products, engage in a wild competition. It is a frantic race where talents are competing to produce as quickly as possible a software (or its promise) that will attract the attention of a predator. The biggest companies buy up the most "innovative" start-ups in a ritualistic, predestined business plan in which a wealthy buyer phagocytes the seller to take over or eliminate its competing product. The industry continues to apply its strategy of "embrace, extend, extinguish" to hide its own misery.

But these mythological terms are so far removed from reality that the capitalists themselves still use an inversion of meaning to hide this fact. Thus, they call "permissive licenses" those that permit predators to avoid the non-reciprocity of the value added by the software produced in common, and "restrictive licenses" those that restrict or abolish any capacity for exclusive appropriation of the software they cover (extraction of its value). But in the face of the abundance of code, the idea that this "commodity" could share the character of scarcity of oil or coal borders on the grotesque; and its *cooperative* mode of production makes any claim to competition futile and derisory. The consideration of cooperative, non-exclusive and non-competitive knowledge technologies invites reflection on the scope of a competitive vision

when resources do not meet the condition of scarcity on which the whole edifice of capitalist extraction is based. In such a context, it is the whole political orientation of society that is turned upside down, making a large part of the software industry obsolete.

Promoting mutual aid

A common – *communalistic* – approach to generalized software production could benefit, for example, the professions of the independent book sector or doctors; an agreement to support the development of free software for their own use would significantly reduce the cost of developing and maintaining common software – considered a common resource; these costs would be much lower in the long run than maintaining an industry designed to extract value rather than provide

it. A professional union could advantageously pursue the invention of a common technical pole for which only the characteristics linked to national differences (e.g. legal) would impose local overheads; most of the functionality forming a common good, the whole profession would benefit from a technical and social innovation – properly *technological* – determining the improvement of the working conditions of all professionals.

The lifting of the smoke curtain of *siliconed* capitalism would reveal all the interest of establishing a public digital infrastructure on free software that would favour its cooperative modes of production and would also satisfy the declared European political will of an open and competitive market between its small and medium-sized businesses, but on the solid

bases of a *technical floor* maintained collectively for the benefit of all participants.

Not only technical, but also organizational, this evolution would accompany the very practices of the user communities and would thereby shape their relationship to technology, allowing them to appropriate it and to consider it from a collective and political perspective. Little by little, the bad habit of abandoning technical choices to corporations would fade away and be replaced by a will of general interest carried by *software syndicates* that would themselves be dedicated to the improvement of the conditions of their own users in proximity. Private interests and their goals of value extraction would be replaced by a logic of improvement of uses, of invention for the general interest, the respect of differences and the taking into account of

singularities otherwise ignored; the value thus created would be a public good, in the same way as science, culture and arts.

Asserting ourselves together

From then on, it would become possible and easier to harmonize legal and soon legislative instruments beyond the specific conditions of each nation; thus, professionals could influence in a much more rational and efficient way the evolution of their profession in a supranational context. The cases of the independent book sector and doctors are helpful in understanding the benefits of standardizing digital tools (software) as a common good. Other domains, such as accounting, architecture, or the relationship of legal persons to administrations, offer a similar opportunity beyond each corporation, in a

context that could be described as *syndical* ; the anti-capitalist struggle for the abolition of the exploitation of workers would naturally find its expression in cooperative, non-exclusive and non-competitive knowledge technologies.

In other words, an approach to software production as a provider of a common infrastructure not only makes the user the central force of proposition, but also renders obsolete the artificial fragmentation of an entire industry based on what can be called an *intellectual racket*. The capitalist mode of production is antithetical to the functioning of a digital commons: the cooperative approach of the commons is infinitely more adapted to software production than is an exclusive approach.

Take me to your leader

by petites singularités

The NGI has no hidden agenda, other than to restore the balance of power at a global scale.

We reject: kings, presidents, and voting. We believe in: rough consensus and running code.

— David D. Clark (1992)

Rough consensus does not mean being presented with a *fait accompli* and having to face up to an unwanted reality. Pushing reality as imposed facts rather resembles a dictatorship.

