

[Osuny case studies: degrowth.net/ + explore.degrowth.net/ \(3/5\)](#)

[yala](#) 1 Juin 23, 2024, 3:47

For a systems developer, nothing is more telling than a documented case. The people that have an informational-technical need will express so in their everyday language, and it the developers task to moderate a conversation, which cuts the case into a methodic form and formalises certain aspects, to fit the needs of a documented case.

This article is part of an article series about [Ergonomy of development experience and common reproducibility questions around maintaining degrowth.net](#).

The previous article in the series is [Working with Osuny out in the open: a replication study to increase the resiliency of the computational Common \(2/5\)](#).

The next article is [Observations and projections from half a year of using Osuny : "Co-construire un commun numérique" pour "Devenir un commun" \(4/5\)](#).

When people express their wants for remediations and highlight urgent needs due to a regression in the system, the developers transform these stories into functional and technical requirements based on behavioural data. They use these stories, called issues or epics, to navigate the field of neccessity and to prioritise interventions.

At the IDN, one of its first acts was to technically appropriate its name, which had been passed on to it from its origin. It was then given a content model, information architecture and visual design, which represented the collective opinion of the sociocratic decision finding in the IDN. Setup, Development and Procuration from many hands have led to the collective milestone of publishing their Osuny web site adapted and filled to their likings.



About ▾ IDN Members Join Us ▾ Assembly 2024 Resources ▾ 🔍

WELCOME TO THE INTERNATIONAL DEGROWTH NETWORK!

We are a global community of individuals, organisations, and degrowth allies committed to creating an equitable and sustainable world for all generations. We're here to build connections across borders while providing a collaborative space to develop alternative pathways toward a better future.

Screenshot degrowth.net above the fold, 2024-06-21

Which are the pieces that are involved in rolling out Osuny to the public, and which journeys do people follow to realise their intents with the web page?

Visitors will come here to watch the outcome of these efforts:

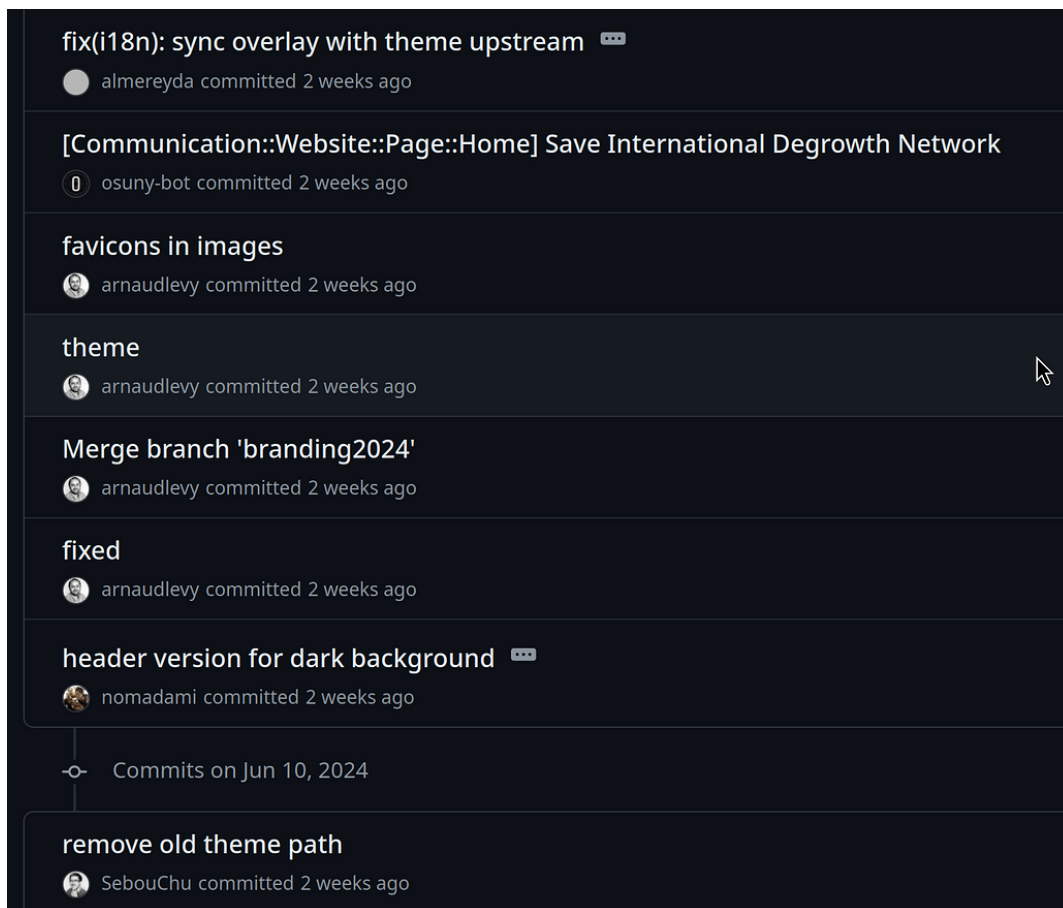
- web page: degrowth.net

Editors will ever only interact with the Osuny web interface for content management.

- web service: idn.osuny.org

Theme and system developers work with the source codes that define the rendering pipeline for the final web site artifacts.

- repository osunyorg/idn-website
- fork degrowth/idn-website
- fork [Commits · degrowth/idn-website](https://Commits.degrowth.net/idn-website)



Screenshot of a part of the so-called commit history of the `idn-website` git repository: Five contributors, of which one is a bot, can be observed interacting in the history of commits to the source code.

Build engineers that set up the running parts of the rendering pipeline in Continuous Integration (CI) will also want to interact with the manifests describing the pipelines to control those systems. They are used to turn the website data into the appealing website you know from above.

After logging in to Osuny and navigating to your website, you are greeted with a status badge carrying information about the last CI build run:



The fork is used for a preview environment of changes, in absence of PR automation. We can follow the different development cycles of the different environments by checking out the closed Pull Requests:

- preview web site next.degrowth.net/
- repository [Pull requests · osunyorg/idn-website](https://github.com/osunyorg/idn-website)
- fork [Pull requests · degrowth/idn-website](https://github.com/degrowth/idn-website)

Using and working with the repository has lead to many interesting discoveries. These were most of the time attached to concrete use cases and journeys a given developer wanted to provide an end to. The situations invited for looking at accessibility and development experience (DX) of the whole tooling that is, whether directly or indirectly, mobilised to build the Osuny web sites.

Accessibility and Development Experience (DX)

There is a whole science attributed to Accessibility and Software Ergonomy of the Development Experience (DX) of a Software development process. What do these points of views add up to what we already know?

I like this sentence, because it is true:

When developing an application, I am also one of its users.

That is, all rationales and values that we apply to User Experience Design also apply to tasks achieved with the development and deployment environments. They have interfaces, they have parameters and they have state, we pass through journeys, have our own stories and try to fit these into a case. And the developers are using those interfaces, writing those stories. Which effectively renders them users of the application, but at a different layer.

Programmers are the users of Application Programming Interfaces (APIs).

This is always good to keep in mind, also when designing collaborative Commons.

What is there to say about accessibility in the Osuny development experience?

Accessibility

Osuny generally makes it easy for you to change basic parameters of your site. Its web accessibility framework does a lot of work for you, in so readers will receive the best possible reading experience, no matter on which screen or screen reader. This is often overlooked in web site design, should but cannot be taken for granted and needs additional highlighting. Visually impaired people will thank you already for delivering your website with Osuny.

The documentation of the project, the application and the theme is highlighting the role accessibility plays in the considerations to make the Software:

1. [Accessibilité | Osuny](#)
2. [Accessibilité | Osuny](#)
3. [Accessibilité – Osuny](#)

When entering the development of an Osuny page via its website repository, you are implementing basic design choices there, see them applied to your website and usually don't return very often. Together with changing your content in the admin interface, it will be your only other interaction point with the Osuny system. Osuny, that is the back office application and my website repository.