So-called benevolent dictatorship[32] is an old school free software governance practice, that sounds very strange to French ears. Brought up in the republican schooling model where they learn that the enlightened absolute monarchy practised in Versailles was in the end dethroned by the very republican revolution, the French quickly learnt that the enlightened model didn't have the effective means to back its claims of absolute power. In fact, the current neo-liberal model of global surveillance has far more reach that any absolute monarch ever had. But still what does it mean when both models are associated, when self-confident old school free software

---

32 *Benevolent Dictator* designates in the free software community the person taking the lead and unilateral decisions regarding a software project, most of the time this situation is correlated to the fact that many projects originate in the idea and action of one person who starts writing the code.

figures, acting as sole decision makers in their domain, partner with the financial leverage of neo-liberal economies, sometimes in governmental context?

Code is politics[33]: governance of the internet was born of the rejection of politics by engineers; the free software movement was born of the rejection of an opaque, nascent software industry; Copyleft appeared as a way to subvert the extractive nature of copyright.

The essential quality of free software is to be developed, maintained, and eventually reproduced and forked by independent human beings supporting each other in this endeavour according to, in the best case, their shared vision of society. Of course no one exists away from capitalist system, long-lasting projects

---

33 Opening sentence of *Software freedom your way*.
   https://ps.zoethical.org/t/sfyw-software-freedom-your-way/19

cannot run on a voluntary basis and require funding. A proper software infrastructure that allows for the multiplicity of free software initiatives to exist and serve the growing needs for reliable, convivial, frugal, and secure digital tools requires public financing.

Currently there is an important and welcome move from European authorities to support developers, software infrastructure, and build some form of European sovereignty. Different programmes are supported that give access to some means and a better visibility to European software, emphasizing open-source and free software, supporting European developer's expertise that has been largely under-recognised until now. Among others the NGI programme and its associated cascading funding – petites singularités acts as a mentorship organization within the NGI Zero

consortium – has benefited many essential free software projects (needless to say, the very large majority of them originates in Northwestern Europe and are led by white men). In fact European authorities clearly voice their objective across this process, to create what they call an "EU Champion", the European equivalent of Silicon Valley giants. Ain't it great? Indeed why wouldn't Europe counter other hegemonic giants in the U.S., China, and probably Russia with one of its own?

Still, is the "EU Champion" model the right one for free software, and what does it entail? Free software values many forms of independence, and despite a blatantly monochromatic milieu, a diversity of visions are still represented, and code generally considered a contribution to the digital

commons towards general public usage; along with the pattern among many free software projects to seek some autonomy from direct State and corporate control. Decentralisation has historically been a critical affordance to support a degree of independence and a diversity of organisational models.

It is more than time to start negotiating the financial conditions of our independent endeavours, to control public resource allocation and discuss openly the attribution criteria. It's time for free software developers to get a hold of their capacity to assess the needs of their communities and express them with agency, withdrawing from the myth that funding is alien.

Not so long ago, the free software movement was in pain and at risk because no one cared that essential pieces of our technical

infrastructure would be maintained voluntarily by a few isolated individuals. Stephen Henson single-handedly maintained OpenSSL, without fair compensation, when Heartbleed struck.

From the grass-roots, a few organised and taught each other about ways and means to access public funding, while philanthropic non-profits created funding infrastructure from scratch, looking at European Commission programmes.

Do not be mistaken: it's not because this money comes from our taxes that anyone can easily access it, as the Commission lobbying structure is official and regulated. You must form an *eligible*[34] consortium involving many partners, fill and submit one or more 70-page

---

34 Theoretically only European organisations are eligible, this can be circumvented by creating a dedicated legal structure in the EU to capture EC funding.

proposals that must be accepted by an independent jury of experts (whom you could join[35]), sign multi-year contracts and a consortium agreement, then process a burden of administrative reporting for the duration of the programme and beyond. This process is theoretically doable by anyone[36], yet in fact, only informed groups have access to it.

In addition, it is a known fact that organisations from the Northwestern European countries have very good access, furthermore they can also build up on their local experience as more national public

---

35 https://commission.europa.eu/jobs-european-commission/experts_en

36 Holger Krekel introduced EU-funding at 32C3 https://media.ccc.de/v/32c3-7300-hacking_eu_funding_for_a_decentralizing_foss_project, and hellekin made a presentation at FOSDEM'18 https://archive.fosdem.org/2018/schedule/event/eufunding that led *petites singularités* to join the NGI Zero consortium.

funding had been entitled to technical infrastructure in NW Europe. Therefore some non-profit philanthropies had the capacity to build upon their previous experience and proximity to the funding sources, they wisely structured while providing the larger community access to dedicated cascading funding.