Pushed my first commit to the website! I changed "site web" to website and "address" to location haha #babysteps. Trying to get a github pages website built as a testing environment for bigger changes but I have hit a tech wall so my developer friend is gonna come over later and help me fix it 😅



Internal message in the IDN OC Tech Circle channel:

« Pushed my first commit to the website! I changed « site web » to website and « address » to location haha #babysteps. Trying to get a github pages website built as a testing environment for bigger changes but I have hit a tech wall so my developer friend is gonna come over later and help me fix it »

Eventually you will need to apply other modifications to your repository, making use of Hugo's internal overlay mechanisms. Or you have come to touch the theme and find yourself in a situation where you want to change that: You can and it's all there for you.

« Dans le thème on a un dossier i18n dans lequel on pose un fichier par langue. Ces fichiers (ou juste certains termes) peuvent éventuellement être overwrite dans un thème précis. »

via [Les clés de traduction : Multilingue – Osuny](#)

At some point you are probably going to hit a technical wall. This may be due to how the theme is integrated as a git submodules, or how changes are supposed to propagate inbetween the different repositories that make up a complete Osuny site.

Keeping a custom fork of the repositories in synchronisation with upstream releases is challenging, not only because it sometimes moves fast and brings many changes, which can be rebased upon, but also due to the many places in which changes can be applied: the admin, the web site *and* now also the theme.

Their models and views need to be kept in sync, to provide meaningful output and to avoid error. Additionally the distributed version control system git is used in an automated manner, which surpasses common and useful code collaboration conventions.

Due to the *bot* pushing directly to the `main` branch of the repository, *the default branch `main` cannot be protected*. This in return invites collaborators to circumvent a reasonably accountable Pull Request workflow by directly pushing to `main`. This removes an important checkpoint and communication channel about changes inside the code base from the code collaboration workflow and invites for trouble.

As we recognise that accessibility is not just a technical term that names technical specifications which must be tested and cared for, we find that the term also describes the ways in which meaningful ways people can *access* and *develop* the distributed system that is Osuny, by interacting with its maintainers, workflows and artifacts. And things are accessible, when they are documented. Which in itself is a good test for the viability of a feature.

That is, good accessibility in development can mean, that we keep promises and expectations in check and in balance, by writing them down.

Development Experience (DX)

Because when things aren't written down, or accessibly so, people need to figure out things on their own and potentially invent unconventional ways to break your workflow. It took us quite a while from discovering the source code repository of the website until we knew what to do with it. The *figuring out* ranges from *randomly changing things to see what happens* to *reverse engineering the application architecture with scrutiny* and can have varying degrees of results.

The care for the development experience of third-parties, which will likely interact with your software, as it is licensed under a permissive license and freely available out in the open, means to understand the design and usage of the APIs as the outcome of a human conversation about needs and affordances to make them work « as expected ». Providing channels to participate in the conversation, agreeing on what can be expected and documenting this decision can already help to avoid many misunderstandings. Even more so when building publicly for the Commons.

After dealing with and learning about the page workflow, we now have an independent environment for previewing changes to the IDN website at <https://...>

This is what we changed so far: <https://github.com/osunyorg/idn-website/compare/main...degrowth:idn-website:dev>

Quite trivial changes, but already collecting a few hours for multiple people to reverse engineer the un(-der)documented Osuny artifacts. (bearbeitet)

Internal message in the IDN OC Tech Circle channel:

« Quite trivial changes, but already collecting a few hours for multiple people to reverse engineer the un(-der)documented Osuny artifacts. »

There are three kinds of developer documentations, none of which could

- [Développer un site – Osuny](#)
- [Contribuer au thème – Osuny](#)
- [Contribuer à l'admin – Osuny](#)

The usage of the term admin here invites for confusion when wanting to distinguish an Osuny admin (of the application, in the application) of the systems administrator that runs the system.

To the User Manual and the Developers Manual I would suggest to add an Administrator Manual.

There are two terms used in multiple contexts with different meaning:

- Developers is ambiguous, because it can mean the people developing an Osuny site, or the people developing Osuny.
- Admin is ambiguous, because it can mean the people that are equipped with administrative privilege inside an Osuny site, or the people that run the systems on which Osuny runs.