For the last five years European free software communities have increasingly been funded by cascading funding from consortiums appointed by European Commission's NGI programme. The facility put in place is a great success and a relief for most developers, a simple, lightweight *do-autocratic* milestone-based model matching the development workflow, where you get the money when the task is done, with no administrative burden: *the keyword here is efficiency.*

Still here comes the issue with the "European Giant": when many different organisations are united under a sole model, diversity is at risk. As many resort to the same centralised preset resource allocation via a unique organisation, as benevolent and supportive as it may be, different issues can arise.

What happens if our benevolent dictator is hit by a bus? We have no direct access to the financial infrastructure, since we have delegated all our representation into the hands of one entity. Although we're dealing here with European tax payer money, public funding, none of the beneficiary individuals and organizations have any sort of institutional representation at EU level. We trust the benevolent dictator to do the right thing, but when many different organisations are united under a single banner, it is simply very easy to

replace a key person by any corporate fiend, and simply lose all our own infrastructure at once. You've seen that before, haven't you?

Another issue is the lack of visibility over both attribution conditions[37] of funding and the decision structure of the Commission programmes. Both the Commission and beneficiaries seem to disregard the conditions of attribution of cascading funding; there is no report over the composition of expert reviewer juries nor any feedback channel to discuss ongoing choices; claims voiced on public channels often remain without response. It is symptomatic that many grantees do not even know that the cascading funding they receive is attached to a consortium. Consortium members do not discuss their consortium's own orientations internally. There is no

---

37   Not to be confused with eligibility conditions.

momentum nor place where free software actors organize to discuss funding mechanisms, their pertinence, actual diversity of needs within the movement, in order to formulate demands towards funding structures.

Despite its impressive success in reaching out to the free software movement and supporting development, NGI project-based cascading funding does not support medium or long term views, and only grants money to hardcore coding: no money for community building, nothing for care and few for maintenance[38]; it doesn't really matter if the code will be used, as long as it is able to display its technical existence, as ephemeral as it might be.

---

38 E.g., software packages may be created on-demand as a service to developers via the cascading funding mechanism, but these may not be maintained because the developers' community members are not involved as package maintainers, and the original packager does not use that software either.

Rationalisation of the funding schemes seem to be in order to fit all in one size, leaving no time to discuss collective objectives, diversity of needs, and upcoming processes; only code is law. This is a dangerous path for diversity, as reported biases[39] may not be addressed, and therefore contribute to their legitimation and continued reinforcement, and the faster the better.

ASAP though may not mean what you think it does: attentive scrutiny affects power. Sure diversity is bothering and slows down decisional processes, yet in a world trending towards uniformity, it is a guarantee for survival and resilience. A one-stop shop for public funding of free software threatens the identity of free software, as it channels ways and means into a single approach – here an

---

39 https://sleepmap.de/2023/operating-system-bias-in-next-
    generation-internet-and-nlnet/

interpretation of *code is law* – at the service of the neo-liberal world view. Diversity is a strength that belongs to the grass-roots, and Cyclops have been trying to embrace it and quench it many times in history, for it enables people over time to challenge hegemonic ruling.

Now, you might ask, who decides what is financed? Well, our benevolent dictator is the fundamental joint here: recruiting independent experts, onboarding grantees, sometimes reporting[40] to Commission people who drink those words and learn each time better what free software is about. Some think that is *a good thing™* because he knows, he has been doing it successfully for decades to the benefit of the larger free software and open-source communities and we should trust his

---

40 https://nlnet.nl/NGI/reports/NGI-Study-ISBN-
    9789279864667.pdf

experience. But how do we learn from it in order to distribute this amazing capacity across multiple people and organizations?

As software is omnipresent in every aspect of our societies we have long understood that it does not exist only by its source code but also through its inner social organisation that informs technical choices, implementation, and conditions its adoption. As we discuss software syndicalism, financing our endeavours seems a crucial point where syndicalists need to organize beyond the available funding sources, to sit down together and summarize the needs and priorities to be funded. Formulating adequate claims for financing is key to think effective organisational means to access to the funding sources, legislative processes, and ultimately to the overarching economic strategy that will grant free software

an actual status of public digital infrastructure. How could we support each other across projects in clarifying our needs for financing and the diversity of possible sources as we wish to avoid creating dependence on a centralised source? Shouldn't we also share a public space where we discuss our choices for financing models and decentralisation of funding sources?