Classical documentation is often structured in:

- User manual
- Developer manual
- Administrator manual

With regards to the usage of the shared git tree, when there are collaborators outside the core development team of Noesya/Osuny, it appears useful to resort to conventional workflows, find a way to protect the `main` branch and to work strictly from PRs, restricting direct (force-)pushes to emergency hotfixes.

Contribution model

Eventually we will find a contribution model, that can be written down and exchanged between peers. The Software is close to useless, when there is no meaningful way to interact with it in certain circumstances. Different people will demand different kinds of responses from the socio-technical system at hand.

When conventional ways are given, that guide people along their paths, a lot of ambiguity and misunderstanding can be eradicated, by not letting it happen. If we want to play with Osuny in the grander scheme of things, it will be good to introduce conventions, patterns and Standards that are

known to work well in other similar circumstances.

One such convention is the presence of a `CONTRIBUTING.md`.

I'm feeling we're now at a level where we're quite juggling around and more and more people work on the repository, why it seems suitable to come up with a clear contribution workflow.

Maybe a `CONTRIBUTING.md` would be the correct way of establishing shared understanding.

 **Arnaud Levy (noesya)** did not yet and will not join the channel here, will they?

Maybe it can be good to have a separate channel for working on the technicalities of the website, in so a shared log can be maintained, without overusing this channel here? (bearbeitet)

I'm counting four people from two organisations that maintain the code:

<https://github.com/osunyorg/idn-website/graphs/contributors>

- nomadami
- arnaudlevy
- SebouChu
- almereyda (me)

It may be good to establish a direct communication channel between all of us.

Internal message in the IDN OC Tech Circle channel:

I'm feeling we're now at a level where we're quite juggling around and more and more people work on the repository, why it seems suitable to come up with a clear contribution workflow.

Maybe a `CONTRIBUTING.md` would be the correct way of establishing shared understanding.

[Arnaud Levy \(noesya\)](#) did not yet and will not join the channel here, will they?

Maybe it can be good to have a separate channel for working on the technicalities of the website, in so a shared log can be maintained, without overusing this channel here?

I'm counting four people from two organisations that maintain the code:

[Contributors to osunyorg/idn-website · GitHub](#)

- nomadami
- arnaudlevy
- SebouChu

- almereyda (me)

It may be good to establish a direct communication channel between all of us.

In the specific situation a side-channel was needed, when multiple implementations by different developers following disparate intents started to have side-effects on the overall consistency of the repository. The resolution was already there at hands and simple: we choose to use Pull Requests (PRs) to contribute to the repository, which all watching the repository will see and receive notifications for. And contributions were flowing again.

Let's forget all of the above and finally look at the concrete development stories with Osuny, which we happened to follow at the IDN. Their story is told in four acts and starts slow with just activating the website, increases the pace with bringing in own changes, works around the limitations posed by a single source of truth and lastly paves the way for a visual reappropriation of the web site with applying a brand design.

1. Setting up the website on an apex domain with using an ALIAS
2. Appropriating and adapting the website to update the theme and to use custom translations
3. Building the website with a Hugo GitHub Actions workflow into a separate GitHub Pages deployment at the fork
4. Collectively developing a style configuration and integrating it in a custom branding update

We will walk through these in due pace and focus our attention on unexpected behaviours and unintended side-effects.

1. Setting up the website on an apex domain with using ALIAS

Setting up a website is simple, right? You buy a domain, you point your DNS at the webserver, you configure the webserver to serve your domain, turn it on et voilà, you're done.

Not so much here. Thanks to the special way how the DeuxFleurs Garage works, remember the thing form another galaxy, which involves the gravitational field of further celestial bodies, called Nomad and Consul, we need a special kind of DNS Resource Record (RR), because we want to serve the apex of the domain, it's zone root at @ , in globally in the Internet known under the name `degrowth.net` .

Because Nomad will produce roaming IPs for the Garage deployments, their IP addresses are only eventually consistent and can change over time. This would be an easy use case for a `CNAME` record, the canonical name of an address under which it shall be known, and then you change that and the roaming is back in sync with your DNS zone.