A digital society with an internet for humans creates a way to discuss publicly and formalize collectively our technical needs as citizens and as society. Citizens must have ways to influence the allocation of public resources towards world-transforming technical ensembles. Beyond the existing and quite successful code-oriented, project-based funding efforts, we demand infrastructure-based community funding for scopes beyond a

single project, that affect more than one software and demand peer cooperation, and also care-based funding to support community development and integration of free software in social life, that is not limited to blanket surveillance and military apparatus.

Let us discuss! Share a tea at Congress, a beer at OFFDEM, organize working sessions yourselves and report to the community of peers at `offdem.net` and across the Fediverse. Share your concerns, publish your studies and your results to benefit the world. Let us gather and formalize non-profit entities to go beyond the arbitrary project-based division of digital labour, and produce a grass-roots, long term citizen political vision for a digital society that does not support the alignment of forces on runaway industrialism, extractive capitalism and militarily legitimized mass murder.

What is at Stake with Interoperability

by petites singularités

> *Until now I could escape Facebook : I have no interest in "having to" talk to people "on Facebook" or to give my consent to this company's practices incompatible with my ethics. Given its dominant position, I have a doubt in my individual capacity as a citizen to resist an interconnection with Facebook that would be imposed from above.*

Consent & interoperability

The General Data Protection Regulation (GDPR) provides *explicit consent* to data usage. But **within the scope of interoperability,** *refusal to consent* **to any predatory use** *must not interfere with communication.* In other words, the predatory platform, if it becomes interoperable by force of law, must not acquire the capacity

to monitor participants in a conversation between its users and people exterior to its platform: this would be a serious violation of privacy of the people in communication.

Interoperability & interconnection

Yes, interoperability is necessary, but it is not a miracle solution to limit the power of and the capacity to exchange with these services could depend on our identification to them, thus our acceptation of their conditions. imposes *explicit consent* for the treatment of personal data (articles 4.11 and 7) that we, non-users of these predatory services, refuse to give: we won't be able, *a priori*, to interact with these accounts with whom we can only connect to by accepting the unacceptable terms of service of their operators, hence interoperability cannot

work since it resolves into an "interoperability without interconnection.[41]"

Interconnection & data portability

Before rushing on the idea of interoperability of the internet giants with open standards, it is therefore necessary to ensure the implementation of the so that users trapped in the platforms can export their data through the use of standards (e.g. ActivityPub). Thus, by allowing users to regain their digital sovereignty and regain control of their personal data, we can kill three birds with one stone: weaken the giants with questionable practices, strengthen existing European law, and observe the emergence of social media

---

41   On the difference between interoperability and interconnection, see Laurent Chemla, "Interoperabilitay", on February 22, 2020. http://www.non-droit.org/2020/02/22/interoperabilitay/

decentralization in line with European values and the charter of fundamental human rights.

Interoperability, interconnection and consent seem to us to be the nerve center of the debate, however it remains complex and overflows in all directions, for example – and this remains open to discussion without being exhaustive:

- **The existence of interoperable open standards**, such as ActivityPub, XMPP, etc. must be supported, notably to allow users in silos to change services – but without losing information, especially because "personal data", usage history and existing conversations, contacts, etc. cannot be transmitted (cf. the Google Reader precedent).

- **A minimalist approach to authorizations** to grant actors during interconnections – see notably the difference between the theory behind authorization, e.g., OAuth, and their actual implementations of *all or nothing* (or: why do you need access to my contact list to pass a message?)

- **Interoperability does not mean decentralized**, the Facebook algorithms will remain dominant and predatory, and will work in parallel to independent decentralized services.

- **Citizens cannot accept that public services use or impose usage of private services**: why not then start with explicitly exposing the issue of centralization and committing ourselves to unwind their presence

within our institutions, our schools, our health system, our administrative communications. Institutional support to open standards, as practised by the Commission within the scope of Next Generation Internet, would reinforce them instead of giving implicit legitimacy to centralized systems by merely asking those to be compatible with standard protocols.

In other words, interoperability alone remains insufficient, and can even prove harmful.

The threat to free software

An OFFDEM manifesto

by petites singularités

This text was originally published on January 5th, 012020 HE and was revised for this edition.

"It is this sum total of these modern attempts to perpetuate colonialism while at the same time talking about 'freedom', which has come to be known as neocolonialism."

— Nkwame Nkrumah, *NEOCOLONIALISM The Last Stage of Imperialism*, London, Thomas Nelson & Sons, Ltd., 1965

We[42], as a group of free software activists, have decided in 2020 to organize OFFDEM, an intersectional gathering around collective practices and free technology productions. Our reasons have been explained in a friendly post on p.s.: forum: `https://ps.zoethical.org/t/why-offdem/2867`

It now seems about time to voice our feelings towards the astonishing pre-emption of the developers communities by surveillance capitalist corporations. While there has been large social movements to claim "Fuck Off

---

42 OFFDEM was organized by a group of different collectives based in Brussels, such as le `HCKLABXL`, les `GNUragist.es`, `Neutrinet`, la `Voix des Sans Papiers`, la `Maison des Migrants`, `Source` radio show, the `Hashët` collective, and `petites singularités`.
    We thank `Instant City Harbor`, `HSBXL`, `Delta.chat`, `EDri`, ActivityPub `SocialHub`, the `Tor Project`, `CCC` and all the collectives who enjoyed the gathering.

Gaggle", and protests against Amygene/ burst all over the world, large protests in India against Fakebook, those companies and other brands are promoted by our "community" events, which also receive individual donations and are organized by people who are often volunteers.

We find unacceptable that our communities be associated with such companies: instead of supporting people around the world who oppose their domination, we cross their picket line. If free technology producers do not stand up to surveillance capitalist corporations, who will?

OFFDEM is *open for freedom, desire, emancipation, meaning*. It was a first, necessary step to affirm the existence of free software outside of the reach of surveillance capitalism.

Someone had to do it. We took the risk, and all the people who attended with us[43].

Do we need such large events?

OFFDEM vouches for decentralization, as most people only have one life and can achieve a limited number of projects, we do not need to be all at the same time at the place, let's organize and promote smaller and more focused meetings on different topics aiming at practical achievements for the communities they support.

Do we actually reach out to our audiences?

The organization of small events can be done at a lower cost and in better conditions than amassing a large number of professionals without asking the question of the uses, the

---

43   including some who came to OFFDEM who were
     not attending FOSDEM.

intentions nor the modalities of technical developments. Opening up to otherness consists in working in proximity with resistance networks, on the ground, in order to open up to plural realities.

Do we question the power relations in technical production?

The comfort of privilege blinds us to the stakes of technique. We are numerous and we are well informed! However, we are permanently in an *entre-soi* favoured by an individualistic culture that limits the organization. For example, we still do not reach the regions of the world outside of the West, where half of our workforce is located, and who are confronted with even harsher forms of domination: we are hardly aware of the concrete problems generated by our activity, nor of the alliances we could make.

Since 2020, OFFDEM has been proving its relevance, because the moments spent together are materialized in acts, the choice of decentralization opens up possibilities and explores new paths outside of surveillance systems.

We want to affirm the possibility of gathering in a different way, in conviviality, comfort, benevolence and hospitality; we want to remind that this form of convivial gathering reflects the values of our community much more than any intensive corporate event, which FOSDEM has become over the years, shaped by the mentality of surveillance capitalism and Silicon Valley. This is not like us, and we will not let them crash our party.

The factory of technical violence

The material origins of powerlessness

Change comes only through action, but the simplest action seems unattainable. As producers of technology, we remain caught up in the dominant operating models, despite our awareness of the reality of the situation: our actions continue to bring destruction beyond our borders.

The infrastructures that organize our communications and industrial production are in the hands of actors who seem beyond our reach. If contemporary activists have never had so many means to organize, the scales of destruction and violence of the military-industrial complex are exponential, leaving barely any interstices for our action.

We also know from experience that the recurrent re-appropriation of all our community models by an insatiable system shows the power of our associative capacities. This pattern of co-option by industry repeats itself over and over again. "Their resource radar detects what can be pumped out for free and comes to suck the energy, according to the famous principle: Embrace, Extend, Extinguish.[44]"

From systemic violence to technological cannibalism

We can observe, in our ultra-liberticidal, madly capitalist world, that everything our production system touches is immediately destroyed, just like the legendary King Midas who turned everything he touched into gold,

---

44 *Embrace, Extend, Extinguish.*
//en.wikipedia.org/wiki/Embrace,_extend_and_extinguish

until he could no longer eat or drink. These fatal consequences are borne by all ecosystems and by the most vulnerable people in our society.