Not so much for an apex record, which comes with different consistency guarantees and different

operational constraints. Why ALIAS records were invented, a semi-standardised not-so-much extension of the DNS protocol, which some DNS servers support. Luckily for us at eco:bytes, who maintain the IDN zone, we were already using the PowerDNS API, which has this feature available and which was only one click and two settings of redundant resolvers away.

Zone Records - degrowth.net

Zone Editor

15 records

Name	Type	Status	TTL	Data
@	ALIAS	Active	3600	garage.deuxfleurs.fr.

Set up this way, when a client requests the address `degrowth.net` we will respond with the IP addresses which we resolve for `garage.deuxfleurs.fr` in the background. This way we will always be up to date, also in the situation when they are expected to change.

Screenshots from querying the DNS for the name `degrowth.net` :

A	AAAA
<pre>\$ dig A degrowth.net ; <<>> DiG 9.18.26 <<>> A degrowth.net ;; global options: +cmd ;; Got answer: ;; ->>HEADER<<- opcode: QUERY, status: NOERR, flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 0 ;; OPT PSEUDOSECTION: ; EDNS: version: 0, flags:; udp: 65494 ;; QUESTION SECTION: ;degrowth.net. IN A ;; ANSWER SECTION: degrowth.net. 300 IN A 91.121.121.121 degrowth.net. 300 IN A 91.121.121.121 ;; Query time: 55 msec ;; SERVER: 127.0.0.53#53(127.0.0.53) (UDP) ;; WHEN: Sat Jun 22 00:13:57 CEST 2024 ;; MSG SIZE rcvd: 73</pre>	<pre>\$ dig AAAA degrowth.net ; <<>> DiG 9.18.26 <<>> AAAA degrowth.net ;; global options: +cmd ;; Got answer: ;; ->>HEADER<<- opcode: QUERY, status: NOERR, flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 0 ;; OPT PSEUDOSECTION: ; EDNS: version: 0, flags:; udp: 65494 ;; QUESTION SECTION: ;degrowth.net. IN AAAA ;; ANSWER SECTION: degrowth.net. 212 IN AAAA ::: degrowth.net. 212 IN AAAA ::: ;; Query time: 63 msec ;; SERVER: 127.0.0.53#53(127.0.0.53) (UDP) ;; WHEN: Sat Jun 22 00:15:19 CEST 2024 ;; MSG SIZE rcvd: 97</pre>

```
$ resolvectl query garage.deuxfleurs.fr
garage.deuxfleurs.fr: 2a01:e0a:2c:540::31 -- link: enp60s0u1u4u4
                    2a01:e0a:e4:2dd0::41 -- link: enp60s0u1u4u4
                    82.67.87.112      -- link: enp60s0u1u4u4
                    82.65.41.110      -- link: enp60s0u1u4u4
                    (global.site.deuxfleurs.fr)
```

Screenshot from querying the name `garage.deuxfleurs.fr`, which is itself also only a canonical name for `global.site.deuxfleurs.fr`. They have added another layer of indirection, probably to prepare for failover scenarios.

This was fun to set up and when the setting was made and the link between our systems was established, the site went live from its new server:

[International Degrowth Network | International Degrowth Network](#)