It is a widely accepted fact that we have blood on our hands, that every day our comfort is provided by soldiers who monitor the mines, fly the drones that will erase lives through a screen, out of sight, out of mind, funded by "structural programs" to "defend" access to "our territories," by politicians who delegate atrocities to "regimes" set up and maintained by "diplomatic", "commercial", and "democratic" coercive forces. The propaganda is unquenchable on the benefits of "civilization" — ours, but remains silent on its mass graves — theirs. Similar is that which resembles us and which industry can assemble; beyond this utilitarian distinction, we fall into

the implausible – yet the norm. Yet this reality is most often impossible to address at the heart of our organizations, and we have just passed a point where the discourse of power "saves lives one by one" while, in the meantime, technological solutions to social problems are still being sought, in vain. For the propaganda strives, with disconcerting ease and success, to put all the weight of the responsibility of the system on individuals — not collectively, en masse, but in isolation, in a detached way, intimately accused – on the individual atomized by this system that deconstructs him.

The proposal to OFFDEM is to trust in the capacities and knowledge of our networks of resistance, the only ones capable of inhabiting the interstices, of forging links according to other modalities, lively, perennial; in the face of insurmountable pressure, to take a step

aside and erase the burden by letting it fall under its own weight in order to consider the facets that usually remain invisible: those strings pulled and frayed, those empty words, those shareholders without action, those financiers without thickness, those one-way mirrors where the emptiness of accusing and demotivating speeches is reflected ; then watch, from the embankments where we are, the train of progress and growth pass by, hurtling towards a mountain whose tunnel at the end of the rails, however real, has just been painted by a mischievous *Geococcyx californianus* who will revel with the audience in the devastating compaction of the ultimate crisis – if the audience ever survives it. On the embankments grow grasses, fragile and thirsty, carrying the whistle of the winds even after the catastrophe.

OFFDEM and THX are the points of connection where we will continue to think together and build the tools, the methods, the collective, that will allow us to take that side step we talk about all the time; to branch off, to pull the brake, to continue elsewhere, differently, here and now...

What are the conditions for escaping from systemic pressure? We imagine them in the collective. We wish them out of the compromise. We know that they are subject to the inertia and tentacles of reality. Far from summing them up as a life recluse behind a screen, between four walls, behind an explosion engine or next to another jet engine, or surrounded by men in arms, we conceive them, intimately, also as propitious to another relation to the world, subjected to a desire to live together, to the will of a good life.

What is alive is dephasing to be no longer oneself – and this is how it remains. The side step, it is this dephasing, this force of the living to accept nothing of the ineluctable, to make it null and void every time that it comes to announce its triumph. The life is what resists entropy, to the ultimate homogenization, the uniformity of the world towards the sand of time, the announced end of the universe; but in the meantime, we are there, here and now, everywhere to affirm compossible living worlds.

O$_4$FFDEM Call for Presence

OFFDEM is an intersectional festival about collective practices & free technologies production.

O$_4$FFDEM will happen on **the first week-end of February 2024 in Brussels**, as you might expect.

This is not a *Call for Participation* but a *Call for Presence*, since this is that quality we expect from all people involved with OFFDEM.

At OFFDEM everyone contributes to the making of the event. we do not hold individual presentations but conversations, all the propositions sent will be shared on our forum and developed collectively.

We welcome proposals from collectives interested in exploring the following topics and sharing their experiences with others.

2-3 February 2024, Brussels

OFFDEM originates from the free software movement and considers freeing social and intellectual production as well as collective care and maintenance of software.

The Call For Presence is Open

OFFDEM already started on our forum as we value preparing the event together to make sure our common time is fruitful and agreeable, spent in resolving issues and planning actions rather than listening to formal presentations.

Please send the proposal for a contribution by email to `offdem.0x04@offdem.net`, explaining what you have in mind, the people or groups you will engage, and your needs. This e-mail

will be published in the section of `OFFDEM` forum[45] dedicated to O$_4$FFDEM that all logged in person can see. You should list the personal emails of all people in your collective who want to participate, so each will be added to the `@offdem.0x04` group before the event to prepare material and structure your contribution, eventually people from the OFFDEM community with shared commonality of interest will join in the process. We will together discuss the details, including how it articulates with other proposals. When applying, please be prepared to give some time to follow up with your application during the preparation process.