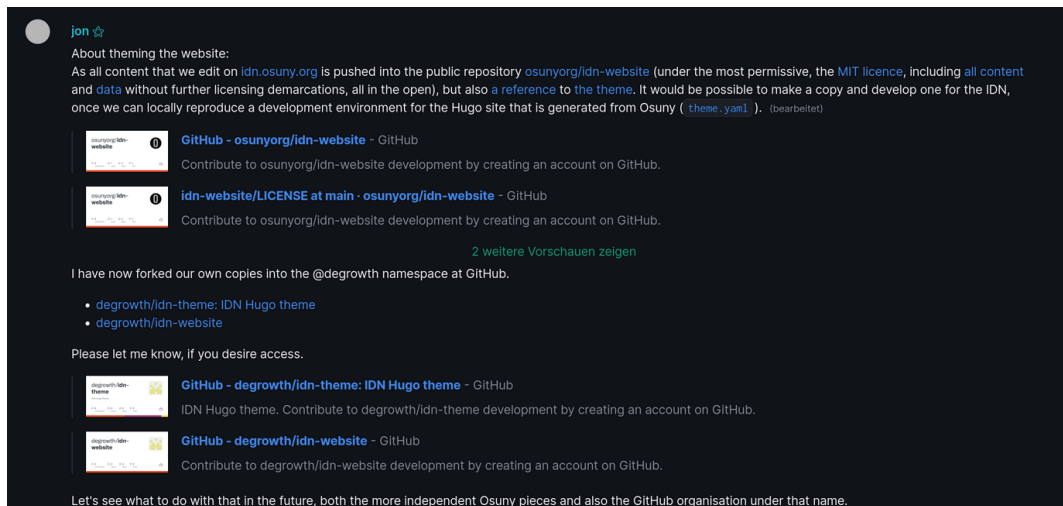
We are a global community of individuals, organisations, and degrowth allies committed to creating an equitable and sustainable world for all generations. We're here to build connections across borders while providing a collaborative space to develop...

2. Appropriating and adapting the website to update the theme and to use custom translations

Next we knew we could change the site to our liking, by modifying its underlying source code, the ideas started to spring. The other circles in the IDN were coming up with a brand design, in order to support the ongoing effort to substantialise the operation with the inauguration of legal body.

The source code posed another surprise. While being technical in nature, it also contained a complete copy of the Osuny content repository from the [idn.osuny.org](#) admin in text form. The source code had a permissive license, and there was no README around at the time, which could have told us more.

The situation was alleviated with introducing a copyright for all non-source-code assets in the repository, until the IDN had chosen to publish the content otherwise. Different Creative Commons constructs are available to choose from.



Screenshot from a post in the internal channel:

About theming the website:

As all content that we edit on idn.osuny.org is pushed into the public repository [osunyorg/idn-website](https://github.com/osunyorg/idn-website) (under the most permissive, the [MIT licence](https://en.wikipedia.org/wiki/MIT_License), including [all content](#) and [data](#) without further licensing demarcations, all in the open), but also [a reference](#) to [the theme](#). It would be possible to make a copy and develop one for the IDN, once we can locally reproduce a development environment for the Hugo site that is generated from Osuny ([theme.yaml](#)).

I have now forked our own copies into the [@degrowth](#) namespace at GitHub.

- [degrowth/idn-theme: IDN Hugo theme](#)
- [degrowth/idn-website](#)

Please let me know, if you desire access.

Let's see what to do with that in the future, both the more independent Osuny pieces and also the GitHub organisation under that name.

Our own playground allowed us to gain confidence in juggling the movable parts we were presented with. This helped us to come up with little tweaks to test our approach. We used the gained independence from forking the repository to update a few language strings and to introduce a preview environment.



[Feat: add preview environment & update theme](#)

osunyorg:main ← degrowth:feat/add-preview-environment

ouvert 04:58PM - 05 Jun 24 UTC  [almereyda](#) +459 -340

Supersedes #1

This PR builds the IDN website from a synchronised fork with an [...](#)

Another surprise waited for us in the repository. The website itself is a regular Hugo page, which uses a heavily adaptable theme to parametrise and templatisé its configuration state and display view. This gave even more confidence, because we learned that we were operating in known terrain.

The future looked bright, not just for the website repository, but also for the newly inaugurated GitHub organisation under the `degrowth` label.



[degrowth](#)

degrowth has 6 repositories available. Follow their code on GitHub.

3. Building the website with a Hugo GitHub Actions workflow into a separate GitHub Pages deployment at the fork

The next endeavour was upstreaming the preview environment, which was briefly mentioned on the side before. The code of the repository was amended with a configuration that introduces a technical convention for dealing with changes, that allows to create independent copies of the page for preview purposes.

- [add Hugo workflow for GitHub Pages by almereyda · Pull Request #1 · degrowth/idn-website](#)
- [Fix: Hugo workflow by almereyda · Pull Request #5 · degrowth/idn-website](#)

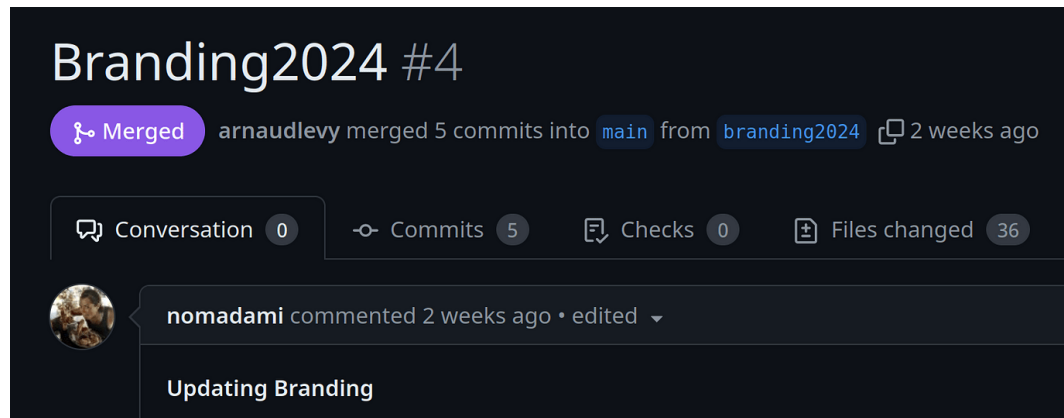
The repository also had a configuration file for a commercial service (Netlify), which was unused and deleted. Though a bit clunky in use, the fork with its web pages published on a separate domain allows to run experiments and prove certain hypothesis, before rolling out larger change sets to the public.

[International Degrowth Network | International Degrowth Network](#)

We are a global community of individuals, organisations, and degrowth allies committed to creating an equitable and sustainable world for all generations. We're here to build connections across borders while providing a collaborative space to develop...

4. Collectively developing a style configuration and integrating it in a custom branding update

The final draw that all of this year's interventions had led up to, was the condensation of the newborn IDN design language into website code and media artifacts for publishing the website under a custom theme. This brought changes to fonts, colours, logo, icons and website style.



This concluded just a week before this year's Degrowth conference in Pontevedra and probably made some people a little happy.

It certainly will affect the future visitors of the site.



Which concludes the round of little introspection into the steps that lead up to the current website of the International Degrowth network. I hope you had unexpected encounters with new ideas and perspectives, else the effort wouldn't have worth it.

Shout outs to anyone involved at the IDN, who held the discussions around branding, public communication and website content alive and well. You're the ones this stuff is here to provide a face and voice to for participating in the global chatter that is the World Wide Web.

Case study: explore.degrowth.net/

The second website was rather easy to set up, as it existed already and it was deemed for a subdomain of degrowth.net, no apex or ALIAS involved.

1. An existing Osuny website was already existing, completely set up and a sub domain name under

the apex domain name `degrowth.net` was chosen.

2. Setting the name of the chosen subdomain in the CNAME record for the canonical name of `explore` to `garage.deuxfleurs.fr`.
3. Sending an email to Noesya and DeuxFleurs, that the setting has been made and the switch was done immediately.

It can't all be that complicated.



[So you've heard about Degrowth. | Explore degrowth](#)

Here, we aim to provide you with:  Knowledge on degrowth that is both diverse and accessible  Tools to make communicating degrowth fun and easy  Inspiration to help you imagine the good life with less

This article is part of an article series about [Ergonomy of development experience and common reproducibility questions around maintaining degrowth.net](#).

The previous article in the series is [Working with Osuny out in the open: a replication study to increase the resiliency of the computational Common \(2/5\)](#).

The next article is [Observations and projections from half a year of using Osuny : "Co-construire un commun numérique" pour "Devenir un commun" \(4/5\)](#)

[Ergonomy of development experience and common reproducibility questions around maintaining degrowth.net](#)

[Observations and projections from half a year of using Osuny : "Co-construire un commun numérique" pour "Devenir un commun" \(4/5\)](#)