---

45   https://oxygen.offdem.net/

Awareness of war-ness calls wariness. While battlefields tend to multiply, governments are increasing their military budget to levels unseen since the Cold War. These industrial investments are not only engaging armies into massive killings but are also threatening civic movements and divergent minorities. The militaro-industrial complex is not a foreign space, it is the means of existence of Occident. As technologist we know its not so far and if we dig a little into our network and resources we quickly encounter military affiliation.

We can be concerned in two main ways. In solidarity with the people aimed by both the software and hardware ±high-tech weaponry developed in our wealthier countries. Then in consideration for our own safety and our capacity to face potential aggression from

inside and outside those. In the shadow of highlighted international *binarism* ("Either you're with us, or you're against us."), a vast complex irrigates both sides of every battlefield, taking profit in return of their spreading.

In this context, growing *strong* bonds, sharing intelligence and *joyful* care between communities is getting vital. **While the industrial economy is accelerating its destructive course against our environment, killing all forms of life for the profit of the capitalists, improving both our knowledge and our cohesion contributes to disarming our enemies, towards emancipation.**

Whatever the walk of life that brings you addressing those concerns altogether, let's face it!

OFFDEM topics

> *It is necessary to consider that there is a machinic essence which is going to incarnate in a technical machine, but also in the social, cognitive environment, tied to this machine – the social ensembles are also machines, the body is a machine, there are scientific, theoretical, informational machines.*

> – Félix Guattari, *Chaosmose*, p.72, Éditions Lignes, 2022 (own translation) (1st edition Galilée, 1992)

Local Organization

Don't look up! Look around: the military industry

Resistance OPSEC: working together safely

Protecting *life essentials*: water and air

Collective Data Sovereignty

Data hacktivisim

Removing functionality: serene minimalism to save energy

Joining forces to compose programs that work well together

Interoperability & Power Relations

Right to connect: surviving internet shutdowns

Public *interoperability poisons* and remedies

Internet from the military to a public digital infrastructure, what public agency?

Where

OFFDEM is organized in collaboration with collectif Zone Neutre and la Voix des Sans Papiers, both uniting undocumented citizens in Brussels, in VSP's occupation rue Fritz Toussaint, at walking distance of the FOSDEM venue near Ixelles Cemetery. This is where we're going to host OFFDEM — this is the venue of the first OFFDEM! We have a small number of small rooms, we will host a single track and leave a little space for people who need to informally gather and work further topics. Be sure to notify your proposed contribution as explained in the Call For Presence.

Why

Key points:

- Free software *requires community*, away from surveillance capitalist sponsors;
- Free software needs to *reconnect with the rest of society*;
- Among free software community, we *promote cooperation and decolonization* (starting with our own minds.)

~~Post-truth~~ Afterword

Just before the world turned upside down in March 2020, talking about the artificializing of the world towards a digital whole, which is draining life and attacking relationships as it brutally removes ore from the bowels of the earth, was both far from us and yet remarkably prescient. Achille Mbembe associated it with an Africanisation of the world whose brutalism threatens our social structures in favour of systems of domination.[46] The smooth society[47] presented on our screens detaches us

---

46 Achille Mbembe, *Brutalisme*, Paris, La Découverte, 2020

47 The *global wage report 2020-2021* by the ILO states: In times of crisis, the level of the average wage can change significantly simply because of major changes in the composition of employment, the so-called "composition effect". When most of those who lose their jobs are low-paid workers, the average wage that is calculated for the rest of the employed automatically increases.

from real conditions. The digital divide[48] puts many people at risk of being marginalised and excluded from the digital world. The digital divide offers the system in place a population of abusable people who are indispensable to maintaining the materiality of the world, because not everyone can be dematerialised. This is part of the illusion of progress. The question remains of how these margins will form a society and associate another relationship with the world and with the living.

It would seem that the conditions of Mbembe's analysis have been consolidated as the dominance of the digital, i.e. the digitalisation of processes essential to society in our contemporary world, has strongly

---

48 We are considering here the second degree of the digital divide, linked to the *use of technologies*. We will come back to this subject in a future *opus*.

penetrated the fields of administration, leisure, business, the university, creation and the social link. This process, recognised and questioned by many, has nevertheless been imposed without discussion thanks to a "crisis"; supposedly temporary "crisis" digital devices such as teleworking, videoconferencing, online or contactless payments, dematerialised and intermediated relations, health checks or compulsive identification find themselves anchored at the very heart of a societal upheaval.

For those of us who "come from the Internet", it is imperative to reflect on the modalities that are imposed on us.

First of all, we must affirm, although it is obvious, that there is no equivalence between remote digital means of communication and a physical meeting: one does not replace the

other. Different technical means offer different possibilities; if synchronous means of communication are useful from time to time, it makes no sense to use them between neighbours, both technically and energetically. Other means can be much more effective in allowing the expression of voices that for various reasons cannot travel: passing on thoughts before meetings, reading the minutes, commenting afterwards, etc.; working over a long period of time; the methods of participation are certainly different, but allow respectful exchanges at a distance. Preferring rigorous organisation to a video *patch* is much more productive for a group which, out of respect for people who cannot join them immediately, must take the time to synthesise and read rather than lose itself in the immediacy of technology.

The myth of the digitisation of the world, which is attached to the idea that the media can compensate for the lack of relationships, allows the underlying acceptance of the continuation of the myth of progress. This preoccupation is reflected in the words: "the world before", "the world after"... Yet the proclaimed urgency avoids any questioning and seems to have no other aim than to cling to a single vision whatever happens, even if it means locking everyone up at home. This choice to sacrifice populations to the altar of globalised commercial and digital circuits has serious consequences.

Is replacing a meeting with a video conference relevant? Isn't adding technological elements: high-definition jingles, home studios and other DIY proposals to make the space more interesting a second-best solution? Such

additions increase the technological debt, the energy cost, and the inequalities between those who can afford such systems and those who cannot...

The urgency invoked, since March 2020, for the widespread use of digital tools has largely defeated our capacity to act, both in terms of reflection and coordination of efforts. We were dismayed by an unexpected change that was based on screen addiction with its corollary consequences — loss of sensitivity ("people no longer feel themselves", loss of a sense of time, screen overdose, dysfunction of the circadian cycle, etc.), loss of reference points ("they die alone in their corner"). However, among the resistance groups, it was also an opportunity to get together, to find themselves outside the time imposed by another urgency, that of everyday life, which had suspended all

integration of digital techniques, left to the goodwill of corporations.

The state injunction forced us to give up things without having taken the time to decide, while the society of the spectacle engages us in a fear of the void (FOMO, Fear of Missing Out). Thus, in order to lock us up at home, the authorities relied on a digital palliative and the extent of what we were experiencing escaped us. It's hard to believe, but it seems that few people perceived that this was a societal shift.

Nevertheless, as in a good cyberpunk scenario, in this process of Africanisation of the world, the marginalised, whether willing or not, have often unexpected resources at their disposal. The long history of the relationship between empires and their beggars, their serfs and witches, their barbarians, their mongrels

and maroons, their fugitives, their bohemians and their undocumented migrants, remains unknown and hopeful.

Countermeasures

After almost two years of change, are we not ready to catch our breath, to evaluate the consequences, to regain control of our technical gestures?

Let's take the time to reflect on this imposition of the all-digital world, to observe our existing knowledge — did we not create the Internet without the intrusive instrument of the camera — and to ask ourselves what other possibilities exist that are not just imposed palliatives but tools that allow for the organisation and consolidation of collectives. Thus, asynchronous exchanges, the enhancement of our close networks, the links

that we want to weave across distance and the way to weave them durably through the use, thought out with parsimony, of technical means allowing us to avoid the trivialisation of the exchange.

The emotional band-aid of 'containment drinks' cannot be the basis of a societal choice. To address this situation we must now take an active stance; when we organise an online meeting, the first thing is to recognise the difference, it is a possibility of a different order than the time of a meeting in the same shared place. If we have to exchange orally with an intelligence that is on the other side of the planet, let us do it with joy as a precious thing that we will prepare, document and whose effects we will try to preserve in the long term.

It is essential to distinguish between the new possibilities offered by digital technology and its imposition on our privacy. It is possible to perform an act, a form of ritual. Ritualising as in distinguishing the benefit of the digital when it offers a new possibility of encounter; and also making it an exceptional moment of 'synchronous intensity' which leads to asynchronous follow-up or prolongs it; acting as an ice-breaker: the opposite of an obligation of (omni)presence of/to the camera.